A Parsimonious Method for Offline Freeway Travel Time Estimation From Sectional Speed Detectors

WILLEM W. E. HIMPE,1 CHRIS M. J. TAMPÈRE,1 and BIEKE MOELANS2

1Center for Industrial Management/Traffic & Infrastructure, Faculty of Mechanical Engineering, KULeuven, Leuven, Belgium
2Departement Mobiliteit en Openbare Werken, Antwerp, Belgium

Online travel time estimation is receiving increasing attention. However, offline algorithms remain important, as databases of historical travel times are needed for many intelligent transportation systems applications. In this paper a parsimonious method is presented for estimating off-line travel times based on local speed measurements along a route with uninterrupted flow, for example, on motorways. The interpolation between the different detector locations is done by reconstructing trajectories based on traffic flow theoretical principles. The proposed method is efficient, easy to calibrate, and shows reliable results even when detector spacing is relatively high. The method is compared to other existing methods and validated on a 40-km highway.

Keywords: Dual Loop Detectors; Motorway; Offline Travel Time Estimation; Speed Interpolation; Traffic Flow Theory; Vehicle Trajectories

INTRODUCTION

Historical travel times along a route are important inputs for a wide range of intelligent transportation systems (ITS) applications like route navigation and travel time prediction. Recent large-scale application of complex data-gathering techniques like floating car data and vehicle recognition systems provide us with individual recordings of trips and by consequence allow for a direct way to measure the travel time along one specific corridor. However, in practice these new systems may have limitations, for instance, because they use a limited or biased sample, have limited accuracy, or are privately operated and hence expensive and subject to property rights issues. Meanwhile, other data are abundantly and publicly available. Local or spot detectors like single loop (SL) detectors, double loop (DL) detector stations, and automatic incident detection (AID) camera systems, on the other hand, are applied along almost all road networks. Because these systems have been recording the traffic state for a decent time, large historical data sets are available for analysis. However, local observations do not provide information in between detectors, so interpolation algorithms are indispensable to infer travel time information.

Over the years, different interpolation algorithms have been proposed in literature. It has been shown that interpolation algorithms that do not account for traffic dynamics and that queue evolution lead to poor travel time estimations, especially if detector spacing becomes larger (Soriguera & Robusté, 2011a). The method discussed in this article elaborates a method originally proposed by Coifman (2002), who reconstructs local vehicle trajectories in the neighborhood of a local detector according to traffic theory. Our contributions are the following:

• The method developed by Coifman, originally conceived for use with a single detector and individual speed observations, is reformulated and elaborated for travel time estimation over an entire corridor, based on speed observations in fixed time intervals. This so-called “average trajectory speed based” or ATSB method allows for a compact formulation that is now explicit, rather than the original implicit calculation.
• Even though the algorithm still contains some traffic flow theoretical simplifications, careful empirical validation of...
Apart from its computational efficiency, which may be of little importance for offline application, the main advantage of the proposed ATSB method over alternative methods that also consider traffic flow theoretical interpolation is parsimony: ATSB basically relies on only one parameter with a clear physical meaning, the maximum backward wave speed. Moreover, it is shown that estimation performance is not very sensitive to the exact value of this parameter. The method uses two more parameters for preprocessing of speed measurements, which are, however, needed for any travel time method building on local speed measurements and could be replaced by any more advanced method.

In the next section, it is explained why travel time algorithms in general are important and an overview is made of the most known traditional speed interpolation methods. In the third section, the ATSB method is described. The fourth section presents an impact analysis of some simplifications in the algorithm and their effect on estimation performance. Next, in the fifth section, a real-world experiment is set up. We have implemented four methods to be tested on an existing highway that is equipped with speed sampling stations. The sixth section compares the estimation performance of the different methods. In the seventh section, the results are discussed and we conclude with our recommendations for the implementation of the method.

OVERVIEW OF TRAVEL TIME ESTIMATORS

There is an increasing demand for travel time calculation methods both for offline estimation and for online prediction. The difference between the two is the discrepancy in the available information that is used as input to the model. Online calculation methods are limited to historical and instantaneous traffic information, while offline methods can use all available information, both before and after the departure time for which the travel time is calculated. As a consequence, offline algorithms can be used only in hindsight and are not applicable in real time.

The offline travel time estimation problem, which is handled in this article, is essential for a number of applications justifying its continuous development and analysis. First, offline travel time estimates are an important performance measure for policymakers. For example, the Flemish authorities are currently reconstructing and storing the daily travel times on their entire highway network. Initiatives like this are necessary to monitor the performance year after year and to evaluate traffic managing implementations that influence accessibility and travel time reliability (Chen, Yang, Lo, & Tang, 2002; Chen, Skabardonis, & Varaiya, 2003; Kraan, van der Zijpp, Tutert, Vonk, & van Megen, 1999). A second use of offline travel times is for calibrating travel and traffic models (Ashok, 1996; Cipriani, Florian, Mahut, & Ngro, 2010). These models are being used to evaluate implementations of advanced traveler information systems (ATIS) and intelligent transportation systems (ITS).

The method proposed in the next section belongs to the widely used class of trajectory methods. Trajectory methods simulate fictive vehicles that travel along a route in time. The result is a path in space and time, also called a trajectory, and a direct estimate of the total travel time. As highlighted in the introduction, there is no speed measurement available for every location along a route. In between detector stations there is no direct information on the actual speed and one has to make speed estimations. The most basic approach to the estimation problem is making some sort of interpolation of the speed in between two detectors.

The piecewise constant speed-based (PCSB) method (with midpoint algorithm being a special case) is the simplest interpolation algorithm (Thijs, van der Zipp, & Bovy, 1999). It assumes that around the detectors the traffic conditions are equal to the measured state. Next it is assumed that in the middle between two detectors, the traffic conditions change abruptly. The method then simply integrates the speed along a path through this discrete speed field. Piecewise linear speed-based (PLSB) and piecewise quadratic speed-based (PQSB) methods are similar but use continuous functions to interpolate the speed between two detectors, a linear interpolation function in case of the PLSB method (Van Lint & van der Zijpp, 2003) and a quadratic interpolation for the PQSB method (Sun, Yang, & Mahmassani, 2008). The PCSB, PLSB, and PQSB methods only make assumptions on how traffic conditions are spread in space. These methods do not take into account the time-dependent relations of traffic conditions. Making such unrealistic assumptions will lead large estimation errors (Soriguera & Robusté, 2011a), especially when the distance between detectors increases (as is illustrated in our case study later).

The filtered inverse speed-based or FISB method (Van Lint, 2010) is a first method that incorporates time-dependent relations. Traffic between detectors is interpolated according to the adaptive smoothing method proposed by Treiber and Helbing (2002). This filter approximates the speed at every position in space and time based on spot speed measurements along kinematic waves. With some basic manipulation, this filter can be used to refine a grid of interpolated speeds, on which trajectories can be calculated similar to the PCSB method. As we confirm further in this article, the results obtained with this method are superior to the more basic PCSB, PLSB, or PQSB, especially when detectors are far apart. The downside of the FISB method is that it relies on a relatively large number of

intelligent transportation systems vol. 18 no. 1 2014
parameters (kinematic wave speeds, speed threshold, smoothing factors, and sample grid size), the influence of which is not always straightforward, as most do not have any physical meaning.

Coifman (2002), exploiting the same principle of kinematic waves, proposed a travel time estimator using data from a single dual-loop detector. Following the widely accepted fact that in congestion speed information, like perturbations, moves upstream at a constant kinematic wave speed (Kerner & Rehborn, 1996; Lighthill & Whitham, 1955), it is possible to derive the average speed upstream or downstream of a detector over a limited length for each vehicle that passes the detector. By integrating these speeds, Coifman reconstructs trajectories and ultimately travel time. The algorithm discussed in this article is a variant of Coifman’s method, which following the nomenclature considered so far will be denoted the average trajectory speed-based (ATSB) method.

Apart from all methods cited so far, which are based on speed interpolation in between detectors, one can also formulate algorithms that rely on density and flow of traffic instead of speed. These methods rely on Little’s law, which states that the average delay in a queuing system equals the number of vehicles in the system divided by the exiting flow. An example of such a travel time estimator is the method of Westerman, Hoogendoorn-Lanser, and Van der Vlist (1995). Some methods combine the two methodologies (de Ruiter, Schutten, & Frijdal, 2002). Recent research (Hellinga, 2002; Herrera & Bayen, 2008; Van Lint & Hoogendoorn, 2010) showed that fusing different sources of data to estimate the travel time is advantageous. Other approaches to the estimation problem consist of using general mathematical models for time-series analysis (Yang, 2005) like, for example, artificial neural networks (Dharia & Adeli, 2003; Van Lint & Hoogendoorn, 2010) or even support vector machines (Wu, Ho, & Lee, 2004). The aforementioned methods have been applied in short-term forecasting, but it is rather easy to reformulate all of them in the offline format.

**METHODOLOGY**

In this section, the underlying theory of the ATSB methodology and some practical implementation issues are handled. The algorithm is briefly outlined, after which each step is elaborated. The first step consists of estimating an average travel time in between two detectors based upon either the upstream or the downstream detector. Next, these two are combined into an estimate for the average speed for the segment between two or the downstream detector. Next, these two are combined into an estimate for the average speed for the segment between two detectors. The final step is to calculate the total travel time on a route consisting of multiple segments.

**Speed Interpolation in Congested Segments**

First consider a road segment between two detectors and assume congested traffic conditions. The upstream detector is called $U$ and the downstream detector $D$. ATSB assumes traffic to behave as described by kinematic wave theory (KWT). In congested regime, perturbations move upstream with characteristic speed $w$ (see Figure 1). The characteristic speed $w$ is assumed constant (i.e., no function of the congested speed or density), corresponding to simplified KWT (Newell, 1993) and empirical research that found values of $w$ ranging from 19 to 25 km/h (Mauch & Cassidy, 2002; Windover, 1998).

Figure 1 shows the relationship between a vehicle trajectory, its average speed $\bar{v}(t^*)$, and the speeds in the congested area that propagate upstream with speed $w$. Along its trajectory, the vehicle’s speed is assumed to be piecewise constant, which breaks down the trajectory in $n$ piecewise linear segments with speed $v_i$. The figure shows that speed $v_i$ can be observed at both detector locations $U$ and $D$. These detectors record average speeds in fixed time intervals $dt$, which actually define the duration of the constant speed partial trajectories $i$. For now, we assume these average speeds $v_i$ to be representative for the partial trajectory $i$ of an average vehicle in the flow (see later discussion).

The travel time of partial trajectory is defined as $T_i$ and the distance traveled as $X_i$. For a departure at time $t^*$, the total travel time and total distance traveled between $U$ and $D$ is

$$T_{UD}(t^*) = \sum_{i=1}^{n} T_i$$

$$\Delta x_{UD} = \sum_{i=1}^{n} X_i$$

Let us now look at one single partial trajectory (e.g., the first one). The distance $X_i$ traveled in one partial trajectory $i$ can be calculated in two ways. On the one hand, it is the distance traveled by the vehicle along the trajectory with constant speed $v_i$:

$$X_i = T_i \cdot v_i$$

On the other hand, as can be inferred from Figure 1, $X_i$ is also the distance covered by the characteristic with speed $w$ starting in $Y_i$ during the time $dt - T_i$:

$$X_i = w \cdot (dt - T_i)$$

Out of these two equations the travel time $T_i$ is

$$T_i = \frac{w \cdot dt}{w + v_i}$$

Finally, the total average speed across road section $UD$ with $n$ piecewise constant speed areas will be

$$\bar{v}(t^*) = \frac{\sum_{i=1}^{n} X_i}{\sum_{i=1}^{n} T_i} = \frac{\sum_{i=1}^{n} \frac{v_i}{w + v_i}}{\sum_{i=1}^{n} \frac{1}{w + v_i}}$$
Figure 1 Average speed $v(t^*)$ of a vehicle that starts moving from $U$ to $D$ at $t^*$ (assuming congested conditions in the link).

Note that this definition will never produce zero speeds and hence infinite segment travel times (as is often the case in some online algorithms with instantaneous sums rather than trajectories). Indeed, in Eq. 6, a $v_i$ equal to zero yields a zero term in the sum that will always be strictly positive; otherwise, the trajectory could never cover the distance between the detectors.

It is clear that the number $n$ of partial trajectories that have to be summed depends on the travel time and hence on the result of Eq. 6. The resulting formulation is thus implicit, making the solution algorithm relatively costly. An approximate explicit solution is to use the average speed $\bar{v}(t^* - dt)$ from a previous time step to define the number of necessary partial trajectories at the present time step. The errors made by making this simplification are small compared to the gain in computation time. The position of the first ($i = 1$) and last ($i = n$) measured speed to be used relative to the present time step is easily found by comparing Figure 1 with Table 1. Note that in Figure 1, arrival of the trajectory at detector $D$ happens to coincide with a time interval boundary, which causes the sums in Eqs. 1–6 to be exact in this special case. In general, the trajectory ends somewhere in between time interval marks, and the last term in the sums should be weighted by the fraction of the last time interval needed to exactly arrive at $D$ as Coifman (2002) suggests. In the validations described in this article that considered time increment $dt = 1$ min, this turned out to only marginally change the travel time estimation and was therefore neglected. However, when applying larger time increments (or shorter detector distances), these weights might have larger impact. In correspondence with this approximation, the continuous times of Table 1 need to be mapped onto discrete time intervals of the measurement data.

**Applicability to Non- or Partially Congested Segments**

So far, the derivation of the algorithm assumed congested conditions along the link. The question that arises now is: Will this simple formula (Eq. 6) with the explicitly defined summation ranges of Table 1 also apply when traffic is free flowing or when there is a shock wave front within the segment (i.e., only part of the segment between detectors is congested)? We discuss these two cases next, and illustrate the behavior of the resulting estimator in the next section.

First, let us look at free-flow traffic conditions. Here there are no characteristics moving upstream and theoretically the approach outlined so far does not make sense. Free-flow traffic states will all be moving forward, presumably at the maximum travel speed (in simplified KWT). The free-flow measured speeds $v_i$ are always more or less equal to the free speed $v_f$. It is not important what speeds or interval $[1,n]$ we will be looking at, or what average we are making; the result will always be $\bar{v} \approx v_f$, the maximum travel speed. Hence, although applying Eq. 6 is theoretically wrong, in practice it will hardly introduce errors, as validated by Coifman (2002). However, it keeps the algorithm very simple, as it does not need to distinguish between traffic states.

A second traffic state that requires closer examination is the transition between free flow and congestion with a shock wave present in the segment (see Figure 2). The shock wave speed is always larger than or equal to $-w$. The average trajectory speed according to Eq. 6 calculated with measurements from detector.
Figure 2  Average speed \( v(t^*) \) on the transition between free flow and congestion.

\( U \) will now no longer be approximately the same as the one from detector \( D \), as both will have a substantially different intervals with slow and fast speeds in the sum of Eq. 6. Figure 2 shows that both (dashed lines) are a bad approximation of the true average travel speed (bold full line). The ATSB algorithm assumes that the average speed at each detector is valid for half the distance between \( U \) and \( D \), yielding the dash-dotted line, which is clearly a better approximation in this case. Admittedly, this assumption is pragmatic and not founded on solid traffic flow theoretical ground. As a consequence, and as can be verified from the theoretical cases in the fourth section of this article, the deviation between estimated travel time and the (traffic flow) theoretical travel time value may become large in some pathological cases. Theoretically, algorithms that fully rely on traffic dynamics, like Kalman filters with traffic flow models embedded (Tampère & Immers, 2007), capture these conditions more accurately. Yet during empirical validation our pragmatic assumption appeared to cause no significant problems, so we see no reason to use substantially more complex algorithms, at least for the offline travel time estimation problem (see also fourth section of this article for a possible explanation why the seemingly poor theoretical assumption performs well in practice).

The assumption of up- and downstream trajectory speeds being valid for half of the segment is analytically equivalent to calculating average trajectory speed as the harmonic mean of the trajectory speeds from detectors \( U \) and \( D \) as computed using Eq. 6:

\[
\overline{v}(t^*) = 2 \cdot \left( \sum_{X \in \{U, D\}} \frac{1}{\bar{v}_X(t^*)} \right)^{-1}
\]

Note that we can use Eq. 7 also in the homogeneously congested or free-flow cases, since then \( \bar{v}_U \) and \( \bar{v}_D \) are in principle identical, so that applying Eq. 7 will hardly make a difference. Rather, it avoids unnecessary complexity of the algorithm since no traffic regimes need to be distinguished. ATSB thus uses Eqs. 6 and 7 irrespective of the traffic regime.

**Integration of Segment Travel Times to Path Travel Times**

In a final step, the method sums up travel times over successive road sections. A simple integration of the speeds into a trajectory is used. This is very similar to PCSB. The average speed and travel time for each road section is calculated according to Eqs. 6 and 7 at discrete time intervals \( dt \) and interpolated linearly between these time marks. Note that opposed to the PLSB method, which uses linear speed interpolation between detector stations, this linear interpolation is now made for travel time over the temporal axis, which hardly introduces errors. It is also possible to use a piecewise constant interpolation or some other interpolation method here, but this has negligible effect in the outcome if small discrete time intervals \( dt \) (typically 1 minute) are used.

After interpolation, a travel time is available for any conceivable arrival time at the upstream end of each section. The path travel times can thus be obtained by recursively summing up all section travel times evaluated at the time of arrival at the upstream detector of that section. Denoting with \( T_{D_i} \) the travel time for a trajectory starting upstream of the path at \( t_0 \) and ending at the downstream end of section \( i \) of length \( \Delta x_i \), and initializing \( T_{D_0} = 0 \), the following recursive equation applies:

\[
T_{D_i} = T_{D_{i-1}} + \frac{\Delta x_i}{v_i(t_0 + T_{D_{i-1}})}
\]

**Preprocessing of Local Speed Measurements**

So far, we have assumed average speeds \( v_i \) to be representative for the partial trajectory \( i \) of an average vehicle in the flow.
However, detector measurements are typically infected with numerous errors (Ki & Baik, 2006) that need to be corrected or discarded during preprocessing. We consider this no part of our method.

However, local detector data also exhibit structural biases introduced by the very nature of the measuring system itself. For one, it is well known that time mean speeds are a worse representative of the average travel speed of the vehicles than space mean speed (Daganzo, 1997,) for which correction methods exist (Soriguera & Robúste, 2011b; Yuan et al., 2010). Second, however, even space mean speeds tend to be biased toward higher values if traffic conditions are not spatially homogeneous in the neighborhood of the detector. By definition, local detectors are only able to measure the vehicles moving at one spot (or over the 3–6 m length of a typical detector). It means that a complete or near standstill in between 2 detector stations will never be registered. As a consequence, lower spatial speeds, where standstill by individual vehicles is more likely to occur, will inevitably be overestimated by local speed detectors.

In our method, we account for the latter bias by a simple linear correction of the lower speeds. A threshold \( v_{\text{bias}} \) is introduced, below which speeds are multiplied by a factor \( f(v) \). To avoid discontinuities, we set \( f(v_{\text{bias}}) = 1 \). As the speed measurement is lower, a stronger correction is applied:

\[
    f(v) = \begin{cases} 
    f_{\text{min}} \left( 1 - \frac{v}{v_{\text{bias}}} \right) + \frac{v}{v_{\text{bias}}} & v \leq v_{\text{bias}} \\
    1 & v > v_{\text{bias}} 
    \end{cases} \tag{9}
\]

In fact, as Eq. 8 corrects for a bias in the detection system, one could argue whether this correction, along with its two parameters \( v_{\text{bias}} \) and \( f_{\text{min}} \), belongs to the algorithm itself or to preprocessing. For this reason, we present in the remainder results with and without this correction separately.

**Algorithm Summary**

Next, the complete ATSB algorithm is described in pseudo code. The road sections between detectors are called segments. If a segment is only bounded by one detector, like at the beginning or ending of the route, only one detector is used for estimating the average speed.

**Algorithm 1: ATSB**

Step 0: Speed measurement preprocessing (Eq. 9)

Step 1: Fill speed matrix

FOR each timestep

FOR each segment

IF both segment boundaries have detectors

- calculate trajectory speed over segment using upstream detector (Eq. 6)

ELSE

- calculate trajectory speed over segment using available detector (Eq. 6)

average segment speed := interpolate average trajectory speed matrix at timestep + total travel time

total travel time := total travel time + segment length / average segment speed

END IF

END FOR

END FOR

Step 2: Calculate trajectories

TO THEORETICAL ANALYSIS OF ESTIMATION BEHAVIOR

This section presents some detail on the calculation results of the ATSB algorithm outlined in the previous section. In homogeneous and stationary conditions (either congested or free flowing), travel time estimation is rather straightforward and no problems are expected. Hence, the analysis here focuses on transient conditions. We analyze the type of transients where a shockwave traverses the segment between the up- and downstream detector, as considered before in Figure 2. Also, inverse transients (shockwaves traveling downstream) or stationary waves in between detectors are discussed.

In transient conditions, estimation of travel time within the segment \( UD \) substantially differs dependent on whether Eq. 6 is applied to the up- or downstream detector. As an extension to Coifman (2002), Eq. 7 proposes a harmonic average of these two estimates. Moreover, Eq. 6 is an explicit rather than implicit scheme according to the summation ranges of Table 1. Finally, we map the continuous times of Table 1 onto discrete time intervals of the measurement data. It is now investigated how these aspects affect estimation in transient conditions. It is then important to define unambiguously which travel time \( \hat{v} (t^* - dt) \) is used in Table 1 to determine the summation ranges. After all, as detected speeds over the up- and downstream detectors significantly differ in transient conditions, so will the average trajectory speeds calculated from them. So should Table 1 then be evaluated using \( \hat{v}_X (t^* - dt) \) (where \( X \) is \( A \) or \( B \), respectively) or \( \hat{v} (t^* - dt) \), the harmonic combination?
For each of these aspects, we now illustrate its impact by selecting a “worst case” situation where the approximation can be expected to have a relatively large impact. Let us start with the latter issue: using separate speeds for up- and downstream estimates, or the harmonic combination to evaluate Table 1. We analyze a scenario where a slowly moving shockwave travels between the up- and downstream detector, as was depicted in Figure 2. Such a situation can be worst case in that the shockwave travels at substantially slower speed than the speed — _w_ — that is assumed for all perturbations in the flow.

Figure 3 shows the impact of using _v_ (t∗ − _dt_) (left graph) instead of separate trajectory speeds _v_ (t∗ − _dt_ ) for the up- and downstream detector (right graph). A “worst case” scenario was chosen with a low speed in the congested region after the shock wave has passed,1 as this tends to maximize the difference in trajectory speeds on both detectors and hence the harmonic average will maximally differ from each of them. The figure shows the exact solution _tt_ ref in the long-dashed line, the trajectory speeds according to Eq. 6 based on up- and downstream detectors (respectively, _tt_ U in short-dashed line and _tt_ D in dotted line), and their harmonic average _tt_ avg according to Eq. 7. With separate summation ranges, the downstream detector tends to consider longer summation intervals, showing a rapidly increasing influence of the slow speeds detected earlier on this detector. For the upstream detector the inverse applies. With the sum based on the averaged speed, an average range is also considered, so that the downstream travel time estimation rises later and the upstream estimation rises earlier. The gross effect is relatively small and remains close to the exact solution for both cases. As we were considering a worst-case scenario, this may be expected to hold equally for less pessimistic cases. The conclusion is that considering two additional auxiliary variables _v_ (t∗ − _dt_ ) would not significantly increase estimation accuracy; hence this additional complexity is omitted from the ATSB method, which works with _v_ (t∗ − _dt_ ) for evaluating Table 1.

A second simplification that needs to be verified is the use of explicit rather than implicit travel time definition. More specifically, one could opt to evaluate Table 1 using _v_ (t∗) for an implicit formulation or _v_ (t∗ − _dt_ ) for an explicit formulation. In the former case, an iterative procedure is needed to compute Eqs. 6 and 7, as the summation ranges in Table 1 depend on the result of Eq. 7, which can only be evaluated once Eq. 6 has been computed with correct values from Table 1. As a worst-case scenario, a rapidly changing travel time is considered. Indeed, in such case, the previously calculated average trajectory speed will be a relatively bad approximation compared to the one currently calculated.

Figure 4 shows the difference between the two approaches. Indeed, the explicit formulation (in the left part of the figure) is somewhat later in following the rising trend in travel time than with implicitly defined ranges (right part). Despite the worst case character, the error remains relatively small and the increasing trend is rapidly restored. Comparison with Figure 3 (left) reveals that in less extreme theoretical cases (slower varying travel time), the delay with the explicit formulation is indeed even less problematic. Also in the empirical case study presented in the next section, we have experimented with implicit versus explicit formulations and found no significant difference in the results (not shown in this paper). Even though there are faster shock waves in the empirical data set, the low speeds are rarely as low as in the theoretical cases of Figures 3 and 4, which also limits the change rate of the travel time, and herewith also the delay related to explicit formulation. We conclude that the simpler explicit scheme is sufficient for any reasonable applications of the ATSB method.

A third approximation to be verified is the discrete mapping of continuous Table 1 values onto time marks _dt_ of the time-aggregate speed measurements rather than using an exact weight for the last term in the sum. A worst-case scenario is found as follows. The (nontruncated) number of intervals _dt_ that needs to be considered according to Table 1 equals _n_ = _ΔvU_ / _v_ (t∗) _Δt_ (1/2 + _x_).

The truncation error (rounding up) is maximized if _n_ is small and only a fraction larger than the first smaller integer (e.g., _n_ = 1.01 being truncated to 2 yields a 99% error in _n_). For a given _dt_ , _n_ is minimized with smaller segment length. Whether _n_ just exceeds the nearest integer depends on the value of the average speed. For minimal _n_ , we require _v_ to be large and hence select a segment length accordingly in Figure 5.

Figure 5 shows that for the higher speeds (lower travel times, hence smaller _n_ ) there is indeed a small error due to truncation (downstream and average travel time rising later). As speeds drop and _n_ increases, the influence of the truncation soon vanishes. Compared to other sources of error and also in absolute terms, truncation has a negligible influence and hence ATSB does not need to apply weights in sum (Eq. 6).

The scenarios considered so far all involved an upstream moving shockwave. Now let us consider behavior of the algorithm with stationary and downstream moving shockwaves. Stationary waves are the simpler cases. Here, one detector is covered by a congested state, whereas the other detects free flow conditions. Obviously, ATSB will estimate travel times as if the stationary wave is positioned exactly halfway in between detectors, irrespective of the real position. Hence, errors are maximized when the real wave is close to either detector, for instance, an incident just upstream of the downstream detector (causing underestimation of travel time) or just downstream of the upstream detector (overestimation). The potential error hence becomes larger as detector spacing increases. Obviously, estimating the correct position of the stationary wave could be done to some extent by applying incident detection techniques based on shock-wave propagation during the onset of the incident-related congestion. However, this

1Note that in these scenarios, we select traffic conditions up- and downstream of the shock wave as well as the shock wave speed independently from one another, whereas according to kinematic wave theory, they should be related. However, this allows us to construct more freely “worst cases” to illustrate the algorithm’s approximations.
would introduce substantial complexity in the algorithm that in our judgment is not justified for a limited error in quite rare conditions.

By far the most pathological case for ATSB algorithm is the downstream moving shock wave, which occurs when congestion dissolves thanks to a reduction of the demand upstream (e.g., end of rush hour). This causes the upstream detector to detect free flow speeds first after an episode in which both detectors were covered by congested traffic, which clearly contradicts the assumptions visualized in Figures 1 and 2. As travel times in congestion can be long, so is the summation interval for the upstream detector according to Figure 1 or Table 1. Although initially the entire interval still contains congested speeds, at some point, the last speeds in the summation will be free-flow speeds, even though the trajectory in reality travels the entire segment between detectors in congested speeds. Hence, estimated travel times decrease toward free-flow values way before the real travel times do, and this offset is in principle unbounded as it is proportional to the congested travel time that can attain any (large) value. Figure 6 provides evidence of such a pathological case. Although the shock enters the segment only at time \( t = 0 \), the travel time estimate drops already 23 minutes earlier. Once the shock actually reaches the segment, estimation is soon restored to acceptable values.

The question now arises of whether this clearly undesired behavior actually jeopardizes the applicability of the ATSB algorithm. It appears that in practice, the theoretical case of Figure 6 hardly occurs. Actually, we discovered the pathological behavior with downstream moving shocks only during the theoretical analyses in preparation for this article and never observed it in empirical work, neither in the case study presented hereafter, nor in any other practical application of ATSB in our laboratory or in the Flemish traffic center that applies the algorithm in practical \emph{ex post} studies regularly since 2005.

The reason is that first-order (i.e., homogeneous) congestion patterns with forward-moving upstream shocks hardly occur in practice; forward shocks do occur frequently, but then the congestion breaks down in stop-and-go waves. Homogeneous congested traffic only occurs for very strong bottlenecks causing congestion near-maximum density and zero speed (Schoenhof & Helbing, 2004; Treiber, Hennecke, & Helbing, 2000). This
means flow is almost zero and hence shock wave speed can only be positive for even lower inflow, which is not very likely. Instead, such situations usually dissolve from downstream when the (usually incidental) cause of the queue was removed. In all other cases, congestion is at least oscillatory or consists of stop-and-go-waves, which corresponds much better to the assumptions made in ATSB that speed fluctuations travel with speed \( w \) upstream. Figure 7 illustrates various empirical congestion patterns and the oscillatory character in light congestion. Even heavier incidental congestion on this site exhibits oscillations, when dissolved from either up- or downstream. Note that the data in this figure have been processed using an adaptive smoothing filter (Treiber & Helbing, 2002) that makes similar assumptions on propagation of perturbations like ATSB and therefore might bias the observations. However, although this indeed causes some artifacts (e.g., stair-type deformation of shock waves, waves occurring or disappearing halfway between detector locations), the nature of the oscillatory patterns and its contours is undeniably revealed, which is the main purpose of the figure. We conclude from this analysis that for traffic patterns occurring in practice, the ATSB method performs well, even though there may be theoretical first-order cases (forward moving shock waves) where ATSB would make large errors.

Overall, the conclusion is that ATSB, despite many simplifications, performs well in all relevant traffic conditions. The estimation accuracy may be improved in some specific cases like stationary waves and forward moving shocks; however, we prefer a simple, parsimonious algorithm. In the case study presented in the next section, this simpler algorithm does indeed prove to be sufficient for practical application and to outperform available alternative algorithms.

**EXPERIMENTAL SETUP AND CALIBRATION**

In the following section, the PCSB, PLSB, FISB and ATSB methods are compared on a data set supplied by the Flemish traffic center. The test bed is the highly congested E313 highway between Geel-Oost and Antwerpen-Oost in Belgium and covers a distance of 41.87 km. Along the motorway, 48 double-loop (DL) detectors are situated with an average spacing of 870 m. To simulate the fact that not all highways have a similar high density of detectors, a second run of the methods is made with only 15 available detectors on the same route. This leads to an average spacing between the detectors of 2920 m. Both runs are compared to test the robustness of the different methods with respect to detector density. The data consists of 1-minute aggregated harmonically averaged speeds over a period of 2 months (September and October 2010). No details were available on the preprocessing method that was applied to detect outliers and complete missing data records (excluding correction of Eq. 8). However, the same preprocessed data were used for all analyses hereafter, so one may assume all algorithms to be equally affected (positively or negatively) by potential errors in the preprocessing.

The ground-truth travel times used for the validation of the methods are direct measurements of the travel time collected from individual vehicles by means of highly reliable automatic

![Figure 5](image)

**Figure 5** Influence of truncation of the summation range. Left: ATSB method with truncated summation range; right: variant using a weighted fraction for the last term in the summation range (\( L_{AB} = 520 \) m; \( v_{fast} = 30 \) m/s; \( v_{slow} = 1 \) m/s; \( c_{shock} = -0.5 \) m/s).

![Figure 6](image)

**Figure 6** Pathological case of a downstream moving shock wave (\( L_{AB} = 2000 \) m; \( v_{fast} = 30 \) m/s; \( v_{slow} = 2 \) m/s; \( c_{shock} = 2 \) m/s).
license plate recognition (ALPR). All the individual records are ordered according to entering time and next aggregated into separated one minute time intervals.

Out of the 8-week data set, a random sample of 20 days was taken for calibration; the rest was used for the validation discussed in the next section. Calibration of the various reference methods was performed by discretizing all parameters in their relevant range and simply selecting the values yielding lowest overall error on the calibration data set (see next section for error measures). This included the semipreprocessing parameters $v_{bias}$ and $f_{min}$, which thus potentially adopted different parameter values depending on the method. It appeared however that optimal values were equal for PCSB and ATSB methods at 40 km/h and 0.5, respectively. PLSB required stronger corrections; however, here we suspect that the joint optimization of parameters tends to compensate for a structural bias (underestimation) in the calibration data set and doesn’t need the additional correction of $w$.

**VALIDATION RESULT**

As means of comparison, the root mean squared error (RMSE) and the mean average percentage error (MAPE) of the estimates $\hat{y}_i$ with respect to the reference values $y_i$ are used:

$$ \text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2} (10) $$

$$ \text{MAPE} = \frac{100}{N} \sum_{i=1}^{N} \frac{|y_i - \hat{y}_i|}{y_i} (11) $$

If these statistics were applied straight away, the result would be biased toward traffic situations that occur more often. Most of the time, vehicles on the observed highway will move in free flow. For such traffic states, travel time estimation is easy, as the average travel time is just related to the length of the highway and the maximum speed limit. In congested traffic states, on the other hand, travel times are harder to estimate correctly. To overcome the positive bias by the vast amount of free-flow data, the travel time estimates were stratified by first categorizing them into bins according to the corresponding ground truth travel time (as an indicator for traffic state). Here we use 10-minute intervals to group the data, yielding eight bins for this data set. After calculating RMSE and MAPE for each bin the mean is taken over all bins. In Table 2 the distribution of observations over bins is shown, together with error statistics for the PCSB method. From the last column it is clear that in the second bin most observations are made. For this bin also the RMSE and MAPE have the lowest value. This clearly illustrates the positive bias that would have been obtained without stratification.
In Table 3, the results for the different methods on a network with a high detector density are presented. The numbers in this table and the next table are stratified averages, like the last row in Table 2. The average spacing between the detectors is 870 m. PCSB and ATSB results are shown without and with preprocessing of the speeds to correct for the structural bias of the DL detectors. The last rows show the sensitivity of ATSB method for the exact value of the characteristic wave speed \( w \), where values \( \pm 5 \) km/u are tested with respect to the calibrated value 19.5 km/u.

In Table 4, the performance of the various algorithms is shown for low detector density. The average detector spacing here is 2920 m. In Figures 8 to 10, an illustration is made of the different travel time algorithms for a typical morning peak. The resulting travel times in a network with high detector density and low detector density are shown, along with the reference measurements.

**DISCUSSION**

Figures 8 and 9 clearly confirm that algorithms based on simple interpolation rather than traffic dynamics exhibit unrealistic results, especially for large detector spacing (cf. Soriguera & Robusté, 2011). Another typical problem with these algorithms is visible. The PCSB and PLSB methods behavior in congestions is characterized by a sawtooth profile. This is the result of trajectories bundling as a consequence of the discrete speed field. In this case, trajectories for the PCSB and PLSB are calculated with equidistant departure times (every minute a new trajectory starts upstream running all the way downstream). It is easiest to understand what happens by considering PCSB, which integrates trajectories through a piecewise constant speed field. Now consider a brief time interval (say 1 minute) where the speed in a segment equals zero and attains some positive value after that. Consider also two trajectories reaching this segment at the very beginning and end of this interval. The first trajectory doesn’t make any progress during 1 minute; the other is not delayed, as it arrives just when speeds turn positive again. Hence, whereas they arrived with 1 minute headway, they overlap from now on and reach the final segment of the route simultaneously. Similar to this extreme case of standstill, every low speed causes the headway between trajectories to decrease, so that at arrival, the trajectory curves are no longer equidistant. Every tooth represents a bundle of trajectories arriving at short headways, even though they departed with 1-minute intervals. Because of their different discretization and integration mechanism, ATSB and FISB don’t experience these problems. A possible solution for PCSB and PLSB is to calculate the trajectories starting from the middle of the path or at the end and averaging the results. Other solutions exist in smoothing the data more (like the ATSB and FISB methods).

Despite the relatively simple way in which traffic dynamics are taken into account (and its corresponding theoretical limitations as discussed in the fourth section), ATSB exhibits very small estimation errors, especially when speeds are adequately preprocessed. The output signal is smooth (no sawtooth or other noise) because of the average speed formulation (see Figure 10), meaning that the outputs can readily be used; without further postprocessing (e.g., smoothing for compensation of the saw-tooth effect). A second and very important property of the methods based on traffic flow theoretical principles is observed when we compare Table 3 with Table 4. It is obvious that the ATSB method (just like the FISB method) hardly degrades with increased detector spacing, whereas simpler interpolation methods like PCSB that performed reasonable with high detector density fail with longer segments. We conclude from these empirical results that methods exploiting simple traffic flow theoretical principles (both the ATSB and FISB method) are statistically better than the other methods (PCSB and PLSB) in all practical traffic conditions.

Our validation results show similar performance of ATSB and FISB. However, the ATSB algorithm is far more

---

**Table 2** Evaluation of every bin for ATSB with preprocessing.

<table>
<thead>
<tr>
<th>BIN (min)</th>
<th>RMSE (min)</th>
<th>MAPE (%)</th>
<th>#OBS</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;20</td>
<td>2.734</td>
<td>13.032</td>
<td>2664</td>
</tr>
<tr>
<td>20–30</td>
<td>1.890</td>
<td>5.780</td>
<td>23250</td>
</tr>
<tr>
<td>30–40</td>
<td>2.908</td>
<td>6.171</td>
<td>1564</td>
</tr>
<tr>
<td>40–50</td>
<td>2.977</td>
<td>5.136</td>
<td>1272</td>
</tr>
<tr>
<td>50–60</td>
<td>3.525</td>
<td>4.780</td>
<td>1123</td>
</tr>
<tr>
<td>60–70</td>
<td>4.601</td>
<td>5.512</td>
<td>770</td>
</tr>
<tr>
<td>70–80</td>
<td>3.477</td>
<td>3.388</td>
<td>547</td>
</tr>
<tr>
<td>&gt;80</td>
<td>4.132</td>
<td>3.511</td>
<td>140</td>
</tr>
<tr>
<td>Average</td>
<td>3.280</td>
<td>5.914</td>
<td></td>
</tr>
</tbody>
</table>

**Table 3** Comparison of different methods (high detector density).

<table>
<thead>
<tr>
<th>Method</th>
<th>RMSE (min)</th>
<th>MAPE (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCSB</td>
<td>5.365</td>
<td>8.852</td>
</tr>
<tr>
<td>PCSB with preprocessing</td>
<td>4.237</td>
<td>7.320</td>
</tr>
<tr>
<td>PLSB</td>
<td>7.869</td>
<td>12.914</td>
</tr>
<tr>
<td>FISB</td>
<td>3.034</td>
<td>6.067</td>
</tr>
<tr>
<td>ATSB</td>
<td>5.398</td>
<td>9.027</td>
</tr>
<tr>
<td>ATSB with preprocessing</td>
<td>3.280</td>
<td>5.914</td>
</tr>
<tr>
<td>ATSB with preprocessing: ( w = 15 ) km/u</td>
<td>3.709</td>
<td>5.871</td>
</tr>
<tr>
<td>ATSB with preprocessing: ( w = 25 ) km/u</td>
<td>3.721</td>
<td>6.397</td>
</tr>
</tbody>
</table>

**Table 4** Comparison of different methods (low detector density).

<table>
<thead>
<tr>
<th>Method</th>
<th>RMSE (min)</th>
<th>MAPE (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCSB</td>
<td>8.973</td>
<td>13.998</td>
</tr>
<tr>
<td>PCSB with preprocessing</td>
<td>7.272</td>
<td>11.261</td>
</tr>
<tr>
<td>PLSB</td>
<td>11.393</td>
<td>18.156</td>
</tr>
<tr>
<td>FISB</td>
<td>3.197</td>
<td>6.124</td>
</tr>
<tr>
<td>ATSB</td>
<td>5.470</td>
<td>9.233</td>
</tr>
<tr>
<td>ATSB with preprocessing</td>
<td>3.560</td>
<td>6.255</td>
</tr>
</tbody>
</table>
Figure 8  Travel time PCSB: left, without preprocessing; right, with preprocessing.

Figure 9  Left: travel time PLSB; right: travel time FISB.

Figure 10  Travel time ATSB: left, without preprocessing, right, with preprocessing.
parsimonious with only one parameter (with a well-known physical meaning and to which the results are little sensitive) or three if one includes the preprocessing parameters, versus eight for FISB (some of which have no physical meaning; moreover, the required level of preprocessing depends on the filter parameters). As a consequence, ATSB is easier to calibrate. Moreover, ATSB is less computationally demanding since it does not require calculating speeds on a grid of intermediate locations between the detectors.

CONCLUSIONS

This article presents the ATSB algorithm for calculating offline travel times from local speed measurements. The algorithm is based on simple traffic flow principles. Travel time estimation is smooth and accurate, even when the detector density is low (tested up to 3-km segments). As such, it overcomes well-known limitations of methods based on simpler speed interpolation that are widely applied in practice. Calibration is limited to one parameter that is easy to interpret, as it is the characteristic speed of perturbation waves in congestion. In comparison to alternative methods based on traffic flow theoretical principles that have far more parameters, the performance is similar while the computation and calibration effort is smaller. The method presented is currently being used by the Flemish Traffic Center to maintain a database of historical travel times on the entire highway network of Flanders (∼1000 km). Applications of such historical data include calibration of traffic models, policy evaluation and monitoring, and as a basis for online methods in the context of ATIS.

It feels natural to look for a way to apply the presented methodology in an online setting. However, this is far from trivial, as the offline version uses for estimation of the travel time for a vehicle departing at $t$ mostly data recorded after $t$, which is impossible in real time. On the other hand, the method can be applied (to the downstream detector only) to reconstruct in real time a trajectory that has just arrived downstream at $t$ since for that only data recorded prior to $t$ would be needed. Although this yields a relatively smooth and accurate estimation for the travel time just realized, this is still is only a lagged estimate and hence not representative for trajectories that currently leave from the upstream detector. In preliminary experiments we attempted to use simple time-series methods with linear extrapolation to compensate for this lag, but so far unsuccessfully. Still it remains a relevant topic for further research on whether more advanced prediction methods than the simple linear one tested so far could overcome these problems and produce an applicable online ATSB variant.

ACKNOWLEDGMENTS

The authors thank the Flemish Traffic Center for the supply of the different data sets and three anonymous reviewers for valuable comments that helped substantially improving the article.

REFERENCES


