Title
Vehicle Reidentification and Travel Time Measurement Using Loop Detector Speed Traps

Author
Coifman, Benjamin Andre

Publication Date
1998-12-01

Institute of Transportation Studies
University of California at Berkeley

Vehicle Reidentification and Travel Time Measurement Using Loop Detector Speed Traps

Benjamin André Coifman

DISSERTATION SERIES
UCB-ITS-DS-98-2

July 1999
ISSN 0192 4109

Vehicle Reidentification and Travel Time Measurement Using Loop Detector Speed Traps

by

Benjamin André Coifman

B.E.E. (University of Minnesota) 1992
M.S. (University of California, Berkeley) 1995

A Dissertation submitted in partial satisfaction of the requirements for the degree of

Doctor of Philosophy in

ENGINEERING:

Civil Engineering

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Michael Cassidy, Chair
Professor Carlos Daganzo
Professor Gordon Newell
Professor Pravin Varaiya

Fall 1998

This dissertation of Benjamin André Coifman is approved:

Chair Date

Date

Date

Date

University of California, Berkeley

Fall 1998

Vehicle Reidentification and Travel Time Measurement Using Loop Detector Speed Traps

by

Benjamin André Coifman

Abstract

Vehicle Reidentification and Travel Time Measurement Using Loop Detector Speed Traps

by

Benjamin André Coifman

Doctor of Philosophy in

ENGINEERING: Civil Engineering

University of California, Berkeley

Professor Michael Cassidy, Chair

This dissertation presents a vehicle reidentification algorithm for consecutive detector stations on a freeway, whereby a vehicle measurement made at a downstream detector station is matched with the vehicle's corresponding measurement at an upstream station. The algorithm should improve freeway surveillance by measuring the actual vehicle travel times; these are simply the differences in the times that each (matched) vehicle arrives to the upstream and downstream stations. Thus, it will be possible to quantify conditions between widely spaced detector stations rather than assuming that the local conditions measured at a single station are representative of an extended link between stations.

The method is developed using vehicle lengths measured at dual loop speed traps. These detectors are quite common, often placed at half mile spacings or less on urban freeways. The proposed approach is a milestone in highway research because no previous work uses the existing detector infrastructure to match vehicle measurements between detector stations. The work is also transferable to other detector technologies

capable of extracting a reproducible vehicle measurement, i.e., a vehicle signature, such as video image processing.

The contribution to the field of traffic surveillance should prove to be significant since the vehicle reidentification algorithms will allow the study of travel time applications (e.g., incident detection and dynamic trip assignment) without deploying an expensive detection system. This will enable cost-benefit analysis before investing in a new detection system. If travel time measurement proves to be beneficial, the system could be deployed using speed traps, or the algorithms could be transferred to emerging detector technologies with better measurement resolution. The methodology should prove beneficial for research purposes as well, by yielding better insight into traffic dynamics between widely spaced detector stations.

Professor Michael Cassidy

Committee Chairman

Contents

List of Figures

List of Tables

Acknowledgements

First I would like to thank my advisor, Professor Michael Cassidy, for supervising this work and for the input, guidance and support he has given me over the years. Professor Carlos Daganzo, Professor Gordon Newell and Professor Pravin Varaiya have also helped me shape this work through course work, personal interaction and their participation on my committee. It has been an honor working with each of them.

I have been very fortunate to have the privilege to attend the University of California at Berkeley and work within three strong departments (Civil Engineering, Electrical Engineering, and Computer Science). I wish to extend my gratitude to Professor Martin Wachs, Samer Madanat and Peter Bickel for their input during my qualifying exams. I would also like to take this opportunity to thank Professor William Garrison for sharing his unique insights on transportation past, present and future (sometimes, common sense is uncommon). Of course, I would not have made it this far if it were not for the solid undergraduate education I received at the University of Minnesota. I would like to thank my undergraduate advisors, Professor Dennis Polla and Professor Ted Higman, for starting me on the path to becoming a researcher.

I would like to acknowledge the PATH program for supporting my research. I would also like to thank Caltrans and the motoring public in the state of California for providing traffic data for this research. I want to specifically thank Sean Coughlin, Joe Palen and Brian Simi for going above and beyond their normal duties with Caltrans. Their friendship, assistance and feedback have been invaluable.

Of course I could not have finished this dissertation without the support of my friends and family. In particular, I would like to mention the camaraderie of my fellow transportation engineering students (see you at TRB). Finally, I would like to thank the Minnesota Transportation Museum for helping me realize my love of transportation which eventually caused me to deviate from my original plans to pursue a PhD in Electrical Engineering.

1. Introduction

This dissertation presents a vehicle reidentification algorithm for consecutive detector stations on a freeway, whereby a vehicle measurement made at a downstream detector station is matched with the vehicle's corresponding measurement at an upstream station. The algorithm should improve freeway surveillance by measuring the actual vehicle travel times; these are simply the differences in the times that each (matched) vehicle arrives to the upstream and downstream stations. Thus, it will be possible to quantify conditions between widely spaced detector stations rather than assuming that the local conditions measured at a single station are representative of an extended link between stations.

The method is developed using effective vehicle length[1] measured at dual loop speed traps. These detectors are quite common, often placed at half mile spacings or less on urban freeways. The proposed approach is a milestone in highway research because no previous work uses the existing detector infrastructure to match vehicle measurements between detector stations. The work is also transferable to other detector technologies capable of extracting a reproducible vehicle measurement, i.e., a vehicle signature, such as video image processing.

Because the proposed algorithm was developed with conventional loop detectors in mind, it uses the (effective) length measurements to distinguish vehicles. Notably, a length measurement may be accurate to only 2 feet due to resolution limitations, making difficult the task of matching pair-wise measurements at upstream and downstream detector stations. However, if the difference between two measurements exceed this measurement resolution, then the pair of measurements probably did not come from the same vehicle. After applying this resolution test to each pair of upstream and downstream measurements (for some specified group of vehicles), the remaining pair-

[1] The effective vehicle length is the length as "seen" by the detectors; i.e., the physical vehicle length and the length of the detection zone.

wise comparisons that can not be eliminated are considered *possible matches*. For example, the upstream and downstream length measurements from the same vehicle should pass the resolution test and the pair will be labeled a *possible match*. Frequently however, one vehicle's measurement downstream will be a *possible match* to a different vehicle's measurement upstream because this pair of measurements likewise passes the resolution test. Clearly, these possible, but incorrect, matches are false positives.

Toward eliminating these false positives, the algorithm uses a simple trick: it matches platoons whenever the vehicles pass both detectors in the same relative order. The sequence of measured lengths in a platoon provides more information than do the individual measurements. For each vehicle in the platoon, the resolution test applied to the correct (but unknown) pair of upstream/downstream measurements should yield a *possible match* and the entire platoon should produce a contiguous sequence of *possible matches* in the space of pair-wise length comparisons. The problem is complicated in that the false positives can form spurious sequences of *possible matches* in the pair-wise comparison space. However, as a sequence of *possible matches* increases in number, the probability that it is due to false positives decreases. The vehicle reidentification problem becomes a matter of searching the pair-wise comparison space for contiguous sequences of *possible matches* that are long enough so that they are probably not caused by false positives.

The research presented in this document has investigated three different approaches to searching the pair-wise comparison space. The results suggest that it is possible to extract a sufficient number of platoons for traffic surveillance applications, while accepting few, if any, false positives.

1.1 Overview

Before addressing the vehicle reidentification algorithms, the motivation for this work is presented in chapter 2 and other surveillance methods that are relevant to travel time measurement are described in chapter 3. Chapter 4 presents the vehicle reidentification algorithms in detail using a pilot study to illustrate the steps. Chapter 5 examines three large examples to quantify the algorithms' performance. Chapter 6 discusses extensions and future research projects based on this work. Following the conclusions in 7 and a list of references, there are three appendices that explain in a step-by-step fashion how to implement the algorithms.

2. Motivation

This work could facilitate travel time measurement using existing detector infrastructure on freeways and would require minimal communications compared to other vehicle reidentification systems. Although the benefits of travel time measurement may be inflated in some of the literature, it still is a promising surveillance tool for traffic engineers. Travel time data could improve existing surveillance applications such as incident detection, control at ramp meters, and traveler information via existing technologies (e.g., changeable message signs and highway advisory radio). The travel time data could also serve as input to emerging technologies such as dynamic traffic assignment (DTA). More importantly, the data could be used to quantify the benefits from these emerging technologies using real traffic data, off-line, before making significant infrastructure investments. Such analysis will allow for quantifying the necessary level of accuracy for a given application. As accuracy increases, the marginal costs for further improvements will likely increase. Thus, a municipality can deploy the least expensive detection system that meets the these specified requirements.

Finally, there are applications which might benefit from the vehicle reidentification or travel time data, although on their own, probably do not justify the deployment of a vehicle reidentification system. For example, the travel time data could be useful for planning applications and the reidentification algorithms could be used to study individual driver dynamics over time and space. The remainder of this section will examine four applications: incident detection, DTA, delay measurement for planning purposes, and for studying driver dynamics.

2.1 Incident detection

A recent report from Caltrans [1] stated that, “Incidents are, by definition, perturbations in the normal operating characteristics of a transportation system, chief of which is travel

time." The potential benefits of incident detection have been known for years [2-6]. Faster response to an incident can reduce the number of drivers affected and reduce the average delay for those who are affected. By reducing total delay, other costs associated with the incident, such as wasted fuel and increased emissions, will also decrease.

Countless automated incident detection strategies have been proposed, but most of these systems suffer from high false alarm rates and/or long detection times. A reliable incident detection system using speed traps has been demonstrated by Lin and Daganzo [7]. The system uses two widely spaced detector stations to detect two "signals" that propagate through the traffic stream. The two signals, a backward moving *shock wave* and a forward moving *drop in flow*, are indicative of an incident between the stations. As noted in [7], "Detection of an incident can happen only when both signals have been received...." Although the *drop in flow* travels at the prevailing traffic speed, this earlier work estimated the *shock wave* speed to be on the order of 8 mph.

Fortunately, the *drop in flow* reflects the fact that vehicles are being delayed behind the incident. All vehicles that arrive at the station after the drop should experience increased travel times over the segment. Thus, an incident detection system based on travel time may not have to wait for the slow moving *shock wave* to reach the upstream station before detecting the incident.

When using travel time to detect incidents, it is necessary to localize the source of delay. It could either be caused by an incident within the link[2], or by queues backing up from some event downstream of the link. In the former case, the downstream detector station will be downstream of the bottleneck and should observe free flow vehicle velocities; while in the latter case, the downstream detector station will be observing congested traffic with lower vehicle velocities.

[2] Assuming that there are no recurring bottlenecks within the link

2.2 Dynamic Trip Assignment

Many researchers are investigating DTA as a means to reduce traveler delay. As proposed, a DTA system would observe current [8-16] and historical traffic conditions [8-15, 17-18], estimate travel times over the network and then route vehicles with the goal of reducing traveler delay.

Typically, the travel time forecasts are based on traditional traffic parameters (such as flow, velocity, and occupancy) measured at discrete point detectors [8-16]. Usually, the point measurements are averaged over fixed time periods (20 seconds-15 minutes) to smooth out transients and then generalized to a link of significant length (0.5-5 miles long). However, the fixed time periods generally do not correspond to a single steady state condition. Instead, a sample may include multiple traffic states and the fixed time average may not reflect any conditions that actually occurred at the detector [19].

Unfortunately, there is not a one-to-one relationship between travel time over an extended link and traffic parameters measured at a discrete point within that link. The DTA literature does not appear to consider the option of measuring travel time directly, but the use of direct travel time measurements should improve the performance of a travel time forecasting algorithm both through real time data, and by providing a set of historical data.

Although the promoters of DTA systems forecast significant benefits, the systems have only been tested in simulation or in very limited field studies [20]. The proposed travel time measurement system could be used for much-needed evaluation under real-world conditions.

2.3 Planning applications

If a travel time measurement system is deployed for ATIS (Advanced Traveler Information Systems) applications or incident detection, the system could prove

beneficial to planning applications as well. Three such applications are considered below, quantifying congestion, model validation and calibration, and tracking freight movements.

2.3.1 Quantifying congestion

Congestion and the associated costs from delay, wasted fuel and increased pollution, have become significant problems for transportation users and non-users alike. Tracking congestion trends can help planners assess how fast problems are growing. The trends can also be used to quantify the benefits of congestion countermeasures.

The state of the practice for quantifying delay and congestion on the metropolitan area level is to use average daily "volume/capacity" measured at discrete points to estimate delay over extended links [21]. As noted in subsection 2.2, there are many problems with using point measurements to estimate travel time or in this case, delay. It would be better to measure delay directly, i.e.,

(actual travel time) - (travel time at posted speed limit)

2.3.2 Model validation and calibration

Model validation and calibration is an important task for the traditional four step planning process as well as the on-going Travel Model Improvement Program which seeks to replace this process with microsimulation models. For example, the TRANSIMS designers at Los Alamos National Labs note that "The most important result of a transportation microsimulation in [the planning] context should be the delays..." [22]. It will be important to verify and calibrate these models to real networks, a task that is well suited to the travel time measurement system.

2.3.3 Tracking freight movements

Finally, because the vehicle reidentification method works particularly well with trucks, it should allow for generating origin-destination (O/D) data on freight movements, and thus, track these movements through the urban freeway network. This point is significant since researchers estimate that freight movement accounts for nearly 1/2 of all transportation costs, but these movements are virtually excluded from the Urban Transportation Planning Process [23]. Because trucks are a primary factor for pavement degradation, the O/D data on freight movements should prove to be significant when forecasting future pavement needs.

2.4 Driver dynamics

Using traditional surveillance methods, it is difficult to examine individual driver dynamics over extended distances. Usually, driver dynamics studies rely on aggregate traffic parameters at multiple sites or restrict the scope to a small number of drivers to overcome the difficulties associated with following vehicles over large distances. The proposed vehicle reidentification system could be used to match observations from the same driver at multiple sites along an extended highway segment. Thus, it will be possible to study behavioral trends over time and space by examining the driver parameters (e.g., headway and velocity) at multiple locations.

3. Other Surveillance Methods

This chapter discusses preceding research related to vehicle reidentification or travel time measurement systems. First, complementary detector technologies are presented in section 3.1, then competing vehicle reidentification systems are presented in section 3.2.

3.1 Complementary technologies

Although this dissertation focuses on measured vehicle lengths from speed traps, the proposed reidentification algorithms could be applied to other signature based detector systems. There are four emerging detector systems under Caltrans sponsorship that promise to yield more robust vehicle signatures while being compatible with the reidentification algorithm:

1. Magnetic Vehicle Signatures from Loop Detectors: Stephen Ritchie, University of California, Irvine [24].
2. Vehicle Dimensions and Velocity From Scanning Laser Radar: Harry Cheng, University of California, Davis [25].
3. Vehicle Dimensions and Velocity From Overhead Video Detectors: Art MacCarley, Cal Poly, San Luis Obispo [26].
4. Visual Vehicle Signatures from Wayside Cameras: Jitendra Malik, University of California, Berkeley, [27].

For example, item 2 above is designed to measure vehicle length with an error of 1 inch at free flow traffic speeds (versus 24 inches with the speed traps).

3.2 Competing technologies

Several systems have been proposed for measuring travel time directly using vehicle signatures [24, 28-38]. These emerging technologies use specialized hardware to extract vehicle signatures that are more descriptive than effective length. In most cases, the systems have only been installed on small test sites. Some of the systems use automatic vehicle identification (AVI), e.g., machine readable "license plates", [28-34] that make vehicle reidentification trivial, but the systems may compromise personal privacy. Furthermore, the AVI systems do not measure local velocities at the detectors, so, an incident detection system based on AVI technology would require three stations to localize the source of delay (see section 2.1 for more information).

Other surveillance systems have been proposed for estimating travel time from aggregate traffic parameters [39-40]. Although these systems appear promising for free flow and lightly congested conditions, they currently do not perform well under heavy congestion.

Another approach for measuring travel time is to match vehicles simply based on the cumulative arrivals at successive detector stations [41-42], i.e., the n-th vehicle at one station is matched to the n-th vehicle at the next station. To counter detector drift between stations, these systems use aggregate measurements to recalibrate during free flow conditions. Unfortunately, congestion can last several hours, leading to significant measurement drift between recalibrations.

Tables 3-1 & 3-2 compare the various travel time measurement systems. The reidentification rate based upon vehicle length measurements at speed traps is not as high as the emerging signature extraction technologies. But, because the former can be implemented using the existing detection hardware, the benefits of travel time measurement can be quantified before a jurisdiction commits to purchasing a travel time measurement system.

TABLE 3-1: Comparison of the infrastructure requirements for various travel time measurement systems

Mode	Primary Correlation Feature(s)	Wayside Detectors	Wayside Control Hardware
Vehicle mounted transponders / license plate readers	vehicle	new	new
Visual signature	vehicle	new	new
Magnetic signature	vehicle and platoon	existing single loops	new
Inferred from aggregate, point based measurements	features in aggregate measurements	existing single loops	existing
Cumulative arrivals	aggregate measurements	existing single loops	existing
Measured length signature	platoon	existing paired loops	existing

Although this section presents competing technologies for measuring travel time, it is not intended to give the reader the impression that any one of the technologies is better than the others under all conditions. In fact, a hybrid between two or more systems will likely yield better performance than any one of the systems operating independently.

TABLE 3-2: Projected performance of various travel time measurement systems

(1=least desirable, 5=most desirable)

Mode	Accuracy	Proportion of Vehicles Reidentified	Communications Bandwidth	Privacy	Cost	References
Vehicle mounted transponders	5	5 [a]	4	1 [b]	1 [b]	[28-31]
Video image processing license plate readers	3-4	2-4	4	1	1	[32-34]
Visual signature	1-4 [c]	2-3	2	4	2 [d]	[35]
Magnetic signature	4 [e]	3-4	2 [e]	4	3	[24, 36-38]
Inferred from aggregate, point based measurements	2-3 [f]	n/a	5	5	5 [g]	[39-40]
Cumulative arrivals, without recalibration.	1 [h]	n/a	5	5	5 [g]	[43]
Cumulative arrivals, recalibrated under free flow conditions	1-3 [h,i]	n/a	5	5	5 [g]	[41-42]
Measured length signature	3	3	4	4	4	This dissertation

a: Almost 100% of the vehicles equipped with transponders can be reidentified; however, to effectively measure travel time, the system requires significant portion of the vehicles be equipped with transponders.
b: Requires public participation to install and maintain transponders
c: Accuracy depends on lighting conditions, occlusion, camera angle, correctly segmenting vehicles from background, etc.. Nighttime and darkness appear to be a significant problem.
d: These systems are computationally intensive, cost should reduce with lower cost of computing power.
e: Bandwidth and accuracy are inversely related
f: The analysis did not provide ground truth verification against measured travel times
g: The system only requires single loops and has the lowest hardware requirement
h: Cumulative arrivals at successive sites tend to “drift” due to detection errors. Without recalibration, this method rapidly breaks down.
i: Unfortunately, congestion can last several hours, leading to significant measurement drift even with recalibration during free flow conditions

4. Vehicle Reidentification Algorithms

This chapter presents three closely related algorithms for matching vehicles at widely spaced detector stations using the measured values of effective vehicle length (i.e., the length "seen" by the detectors). A vehicle's measured length is not unique, it is subject to resolution constraints and it may be affected by measurement errors. However, a sequence of measured lengths rapidly becomes distinct and the sequence can potentially be reidentified at successive detectors. The three algorithms look for short sequences of measured vehicle lengths that exhibit a strong correlation between two stations. Lane changes and measurement errors disrupt the sequences, so the algorithms are specifically designed to match vehicles between these disruptions.

This chapter begins with an example of manual vehicle reidentification in section 4.1, where a human observer matched vehicles using visual comparisons between measured lengths at two successive detector stations. The example presents the basic strategies used by each algorithm to match vehicles and introduces notation used throughout the remainder of the chapter. The remainder of the chapter, section 4.2, describes each algorithm in detail[3] and compares them.

4.1 An example of manual vehicle reidentification

The following example uses data collected at two successive detector stations on March 10, 1993 [44]. Both stations have dual loop speed traps in each lane and the example uses the two speed traps shown in Figure 4-1.

Figure 4-2 shows just over two minutes of time series vehicle length data extracted at the two stations[4]. The upstream and downstream series were observed at different times to account for the vehicle trip times between stations. These length measurements are

[3] See Appendix A for an explicit step by step description of each algorithm.

[4] The reader can refer to Appendix C for details on how these lengths were calculated.

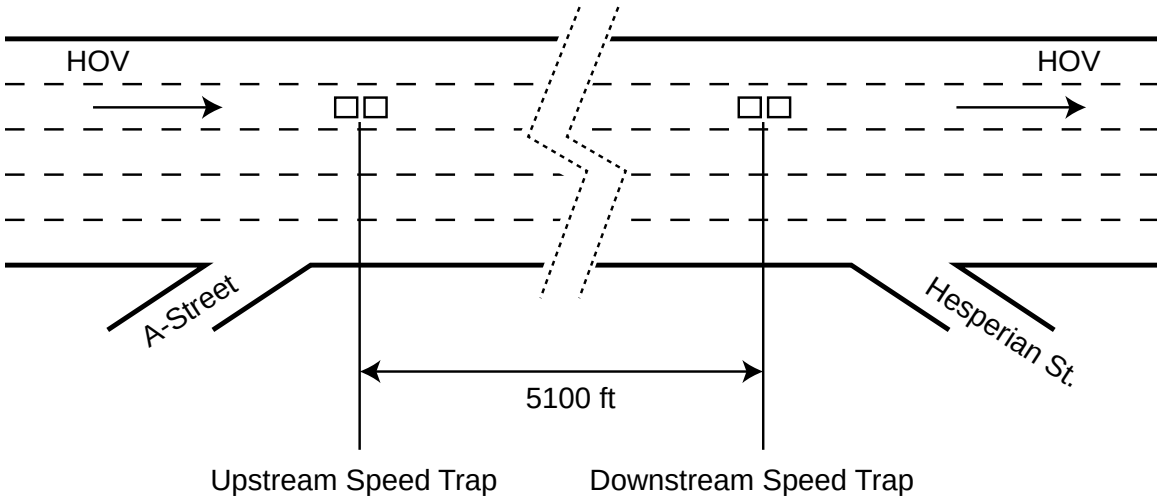


FIGURE 4-1: Region of pilot study on Interstate-880, south of Oakland, California

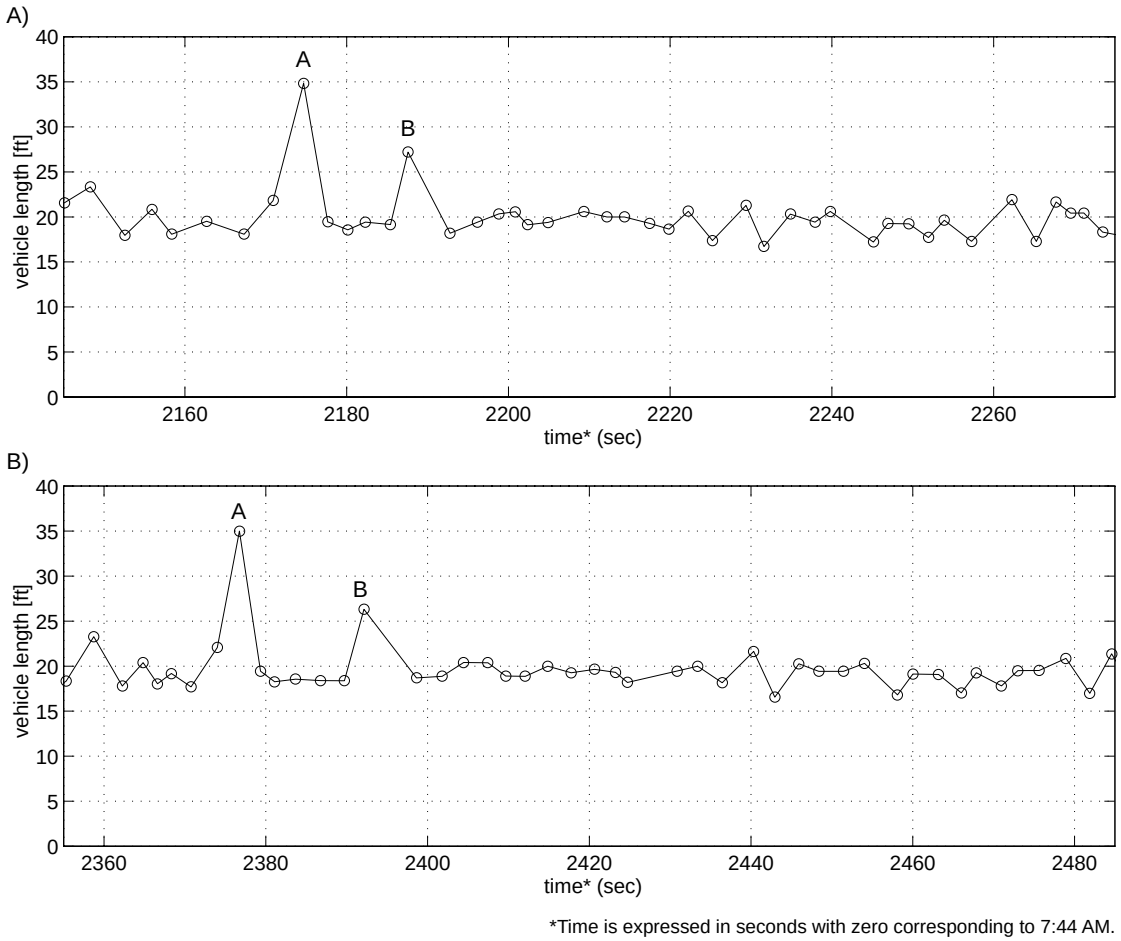


FIGURE 4-2: (A) Detail of the upstream vehicle length time series, (B) Detail of the downstream vehicle length time series

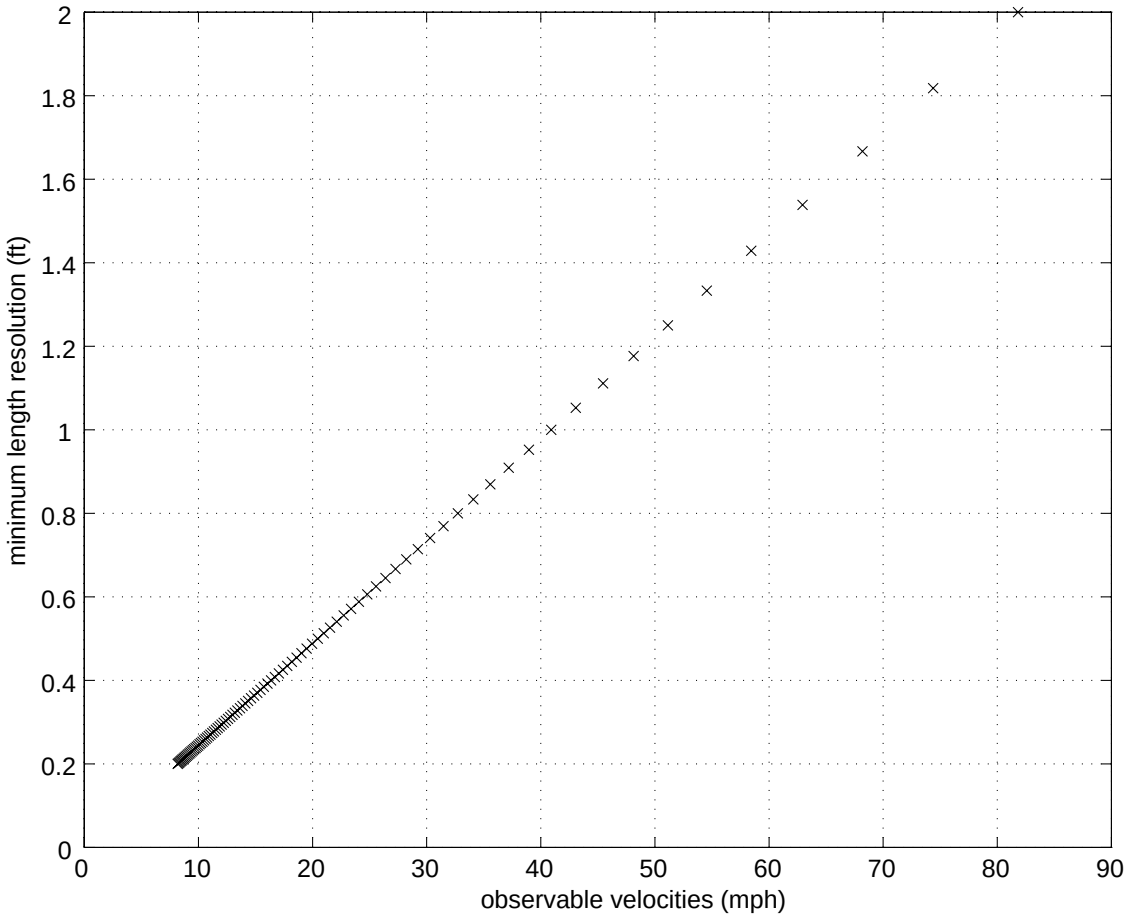


FIGURE 4-3: Minimum length resolution as a function of velocity

subject to resolution constraints that are a function of the loop separation within the given speed trap, the controller sampling rate and the vehicle velocity. Because the loop separation and sampling rate are fixed, vehicle length resolution ranges from 0.5 ft at 20 mph to 2 ft at 80 mph and this relation is shown in Figure 4-3. In addition to the resolution constraint, measurements are subject to external noise caused by misdetections and vehicles changing lanes over the detector station.

Indexing these vehicles by arrival number[5] rather than time, Figure 4-4A shows the two vehicle length sequences superimposed on the same plot while Figure 4-4B

[5] These numbers simply reflect the order that vehicles pass the given detector station and the arrival numbers at one station are not directly related to the arrival numbers recorded at any other station.

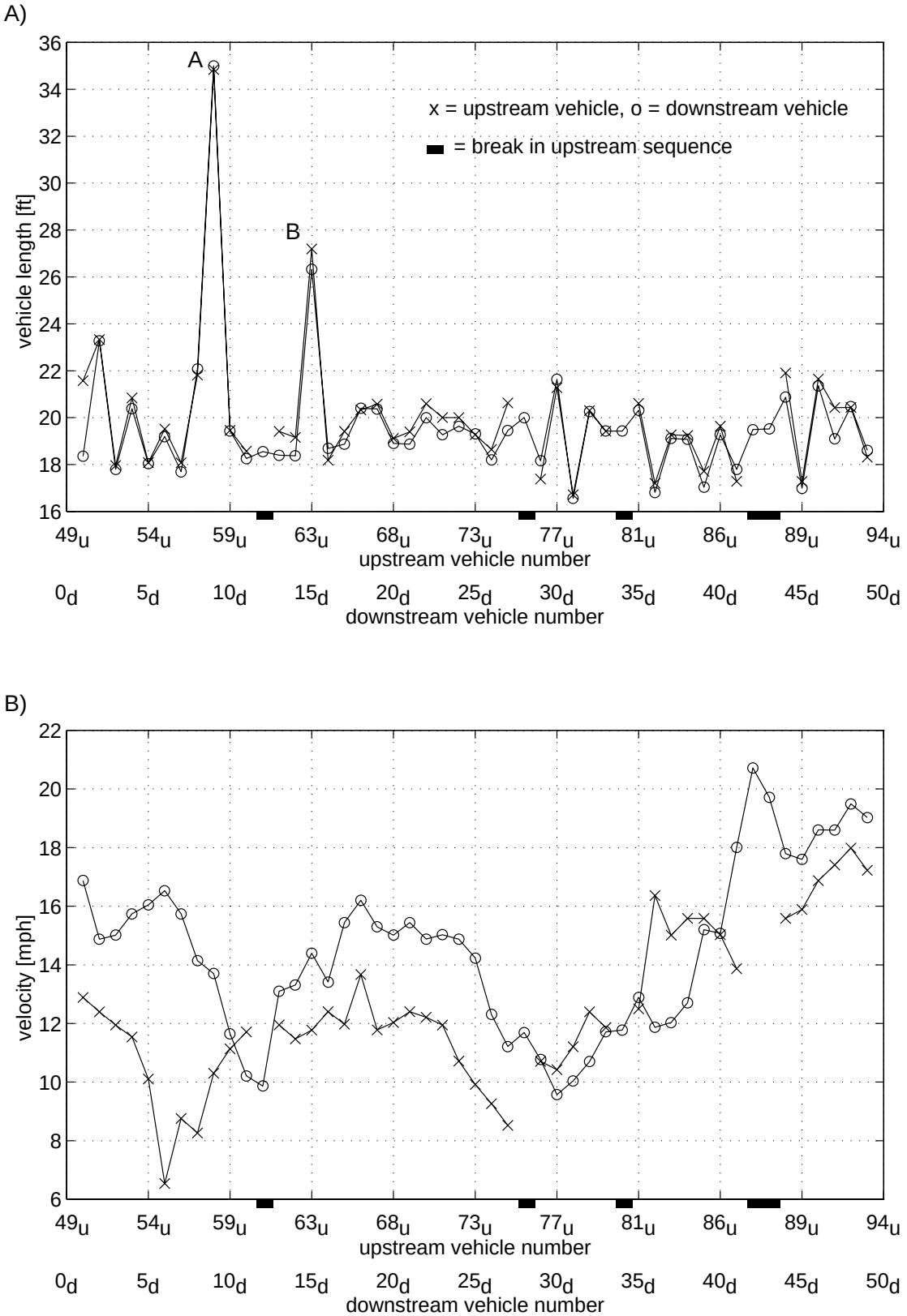


FIGURE 4-4: Manual reidentification, (A) superposition of the vehicle lengths from Figure 4-2. (B) The corresponding measured vehicle velocities.

shows the corresponding velocities for reference. To simplify later steps in the discussion, the upstream sequence starts with vehicle number 50. In this example, subscripts have been added to the vehicle numbers to differentiate between the two stations: "u" for upstream and "d" for downstream. For each match the human observer found, the upstream and downstream measurements are plotted at the same horizontal position, e.g., downstream vehicle number 35_d is matched with upstream vehicle number 81_u. As part of the matching process, the observer inserted four breaks in the upstream sequence, where a break is simply a horizontal shift in one of the sequences. A break in one sequence is analogous to deleting a vehicle that does not have a match from the other sequence; i.e., breaks represent lane changes that occurred between the detector stations and/or detector errors at the stations. The breaks were inserted strictly on the basis of improving the match between the upstream and downstream length measurements. The difference between the upstream and downstream length measurements is less than 1/2 foot for approximately 75 percent of the matches in this figure. The strong similarity between the two sequences, in conjunction with the correlation of the two long vehicles (labeled A and B in the figure), point to the feasibility of reidentifying vehicles from sequences of measured vehicle lengths.

Replotting the matches from Figure 4-4A with respect to the arrival number at each station yields Figure 4-5. The vertical axis is increasing downward in this figure because it was plotted using matrix notation. The matches tend to fall into diagonal sequences at -45 degrees[6], with occasional deviations due to lane changes. Thus, for all of the vehicles in a platoon between two successive deviations, the upstream arrival number differs from the downstream arrival number by a fixed offset.

[6] In other words, a match will usually be to the right one column and down one row from a preceding match in this figure.

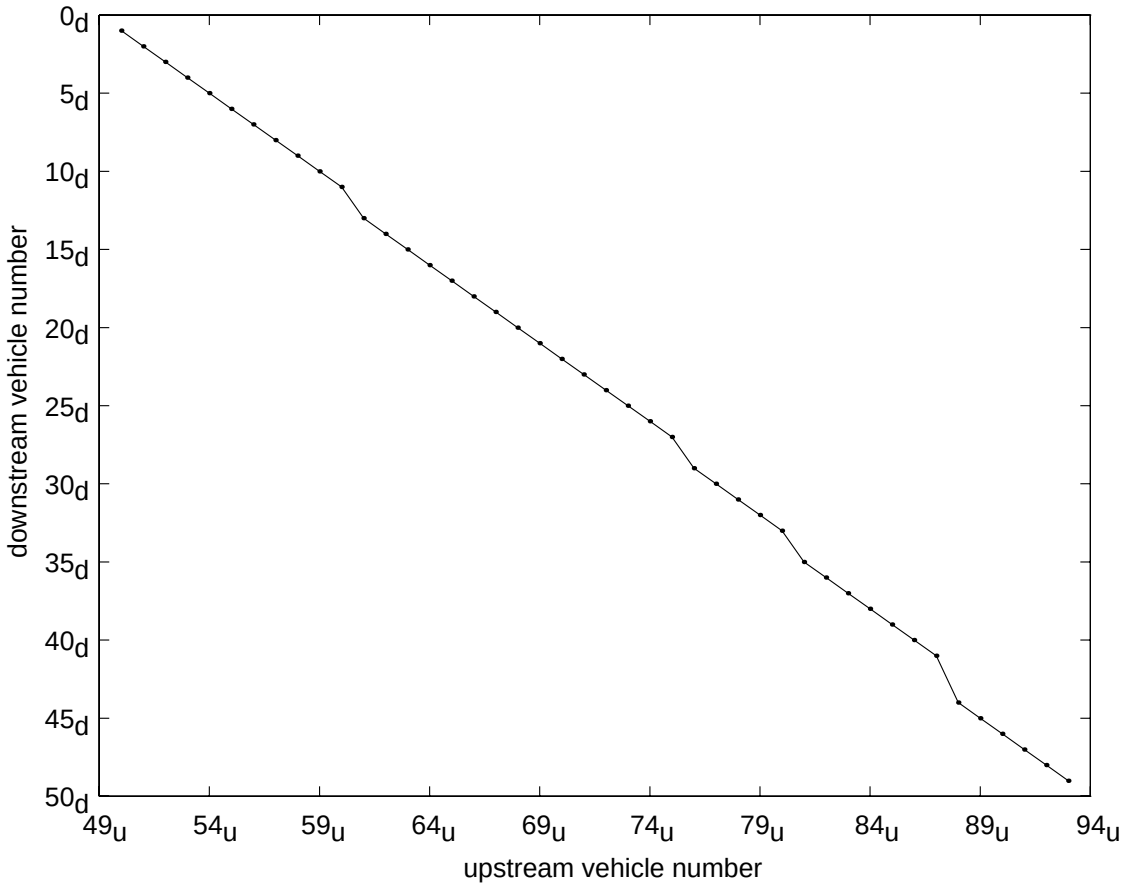


FIGURE 4-5: This figure shows the matches from Figure 4-4A plotted with respect to the arrival number at each station. Note that the vertical axis is increasing downward in this figure because it was plotted using matrix notation.

4.2 Algorithm description

The three approaches to reidentifying vehicles automatically are presented in this section. First, subsection 4.2.1 presents the Basic Algorithm, which attempts to find an upstream match for every vehicle that passes the downstream station. Under free flow traffic conditions, the vehicle length measurement resolution degrades, making difficult the task of differentiating between vehicles. The Subsampling Algorithm, which only matches distinct vehicles, was developed in response to these deficiencies and is presented in subsection 4.2.2. The Approximation Algorithm presented in subsection 4.2.3 provides a second approach to overcome the same deficiencies. This final approach tries to find the best fixed offset for a group of n vehicles, the group offset is used as an approximation

for each individual vehicle's offset within the group. After presenting the three algorithms, this section concludes with a brief summary contrasting the different approaches.

4.2.1 Basic Algorithm

The basic reidentification algorithm attempts to match each vehicle's length measurement at the downstream station with its corresponding upstream measurement. Of the three algorithms examined, this approach could yield the most information about the traffic stream because it attempts to make an exact match for a large number of vehicles.

The algorithm starts by comparing individual length measurements between the two stations using a resolution test described below. If the difference between the upstream and the downstream measurements exceed the measurement uncertainty (which is a function of velocity, as shown in Figure 4-3) then the observations probably did not come from the same vehicle. The pair of vehicles can then be marked as an *unlikely* match. Otherwise, the pair of measurements can not be eliminated by this test and the pair is marked as a *possible match*.

The algorithm applies the resolution test to each pair of upstream and downstream measurements from some specified group of vehicles. In practice, the group is selected to ensure that the true, but unknown, match for a downstream vehicle will fall somewhere in the upstream set (see Appendix A for more details). The results of these resolution tests can be summarized in a *vehicle match matrix*. The matrix is indexed by arrival number at each station (upstream and downstream) and each element of the matrix is the outcome of a single pair-wise resolution test. Figure 4-6 shows an example of the notation used in the *vehicle match matrix*.

The fixed set of vehicles from Figure 4-4 yield the *vehicle match matrix* shown in Figure 4-7. The horizontal axis is indexed by upstream arrival number and the vertical

(A)

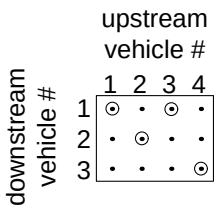
downstream, d: vehicle #	length
1	20
2	30
3	16

upstream, u: vehicle #	length
1	20
2	31
3	21
4	15

(B)

possible matches (d,u)
(1,1)
(1,3)
(2,2)
(3,4)

(C)



"o"= lengths pass the pair-wise resolution test, thus, a possible match

FIGURE 4-6: A simple example of notation: (A) measured vehicle lengths, (B) possible matches with a length measurement tolerance of 1 unit, (C) resulting vehicle match matrix.

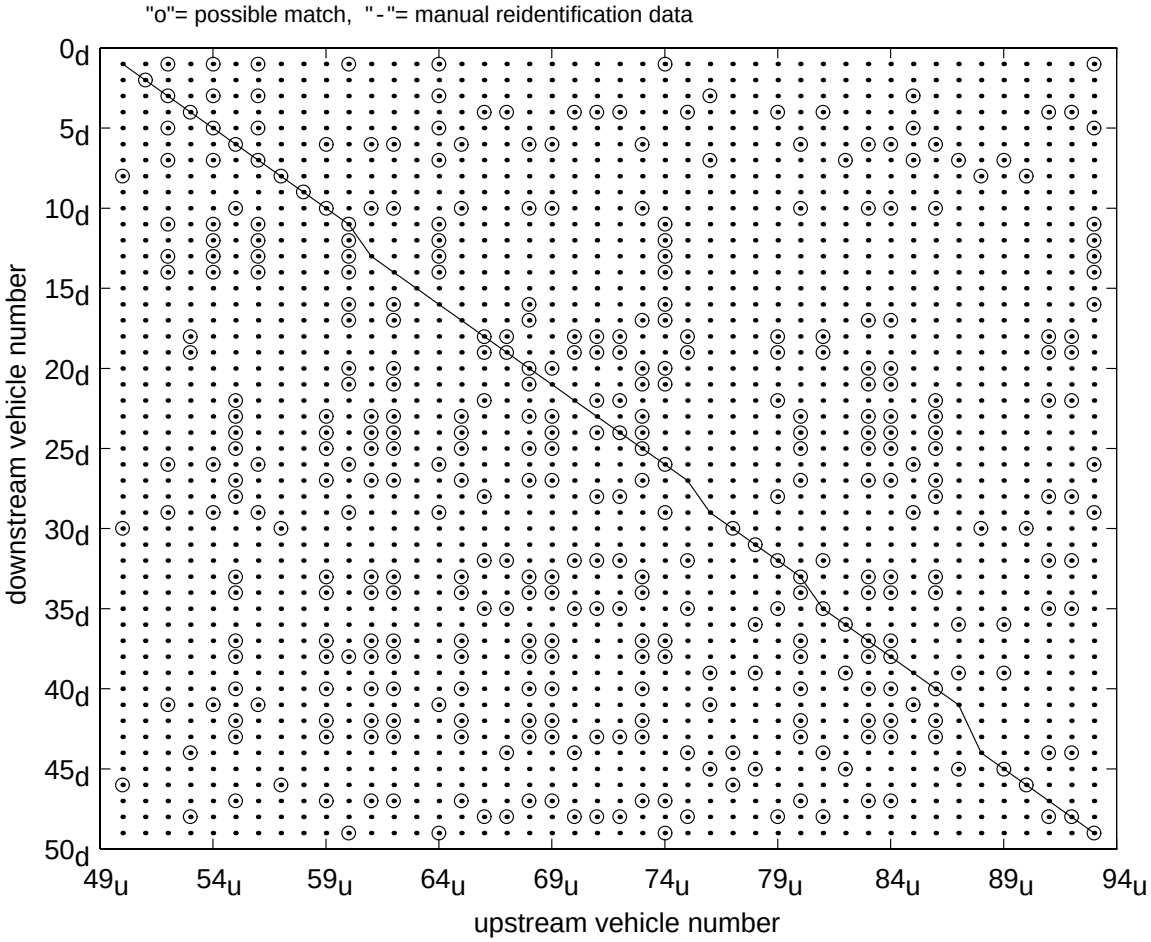


FIGURE 4-7: Vehicle Match Matrix, summarizing the outcome from many successive resolution tests

axis is indexed by downstream arrival number. In this figure, "O" indicates a *possible match* because the two length measurements are within the measurement uncertainty, while all other elements are left empty to indicate that a match is unlikely between the given pair of vehicles. The manually generated reidentifications from section 4.1 are shown for reference with the solid line, but they are unknown by the algorithm.

Many false positives are clearly evident in Figure 4-7 since each vehicle can only have, at most, one true match, yet most rows have more than one *possible match* for the given downstream vehicle. Assuming that any two successive length measurements at a detector station are independent of each other, the false positives are manifest as random noise in the *vehicle match matrix*. If a false positive occurs with probability less than 0.5, a false positive should usually be preceded (moving up one row and shifting left one column in the matrix) by an *unlikely* element. Whereas, if vehicles maintained their order between the two stations and the probability of a false negative[7] is less than 0.5, a true match should usually be preceded by a *possible match* element. Relaxing the order constraint somewhat, the work of John Windover on driver memory [45] has shown that long sequences of drivers often maintain their headway, and thus, their order for extended distances. So, if vehicles usually maintain their order between stations, the true (but unknown) matches should manifest themselves as sequences (diagonal lines at -45 degrees) of *possible matches* in the *vehicle match matrix*. In other words, false positives will typically form short sequences while the true matches will usually form longer sequences in the *vehicle match matrix*. To exploit this property, the algorithm looks for sequences of *possible matches* in the *vehicle match matrix* and tallies how many sequential vehicles matched at both stations. These totals are stored in the *sequence matrix;* in which each element contains an integer totaling the cumulative number of *possible matches* in a sequence up to and including the given element[8]. Figure 4-8 shows

[7] Where a false negative is a matrix element marked as *unlikely* even though the two measurements actually came from the same vehicle.

[8] Thus, unlikely matches are represented by zeros, or for clarity of display, blanks in the graphical format.

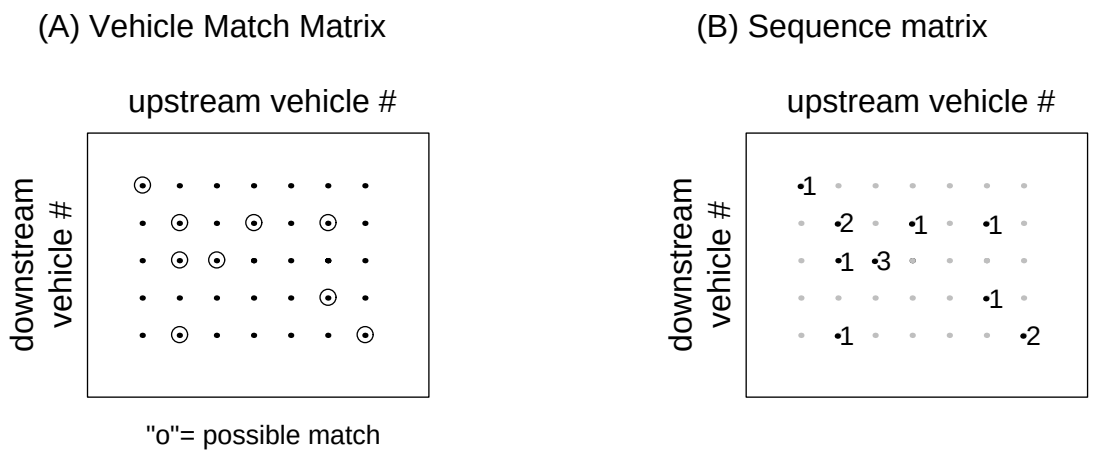


FIGURE 4-8: A simple example illustrating the transition from (A) the Vehicle Match Matrix to (B) the Sequence Matrix. Each non-zero element in the Sequence Matrix indicates the total number of Possible Matches in the sequence up to and including the given matrix element.

a simple example of the conversion to the *sequence matrix*. The *sequence matrix* for the on-going example is shown in Figure 4-9, where elements of length one have been omitted for clarity.

Next the algorithm allows for lane changes and/or misdetections in the sequences. Figure 4-10A-C shows the three lane change maneuvers searched for by the algorithm:

(A) one vehicle exits the lane between stations or a vehicle is not detected at the downstream station, (upstream vehicle n-1 in the example),

(B) one vehicle enters the lane between stations or a vehicle is not detected at the upstream station, (downstream vehicle m-1 in the example),

(C) one vehicle enters and one vehicle leaves the lane between stations or there is a false negative in the data, (vehicles m-1, n-1 in the example).

For each sequence of vehicles in the *sequence matrix*, the algorithm checks the first element to see if it can be linked to an earlier sequence (i.e., a sequence starting with a lower vehicle number) via a lane change maneuver. The procedure is demonstrated using the sequence starting with element (m,n) in Figure 4-10D, the algorithm checks the

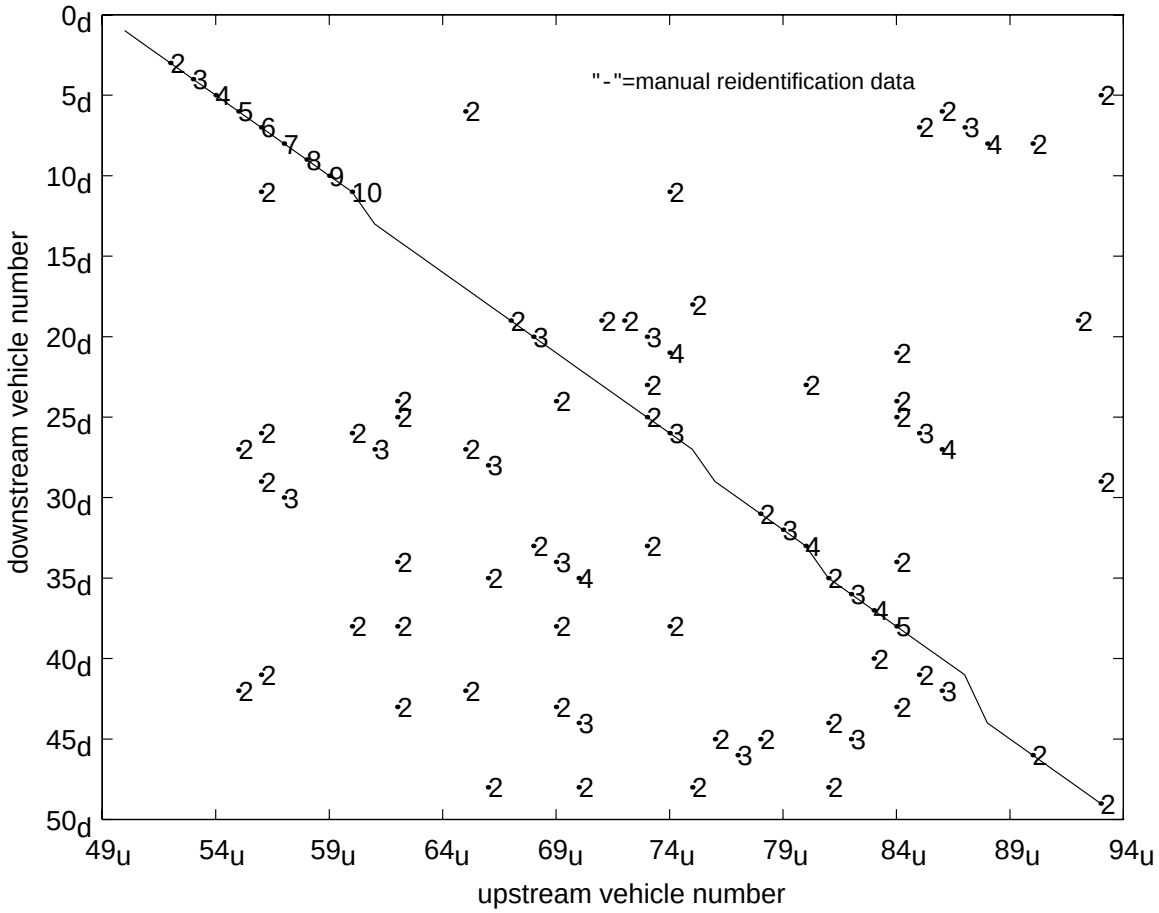


FIGURE 4-9: Sequence matrix, indicating the sequential number of possible matches

sequence matrix to see if there are any earlier sequences passing through one of the three shaded elements, where each element corresponds to one of the lane change maneuvers shown in Figures 4-10A-C. If so, the algorithm increments all elements in the sequence starting at (m,n) by the highest value from the shaded elements in the *sequence matrix*, less a penalty of one vehicle for the lane change, and places the modified-sequence[9] in the *lane change matrix*. The penalty gives contiguous sequences a slight advantage in the final step of the algorithm. Otherwise, if there are no preceding sequences in the shaded elements, then the algorithm simply copies the entire sequence unchanged from the *sequence matrix* to the *lane change matrix*.

[9] "modified-sequence" implies that the sequence was modified because of a lane change.

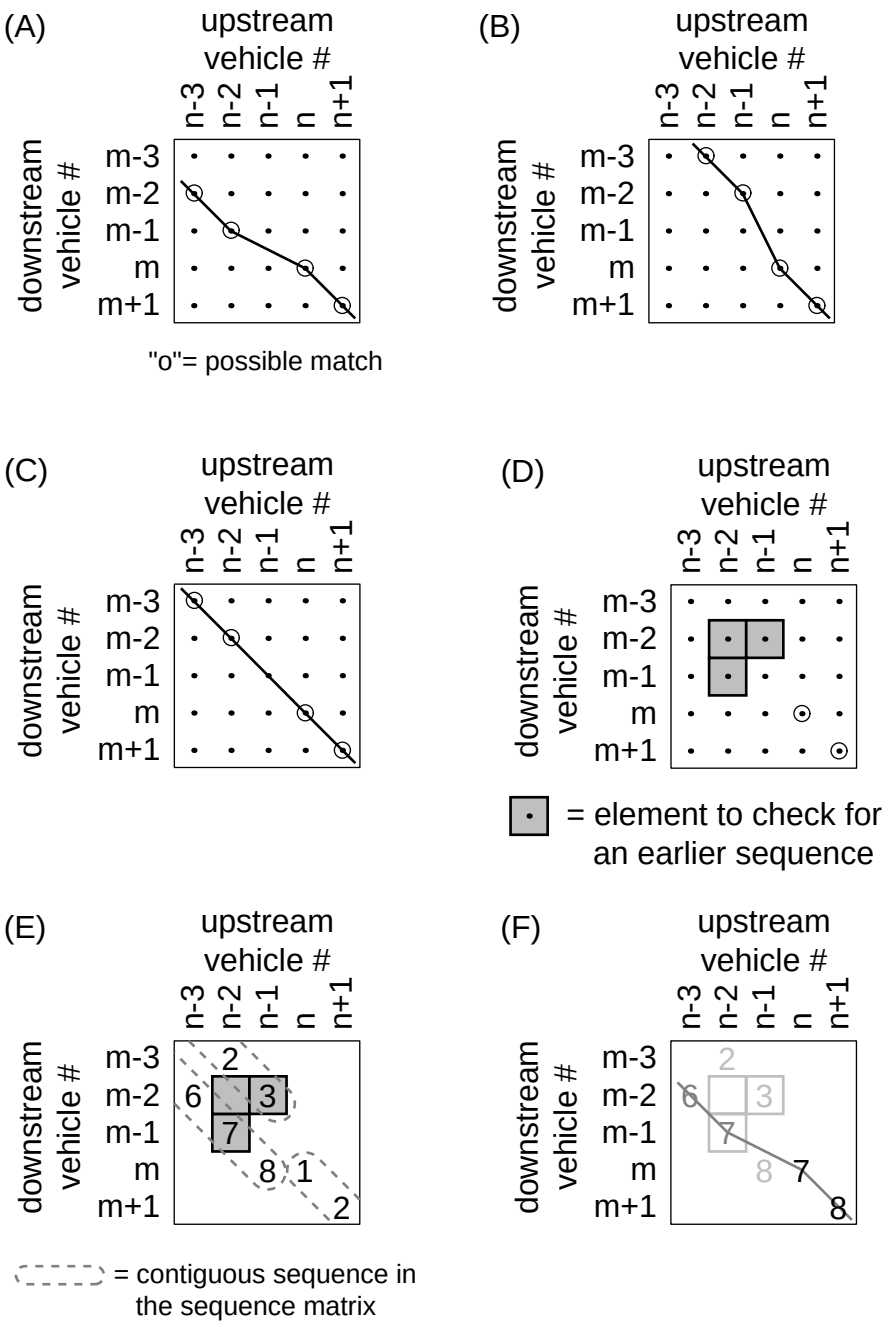


FIGURE 4-10: A simple example illustrating the possible lane change maneuvers recognized by the Basic Algorithm: (A) One vehicle exits the lane between stations, (B) One vehicle enters the lane between stations, (C) One vehicle enters and one vehicle exits the lane between stations, (D) The search region for the sequence starting at element (m,n), (E) a hypothetical sequence matrix with (F) the resulting lane change matrix with a modified-sequence starting at element (m,n) shown in black.

For example, Figure 4-10E shows a hypothetical *sequence matrix* with three sequences, two of which start before downstream vehicle m-3 and are not shown in their entirety. When the algorithm reaches the sequence starting at (m,n), it finds that there are two earlier sequences that pass through the search area (shown in gray). It takes the highest value in the search area, 7, subtracts 1, adds the result to all of the elements in the current sequence and then places the modified-sequence in the *lane change matrix*, shown in Figure 4-10F. Figure 4-11 shows the *lane change matrix* for the on-going example, again, all elements of length one are omitted for clarity.

Finally, the algorithm identifies final matches by extracting all sequences from the *lane change matrix* longer than a pre-specified threshold. Entire sequences (and modified-sequences) are selected from the *lane change matrix*, successively from longest

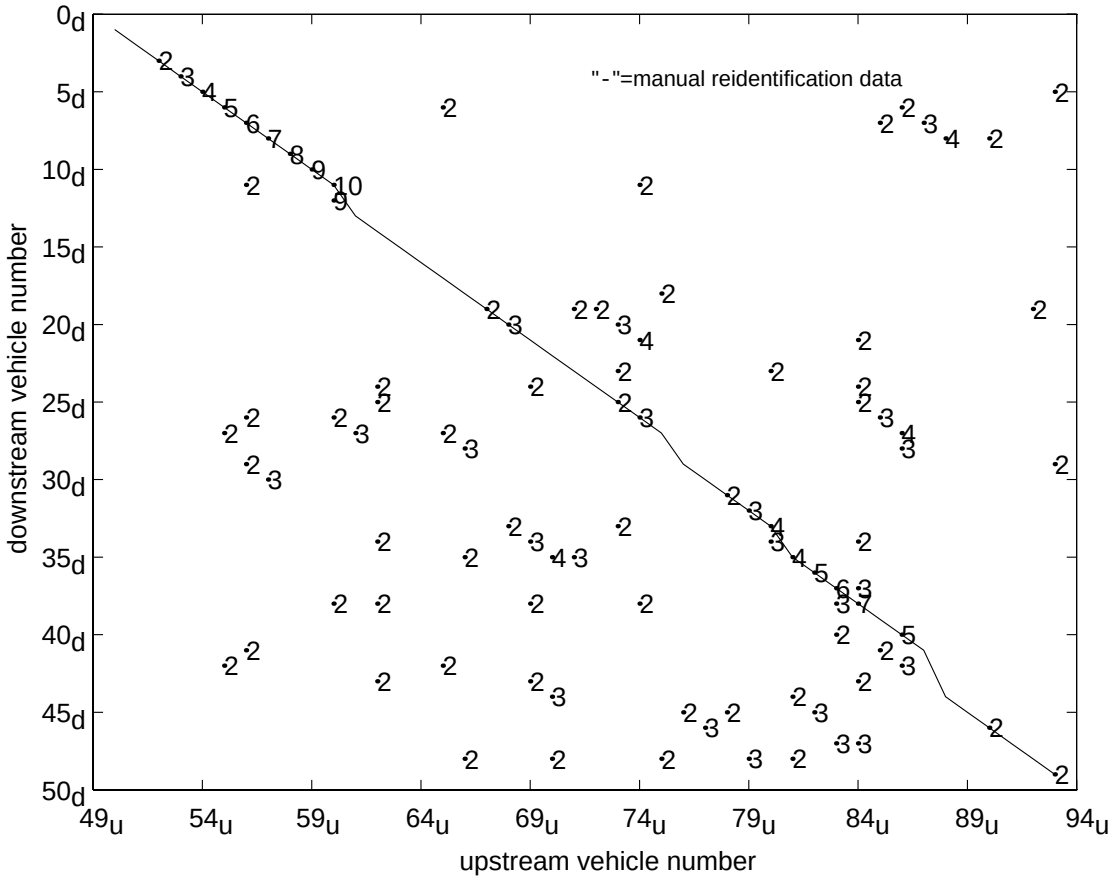


FIGURE 4-11: Lane change matrix, allowing for modified-sequences containing a single lane change maneuver

to shortest[10] and are copied to the final matrix, called the *threshold matrix*. Once a given match has been identified, the corresponding row and column of the *lane change matrix* are removed from further considerations. In the on-going example, a threshold level of five matches for a sequence yields the two platoons shown in Figure 4-12. Note that both platoons fall on the manually calibrated data and almost half of the vehicles that passed the detector stations were reidentified (i.e., matched).

Travel time for a reidentified vehicle can then be measured by taking the difference in known arrival times at the two stations. To estimate travel time during the

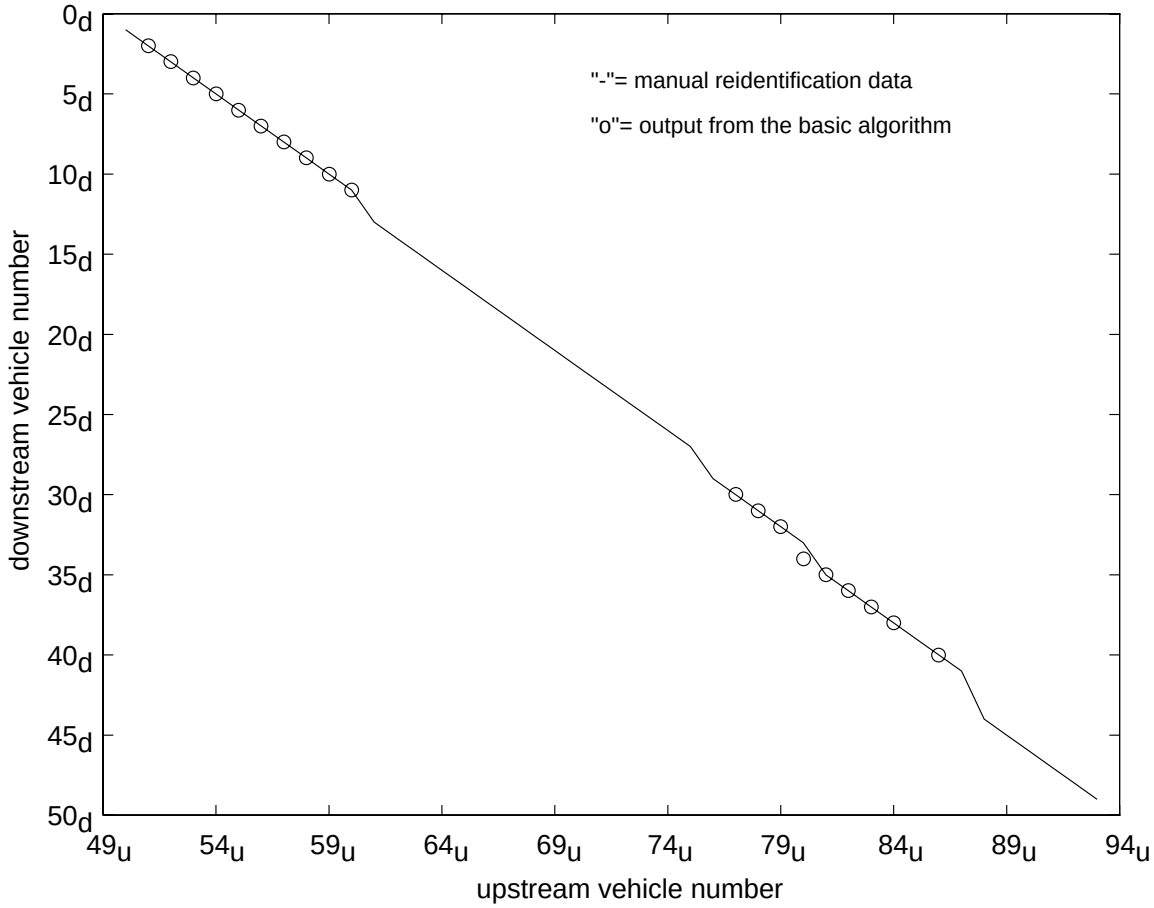


FIGURE 4-12: Threshold matrix, retaining only those sequences longer than a threshold length

[10] Note that a modified sequence starts after a lane change from an earlier sequence. The algorithm will identify the earlier sequence and it will treat the union of the two sequences as if it were a single sequence.

short periods with no reidentified vehicles, the reidentification process can be approximated by pairing vehicles based on the cumulative number to pass each station after the last correlated sequence, i.e., progress through the matrix at -45 degrees from the last match until a new match has been identified.

4.2.2 Subsampling Algorithm

The Basic Algorithm works well under congested traffic conditions. But as previously mentioned, the vehicle length measurement resolution degrades at free flow velocities, causing the number of *possible matches* to increase in the Basic Algorithm. Furthermore, vehicles may be less likely to maintain their order between detector stations in free flow conditions due to frequent opportunities to overtake one another. Subsampling a distinct segment of the total sample can overcome these problems.

Most vehicles on the highway (e.g., sedans, pickup trucks, etc.) are small and have effective lengths on the order of 16-22 ft. The range of these effective lengths is only 6 ft, but the vehicle length measurement uncertainty may be as poor as 2 ft at free flow velocities, making difficult the task of differentiating one small vehicle from another. Consider the observed distribution of vehicle lengths at one detector station, as shown in Figure 4-13A, approximately 80 percent of the measurements fall into the 16-22 ft range. The effective length for long vehicles, on the other hand, can range from 22 ft to over 80 ft[11], e.g., Figure 4-13B. By restricting the Basic Algorithm exclusively to long vehicles, the large range of lengths can offset the degraded measurement resolution. Because the long vehicles make up a small portion of the population, there will frequently be large headways between two successive observations. The large headways reduce the opportunity for overtaking and increase the probability of maintaining the vehicle sequence between detector stations.

[11] The upper limit is a semi truck with two trailers.

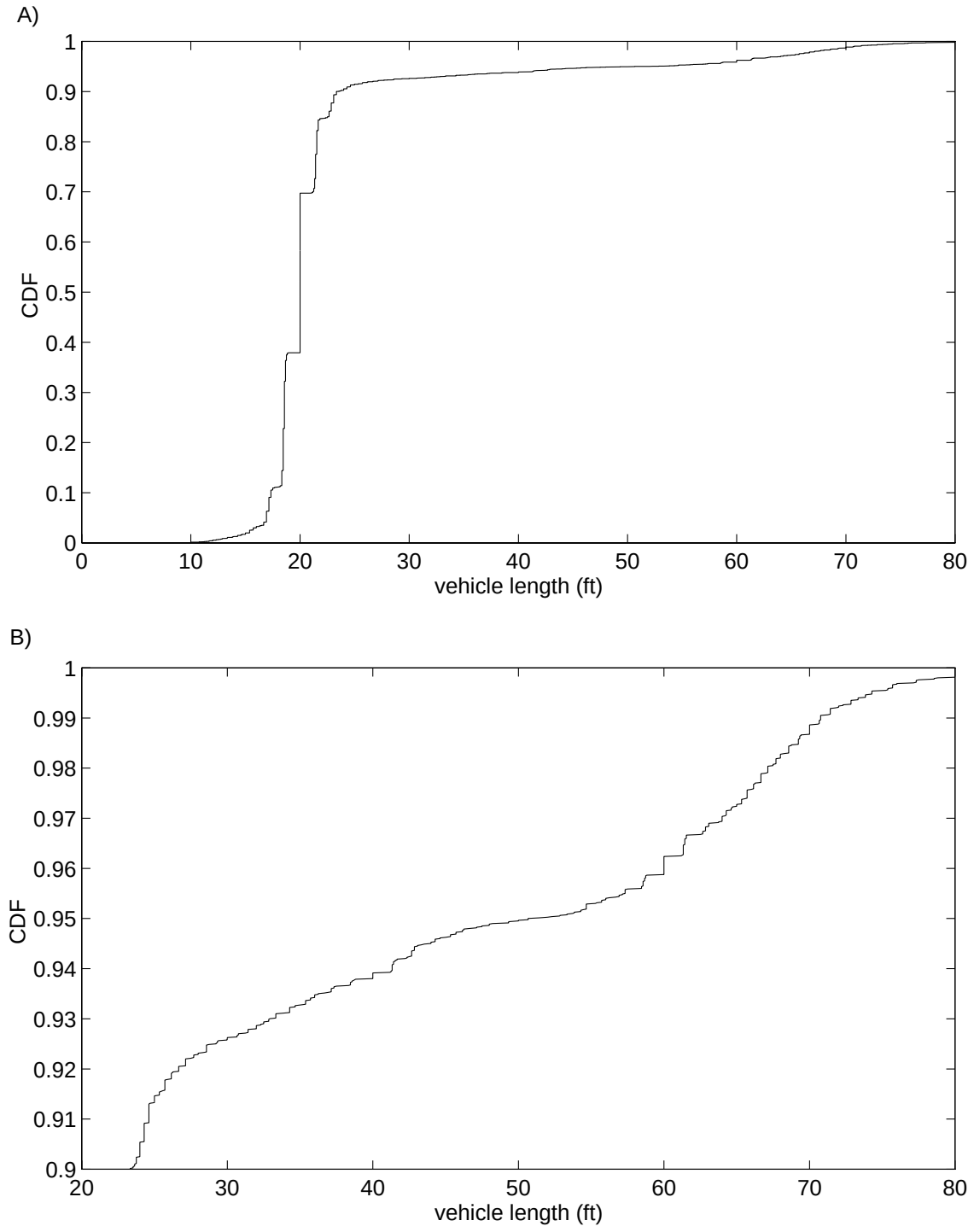


FIGURE 4-13: (A) Cumulative Distribution of measured vehicle lengths in one lane during the evening peek at a detector station. Sample size = 8002 vehicles. (B) Detail of the CDF from part A, showing the large range of lengths observed in the longest 10 percent of the vehicles.

Before comparing measurements from two stations, the algorithm "subsamples" all vehicles longer than some pre-specified minimum length at each station and assigns sequential integers according to their arrival. Using the data in Figure 4-2 and a minimum length of 21 ft, the algorithm subsamples about 20 percent of the vehicles at each station. The Subsampling Algorithm applies the Basic Algorithm only to the subsamples, i.e., it attempts to match all long vehicles by following the steps previously described. First, the algorithm generates a *vehicle match matrix* (Figure 4-14A); second, it identifies sequences of potential matches (Figure 4-14B); third, it allows for lane

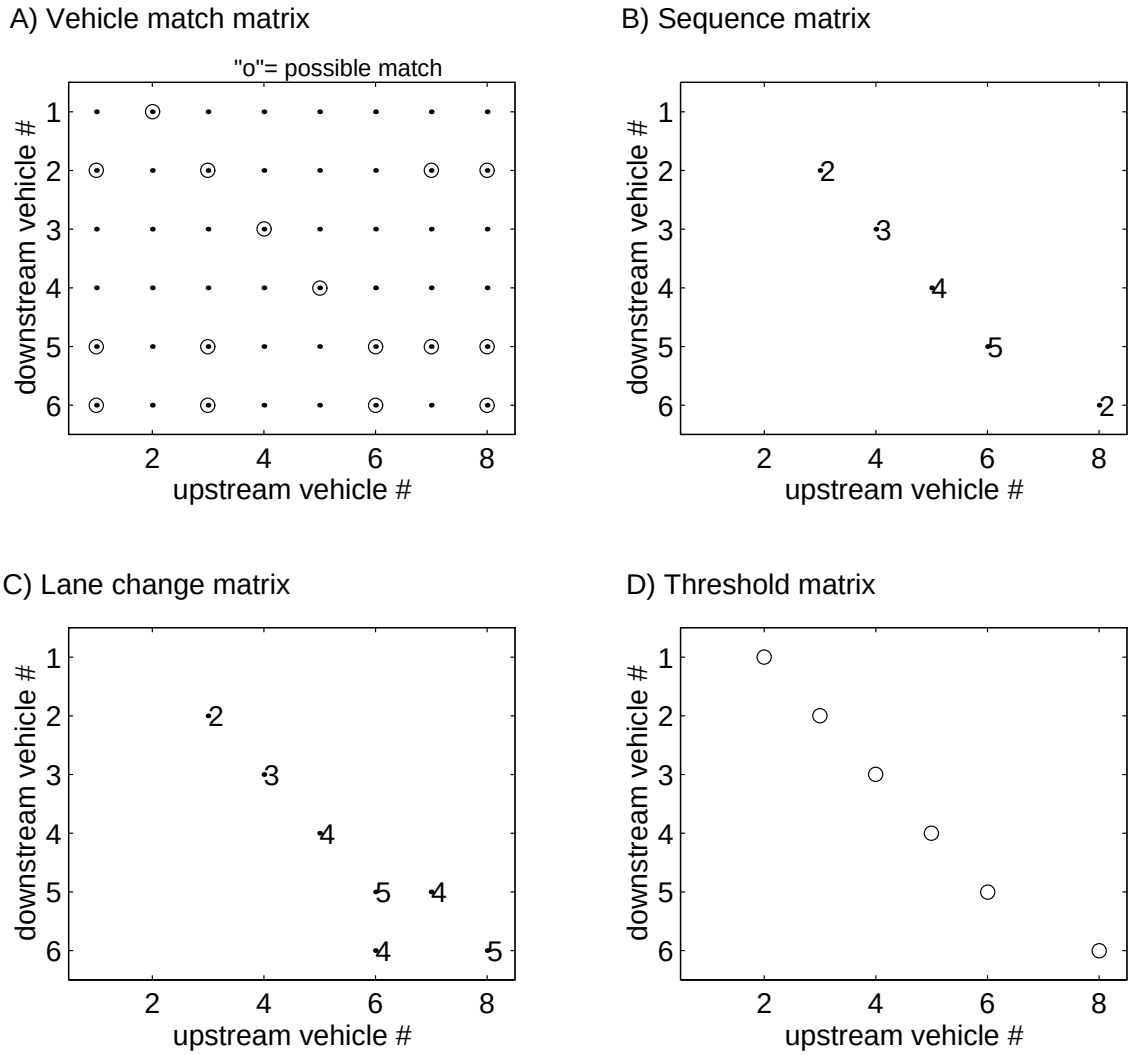


FIGURE 4-14: The Subsampling Algorithm, apply the Basic Algorithm to all vehicles longer than a pre-specified minimum length

change maneuvers (Figure 4-14C); fourth, it keeps only those sequences over a given threshold (Figure 4-14D). Finally, the matches from Figure 4-14D are transposed back to the original sample as shown in Figure 4-15. Note that the Subsampling Algorithm has correctly reidentified two vehicles, downstream numbers 15_d and 46_d, that were not matched using the Basic Algorithm in Figure 4-12. These vehicles fall into short sequences using the Basic Algorithm and they are eliminated, but within the subsample, they fall into longer sequences and they are correctly matched by the Subsampling Algorithm.

Naturally, travel time for long vehicles, i.e., trucks, may not be representative of the entire vehicle population. So, this algorithm is intended for free flow conditions, when local velocity measurements at the detector stations should be representative of the

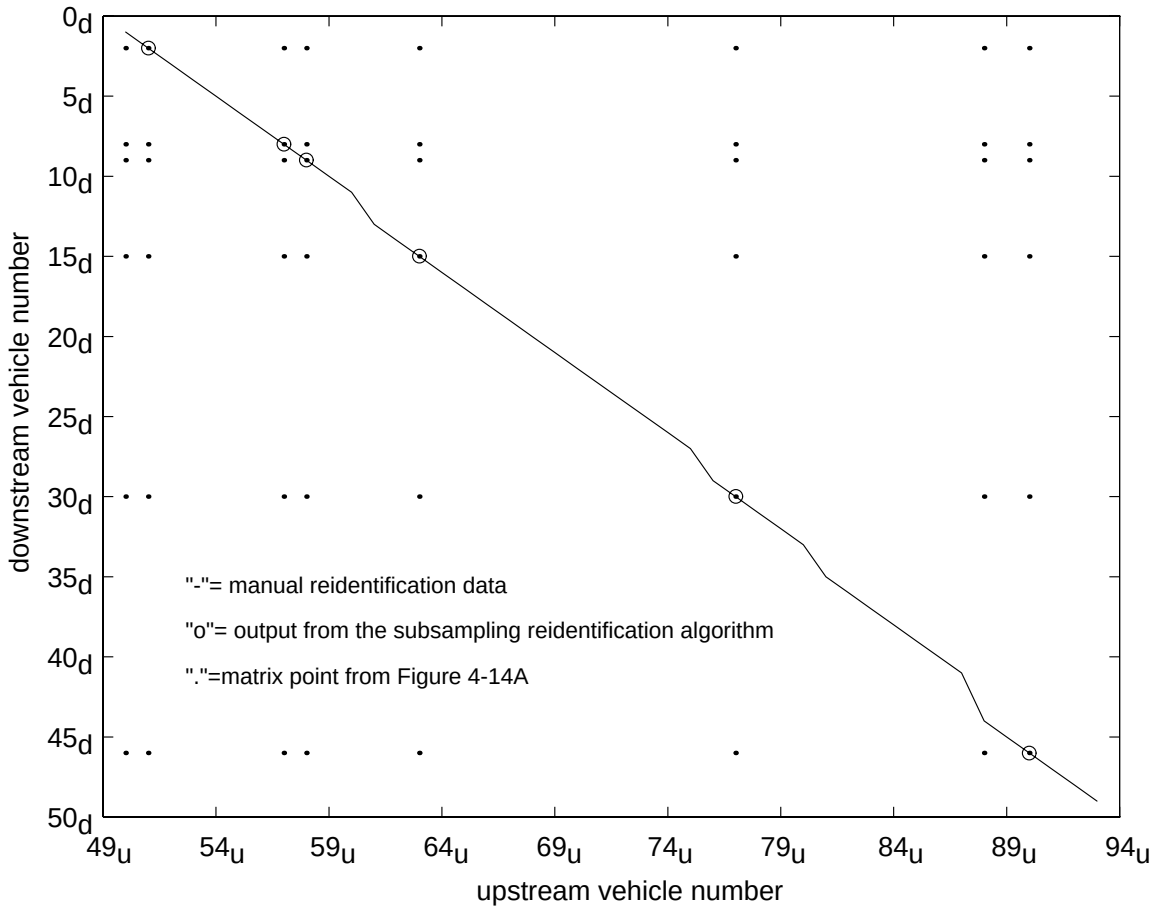


FIGURE 4-15: Transpose the subsample matches back to the original vehicle indices

entire link. The trucks serve as a good diagnostic for the onset of congestion and thus, signal the need to switch back to the Basic Algorithm using the entire population.

Finally, with more advanced vehicle detection systems, it should be possible to subsample vehicles based on other features, such as vehicle color measured from a video image processing system. It is a simple extension of the algorithm to process multiple subsamples in parallel, such as one subsample containing all of the red vehicles and another subsample containing all of the green vehicles[12].

4.2.3 Approximation Algorithm

The Approximation Algorithm provides a second approach to overcome degraded measurement resolution as well as addressing the possibility that vehicles may overtake one another. It attempts to find an approximate match for every vehicle, but it does not provide an exact match. Like the Basic Algorithm, each downstream vehicle is compared to a large number of upstream vehicles using the resolution test to identify all *possible matches*. As previously noted, a given vehicle can have, at most, one true match, with all other *possible matches* being false positives. To reduce the influence from vehicles with many false positives, those vehicles with an uncommon length and thus, few *possible matches*, are assigned greater weight by the algorithm; for a vehicle with n *possible matches*, each *possible match* is assigned the weight 1/n.

The algorithm generates a *vehicle match matrix* to summarize the comparisons of successive downstream vehicles with numerous upstream ones. Figure 4-16 shows a *vehicle match matrix* for a larger set of upstream vehicles than was used in the earlier examples. The larger matrix is necessary because the algorithm calculates the average weight on each diagonal. Columns 50-93 are the same data shown in Figure 4-7. The lower left and upper right hand corners of this larger matrix are blank, indicating that no

[12] Note that the subsamples do not have to be mutually exclusive, the sampling criteria could be selected so that some vehicles will be included in several different subsamples.

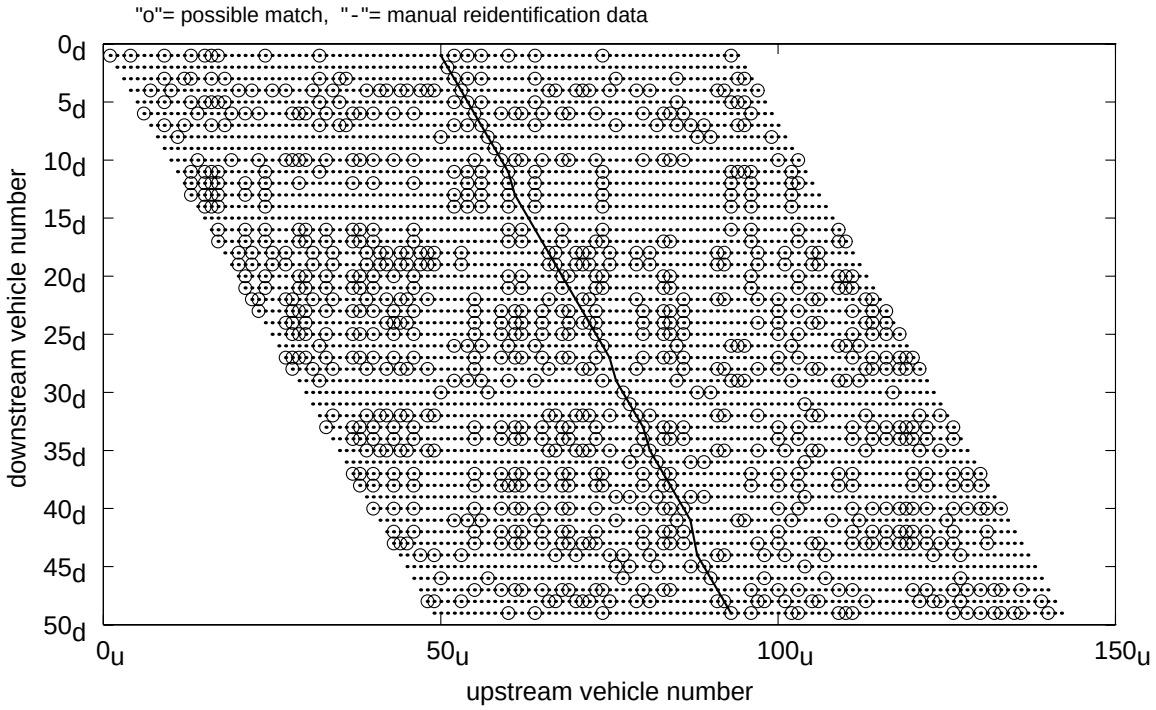


FIGURE 4-16: A larger vehicle match matrix, this time illustrating a moving window search

comparisons were made in these regions. These sections were excluded by design: each row contains the same number of pair-wise comparisons (i.e., 94), but the set of upstream vehicles is shifted to the right by one vehicle in each new row. Because it is easier to write computer code to calculate the average weight by column than by diagonals, vehicles will be indexed by upstream offset rather than upstream vehicle number, where,

upstream offset = upstream vehicle number - downstream vehicle number.

Replotting the data from Figure 4-16 using the upstream offset rather than upstream arrival number yields the *offset match matrix*, as shown in Figure 4-17[13]. In other words, by shifting all rows to the left this step has simply removed the blank space in the lower left hand corner of Figure 4-16. Note that in this new coordinate system, if there were no misdetections or lane changes between the stations, all of the true matches would all fall

[13] Continuing the use of subscripts in this example, the upstream offset is denoted by “uo”.

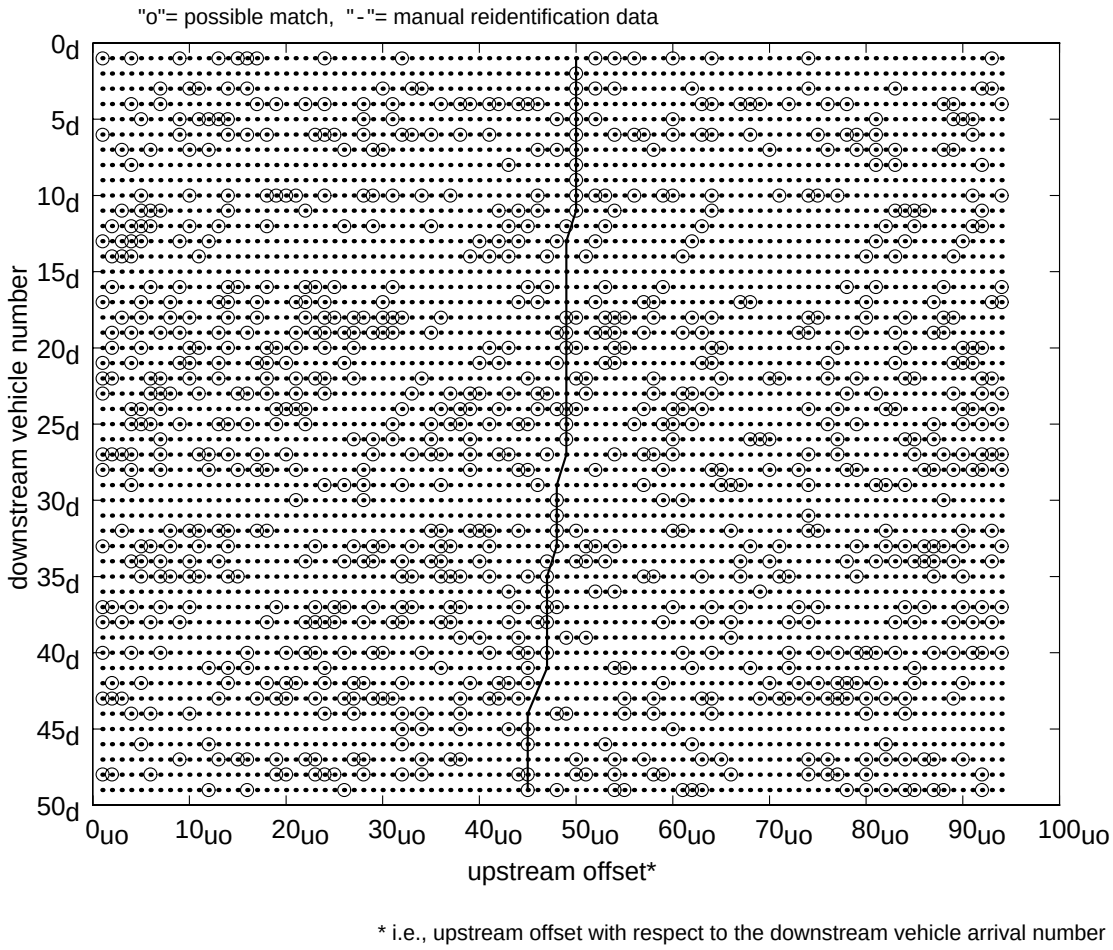


FIGURE 4-17: Offset match matrix

in the same column. Any lane change or misdetection, however, will cause a column shift in the true matches, as demonstrated by the manually matched data (the solid line in Figure 4-17).

Following the same logic presented in subsection 4.2.1, for any downstream vehicle, a true positive is more likely to be preceded (moving up one row in this case) or followed (moving down one row in this case) by a *possible match* as compared with the false positives for that vehicle. To eliminate most of the false positives, the algorithm searches for short sequences of *possible matches* (e.g., only one or two vehicles) and eliminates them from further consideration. The remaining data are stored in the *filtered offset match matrix*. Figure 4-18 shows a simple example of this elimination or

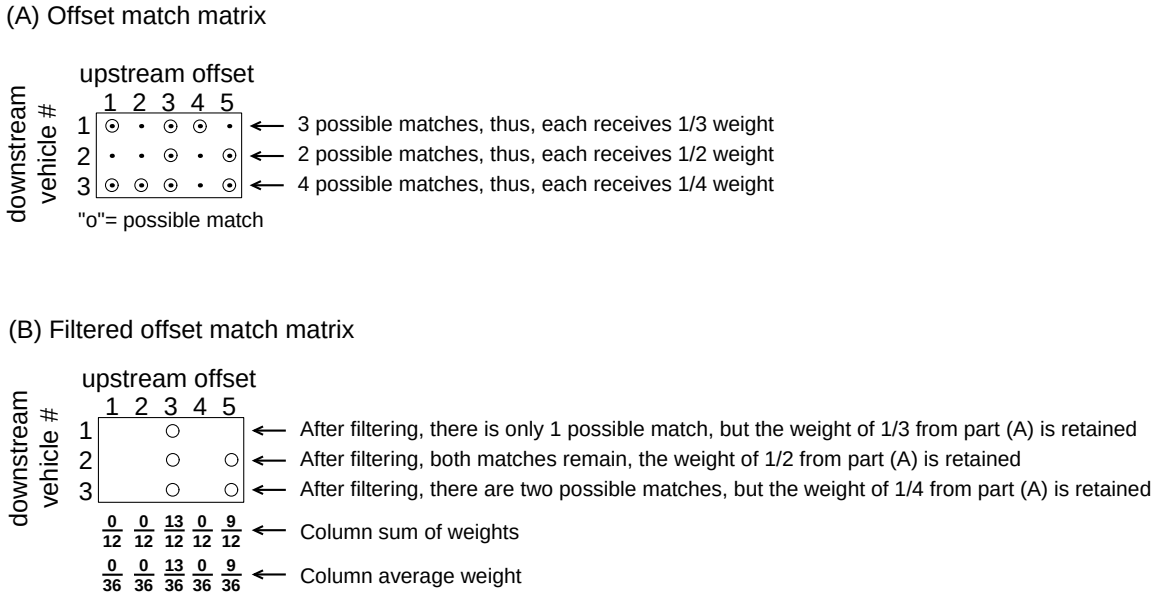


FIGURE 4-18: An example of notation and weight assignment (A) each row in the offset match matrix receives a total weight of 1; (B) all vertical sequences with only one vehicle are removed, yielding the filtered offset match matrix, however, the weights from the original match matrix are retained

"filtering" process while Figure 4-19A shows the data from Figure 4-17 after all sequences shorter than three vehicles have been eliminated.

The algorithm calculates the average weights on each column, this is illustrated at the bottom of Figure 4-18B, while the resulting averages for the on-going example are shown by the "X"'s in Figure 4-19B. The larger averages occur in those columns that include a *possible match* for vehicles of uncommon lengths. Since the algorithm has already shed the false positives for the uncommon vehicles, these high averages indicate that the true matches likely resided in that column for some collection of downstream vehicles; because of lane changing (and detector errors), the true matches will tend to shift columns of the *filtered offset match matrix* as the downstream vehicle number increases. But the column shifts due to lane changes will typically be small relative to the number of upstream vehicles under consideration and the high averages should fall in a small region, e.g., the averages in Figure 4-19B. A moving sum is used to find the

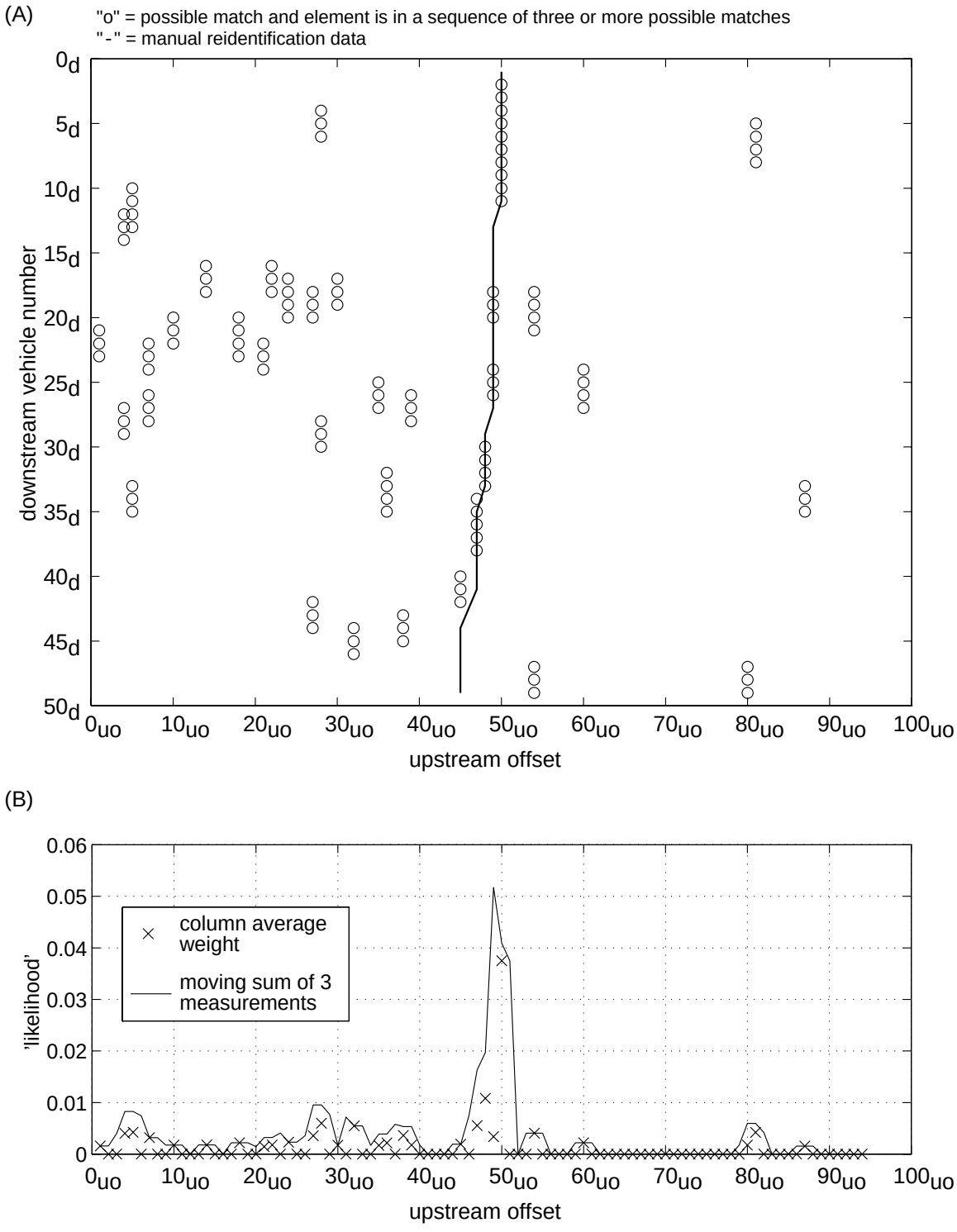


FIGURE 4-19: (A) Filtered offset match matrix, retaining all platoons of three or more vehicles, (B) Localizing the offset between the upstream and downstream by calculating the column average weights from filtered offset match matrix

center of this region and the *group offset* is defined as the upstream offset corresponding to the maximum value of the moving sum, e.g., the *group offset* is 49_{uo} for Figure 4-19B. The *group offset* should be close to the true upstream offset for each vehicle within the group (within +/- 2 vehicles for the preceding example).

This approximation should be sufficient for many applications, e.g., for stations spaced at one mile, free flow travel time will be approximately 60 seconds while the error due to missing the true match by two or three vehicles will only be a few seconds. To find exact matches, the Approximation Algorithm can be used to estimate a small range of *possible matches* for a vehicle. Then, the Basic Algorithm can be applied to this small range to find the exact match.

4.2.4 Summary

The Basic Algorithm attempts to find an exact match for every vehicle. This property is particularly desirable during congestion, when travel times are likely to change rapidly due to disturbances propagating through the traffic. Using 60 Hz speed trap data, the algorithm works well for freeway traffic moving slower than 40 mph. At higher detector sampling frequencies, it should be feasible to apply this algorithm during free flow traffic conditions as well because the length measurement resolution will be improved.

The Subsampling Algorithm works well under all traffic conditions, but it only attempts to match the trucks (up to 20 percent of all vehicles). As previously noted, this algorithm is intended for free flow conditions, when local velocity measurements at the detector stations should be representative of the entire link. The trucks serve as a good diagnostic for the onset of congestion and thus, signal the need to switch to the Basic Algorithm.

The Approximation Algorithm also works well under all traffic conditions. It attempts to find an approximate match for every vehicle, but it does not provide an exact

match. Compared to the Subsampling Algorithm, this algorithm can provide more frequent information during transitions from free flow to congestion (or vice versa) because it incorporates information from all vehicles. To find exact matches during these transitions, the Approximation Algorithm can be used to estimate a small range of *possible matches* for a vehicle. Then, the Basic Algorithm can be applied to this small range to find the exact match. When there are no transitions between the detector stations, the Basic or Subsampling Algorithm should be favored over the Approximation Algorithm because theses provide exact matches; but the Approximation Algorithm could be run in parallel to corroborate the other algorithm(s).

5. Testing and Verification

The examples presented in section 4.2 suggest that automated vehicle reidentification is possible. But proving that the algorithms work requires sufficient ground truth data to verify matches between two detector stations. Generating ground truth data is complicated by the simple fact that vehicle reidentification over extended distances is inherently difficult, both for an automated system and for a human. It is prohibitively time consuming for a human observer to generate exact matches for a large number of vehicles.

Fortunately, it is not necessary to match every vehicle manually to verify a vehicle reidentification algorithm. If the given algorithm is correctly matching vehicles, it will also yield the true travel times for those vehicles. Although travel time over a segment can change dramatically in a short period of time, the travel times for two successive vehicles will be very similar. Thus, the human observer must manually match a sufficient number of vehicles to capture changes in segment travel time, but this can be accomplished using a small fraction of the passing vehicles. Manual verification is still a labor intensive process, but now it becomes feasible to generate ground truth for significant samples.

This study used video data, recorded concurrently with the speed trap data, for manual verification. Two approaches were used to collect the video data. The first approach placed a camera at each detector station and the human observer matched vehicles between the two cameras. The second approach used a single camera view to capture both detector stations and thus, replaced the problem of matching vehicles between two video tapes with an easier task, tracking vehicles on a single tape. The single camera approach required a suitable location for camera placement and was limited to detector stations less than half a mile apart. Above this separation, it was impossible to view both stations while being able to discriminate between vehicles at the distant station.

In either case, it was necessary to synchronize the video and detector clocks before comparing travel times from an algorithm against the ground truth. Coordinating a camera's clock and a detector station's clock is fairly straightforward. Just as a sequence of vehicle lengths rapidly becomes unique when the number of vehicles increases, the sequence of vehicle headways also becomes unique. Because the video includes the same headways recorded by the detector station, the user had to note vehicle arrival times from the video relative to some arbitrary reference vehicle and then find the matching sequence of headways in the detector station data.

Using the preceding steps to generate ground truth matches and coordinate the video data with the detector data, the remainder of this chapter presents each algorithm's performance over a large set of vehicles. The first section examines free flow traffic using the Subsampling Algorithm. The next section examines congested traffic using the Basic Algorithm, while the final section considers the transition from free flow to congestion using the Approximation Algorithm.

5.1 Subsampling Algorithm verification

The Subsampling Algorithm was tested during free flow conditions over a two mile segment of State Highway 99 in Sacramento, as shown in Figure 5-1. The algorithm attempts to match trucks in the right hand lane. Although it may seem counterintuitive, these conditions are very challenging for feature based vehicle reidentification for several reasons. First, vehicles are free to overtake one another, decreasing the probability that platoons will persist over the two miles that span the detector stations. Second, entering and exiting vehicles from the three intervening ramps disrupt the sequence even more. Finally, the high velocities reduce the measurement resolution to approximately 2 feet, further complicating the task of reidentifying vehicles.

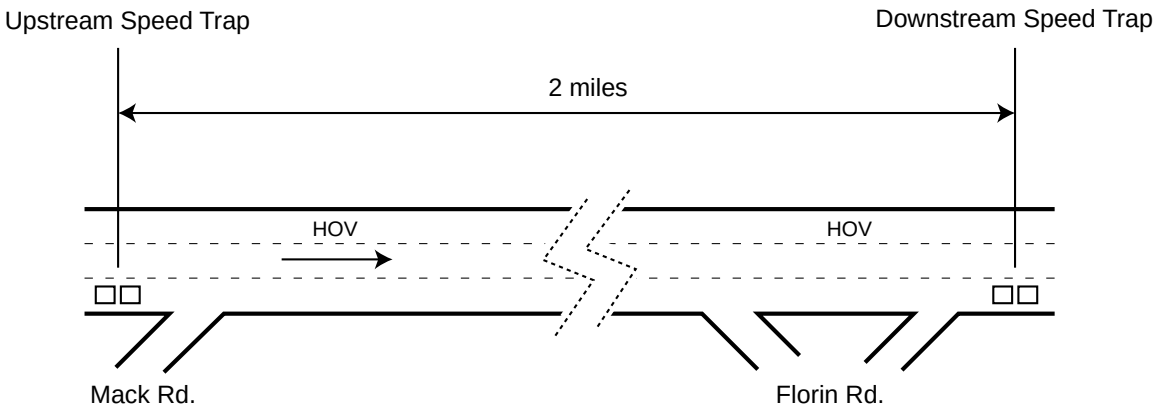


FIGURE 5-1: The segment of State Highway 99 in Sacramento, California used to verify the Subsampling Algorithm.

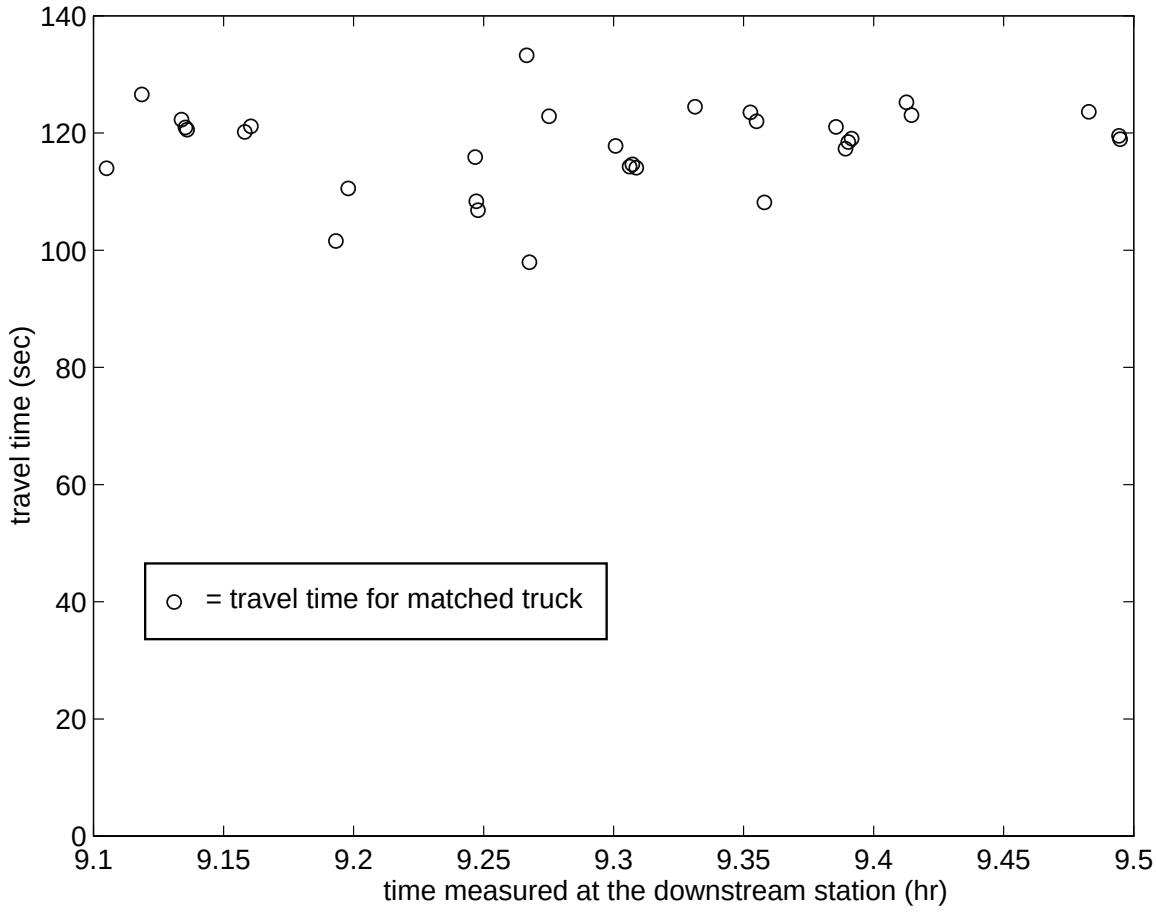


FIGURE 5-2: Travel times for matched trucks using the Subsampling Algorithm.

The algorithm matched 46 trucks, or approximately 60 percent of the trucks that passed the downstream station during the 25 minute study period; the travel time for these matches are shown with "O"'s in Figure 5-2. Travel times ranged between 100 seconds and 135 seconds, i.e., the segment velocity ranged between 53 mph and 72 mph. Using concurrent video, the human observer matched vehicles between the two stations and measured ground truth travel times; yielding the "X"'s in Figure 5-3[14]. Comparing the travel times as measured by the algorithm against the ground truth, we see a good performance by the algorithm. The average measurement error was 0.69 percent, corresponding to an average segment velocity error of 0.5 mph.

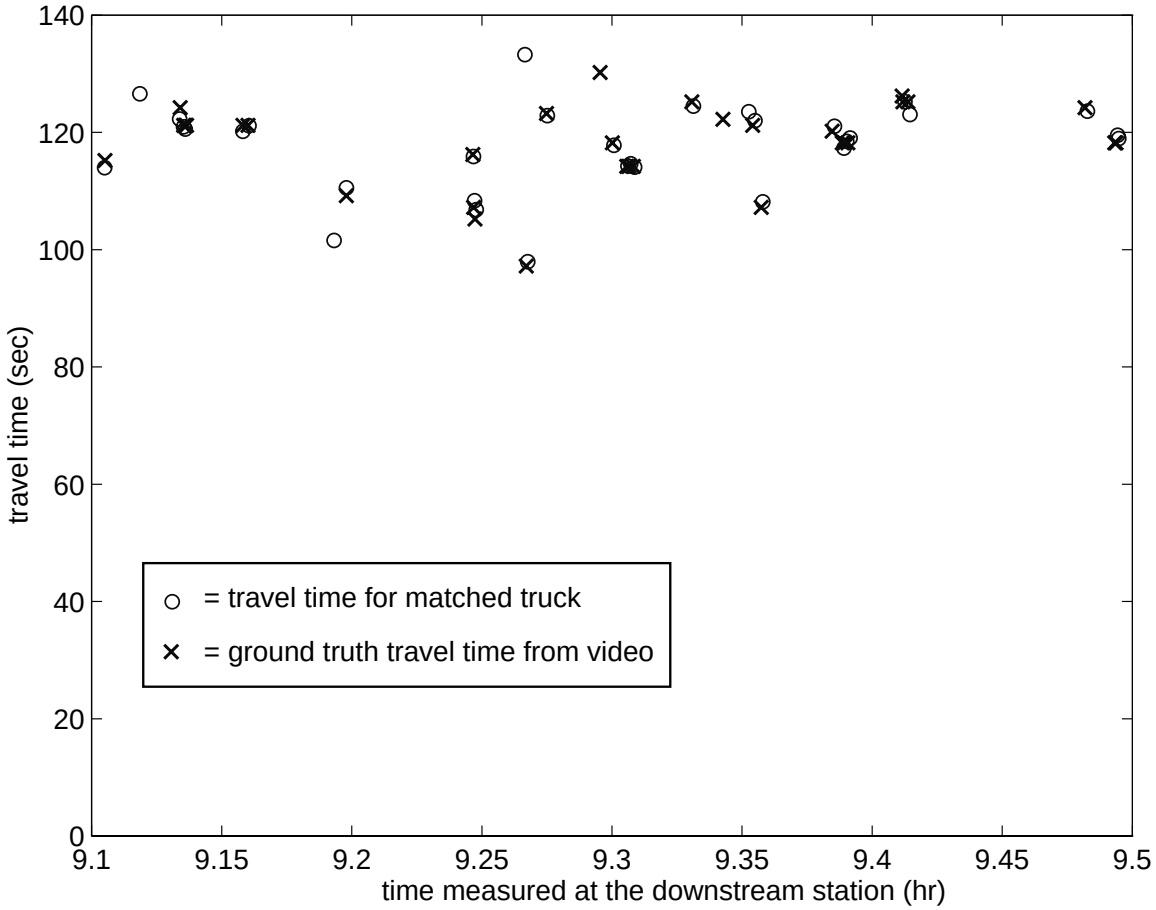


FIGURE 5-3: Comparing travel times for matched trucks using the Subsampling Algorithm against ground truth matches.

[14] Note that the set of vehicles used for ground truth is not identical with the set of vehicles matched by the algorithm because the two processes were independent.

5.2 Basic Algorithm verification

The Basic Algorithm was tested during congested conditions over a 0.35 mile segment of Interstate 80 in Berkeley. The algorithm attempts to match all vehicles in lane two, as shown in Figure 5-4. The study period was approximately 70 minutes. Velocities at the detector stations ranged between 0 mph and 40 mph for this example while twenty five disturbances passed through the segment (the average increase or decrease in travel time due to these disturbances was 28 seconds).

The solid line in Figure 5-5 shows the travel times measured by the algorithm. Travel times ranged between 50 seconds and 130 seconds, i.e., the segment velocity ranged between 9 mph and 25 mph. The algorithm matched 907 vehicles, approximately 60 percent of the vehicles that passed through the segment. The time between successive matches is typically on the order of a few seconds, with 1.5 minutes being the longest period without a reidentification in this example. Figure 5-6 compares the ground truth matches, the "X"'s, against the matches generated by the algorithm. Again, the algorithm traces the ground truth quite well; note how the algorithm follows the increasing and decreasing travel time as disturbances pass through the link. The average measurement error was 2.4 percent, corresponding to an average segment velocity error of 0.4 mph.

5.3 Approximation Algorithm verification

The Approximation Algorithm was tested during the transition from free flow to congestion over a 1.5 mile segment of Interstate 80 in Berkeley. In this test, the algorithm attempts to find an approximate match for every vehicle in lane two, as shown in Figure 5-7. Initially, velocities were on the order of 60 mph, they drop to 20 mph as a downstream queue overruns the segment and then, towards the end of the two hour sample, the velocities drop further to about 15 mph.

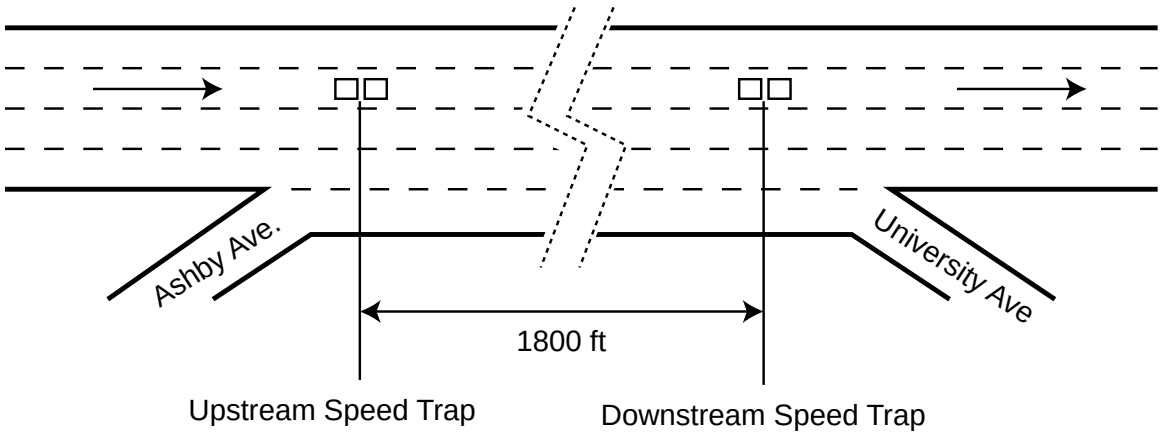


FIGURE 5-4: The segment of Interstate-80 in Berkeley, California used to verify the Basic Algorithm.

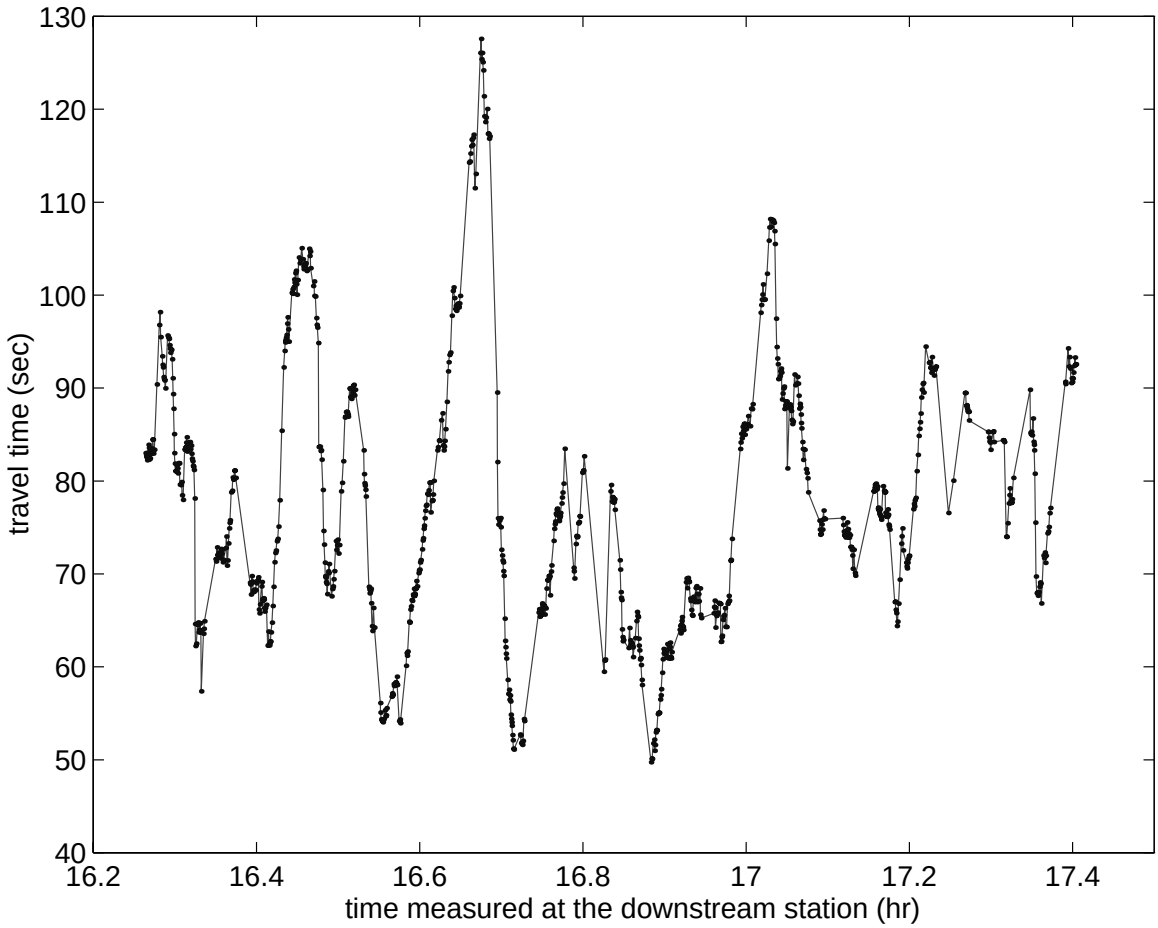


FIGURE 5-5: Travel times for matched vehicles using the Basic Algorithm.

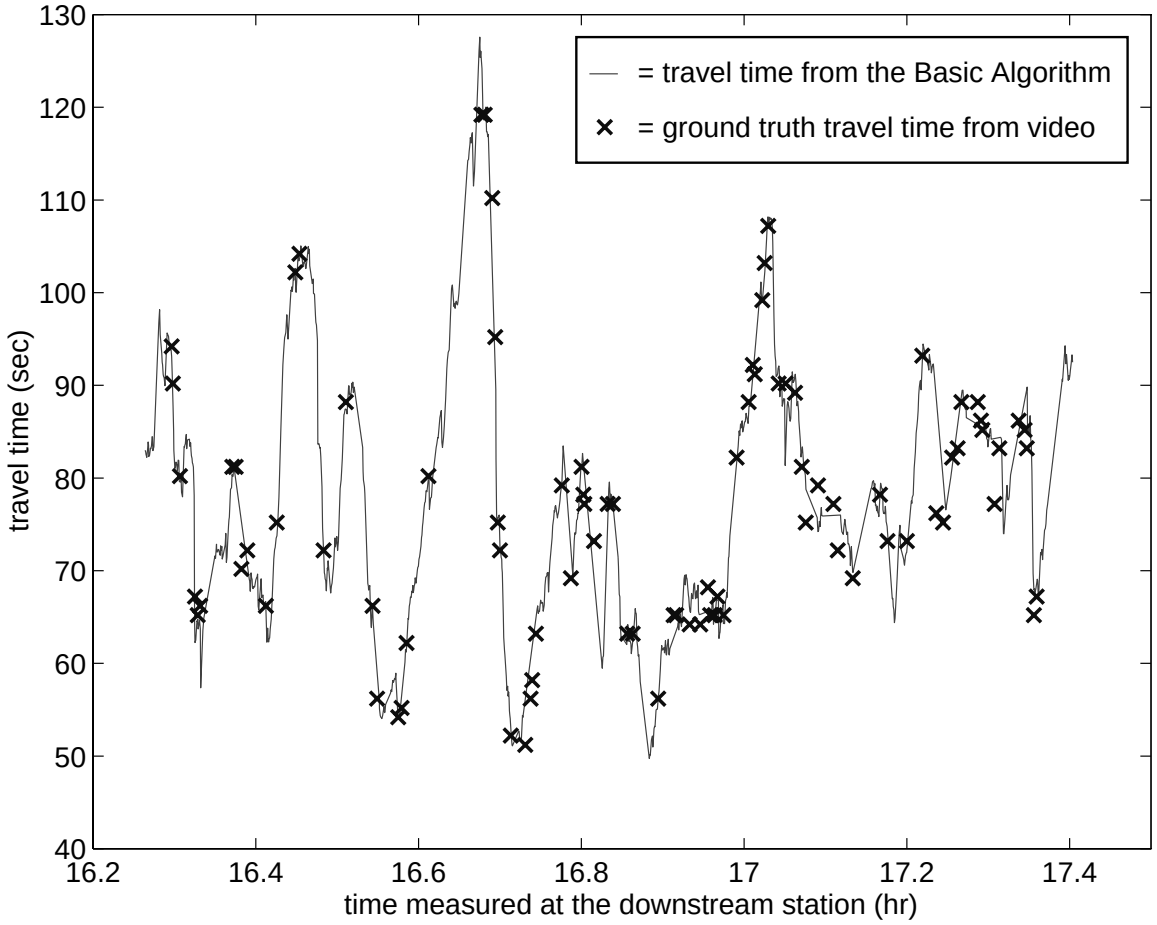


FIGURE 5-6: Comparing travel times for matched vehicles using the Basic Algorithm against ground truth matches.

The algorithm found an "approximate" match for almost all of the vehicles that passed during the study period. Travel times for these matches ranged between 70 seconds and 260 seconds, as shown in Figure 5-8, i.e., the segment velocity ranged between 18 mph and 67 mph. Comparing the algorithm against ground truth travel times, indicated with "X"'s in Figure 5-9, the average measurement error was 4.8 percent, corresponding to an average segment velocity error of 1.5 mph.

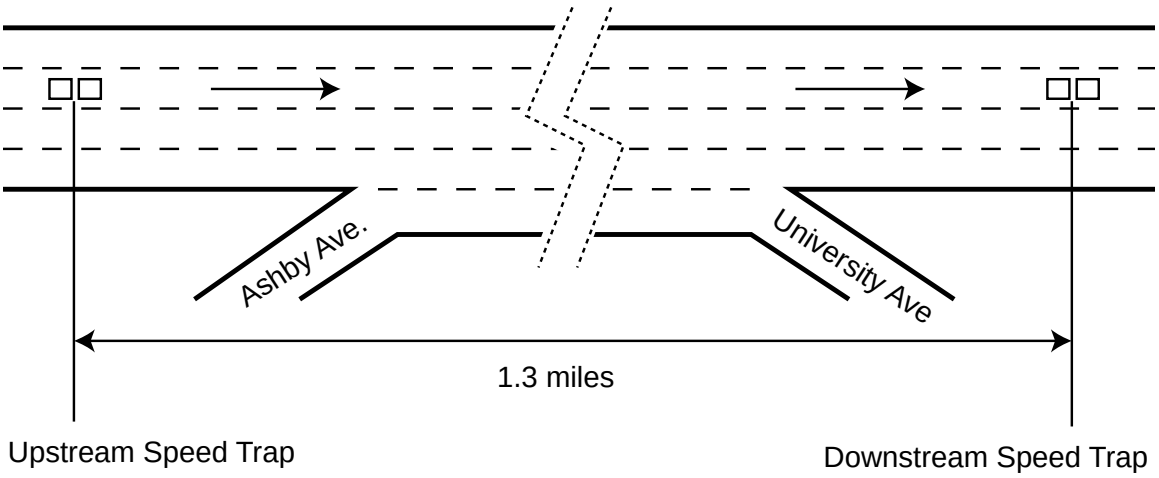


FIGURE 5-7: The segment of Interstate-80 in Berkeley, California used to verify the Approximation Algorithm.

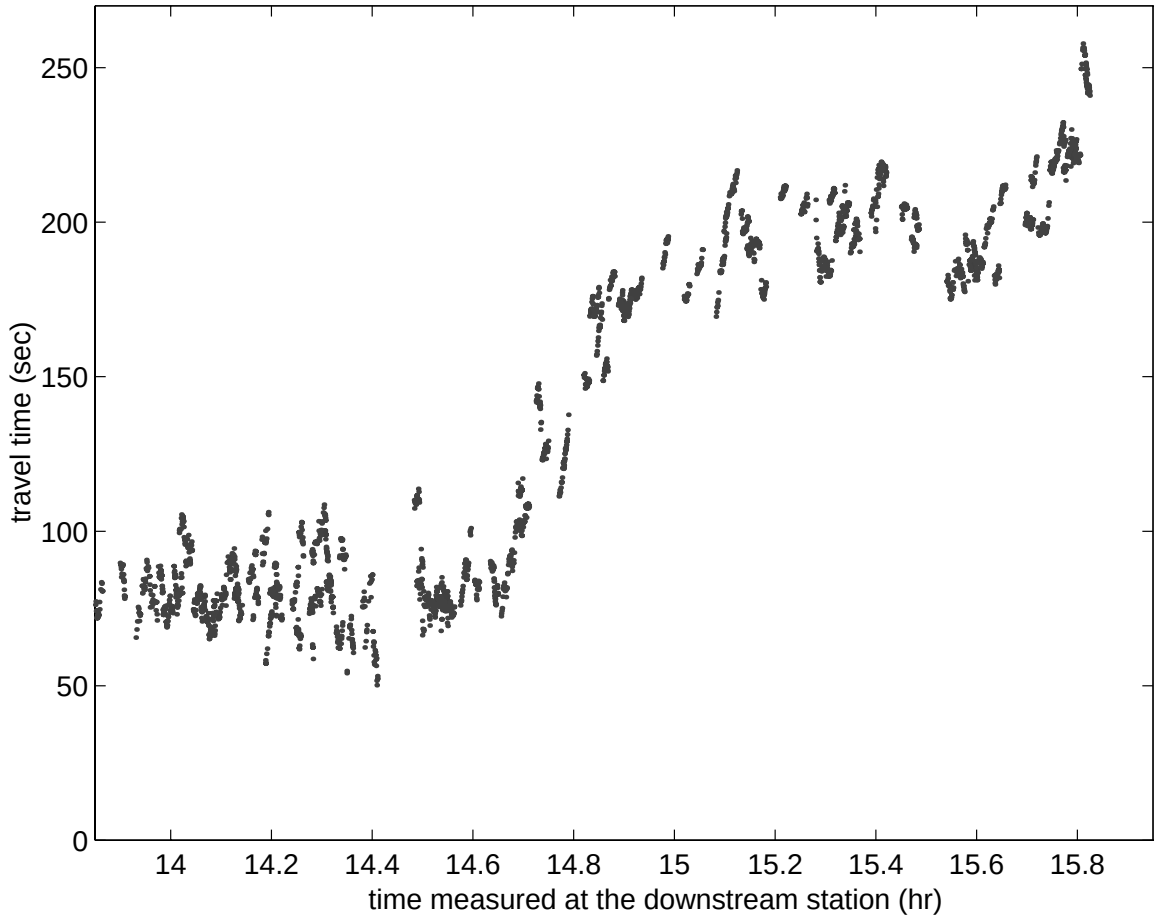


FIGURE 5-8: Travel times for matched vehicles using the Approximation Algorithm.

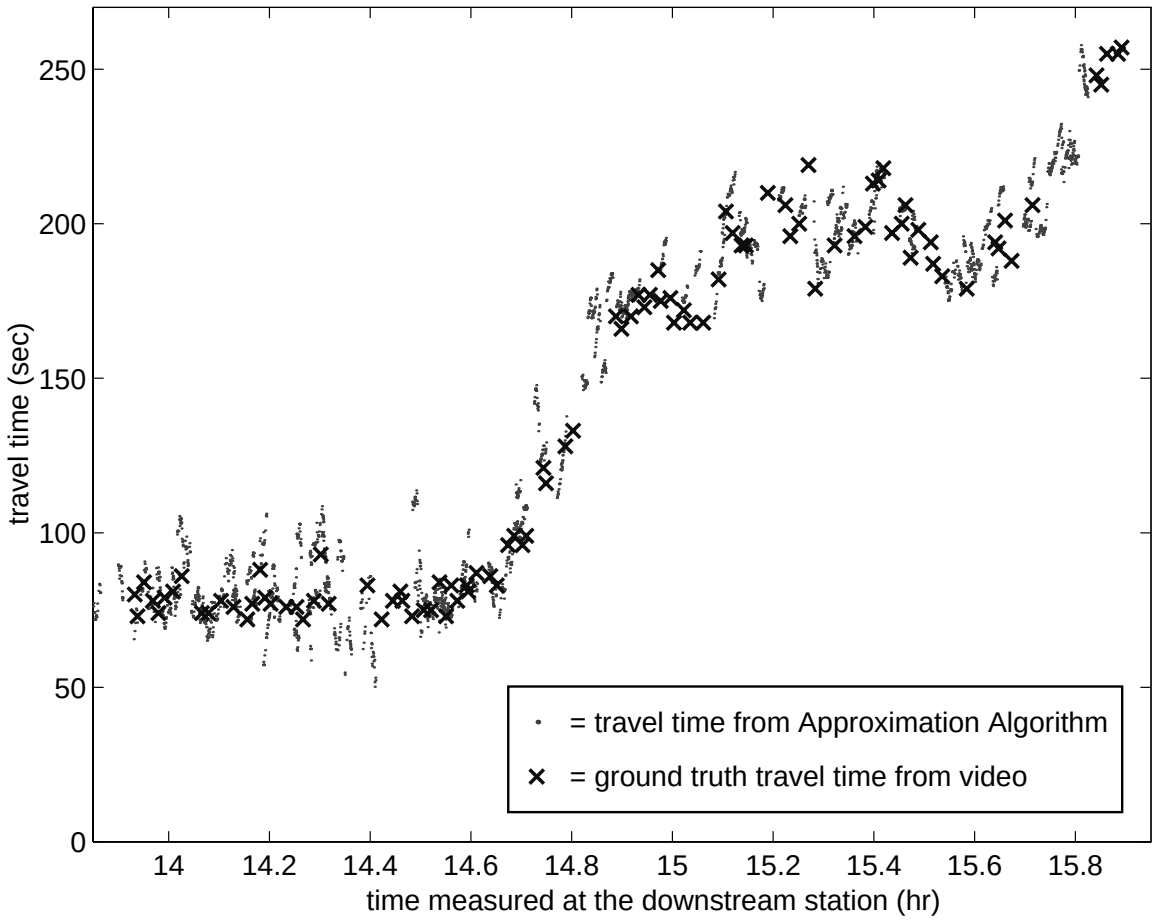


FIGURE 5-9: Comparing travel times for matched vehicles using the Approximation Algorithm against ground truth matches.

6. Extensions and Future Work

This chapter presents several extensions that could be realized through future research projects based on the dissertation work. Section 6.1 discusses the Berkeley Highway Laboratory, which will transmit high resolution speed trap data in real-time to the University of California, Berkeley, and from campus, on to Caltrans. Section 6.2 presents three emerging detector technologies that, in conjunction with the vehicle reidentification algorithms, should improve performance beyond what is possible with speed traps. Finally, section 6.3 discusses a number of applications of the new vehicle reidentification system.

6.1 Berkeley Highway Laboratory

All of the analysis in this dissertation was conducted off-line, that is, the data were collected and then the algorithms were run several hours or several days later. Although the algorithms run faster than real time, Caltrans does not currently have the communications infrastructure to transmit event data[15] in real time. As an extension to the dissertation research, work is underway to develop an inexpensive means to transmit these data using wireless modems.

The new communications infrastructure will be deployed at the Berkeley Highway Laboratory, which consists of eight detector stations along a 2.2 mile segment of Interstate 80 in Berkeley and Emeryville, as shown in Figure 6-1. The communications hardware are currently operational at two stations and should be up and running at all of the stations by early 1999.

The event data will be used as real-time input to the vehicle reidentification algorithms and the measured travel times will be available over the internet in real-time.

[15] The event data is simple the individual loop detector "events" used as input to the algorithms and discussed in Appendix B.

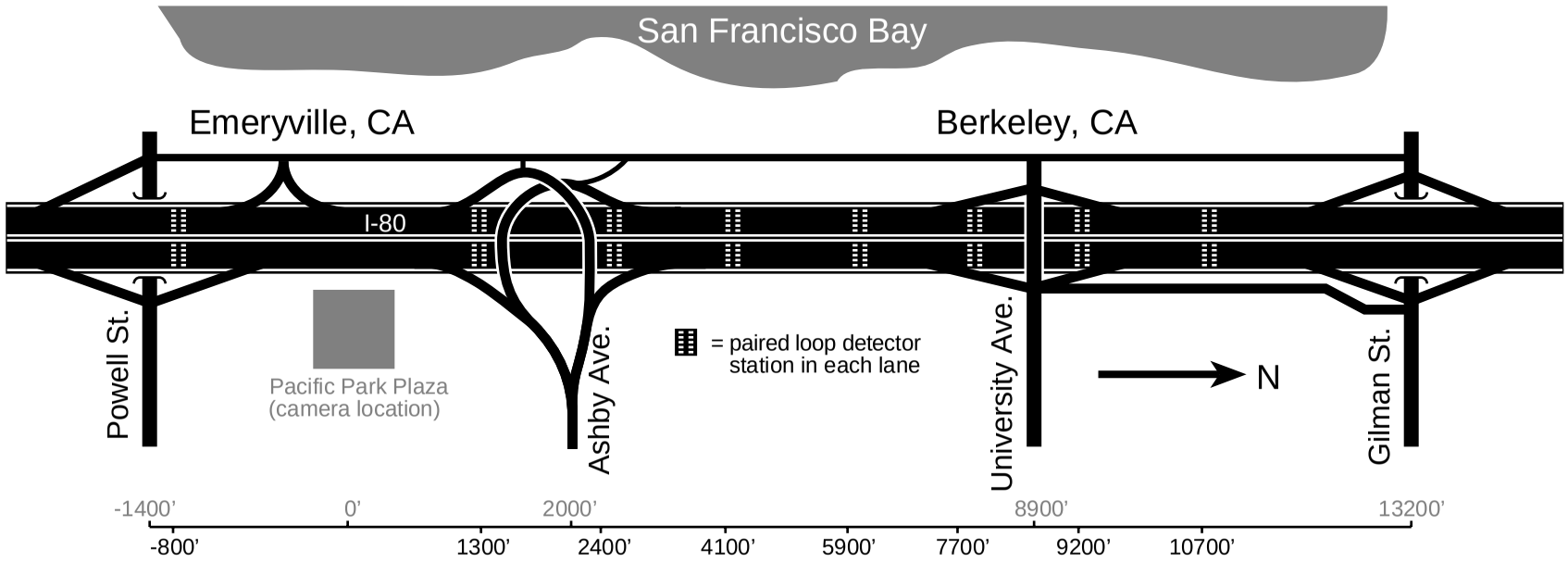


FIGURE 6-1: The Berkeley Highway Laboratory which will include eight dual loop speed trap stations and contiguous video coverage over a 2.2 mile segment of Interstate 80 in Berkeley and Emeryville, California. Video cameras will be mounted on the roof of the 30 story Pacific Park Plaza. The lower portion of the figure shows distances along the highway, dark numbers indicate detector station locations, while lighter numbers indicate the locations of the Pacific Park Plaza and four interchanges.

Travel times over one segment (i.e., between two successive stations) should be on-line before the end of 1998, with the other segments following shortly thereafter. A related project will provide video surveillance, which will be useful for generating ground truth manually and perhaps enabling automated routines for generating ground truth using video image processing to track vehicles [27, 35].

Because the vehicle reidentification algorithms will run 24 hours a day, the Berkeley Highway Laboratory will provide extensive verification of the dissertation work. In addition to manually matching vehicles to verify the algorithms, it is possible to conduct rudimentary tests using the local velocity measurements. During free flow, a vehicle's velocity as measured by its trip time over a segment should be close to its velocities measured by detectors at the upstream and downstream ends of the segment. The segment travel time will increase above the free flow travel time under two conditions: when a disturbance enters the segment by passing over one of the detector stations, or when a bottleneck forms within the segment. In either case, the associated disturbance(s) should eventually be observable in the local conditions at one or both of the detector stations.

6.2 Emerging detector technologies

Through collaboration with controller suppliers, state DOT's, and other researchers, this work could be used to improve vehicle reidentification beyond what is possible with speed traps. As noted earlier, the vehicle reidentification algorithms are compatible with several vehicle detectors. Briefly examining three emerging technologies that should be compatible with this dissertation work, first, controller suppliers are producing new hardware to extract a detailed magnetic vehicle signature. So, rather than having two detector states, "on" or "off", the controller reports a continuous response as a vehicle passes over the loop detector(s).

Next, using a video image processing system, it should be possible to extract vehicle lengths using pseudo-loops in a speed trap configuration. More importantly, the image processing system could be used to extract a multidimensional feature vector, say, color and length. The additional information could be used to make the existing routines more robust, as well as enabling new strategies such as subsampling by color and processing each color group in parallel. For example, the algorithm could process all of the green vehicles in one group, and all of the red vehicles in another.

Finally, other researchers are working to reduce vehicle length measurement uncertainty. Cheng's [25] scanning laser radar is one example. It is designed to measure vehicle length with an error of one inch at free flow traffic speeds, compared to nearly two feet using existing speed trap hardware.

6.3 Applications

After completing this dissertation, the work will be used to investigate several applications of vehicle reidentification and travel time data, including those described below.

- Incident detection strategies that include travel time measurements between detector stations might be able to decrease the time to detection without sacrificing reliability.

- Conventional volume/capacity estimates of congestion fail to account for the temporal and spatial distribution of travel. By comparing traditional congestion measures against actual delay (travel time - travel time at posted speed limit), it will be possible to quantify the performance of the old metrics and perhaps develop new metrics that are more informative measures of congestion.

- By matching individual truck measurements between many detector stations, it should be possible to generate O/D data on freight movements.

- The proponents of ATIS and DTA believe these technologies will provide significant operating improvements on the freeway network, but providing better information does not necessarily reduce congestion. Using measured travel times, it should be possible to quantify the benefits to drivers if they were aware of the most recent travel time measurements, had a perfect prediction of future travel times (i.e., if the current travel times were predicted some time earlier), or had an imperfect prediction of future travel times.

- Examine traffic dynamics in the context of the additional information available from vehicle reidentifications.

The vehicle reidentification algorithms would be used to acquire the vehicle reidentification data from speed traps, but the applications would be designed to be source independent. There are several sources of speed trap data to work with, including the forthcoming Berkeley Highway Laboratory mentioned above and the large pre-existing Freeway Service Patrol (FSP) database [44]. The FSP database contains speed trap data from 20 detector stations, as well as incident data and probe vehicle travel times over 7 miles of Interstate-880 south of Oakland, CA for 50 days.

7. Conclusions

This dissertation has presented the development of three closely related algorithms to match a vehicle's length measurement at a downstream detector station with the vehicle's corresponding measurement at an upstream station. The algorithms rule out unlikely matches and look for sequences of *possible matches* between measurements at the two stations. Each algorithm uses a different strategy to eliminate spurious sequences due to false positives. The algorithms were used to measure travel times on a large data set and the average measurement error for the different algorithms ranged between 0.7 percent and 4.5 percent, corresponding to an average segment velocity error between 0.4 mph and 1.5 mph.

The beauty of the approach is in its simplicity. Matching vehicles between detector stations is a difficult task and some of the best minds have tried to tackle the problem with varying degrees of success. Preceding work emphasized computationally intensive strategies and/or hardware intensive strategies. By creating the solution space of *possible matches*, this research has enabled vehicle reidentification using existing detector hardware and inexpensive computers.

The contribution to the field of traffic surveillance should prove to be significant since the vehicle reidentification algorithms will allow the study of travel time applications without deploying an expensive detection system and thereby enable cost-benefit analysis before investing in a new detection system. If travel time measurement proves to be beneficial, the system could be deployed using speed traps, or the algorithms could be transferred to emerging detector technologies with better measurement resolution. The methodology should prove beneficial for research purposes as well; yielding better insight into vehicle dynamics between widely spaced detector stations without the host of assumptions necessary with simulation.

8. References

[1] Palen, J. “The Need for Surveillance in Intelligent Transportation Systems”, *Intellimotion*, Vol 6, No 1, University of California PATH, Berkeley, CA, 1997, pp 1-3, 10.

[2] Lindley, J. *Quantification of Urban Freeway Congestion and Analysis of Remedial Measures*, Federal Highway Administration, 1986.

[3] Goolsby, M. E. “Influence of Incidents on Freeway Quality of Service”, *Highway Research Record Number 349*, Transportation Research Board, 1971

[4] Messer, C., Dudek C., “Method for Predicting Travel Time and Other Operational Measures in Real-Time During Freeway Incident Conditions”, *Highway Research Record 461*, Highway Research Board, 1973, pp 1-16

[5] Urbanek, G., Rogers, R. *Alternative Surveillance Concepts and Methods for Freeway Incident Management, Volume 1: Executive Summary*, Federal Highway Administration, 1978.

[6] Cambridge Systematics, *Incident Management, Executive Summary*, Trucking Research Institute, Alexandria, VA, 1990.

[7] Lin, W., Daganzo, C., “A Simple Detection Scheme for Delay-Inducing Freeway Incidents”, *Transportation Research-Part A*, Vol 31A, No 2, March, 1997, pp 141-155.

[8] Ben-Akiva, M., de Palma, A., Kaysi, I., “Dynamic Network Models and Driver Information Systems”, *Transportation Research-Part A*, Vol 25A, No 5, September 1991, pp 251-266.

[9] Kaysi, I., Ben-Akiva, M., Koutsopoulos, H., "An Integrated Approach to Vehicle Routing and Congestion Prediction for Real-Time Driver Guidance", paper presented at the Transportation Research Board, 72nd Annual Meeting, 1993.

[10] Ben-Akiva, M., Cascetta, E., Whittaker, J., Cantarella, G., de Ruiter, J., Kroes, E., "Real-Time Prediction of Traffic Congestion", *Proc. of the Third International Conference on Vehicle Navigation and Information Systems*, IEEE, 1992, pp 557-562.

[11] Ben-Akiva, M., Cascetta, E., Gunn, H., Smulders, S., Whittaker, J., "DYNA: A Real-Time Monitoring and Prediction System for Inter-Urban Motorways", *Proc. of the First World Congress on Applications of Transportation Telematics and Intelligent Vehicle-Highway Systems*, Vol 3, ERTICO, 1994, pp 1166-1180.

[12] Ben-Akiva, M., Cascetta, E., Gunn, H., "An On-Line Dynamic Traffic Prediction Model for and Inter-Urban Motorway Network", *Urban Traffic Networks - Dynamic Flow Modeling and Control*, Springer, 1995, pp 83-122.

[13] Mahmassani, H., Peeta, S. "System Optimal Dynamic Assignment for Electronic Route Guidance in a Congested Traffic Network", *Urban Traffic Networks - Dynamic Flow Modeling and Control*, Springer, 1995, pp 3-37.

[14] Hobeika, A., Kim, C., "Traffic-Flow-Prediction Systems Based on Upstream Traffic", *Proc. of the 1994 Vehicle Navigation and Information Systems Conference*, IEEE, 1994, pp 345-350.

[15] Wild, D., "Pattern-Based Forecasting", *Proceedings of the Second DRIVE-II Workshop on Short Term Traffic Forecasting*, TNO Institute for Policy Studies, Delft, The Netherlands, 1994, pp 49-63.

[16] Madanat, S., Krogmeier, J., Hu, S., "An Enhanced Kalman Filtering Algorithm for Dynamic Freeway OD Matrix Estimation and Prediction", *Proc. of the Fourth International Conference on the Applications of Advanced Technologies in Transportation Engineering*, ASCE 1995, pp 423-428.

[17] Hoffman, G., Janko, J., "Travel Times as a Basic Part of the LISB Guidance Strategy", *Proc of the Third International Conference on Road Traffic Control*, IEE, 1990, pp 6-10.

[18] Uerlings, U., "The Prediction System within the Socrates Information Centre", *Proceedings of the Second DRIVE-II Workshop on Short Term Traffic Forecasting*, TNO Institute for Policy Studies, Delft, The Netherlands, 1994, pp 27-47.

[19] Cassidy, M., "Unique Bivariate Relationships in Highway Traffic", paper presented at the Transportation Research Board, 75th Annual Meeting, 1996.

[20] Arnott, R., de Palma, A., Lindsey, R., "Does Providing Information to Drivers Reduce Traffic Congestion?", *Transportation Research-Part A*, Vol 25A, No 5, September 1991, pp 309-318.

[21] Schrank, D. L., Turner, S. M., Lomax, T. J., *Urban Roadway Congestion - 1982 to 1992*, Texas Transportation Institute, Vol 1-2, 1995.

[22] Nagel, K., Stretz, P., Pieck, M., Leckey, S., Donnelly, R., Barrett, C. "TRANSIMS Traffic Flow Characteristics", paper presented at the 77th annual TRB meeting, Transportation Research Board, 1998.

[23] Ogden, K., "Urban Freight Policy", *ITS Review*, August 1992, University of California, pp 2-3, 8.

[24] Kühne, R., Palen, J., Gardner, C., Ritchie, S. "Section-Related Measures of Traffic System Performance", paper presented at the 76th annual TRB meeting, Transportation Research Board, 1997.

[25] Cheng, H., et al, *A Real-Time Laser-Based Prototype Detection System for Measurement of Delineations of Moving Vehicles*. PATH, University of California, Berkeley, CA, 1998.

[26] MacCarley, C. A., *Videobased Vehicle Signature Analysis and Tracking Phase 1: Verification of Concept and Preliminary Testing*, PATH, University of California, Berkeley, CA, 1998.

[27] Coifman, B., Beymer, D., McLauchlan, P., and Malik, J. "A Real-Time Computer Vision System for Traffic Surveillance and Vehicle Tracking", [in publication], *Transportation Research-Part C*, 1999.

[28] Levine, S., McCasland W., "Monitoring Freeway Traffic Conditions with Automatic Vehicle Identification Systems", *ITE Journal*, Vol 64, No 3, March 1994, pp 23-28.

[29] Balke, K., Ullman, G., McCasland, W., Mountain, C., and Dudek, C. *Benefits of Real-Time Travel Information in Houston, Texas*, Southwest Region University Transportation Center, Texas Transportation Institute, College Station, TX, 1995.

[30] Christiansen, I, Hauer, L. "Probing for Travel Time: Norway Applies AVI and WIM Technologies for Section Probe Data", *Traffic Technology International*, Aug/Sep 1996, UK & International Press, Surrey, UK, 1996, pp 41-44.

[31] Oda, T, Takeuchi, K. and Niikura, S. "Travel Time Measurement Using Infrared Vehicle Detectors," *Proc of the 8th International Conference on Road Traffic Monitoring and Control*, Report 422, 1996, pp 178-182.

[32] Seki, S. "Travel-Time Measurement and Provision System Using AVI Units", *Proc. 2nd World Congress on Intelligent Transport Systems*, VERTIS, 1995, pp 50-55

[33] Tanaka, Y., Nishimura, F. "Multiroute Travel-Time Data Provision Systems Operating in Osaka", *Proc. 1994 Vehicle Navigation and Information Systems*, IEEE, 1994, pp 351-356.

[34] Cui, Y., Huang, Q. "Character Extraction of License Plates from Video" *Proc. 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 1997, pp 502-507.

[35] Malik, J., et. al., *Traffic Surveillance and Detection Technology Development: New Traffic Sensor Technology Final Report*, University of California PATH, Berkeley, CA, 1997.

[36] Pfannerstill, E. "A Pattern Recognition System for the Re-identification of Motor Vehicles", *Proc. 7th International Conference on Pattern Recognition*, Montreal, IEEE, New Jersey, 1984, pp 553-555

[37] Kühne, R., Immes, S. "Freeway Control Systems for Using Section-Related Traffic Variable Detection" *Pacific Rim TransTech Conference Proc., Vol 1*, 1993, ASCE, pp 56-62

[38] Reijmers, J. "On-Line Vehicle Classification", *Proceedings of the International Symposium on Traffic Control Systems*, Vol 2B, Institute of Transportation Studies, University of California at Berkeley, 1979, pp 87-102

[39] Dailey, D. "Travel Time Estimation Using Cross Correlation Techniques," *Transportation Research-Part B*, Vol 27B, No 2, 1993, pp 97-107.

[40] Petty, K., et. al. "Accurate Estimation of Travel Times From Single Loop Detectors", paper presented at the 76th annual TRB meeting, Transportation Research Board, 1997.

[41] Westerman, M., Immers, L. "A Method for Determining Real-Time Travel Times on Motorways," *Road Transport Informatics/Intelligent Vehicle Highways Systems*, ISATA, 1992, pp 221-228.

[42] Westerman, M., Litjens, R., Linnartz, J. *Integration of Probe Vehicle and Induction Loop Data- Estimation of Travel Times and Automatic Incident Detection*. PATH, University of California at Berkeley, 1996.

[43] Nam, D., Drew, D. "Traffic Dynamics: Method for Estimating Freeway Travel Times in Real Time from Flow Measurements", *Journal of Transportation Engineering*, Vol 122, No 3, May/June 1996, pp 185-191.

[44] Skabardonis, A., et al, *Freeway Service Patrol Evaluation*. University of California PATH, Berkeley, CA, 1995.

[45] Windover, J., *Empirical Studies of the Dynamic Features of Freeway Traffic*, Dissertation, University of California, 1998.

9. Appendix A

9.1 Implementation

This section details the steps used to realize each algorithm. The initial steps are the same for all of the algorithms. First, each vehicle is processed as it passes a single detector station. Second, for a given vehicle at the downstream detector station, a range of feasible upstream matches is established; i.e., all of the upstream measurements that may have come from the same vehicle are identified. These steps are presented in subsections 9.1.1 - 9.1.2. Next, each algorithm uses a slightly different strategy to go through the feasible upstream matches and identify the final matches. The algorithm specific steps are detailed in subsections 9.1.3 - 9.1.5, where each subsection corresponds to the Basic, Subsampling and Approximation Algorithm, respectively. Although the different algorithms are presented separately, the common steps make it simple to run two or more algorithms in parallel with the same input data.

9.1.1 Common steps for each vehicle at a single detector station

This subsection details the analysis applied to each vehicle that passes a detector station, independent of any other station. Each detector station used in this study has a speed trap in each lane, where the speed trap consists of two loop detectors spaced 20 ft apart.

1) Vehicles are assigned successive arrival numbers as they pass a detector station. The numbers in one lane are assigned independently from the other lanes and the numbers are not directly related to the arrival numbers recorded at any other station. If a vehicle only activates a single loop of a dual loop speed trap, it is not included in the numbering sequence and the vehicle is excluded from further analysis. These discarded detections

typically account for less than one percent of all vehicles. Appendix B provides more information on how unmatched and erroneous pulses are detected.

2) A vehicle's arrival time, velocity, effective length and length uncertainty, as defined in Appendix C, are recorded as it passes a speed trap. The effective length ± half of the length uncertainty bound the *length range* for the vehicle.

9.1.2 Common steps for each vehicle at the downstream detector station

The preceding steps are applied independently at two consecutive detector stations. Starting from step 3, the algorithms use data from the same lane at both stations.

3) The *last feasible upstream match* is identified for a downstream vehicle using the distance between the detector stations, vehicle arrival times at the two successive detector stations and an assumed maximum possible speed of 100 mph. To illustrate this process, consider two stations 1470 feet apart. A vehicle arriving at the downstream station traveling less that 100 mph (147 ft/sec) must have passed the upstream station at least 10 seconds earlier. So, the last vehicle to pass the upstream station in the same lane, at least 10 seconds earlier is considered the *last feasible upstream match* for the given downstream vehicle.

4) A *set of reasonable upstream matches* is identified for each downstream vehicle; where this set is the last n successive upstream vehicles in the same lane ending with the *last feasible upstream match*, The constant, n, should be set large enough to ensure that the true match will always fall within in the set of reasonable upstream matches, while being small enough to allow the computer to process the data. For the examples presented in chapter 5, n was set arbitrarily to 100 vehicles; then, after running the algorithms, it was verified that the true matches were always within 100 vehicles of the *last feasible*

upstream match. In practice, a conservative value of n could be set from estimated jam density and the distance between stations.

9.1.3 Basic Algorithm

The following steps are used to implement the Basic Algorithm and they are specific to this algorithm. To emphasize this algorithm specificity, the steps are indicated with a "b" for "Basic". To implement the algorithm, steps 5b-12b are repeated for each vehicle as it passes the downstream station.

5b) The downstream vehicle's *length range* is compared against the *length range* for each vehicle in the *set of reasonable upstream matches*. For each pair-wise comparison, if the two ranges intersect, the pair is a *possible match*, otherwise, a match is unlikely. The results are stored in a row vector with "1" indicating a *possible match* and "0" indicating that a match was unlikely. Finally, the upstream offset (as defined in subsection 4.2.3) is calculated for each vehicle in the *set of reasonable upstream matches*.

6b) The row vector is placed in a *sequence matrix* where each row is indexed by the downstream vehicle number and the columns are indexed by the upstream offset[16]. For each *possible match*, the value from the corresponding column in the previous row is added (if it exists). Thus, the row stores the total number of consecutive *possible matches* in a sequence up to and including the given downstream vehicle for each upstream offset.

7b) The same row vector is placed in the *lane change matrix* using the indices from step 6b. Again, for each *possible match*, the value from the corresponding column in the previous row is added (if it exists). Each new sequence in the current row of the *lane*

[16] This step deviates from the notation used in subsection 4.2.1. As previously noted, sequences will fall into columns in this coordinate system and thus, it is easier to write computer code to manipulate the matrices compared to working with the non-shifted matrices. This shift has the added benefit of decreasing matrix width.

change matrix (i.e., those of value 1) is examined to see if it can be joined to an earlier sequence in the *sequence matrix* by a simple lane change maneuver, as shown in Figure 9-1A-C[17]. To demonstrate this process using the sequence starting at (m,n) in Figure 9-1D, the three shaded elements of the *sequence matrix* are searched for earlier sequences. If any of these elements contain a value greater than one, the largest value is copied to element (m,n) in the *lane change matrix*. Thus, the row indicates the total

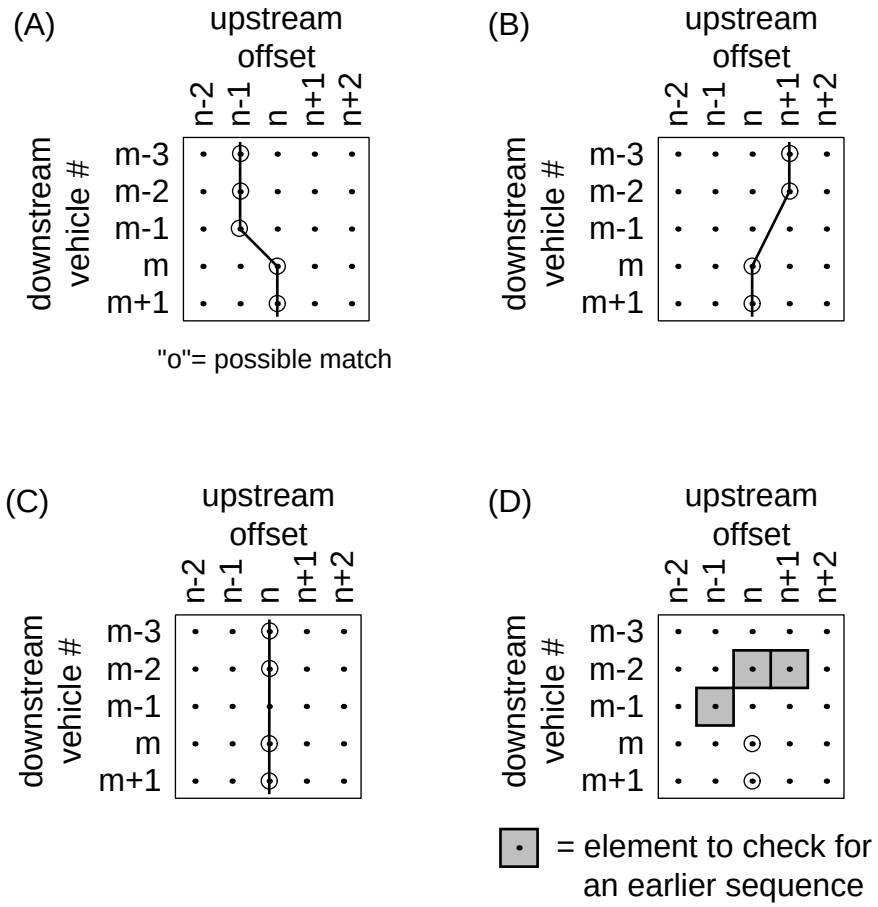


FIGURE 9-1: A simple example illustrating the possible lane change maneuvers recognized by the Basic Algorithm: (A) One vehicle exits the lane between stations, (B) One vehicle enters the lane between stations, (C) One vehicle enters and one vehicle exits the lane between stations, (D) The search region for the sequence starting at element (m,n).

[17] This figure simply shows the lane change maneuvers from Figure 4-10 transposed to the new coordinate system.

number of consecutive *possible matches* for each upstream offset up to and including the given downstream vehicle after allowing for individual lane change maneuvers.

Note that by using values from the *sequence matrix* for the pre-lane change data, the totals in the *lane change matrix* include, at most, one lane change maneuver. For sequences that include a lane change maneuver, the *sequence matrix* contains the portion of the sequence before the lane change and the *lane change matrix* contains the portion of the sequence after the lane change. In this fashion, one *possible match* can be included in several different sequences in the *lane change matrix*. Finally, in the event a sequence does not include a lane change maneuver, both matrices will store the same values for the sequence.

8b) Any sequences that end in the previous row are identified, as exemplified in Figure 9-2A. For each *possible match* in the previous row, r-1 in this example, the current row, r, is checked to see of there is a *possible match* in the same column. If there is no corresponding match in the current row, the sequence has ended. In the example, the sequence in column s has ended at (r-1,s) because (r,s) does not contain a *possible match*. However, at this point in the analysis, it is impossible to determine if the sequence in column s-2 has ended at row r since we do not know if there will be a *possible match* in (r+1,s-2) until vehicle r+1 arrives at the downstream station.

9b) All of the sequences that ended in the previous row and contained a lane change maneuver are identified. Figure 9-2B shows a simple example illustrating this process with a *sequence matrix* on the left and the corresponding *lane change matrix* on the right. If a given sequence includes a lane change maneuver, the final value in the *lane change matrix* will be higher than the corresponding position in the *sequence matrix*, otherwise the two values will be equal[18]. In the example, two sequences end in row r-1. Examining

[18] The reader should note that if a sequence ends in the *lane change matrix*, by definition, it must end in the *sequence matrix* as well.

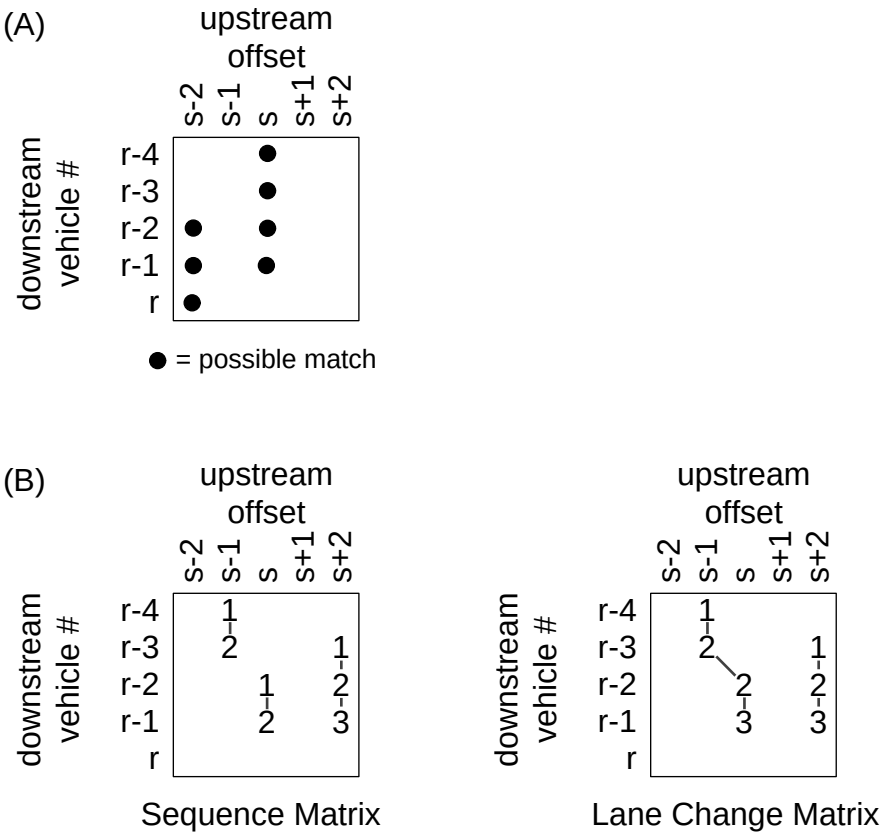


FIGURE 9-2: (A) The end of the sequence in column s can only be detected after the first unlikely element has been observed in the column. In this case once vehicle r passes the downstream station. (B) This figure shows a simple example of how to differentiate between sequences in the lane change matrix that do not contain a lane change from those that do. If the sequence does not contain a lane change, the ending value will be identical to the sequence matrix (e.g., element (r-1,s+2)), otherwise, it will be greater (e.g., element (r-1,s)).

the *lane change matrix*, the first sequence includes a lane change and ends with element (r-1,s). Comparing the value in (r-1,s) between the two matrices, we see that the *lane change matrix* has a higher value. The second sequence, ending at (r-1,s+2), does not include a lane change and thus, the values in (r-1,s+2) are identical for both matrices.

10b) Once a sequence ends, all of the elements in that sequence are set equal to the sequence length, as illustrated in Figure 9-3. Part A shows a *sequence matrix* after the b-th vehicle passes the downstream station. Allowing for individual lane change

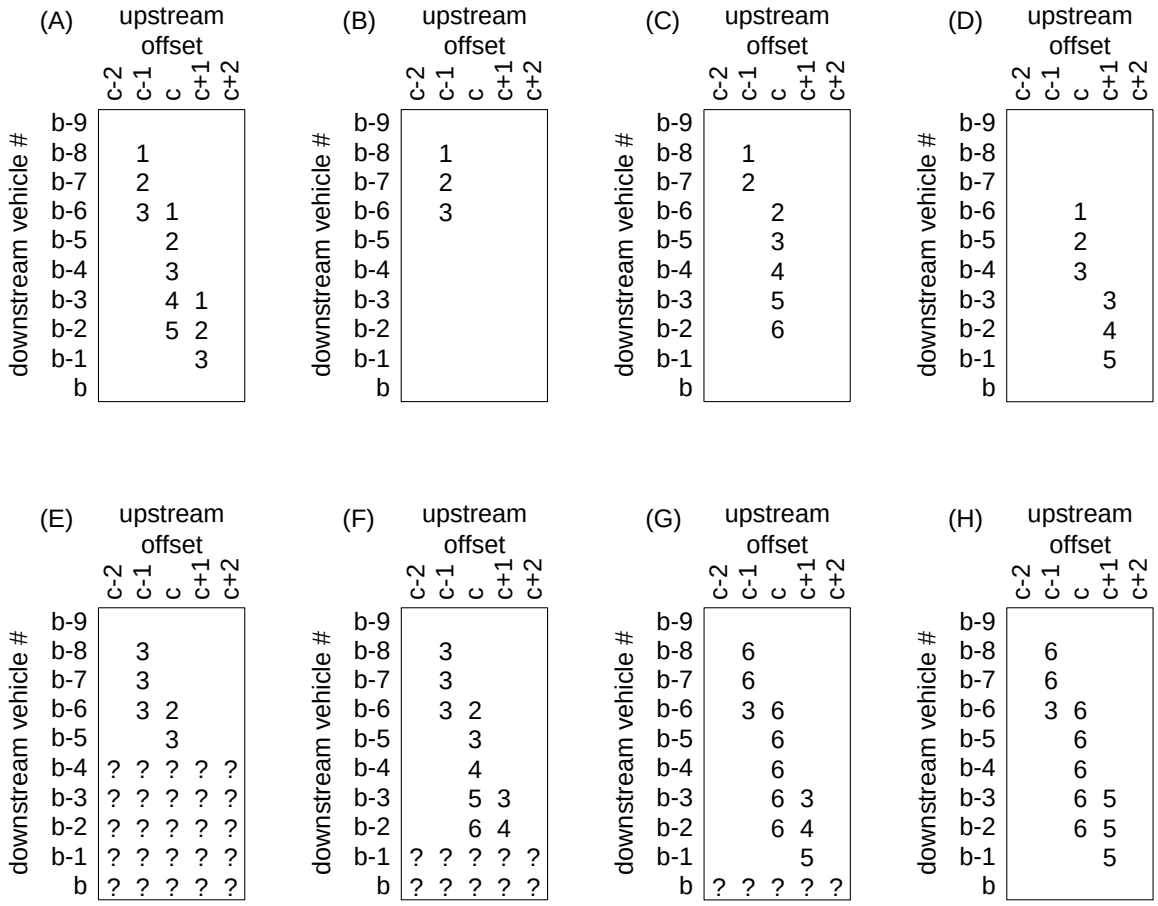


FIGURE 9-3: (A) A simple sequence matrix containing three sequences. After allowing for individual lane change maneuvers, three sequences emerge, as shown in (B)-(D). In practice, the lane change matrix would evolve as vehicles pass the downstream station, as shown in (E)-(H). (E) This is the lane change matrix immediately after vehicle b-5 passes. The end of the first sequence has been detected and all of its elements have been set equal to its length of 3. (F) The same matrix after vehicle b-2 passes, two sequences were extended by this vehicle. Note how the pre-lane change portion of the third sequence (from part D) is obscured by the second sequence (from part C). (G) After vehicle b-1 passes, the end of the second sequence is detected. All appropriate elements are set equal to the sequence length of 6. (G) Finally, after vehicle b passes, the end of the third sequence is detected and again, the sequence length is passed back to earlier elements in the sequence. This time, however, the pre-lane change matches are not changed because they have already been assigned to a sequence of greater weight in part G.

maneuvers, three overlapping sequences would be recorded in the *lane change matrix*. Each of these sequences are shown one at a time in parts B-D. Note that the second sequence (part C) overlaps a portion of the first sequence (part B) and the third sequence (part D) overlaps a portion of the second sequence.

By definition, the last element of a sequence contains the sequence length, while the earlier elements will contain lower values. After a sequence ends, the algorithm searches the *lane change matrix* and finds all of the elements in the sequence. For a sequence with t matches and no lane change maneuver, the sequence will simply be the preceding t elements in the *lane change matrix* and all of these elements are set equal to t. This situation is demonstrated in part E, where the end of the first sequence has been detected and all of its elements have been set equal to its length of 3. If the sequence contains a lane change, the algorithm finds the post-lane change portion by successively stepping back one row at a time in the same column until it finds an element with a value of zero (note that all elements with value zero are left blank in the figures). All of the non-zero elements are set equal to the sequence length, e.g., in part H, all of the post-lane change elements in the third sequence (column c+1) have all been set to 5. Using the same logic from step 7b, the algorithm identifies the pre-lane change portion of the sequence by examining three preceding elements in the *sequence matrix*, as illustrated in Figure 9-1D[19]. Any pre-lane change elements in the *lane change matrix* with values lower than the current sequence length are set equal to this new value, as illustrated by elements (b-7,c-1) and (b-8,c-1) of the second sequence in part G (compare the values of these elements to what they held in part F). However, the pre-lane change elements may already have a higher value, in which case, they will not be changed, as illustrated by elements (b-4,c) to (b-6,c) of the third sequence in part H.

[19] In the event there are two or three possible lane change maneuvers with the maximum value, the algorithm will follow all of the maneuvers that correspond to the highest value.

11b) The active rows of the *lane change matrix* are selected, i.e., all rows containing a sequence that could be extended by subsequent vehicle arrivals are identified. To this end, the algorithm only needs to find the longest sequence that may be modified by subsequent vehicle arrivals. Figure 9-4 is used to illustrate this process. The longest sequence will usually correspond to the highest value in the current row of the *lane change matrix*, as shown in row d of part A. However, there may be a lane change maneuver that skips the current row of the *lane change matrix* (e.g., Figure 9-1B-C) and a sequence ending in the preceding row, d-1, of the *sequence matrix* could be joined to a new sequence starting in row d+1. In the former example, if the next downstream vehicle yields a *possible match* in element (d+1,e), then the entire sequence will be extended and all rows from d-3 onward will be affected. This figure shows the worst case, where the sequence contains a lane change that skips a row, thus, the highest value in row d is two less than the number of active rows. In the latter example, since vehicle d+1 has not arrived yet, the algorithm must consider the largest value in the preceding row of the *sequence matrix* as well. In this case, the value will always be one less than the number of active rows since the *sequence matrix* can not contain a lane change at this point and

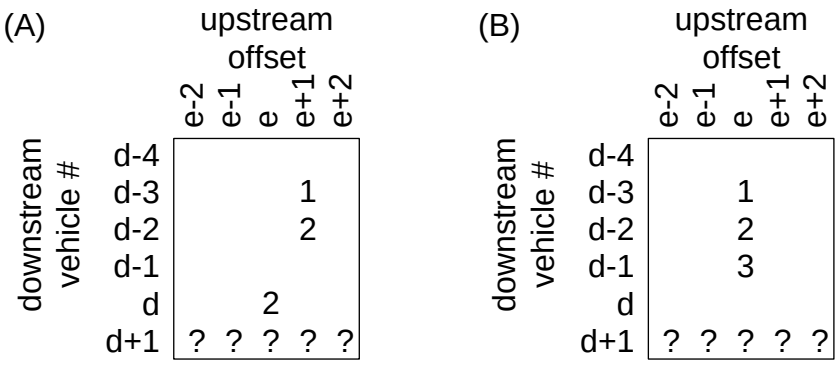


FIGURE 9-4: (A) A simple lane change matrix. (B) A simple sequence matrix

the value does not include the d-th vehicle. In either case, row d-4 can not be affected by vehicle d+1 or any subsequent vehicles. Thus, row d-4 is inactive. In summary, given:

x = the largest value in the current row of the *lane change matrix*

y = the largest value in the preceding row of the *sequence matrix*

z = max(x+2,y+1)

the active rows are the z most recent rows and all preceding rows are inactive.

12b) Final matches are extracted from any inactive row in the *lane change matrix* on a row by row basis, as follows: first, the element with the largest value in the given row is found. If two or more elements contain the largest value, the row is deleted without a match[20]. Next, if the value is less than the pre-specified minimum final sequence length, the row is deleted without a match. Otherwise, the downstream vehicle number and the upstream offset of the element are saved as a final match, then the row is deleted. After deleting a row in the *lane change matrix*, the corresponding row is deleted from the *sequence matrix*. Note that by deleting rows after they have been processed, the matrices are kept small.

9.1.4 Subsampling Algorithm

The Subsampling Algorithm explicitly identifies distinct vehicles that are easier to identify, i.e., the long vehicles. All other vehicles are excluded from the analysis. Then, this algorithm applies the Basic Algorithm to the long vehicles. Continuing the emphasis on algorithm specificity, the steps are indicated with a “s” for “Subsampling”. To implement the algorithm, steps 5s, 7s-8s are repeated for each vehicle as it passes the downstream station, while step 6s is repeated for each vehicle as it passes the upstream station.

[20] When deleting a row, the indices for all other rows are preserved.

5s) All downstream vehicles longer than a threshold length are identified or subsampled. This threshold was set to 23 feet for the example shown in section 5.1. These vehicles are assigned a new set of sequential arrival numbers based on their order in the subsample. The algorithm does not attempt to find matches for any vehicles shorter than the threshold length; thus, steps 3-4 may be omitted for the vehicles excluded from the subsample.

6s) All upstream vehicles that are longer than the threshold length or that have a *length range* that includes the threshold length are subsampled. Similar to step 5s, these vehicles are assigned a new set of sequential arrival numbers based on their order in the subsample.

7s) All upstream vehicles that were not subsampled are removed from the *set of reasonable upstream matches*. Using the resolution test, these removed vehicles would not yield any *possible matches* with the subsampled downstream vehicles.

8s) Indexing rows and columns by the subsample arrival numbers, rather than the arrival numbers for the entire population, the algorithm applies the Basic Algorithm (steps 5b-12b) to the subsamples. Note that the upstream offset in step 6b is calculated with respect to the subsample arrival numbers.

9.1.5 Approximation Algorithm

The Approximation Algorithm is implemented in two parts. First, steps 5a-10a are repeated as each vehicle passes the downstream detector station; note that the suffix “a” is used to denote the steps specific to the Approximation Algorithm. After a fixed number of vehicles pass the downstream station, steps 11a-15a are applied to a large number of downstream vehicles and the downstream vehicle count is reset to zero.

5a) Identical to step 5b of the Basic Algorithm.

6a) Identical to step 6b of the Basic Algorithm.

7a) The Approximation Algorithm counts the number of *possible matches*, n, in the *set of reasonable upstream matches*. The results are stored in a second row vector with 1/n indicating a *possible match* and "0" indicating that a match was unlikely. Thus, the fewer *possible matches*, the greater the weight assigned to each match. Finally, the new row vector is placed in an *offset match matrix* where each row is indexed by the downstream vehicle number and the columns are indexed by the upstream offset.

8a) Identical to step 8b of the Basic Algorithm.

9a) Out of the sequences that ended in the previous row of the *sequence matrix*, any sequence that is shorter than a threshold number of vehicles[21] is eliminated from the *offset match matrix*. In other words, the elements of the *offset match matrix* corresponding to the short sequences are set equal to zero.

10a) Using R to denote the threshold number of vehicles from the last step, the *sequence matrix* must contain enough downstream vehicles to differentiate between sequences shorter than R and those that are not. So, the *sequence matrix* only needs to store the most recent R rows. All preceding rows are inactive and they are discarded from the *sequence matrix* (note that the inactive rows are not eliminated from the *offset match matrix* in this step).

11a) After M vehicles pass the downstream station, the algorithm calculates the average value over the most recent 2*M inactive rows[22] for each column in the *offset match*

[21] This threshold number was set to 3 for the example in section 5.3.

[22] For the example in section 5.3, M was set to 20.

matrix. These averages are placed in a row vector indexed by upstream offset, and the oldest M inactive rows are discarded from the *offset match matrix*.

12a) As previously noted in subsection 4.2.3, a large average indicate that the true matches likely resided in the given column for some portion of the 2*M downstream vehicles; because of lane changing and detector errors, several adjacent columns will typically have large averages. A moving sum of three elements is used to find the center of this region.

13a) The group offset for the 2*M downstream vehicles is defined as the upstream offset corresponding to the maximum value of the moving sum.

14a) If the group offset is measured correctly, it should be similar from one group to the next. In the extreme case where there were no lane changes or misdetections, the true group offset would be constant across groups.

Unfortunately, for uncommon vehicles, step 7a may yield false positives with large weights and occasionally these false positives are not eliminated in step 9a. These false positives may be large enough to disrupt step 13a and the algorithm will calculate a false group offset for the 2*M vehicles. These errors will be random, the false group offset has an equal probability of occurring at any upstream offset within the row vector of average weights.

To eliminate these errors, the current group offset is compared to the group offset for the four preceding, non-overlapping groups[23] as follows. If the current group is number 0 and the preceding groups are numbered 1 through 4, where group 1 is most

[23] Note that the groups contain 2*M downstream vehicles, but a new group offset is measured every M vehicles and two successive groups overlap by M vehicles.

recent and group 4 is the oldest; the algorithm calculates the following four parameters, i.e., the $t_i's$:

$$t_i = \begin{cases} 1, & |G_0 - G_i| \le i \times (2 \times M) \times p \\ 0, & \text{otherwise} \end{cases}$$

Where,

i = group number

G_i = the group offset for the i-th group

p = assumed maximum percentage of vehicles that may change lanes between detector stations, set to 30 percent for the example in section 5.3

$(2 \times M) \times p$ = maximum allowable difference between two successive group offsets.

Then, the current group offset is accepted if,

$$\sum_{i=1}^{4} t_i \ge 3$$

and rejected otherwise. Thus, the current measure must be similar to three out of the four preceding, non-overlapping values of group offset to be accepted.

15a) The algorithm calculates the final matches for each downstream vehicle. As noted in step 14a, each set of M vehicles contribute to two overlapping groups of 2*M vehicles, where each group has its own group offset. For a given set of M vehicles, both group offsets will usually be accepted by step 14a. So in this case, the offset for each vehicle is set equal to the average of the two group offsets for the overlapping groups.

A false positive with large weight may disrupt two overlapping groups, but the vehicles that only fall into one of the two groups will have a second measured group

offset that does not include the false positive. So, for a given set of M vehicles, if one group offset is accepted and the other rejected by the previous step, the offset for each of the M vehicles is set equal to the group offset for the accepted group. On the other hand, if both group offsets are rejected, then the M downstream vehicles are not matched. This redundancy increases the number of vehicles matched in the presence of false positives.

10. Appendix B

10.1 Speed trap data from one detector station

The following section provides a brief review of speed trap operation. A given speed trap records vehicle arrival and departure times from each loop, as shown in Figure 10-1. Typically, these data are aggregated to calculate flow, occupancy and average velocity over a fixed observation period. For this study, however, each vehicle was treated independently. From the four events indicated in Figure 10-1B, t_{RISE_up}, t_{FALL_up}, t_{RISE_down}, t_{FALL_down}, the following parameters were calculated: travel time via the rising edges (TT_r), travel time via the falling edges (TT_f), total time the upstream detector is on (OT_u) and total time the downstream detector is on (OT_d). Specifically:

$$TT_r = t_{RISE\_down} - t_{RISE\_up}$$
$$TT_f = t_{FALL\_down} - t_{FALL\_up}$$
$$OT_u = t_{FALL\_up} - t_{RISE\_up}$$
$$OT_d = t_{FALL\_down} - t_{RISE\_down}$$

10.2 Loop errors at an individual speed trap

Loop detectors are prone to frequent errors. Two common errors prevent simple vehicle length estimation, as described below. First, an unmatched event at one loop (e.g., two consecutive rising edges when the events should alternate between rising and falling). This error is illustrated in Figure 10-2 and occurred approximately once for every 10,000 vehicles in the data set. To address this error, if n consecutive rising edges were recorded from a given loop, the first n-1 rising edges were discarded (n=2 in the example). Likewise, if m consecutive falling edges were recorded from a given loop, the last m-1 falling edges were discarded.

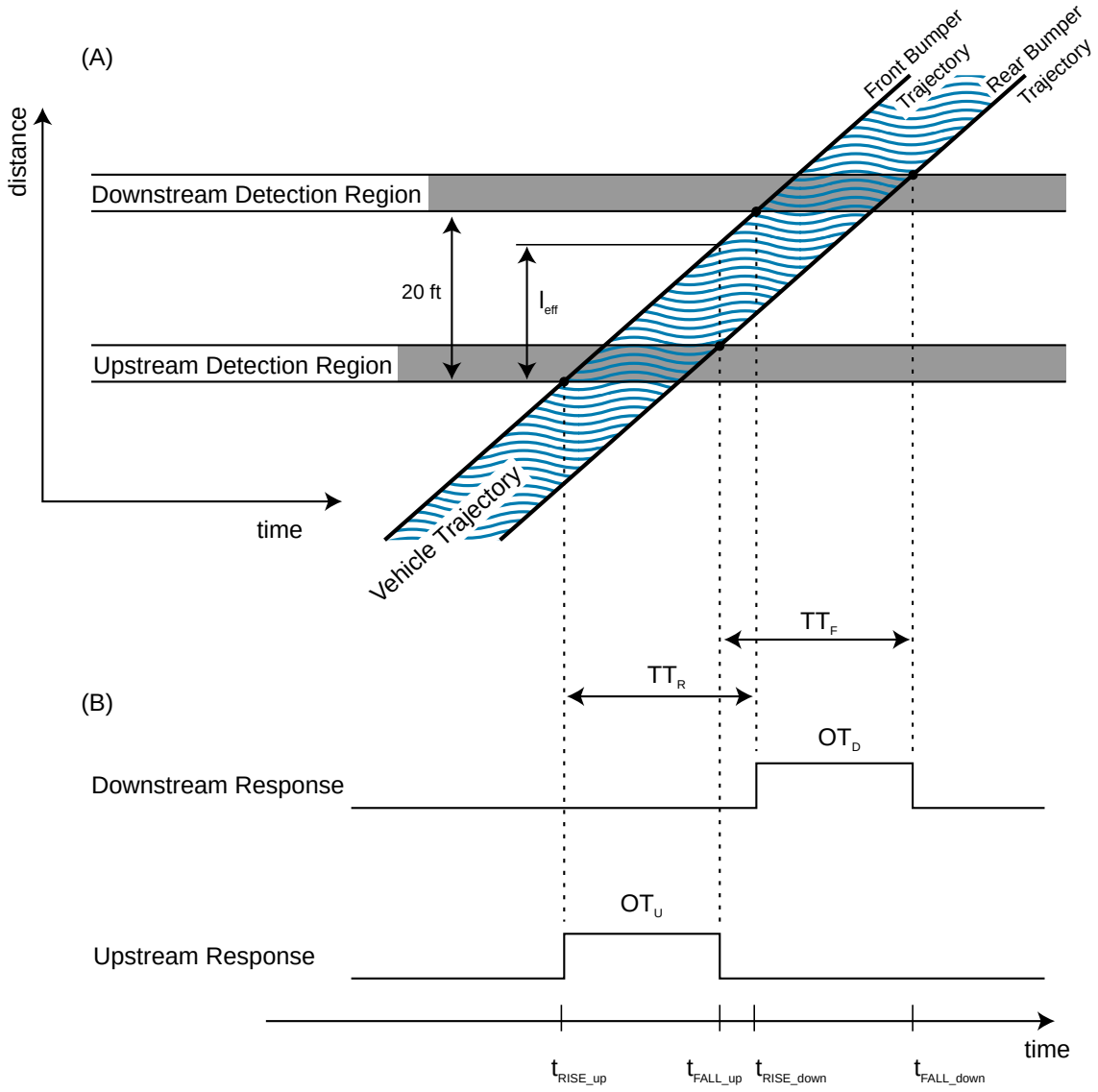


FIGURE 10-1: One vehicle passing over a speed trap, illustrating the four time measurements. (A) Time space representation showing the loop detectors and both ends of the vehicle. (B) Detector output, yielding the upstream and downstream, rising and falling edges at the indicated times.

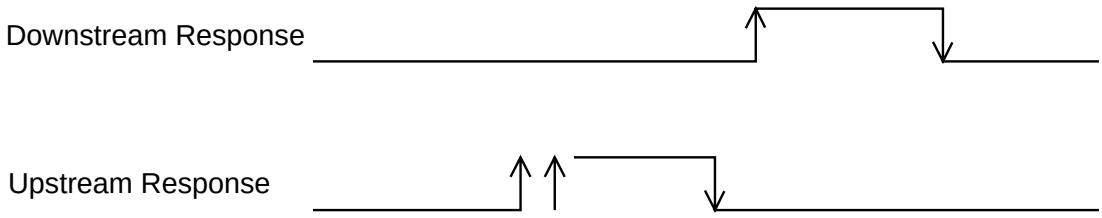


FIGURE 10-2: An example of an unmatched event at one detector, in this case, a rising edge at the upstream detector.

The second, more common error, occurred when two pulses were observed at one detector, but only one pulse was observed at the other detector. These unmatched pulses occur when a vehicle is only detected at a single loop, as illustrated in Figures 10-3A and 10-4A, or when a vehicle activates one loop more than once, as illustrated in Figures 10-3B and 10-4B. Rather than attempting to discriminate between the two sources of error, all questionable pulses were removed, (i.e., all pulses within the dashed circles in Figures 10-3 and 10-4). Following this removal, the modified data would suggest a vehicle changed lanes when in fact it did not. Provided these phantom lane changes were relatively infrequent, they will not disrupt the vehicle reidentification algorithms. This latter type of error occurred approximately once in every 100 vehicles for this study, which is sufficiently low that they did not interfere with the algorithms.

After eliminating these errors, it is possible to establish a one-to-one match between events at the upstream and downstream loops, and thus, match pulses directly. There are other loop errors that are less significant for vehicle reidentification, e.g., missing a vehicle altogether or simultaneously observing a vehicle in two adjacent lanes. These errors do not preclude vehicle length estimation but they will create noise when attempting to match vehicles between stations.

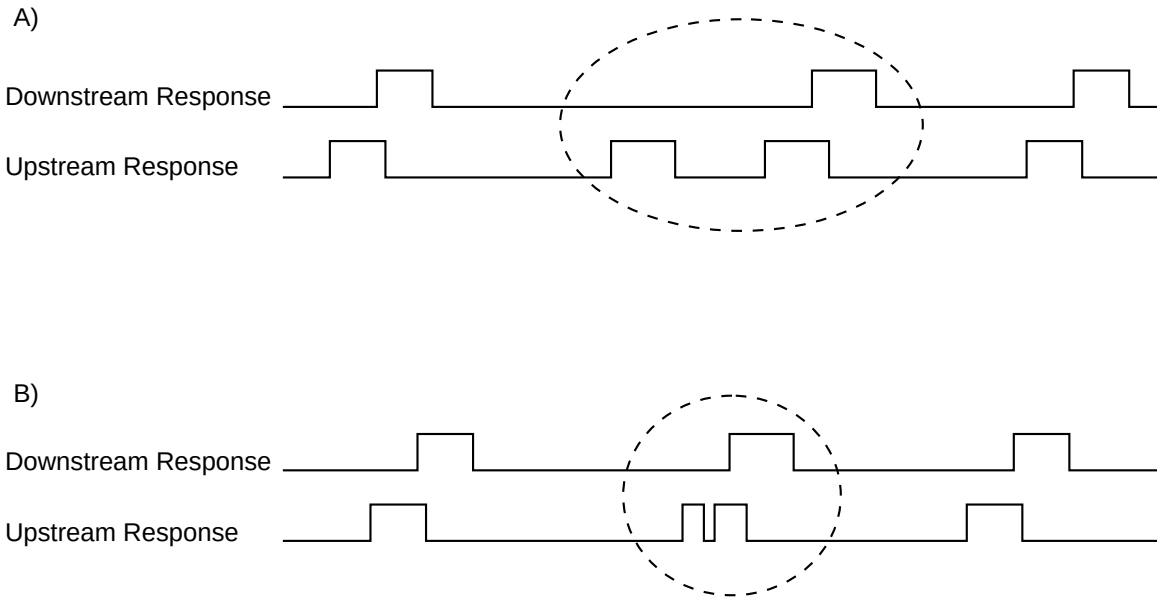


FIGURE 10-3: Two examples of unmatched pulses at the upstream loop, (A) four vehicles activate the upstream loop, but only three activate the downstream loop, (B) a vehicle activates the upstream loop twice but it only activates the downstream loop once.

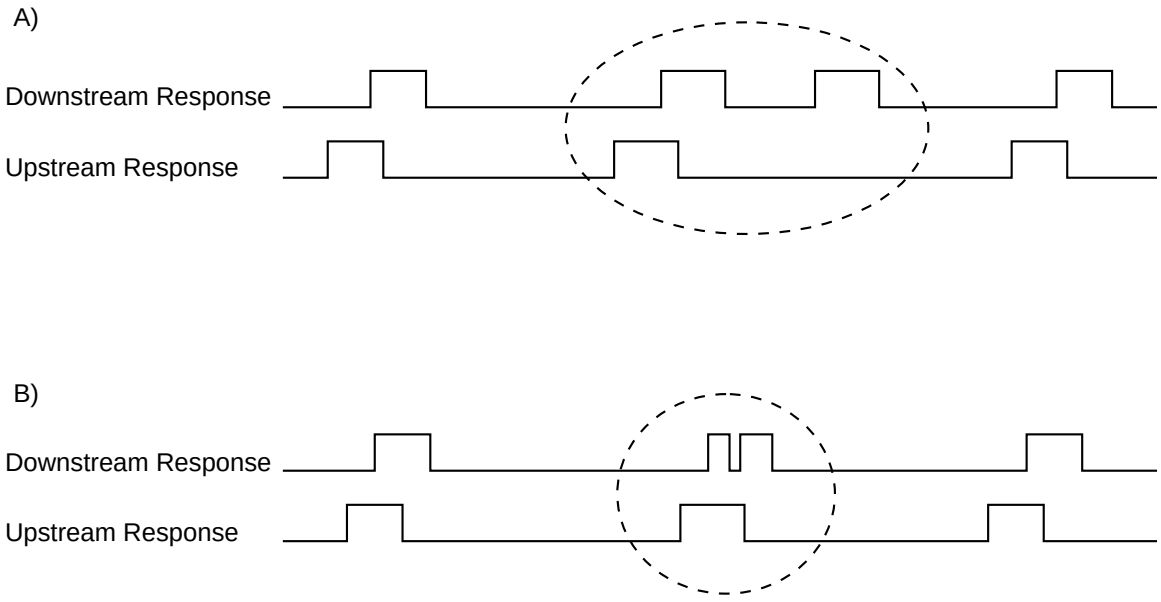


FIGURE 10-4: Two examples of unmatched pulses at the downstream loop, (A) four vehicles activate the downstream loop, but only three activate the upstream loop, (B) a vehicle activates the downstream loop twice but it only activates the upstream loop once.

11. Appendix C

11.1 Vehicle parameter measurement

Once the individual speed trap data were cleaned up by matching upstream and downstream pulses, as per Appendix B, it was possible to measure a vehicle's effective length, L, and the associated measurement uncertainty, L_{err}. Effective length is simply the measured velocity multiplied by the time the detector was on, i.e., the on-time. In practice, for each vehicle that passes the speed trap, there are two measurements of on-time, one from each loop, and two measurements of velocity , one for the front bumper (using the difference between the time each loop is activated) and one for the rear (using the difference between the time each loop clears).

velocity from rising edge: $$V_r = \frac{20}{TT_r} \quad \left[\frac{\text{ft}}{\text{sec}}\right];$$

velocity from falling edge: $$V_f = \frac{20}{TT_f} \quad \left[\frac{\text{ft}}{\text{sec}}\right];$$

where 20 (ft) represents the loop separation, i.e., the spacing between corresponding points on the two loops.

These measurements are used to calculate two estimates of vehicle length: the first uses the front bumper velocity and the upstream loop on-time, while the second uses the rear bumper velocity and the downstream loop on-time.

length measurement #1: $$L_1 = V_r \cdot OT_u \quad [\text{ft}];$$

length measurement #2: $$L_2 = V_f \cdot OT_d \quad [\text{ft}];$$

The logic for pairing the given on-time with the given velocity measurement is as follows: for a short vehicle, such as a sedan, the effective length is on the order of the

spacing between the two loops in a speed trap. Thus, the period that the upstream loop is occupied is roughly concurrent with the time that the front bumper velocity is measured over the speed trap. Similarly, the period that the downstream loop is occupied is roughly concurrent with the time that the rear bumper velocity is measured over the speed trap. For longer vehicles, the period a loop is occupied includes the time of the respective velocity measurement, but exceeds the duration.

The average of the two length measurements is recorded as the effective vehicle length.

effective length: $$L = \frac{L_1 + L_2}{2};$$

Next, three constraints are used to estimate the length uncertainty for the vehicle. First, the difference between the two length measurements yields a length based constraint.

constraint #1: $$C_1 = \max(L_1, L_2) - \min(L_1, L_2) \quad [\text{ft}];$$

Second, the controller samples at 60 Hz, so time measurements are accurate to 1/60th of a second at the detector station. Thus, as discussed in section 4.1, the length resolution degrades as velocity increases. The algorithm generates a velocity based length resolution constraint from the larger velocity measurement

constraint #2: $$C_2 = mL \cdot \max(V_r, V_f) \quad [\text{ft}];$$

where,

mL = minimum allowable L_{err} per 1 ft/sec; currently set to 0.017 sec.

Third, at low speeds, constraint #2 is too restrictive. To counter this problem, a prespecified minimum measurement uncertainty provides the final constraint:

$$\text{constraint \#3:} \qquad C_3 = \min\left(\max\left(dL_{20}, \frac{(L-20)\cdot\left(dL_{80} - dL_{20}\right)}{60} + dL_{20} \right), dL_{80} \right) \quad [\text{ft}];$$

where,

dL_{20} = total minimum allowable L_{err} for a vehicle under 20 ft; currently set to 1 ft,
dL_{80} = total minimum allowable L_{err} for a vehicle over 80 ft; currently set to 10 ft.

Note that the minimum measurement uncertainty increases linearly with vehicle length for vehicles between 20 ft and 80 ft. This increase is to account for two factors; first, the fact that longer vehicles tend to have higher suspensions, increasing the separation between the vehicle underframe and the roadway or loop detectors. The larger separation increases the chance that a loop will "clear" prematurely for these long vehicles. Second, the preceding analysis assumes the vehicle travels at a constant velocity as it passes over the speed trap. By ignoring the possibility of acceleration, there will be some errors in the length measurement. The magnitude of this error increases as vehicle length increases because the time the vehicle occupies the detector increases.

The largest of the three constraints on the length resolution was used as the length uncertainty for the given vehicle, i.e.,

$$\text{length uncertainty:} \qquad L_{err} = \max\left(C_1, C_2, C_3\right) \quad [\text{ft}].$$

Finally, the length range was bounded by:

$$\text{maximum reasonable length:} \qquad L_{\max} = L + 0.5 \cdot L_{err}$$

$$\text{minimum reasonable length:} \qquad L_{\min} = L - 0.5 \cdot L_{err}.$$