

Modelit
Elisabethdreef 5
4101 KN Culemborg



info@modelit.nl
www.modelit.nl

Feasibility of AID algorithms based on Acoustic Sensing applied to the A9

Date November 26, 2022
Author Nanne van der Zijpp
Modelit
Commissioned by Rijkswaterstaat

Content

1	Management summary	1
1.1	Exploration and Verification of OptaSense data.....	1
1.1.1	Usability of the congestion indicator for AID purposes .	1
1.1.2	Usability of the speed indicator for AID purposes	1
1.2	Inspection of OptaSense speed data in more detail	1
1.3	Bayesian Framework for estimating congestion state	2
1.4	Comparing the A58 and A9 results.....	3
2	Exploration and Verification of OptaSense data	5
2.1	Introduction.....	5
2.2	Historizing OptaSense data	5
2.3	Correcting the congestion status	5
2.4	Compare congestion status prior and post update.....	6
2.4.1	Prior to update	7
2.4.2	Post update.....	7
2.5	Visual inspection of the OptaSense congestion indicator	8
2.6	Usability of the congestion indicator for AID purposes	8
2.7	Usability of the speed indicator for AID purposes	9
2.8	Conclusion	12
2.8.1	Usability of the congestion indicator for AID purposes	12
2.8.2	Usability of the speed indicator for AID purposes	12
3	Inspection of OptaSense speed data in more detail.....	13
3.1	Piecewise constant property.....	13
3.2	Possible explanations for the low data refresh rate.....	15
3.3	Implication for fiber based AID algorithms	16
3.4	Patterns in speed innovations.....	16
3.4.1	Pseudo trajectories	16
3.4.2	Initial missing data echo pseudo trajectories	17
3.4.3	Pseudo trajectories do not propagate through congestion	17
3.5	Third indication for congestion: capped pseudo trajectories.....	17
3.6	Detailed inspection – conclusions.....	18
4	Bayesian Framework for estimating congestion state	20
4.1	Introduction	20
4.2	Elements of a Bayesian approach	20
4.2.1	Some notation.....	20
4.2.2	Elements of a Bayesian approach.....	21
4.2.3	Measurement update	21
4.2.4	Time extrapolation	22
4.2.5	Point estimate	22
4.3	Application of the Bayesian framework to AID	22
4.3.1	Measurement update	23
4.3.2	Time extrapolation	24
4.3.3	Point estimate	26
4.4	Application.....	26
4.4.1	Matlab code	26
4.4.2	Tuning the parameters	26

4.4.3	Time range of data	28
4.4.4	Applied methods.....	28
4.5	Results.....	29
4.5.1	In depth analysis of single congestion event.....	29
4.5.2	Full period analysis	33
4.5.3	Congestion code comparison before and after system update	35
4.6	Conclusion	36
5	Comparing the A58 and A9 results	37
6	Introduction	37
6.1	Selected conclusions of the A58 study	37
6.2	Comparison between A58 and A9 results	38
6.3	Visualizations of the A58 data	38
6.4	Possible causes for found differences	41
6.4.1	Possibility 1: difference of scale of the analysis.....	41
6.4.2	Possibility 2: difference in detection accuracy	42
6.4.3	Possibility 3: Calibration	42
6.4.4	Possibility 4: Changes in the OptaSense system	42
6.5	Impact of system update during project	43
6.5.1	Examples before update	43
6.5.2	After system update.....	46
6.6	Conclusions	49
7	References	50
8	Appendix: Verifying OptaSense Raw Speed and congestion data	51
8.1	Location of speed data	51
8.2	Location of congestion data	53
8.3	Drill down to raw data	53
8.3.1	Sample.....	54
8.3.2	Reading JSON messages.....	55
8.3.3	Tabulating sample data	55
8.3.4	Comparing matched positions	56
8.4	Conclusion	56
8.5	Update 25/07 2022.....	57
9	Appendix: scripts and tools to specify and publish incident data	58
9.1	Published data	58
9.2	Script "recordMTMIncident".....	58

1 Management summary

1.1 Graphs

The visualizations that illustrate and underpin the arguments in this report can be viewed with any web browser through:

modelit.nl/compare

Most of the graphs have interactive features. In the future the graphs may move to a new location.

1.2 Exploration and Verification of OptaSense data

1.2.1 *Usability of the congestion indicator for AID purposes*

Based on visual inspections of the OptaSense congestion indicator jointly with FCD data and loop-based AID's we conclude that in the case of A9R the OptaSense indicator cannot be used as a direct replacement for AID's that are generated from FCD or loop data.

1.2.2 *Usability of the speed indicator for AID purposes*

Although the speed data represents a richer dataset than the congestion data, also this data does not allow the derivation of AID's in an obvious way. The main issues are:

- When congestion occurs, more often than not, no speed data are reported at all. Although this implies that missing speed data is indicative of congestion, missing data also frequently appear outside congested periods;
- Speed drops as observed in FCD data often translate in either a speed drop of OptaSense data or to the OptaSense data being missing, but this happens with a considerable delay of 3-5 minutes.

In order to use the OptaSense data as a basis for AID's a considerable amount of postprocessing would be needed. On the one hand this is needed to eliminate false positives; On the other hand, this is needed to extend alarms that have been generated to surrounding locations and time-intervals to reduce false negatives.

1.3 Inspection of OptaSense speed data in more detail

Inspecting the OptaSense speed data in more detail shows a high proportion of missing data (about 57%). There is a strong positive correlation between the absence of observed speeds and the occurrence of congestion, even to the point that the lack of observed speeds is indicative of congestion.

The frequency with which changed data values appear is much less than the 1 second granularity of the data. The time series are therefore referred to as piecewise constant. The update frequency is also much less than the

typical vehicle headway to be expected on the A9. The impression is that the system mainly detects heavy traffic and traffic driving at higher (uncongested) speeds.

The only way to remedy this would be to increase the sensitivity of the system, but the current sensitivity setting is already the result of a careful trade of between being able to observe details on the one hand and producing faulty results as result of noise on the other hand. Apparently this trade off allows insufficient slack to allow for a setting that satisfies both requirements at an acceptable level.

When constructing an AID algorithm based on the OptaSense speed data, it is probably a good idea to base the algorithm on the reduced data set that results if all repeated values are ignored.

It is observed that the time instant / longitude pairs where the remaining data appear are located on lines that we refer to as "pseudo trajectories". Some of these pseudo trajectories coincide with observed FCD trajectories, although nightly data had to be inspected to ascertain this.

When pseudo trajectories reach a congested area they either dissolve or contain low-speed data points. Based on this observation a heuristic has been outlined.

This heuristic has not been investigated any further, but in follow up work to be carried out by the Rijkswaterstaat DataLab group the usage of AI techniques for deriving AID's will be investigated. This project will not only use OptaSense speed estimates but all available data, including the raw data generated by OptaSense system.

Apart from the possible success that the DataLab team might have with an AI based algorithm the detailed data inspection in this chapter has not been successful in showing the way to a robust AID algorithm based on OptaSense speed data for A9R case.

1.4 Bayesian Framework for estimating congestion state

A Bayesian method for estimating AID intervals has been derived, implemented, and applied to OptaSense speed data.

The behavior of the method can be tuned with the following parameters:

Aspect	Tuning option
<i>Propagation</i>	Parameters that describe the probability that congestion propagates to adjacent locations and the probability that a new congestion event is initiated.
<i>Measurement</i>	A likelihood ratio $P[U \text{congested}]/P[U/\text{not congested}]$ is defined for each possible outcome U, including the outcome "no data".
<i>AID threshold</i>	Congestion is indicated if the Bayes probability exceeds this value

These parameters make it possible to adapt the method to a wide variety of circumstances. The method has been applied with an initial best guess for the parameters. Compared to the existing OptaSense congestion indicator, the new Bayes method does not offer a spectacular improvement. The resulting AID intervals are more or less comparable.

The main reason to proceed with the new method lies in the possibilities to tune and extend the method while the existing method is a black box. This may be of importance for combining OptaSense data with other data sources.

During the analysis it has also become apparent that the system update that has occurred in the spring of 2022 has had a significant impact on the congestion estimates that OptaSense produces. In the new version reports of congestion are clearly less frequent compared to the earlier version.

There are a few improvements to the Bayes method that should be possible but could not be implemented due to a lack of time.

- At this moment the method still produces too many false positives during prolonged periods of missing data. A mechanism must be implemented to suppress these;
- The method has not been applied to the new version of the OptaSense system. The most visible difference between new and old version is that the time step has been reduced from 10 to 1 second and the spacing from (about) 50 to 10 meters. Maybe as a result of this the data contain many repeated values. Feeding these data directly into a Bayesian method would be suboptimal because it would violate the assumption that the error in each observation is independent from earlier errors. This doesn't hold at all for repeated values.

1.5 Comparing the A58 and A9 results

At first sight the results of the current study seem less optimistic about the usability of the OptaSense solution for AID purposes than the results of an earlier study on the A58 [3].

Reasons not attributable to system changes

- In the A58 study (graphical) results have been presented at a much higher aggregation than used in the A9 study. Requirements with respect to positional accuracy and latency imposed in the current study cannot be checked at that aggregated level;
- The incidents analyzed on the A58 have longer length and duration compared to those on the A9. Given the same absolute accuracy, it is to be expected that AID's have a better relative performance on the A58 compared to the A9.

Reasons attributable to system changes

- The OptaSense data used in the A9 study contain more missing periods than those in the A58 study. In part this may be due to the higher level of detail (1sec/10meter versus 60sec/50meter) but especially in congested periods the A9 data display a very high proportion of missing data that seems to be absent in the A58 data;

- During the A9 study a system update has taken place. After the update the sensitivity of the OptaSense congestion indicator has dropped significantly, indicating the algorithm that computes the congestion indicator wasn't scaled properly;
- Possibly (but unproven) the A58 system operated under more favorable physical conditions or was better calibrated.

2 Exploration and Verification of OptaSense data

2.1 Introduction

In an earlier study the OptaSense data have been explored and verified already. However, early in 2022 there has been a system update. This system update has had a number of consequences:

- the csv based interface as described in "OptaSense Road System Interfaces Tech Note.pdf" has been discontinued. This system was previously used to historize data and access them for off-line analysis;
- there is no evident replacement for this csv based system, other than querying the OptaSense webservice periodically and historizing these data in an external process. Such a historian has been implemented;
- the time granularity of the OptaSense webservice is 1 second compared to 10 seconds of the csv based system;
- the space granularity of the OptaSense webservice has been refined to approximately 10 meters, compared to 50 m for the csv based system.

2.2 Historizing OptaSense data

Because the OptaSense system is now missing a data historian that can reproduce the data published through the webservice, a stand-alone data historian has been implemented as a process that runs continuously and queries the OptaSense webservice each second. The resulting json messages are logged on disk. The first and last full day that have been logged this way are June 20 and July 30 in 2022.

The uptime of the querying process has been 100%, or very close to that, so no data was lost in this process.

Every few days the data have been parsed so that they become available in a binary format that can be loaded in "fcdview" tool for further analysis. Two types of data have been extracted: speeds and congestion status.

2.3 Correcting the congestion status

OptaSense publishes a congestion indicator that might be helpful in generating AID's. The OptaSense congestion indicator can have 3 values. That we interpret as follows

- 0, No congestion detected;
- 1, Congestion detected with low certainty;
- 2, Congestion detected with high certainty.

The fcdview tool visualizes the congestion indicator with a pale shade of magenta (congestion level 1) or a darker shade of magenta (congestion level 2).

An initial inspection has made clear that the OptaSense reported speeds indicator and the OptaSense congestion indicator do not match properly.

For 27 hand picked incidents a comparison was made between fcd speed, fiber speed and congestion indicator. The time- and longitude intervals for these incidents are shown in:

<https://www.modelit.nl/compare/CongestionEvents.png>

The comparisons can be accessed through:

<https://modelit.nl/compare/fcdvsfiber0.html>

Each comparison features a slider that can be moved to compare two images.

The comparisons show that fcd speed and fiber speed match, but fiber speed and congestion do not match. See for example

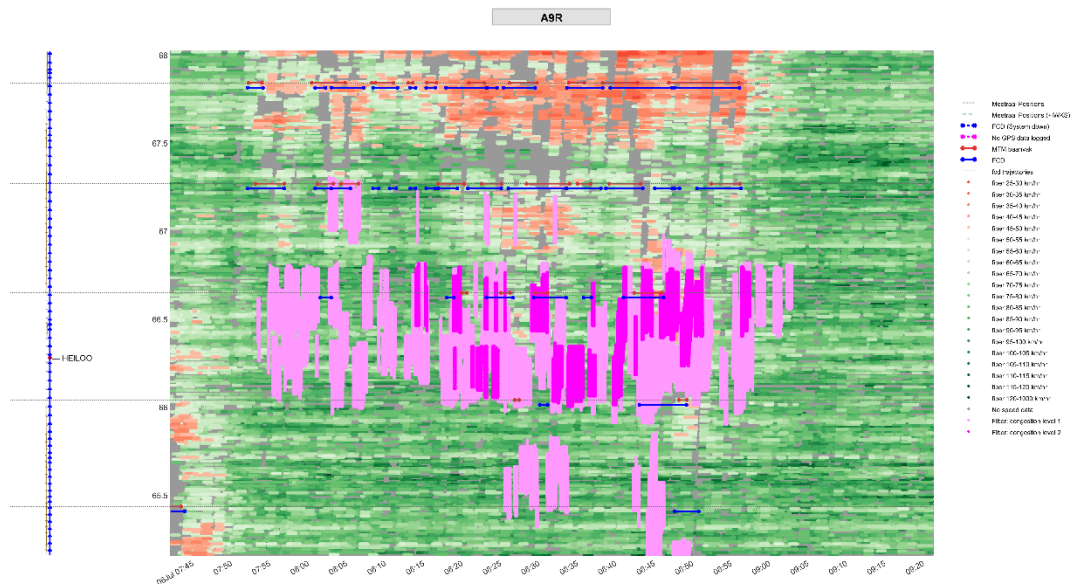


Figure 1 *OptaSense speed (green/red) and OptaSense congestion indicator (purple) shown in single image. A full interactive view is shown at https://modelit.nl/compare/do_compare.html?ref=cng.png&alt=spd.png*

This finding has been further investigated in Appendix 8. This has led to a workaround for obtaining the congestion indicator. This workaround will be used throughout this report.

After applying the work around the data for the 27 selected incidents can be inspected at:

<https://modelit.nl/compare/fcdvsfiber.html>

2.4 Compare congestion status prior and post update

Sections 2.4.1 and 2.4.2 visualize the OptaSense congestion indicator in the last available week before the system update (see section 2.1) and the first available week after the update. Due to a server malfunction, no access to the OptaSense data has been possible during the period in between.

Inspecting the plots indicates that the behavior of the system has changed. Prior to the update there has been a rather high false alarm rate for congestion level = 1.

Although reducing the false alarm rate is a good thing, this also reveals that the system update has had an actual effect on the indicators that the system produces, and also indicates that earlier (pre update) findings cannot be applied to the current (post update) data.

2.4.1 **Prior to update**

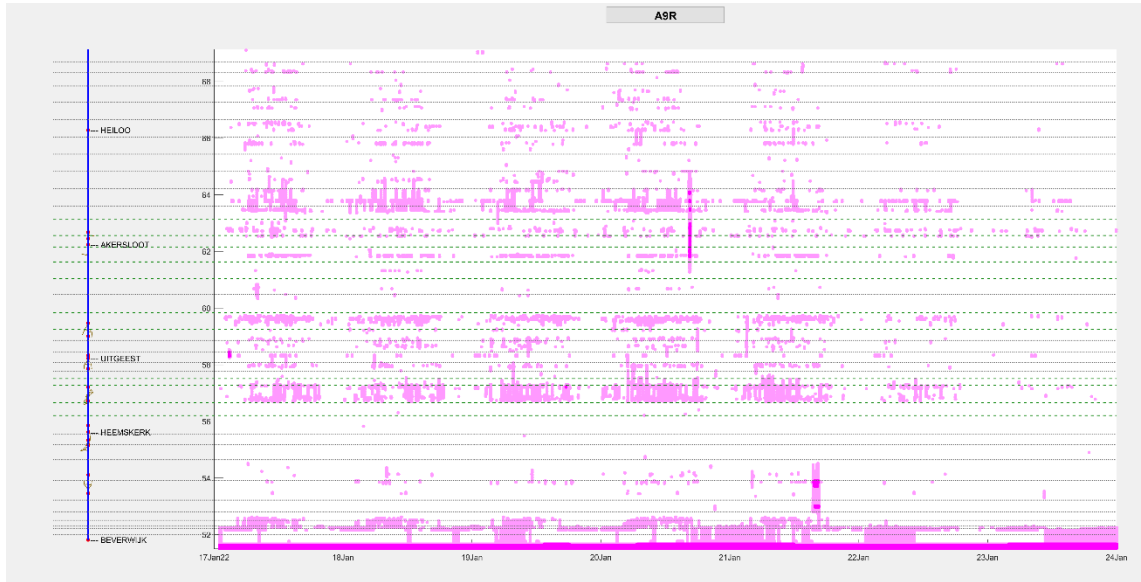


Figure 2: Visualization of the OptaSense congestion indicator in the week starting January 17, 2022.

2.4.2 **Post update**



Figure 3: Visualization of the OptaSense congestion indicator in the week starting June 20, 2022.

2.5 Visual inspection of the OptaSense congestion indicator

Using the loop based MTM-system and the Floating Car Data (FCD) that were obtained from BeMobile as a guideline, all major incidents (27 total) on the A9R between June 20 and Juli 30 have been identified, see <https://www.modelit.nl/compare/CongestionEvents.png>.

Inspection is possible at:

<https://modelit.nl/compare/fcdvsfiber.html>

Use the hyperlinks in the 2nd column to simultaneously view FCD speed data, Fiber speed data and the OptaSense congestion indicator.

After inspection the 27 incidents can be categorized as follows:

Table 1: *Classification of congestion events*

Congestion type	Typical delay	Incident ID's
Light	< 1 min	2, 3, 10, 12, 13, 14, 15, 16, 27
Moderate	1-3 min	1, 4, 5, 7, 9, 21, 24, 26
Heavy	>3 min	6, 8, 20
Very Heavy	>3 min, during >1h	11, 17, 18, 19, 22, 23, 25

2.6 Usability of the congestion indicator for AID purposes

In this section we investigate the option of using the OptaSense congestion indicator can be used as a "drop in replacement" for the AID's that are generated from FCD or loop data. With "drop in replacement" we mean that the data can be used without further manipulation or postprocessing.

To this end the earlier mentioned visualizations have been studied (see <https://modelit.nl/compare/fcdvsfiber.html>). The incidents for each congestion type classification have been reviewed, and a subjective assessment of the fitness for use has been made. These are presented in Table 2

Table 2: *Subjective assessment of fitness for use of congestion indicator for AID*

Congestion type	General finding
Light	Not detected at all
Moderate	Sometimes detected with congestion code =1, detection to late or too much downstream to be useful. Correlation between incident and detection location is poor.
Heavy	Partly detected with congestion code=1. Only small portions of the congestion is detected with congestion code=2, but with long delay.

	Detection is frequently interrupted, especially where speed data report "No Data"
Very Heavy	Largest part of congestion is detected with congestion code=1. Detection is frequently interrupted, especially where speed data report "No Data". Jointly congestion code =1, congestion code =2, and speed= "No Data" cover the congestion event to a reasonable extent.

For none of the congestion types the OptaSense congestion can be used as a drop in replacement for the AID's that are generated from FCD or loop data. Incidents of the categories Light, Moderate and Heavy are not or only partly detected. Very heavy incidents are detected for the largest part with congestion code=1, but this detection is frequently interrupted by spells where the speed is reported as "No Data", so a method should be found to interpret these "No data" spells as congested periods as well.

Also, the condition "congestion=1" also frequently applies in periods without congestion, so using this to trigger AID's would generate a false positives for these periods.

2.7 Usability of the speed indicator for AID purposes

In a similar manner as was done with the congestion indicator, the fitness for use of the speed data can be assessed.

Note that the OptaSense congestion indicator is derived from the OptaSense speed data, so from an information point of view the speed data represents a "richer" set of data. Information about the occurrence of incidents might be hidden in the speed data without being visible in the congestion status that was derived from it.

To find out if this is the case the congestion events that were identified in section 2.5 are once more inspected using the image comparison slider display. This time FCD speed are compared to fiber based speed data. The relevant shortcuts are available in the 7th column of the [table of incidents](#) (see Figure 36).

The subjective findings are reported in Table 3.

FCD vs Fiber speed detection (corrected congestion specification)

Incident ID	Date	Time	Position (downstream) [km]	Incident duration [sec]	Incident length [km]	FCD vs Fiber	FCD vs Fiber (+cong)	Fiber vs Cong
1	20-06-2022	15:46	66.727	1315	5.239	compare	compare	compare
2	20-06-2022	21:50	56.760	422	0.715	compare	compare	compare
3	21-06-2022	10:10	63.894	430	1.828	compare	compare	compare
4	21-06-2022	21:12	59.907	4817	4.529	compare	compare	compare
5	22-06-2022	09:19	64.827	692	2.679	compare	compare	compare
6	22-06-2022	15:27	66.622	1417	3.587	compare	compare	compare
7	23-06-2022	07:07	61.342	611	2.132	compare	compare	compare
8	24-06-2022	17:15	69.180	1777	2.356	compare	compare	compare
9	25-06-2022	11:49	64.696	848	3.301	compare	compare	compare
10	26-06-2022	13:25	66.698	290	0.987	compare	compare	compare
11	27-06-2022	17:22	69.169	5002	5.750	compare	compare	compare
12	27-06-2022	17:29	59.055	336	0.356	compare	compare	compare
13	30-06-2022	13:03	62.101	564	2.832	compare	compare	compare
14	30-06-2022	13:13	66.308	256	0.925	compare	compare	compare
15	03-07-2022	13:23	68.733	2239	0.938	compare	compare	compare
16	05-07-2022	22:19	64.712	769	1.264	compare	compare	compare
17	06-07-2022	07:24	68.660	5616	8.577	compare	compare	compare
18	06-07-2022	16:21	66.671	4767	3.637	compare	compare	compare
19	08-07-2022	15:43	66.627	4828	3.122	compare	compare	compare
20	11-07-2022	16:59	60.005	3594	7.316	compare	compare	compare
21	12-07-2022	17:03	55.359	3634	3.030	compare	compare	compare
22	13-07-2022	12:49	68.416	5365	7.732	compare	compare	compare
23	18-07-2022	15:17	69.180	2565	3.850	compare	compare	compare
24	21-07-2022	15:22	67.856	1382	9.553	compare	compare	compare
25	22-07-2022	17:16	66.734	3578	3.384	compare	compare	compare
26	23-07-2022	14:23	66.888	1826	1.579	compare	compare	compare
27	26-07-2022	20:05	63.813	809	1.284	compare	compare	compare

Figure 4: Table with shortcuts to "FCD vs Fiber plot". Click 7th column to compare inspect incident speed.

Table 3: Subjective assessment of fitness for use of speed data for AID

Congestion type	General finding
Light	<p>Light congestion is not, or barely visible in the OptaSense speed pattern. Occasionally 1 or 2 datapoints indicate a speed below 50 km/h, but the timing and location of such datapoints deviates too much from the location where MTH and FCD detect such an incident to be useful.</p> <p>Light congestion is by definition short lived. The effective detection of light congestion requires a response time of less than a minute.</p>
Moderate	<p>Moderate congestion somewhat visible in the OptaSense speed data in the form of patches with missing data and occasional speed observations below 50 km/h.</p> <p>However, patches with missing data are very frequent, also outside congested periods, and the speed observations below 50 km/h, if any, severely</p>

	<p>(typically 5 minutes) lag behind the speed drop as observed through MTM and FCD.</p> <p>Less than 30% congestion area is covered with speed observations below 50 km/h.</p>
Heavy	<p>Heavy congestion is visible in the OptaSense speed data in the form of patches with missing data mixed with speed observations below 50 km/h.</p> <p>Typically less than 50% congestion area is covered with speed observations below 50 km/h.</p> <p>Because patches with missing data frequently occur outside congested periods as well, and the observations below 50 km/h tend to lag behind the speed drop as observed through MTM and FCD, there is no evident logic that can be applied to derive the congestion location from the OptaSense speeds in this case.</p>
Very Heavy	<p>The observations that apply to moderate and heavy congestion types also apply to this congestion type. The general findings are:</p> <ul style="list-style-type: none"> • The congested area as seen in the FCD data set, translates to either missing data or low speed observations in the OptaSense; • This process typically lags the FCD data by 3-5 minutes; • Typically less than 50% congestion area is covered with speed observations below 50 km/h. <p>Because very heavy congestion extends over a larger time period and longitude range than less severe categories of congestion the tendency is to view this type of congestion on a more aggregate level. At first sight this improves the similarity between the OptaSense- and FCD speed patterns.</p> <p>However if one zooms in to a scale similar to what is used to inspect the other congestion types the one is confronted with issues similar to the ones observed with these congestion types, which are:</p> <ul style="list-style-type: none"> • speed drops in fcd data translate in either a speed drop of OptaSense data or to the OptaSense data being missing; • this happens with a considerable delay of 3-5 minutes; • patches of missing data occur outside congested periods as well. <p>Also in this case there is no evident logic that accurately derives the congestion location from the OptaSense speed data</p>

2.8 Conclusion

2.8.1 Usability of the congestion indicator for AID purposes

Based on visual inspections of the OptaSense congestion indicator jointly with FCD data and loop-based AID's we conclude that in the case of A9R the OptaSense indicator cannot be used as a direct replacement for AID's that are generated from FCD or loop data.

2.8.2 Usability of the speed indicator for AID purposes

Although the speed data represents a richer dataset than the congestion data, also this data does not lend itself for the derivation of AID's in an obvious way. The main issues are:

- When congestion occurs, more often than not, no speed data are reported at all. Although this implies that missing speed data is indicative of congestion, missing data also frequently appear outside congested periods;
- Speed drops as observed in FCD data often translate in either a speed drop of OptaSense data or to the OptaSense data being missing, but this happens with a considerable delay of 3-5 minutes.

In order to use the OptaSense data as a basis for AID's a considerable amount of postprocessing would be needed. On the one hand this is needed to eliminate false positives; On the other hand, this is needed to extend alarms that have been generated to surrounding locations and time-intervals to reduce false negatives.

If such an approach can be a success is difficult to judge at this point. The next chapters of this report will further investigate this.

3 Inspection of OptaSense speed data in more detail

3.1 Piecewise constant property

Every second, data are reported for a set of locations that are, on average, are 10 meters apart.

Inspecting these data in more detail show that for each location the time series of speed data can best be described as “piecewise constant”. Using the data of July 18 (Monday) as an example, locations can either report a valid speed (42.5%) or a missing value (57.5%). Given that a valid speed is reported for a specific period, the probability that the same value will be reported in the next period is 96.74%. Likewise if a missing speed is reported, the probability that the next period reports a missing speed is 99.32%. See also the table below.

Current period <i>(July 18, 2022)</i>	Next period compared to this period
Speed reported 42.5%	Identical speed reported 96.74%
	New valid speed value reported 2.34%
	Missing speed reported 0.92%
Missing speed reported 57.5%	New valid speed value reported 0.68%
	Missing speed reported 99.32%

How this looks when visualized is shown in Figure 5 and Figure 6. A better way to view both figures is the interactive version that can be viewed at:

https://modelit.nl/compare/do_compare.html?ref=Reported.png&alt=Innovations.png

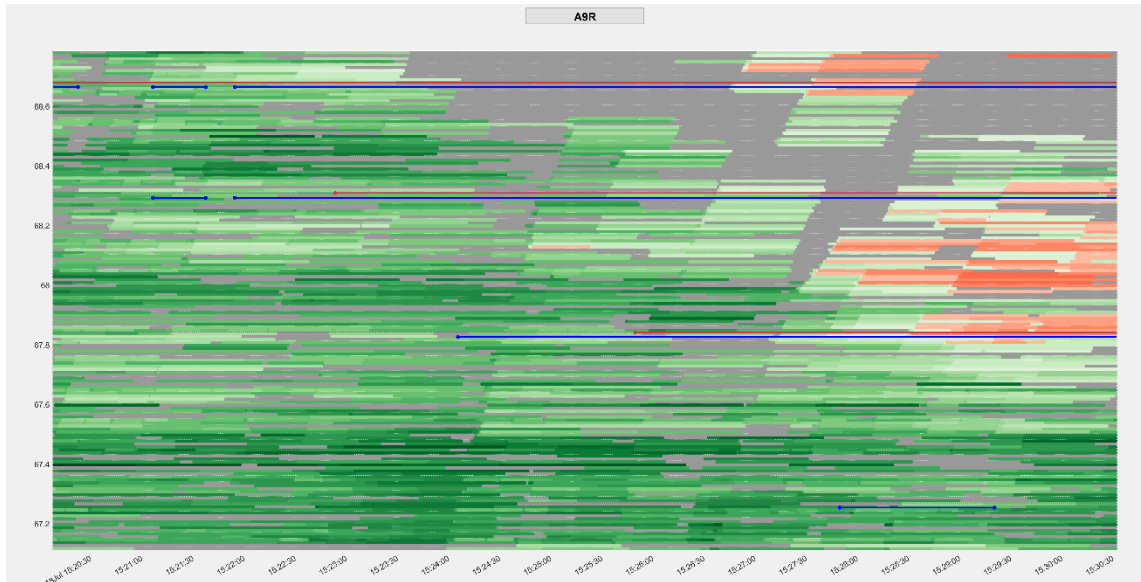


Figure 5: Snapshot of reported speeds for A9R longitude 67.2-68.2 (y-axis) and period 15:20-15:30 (x-axis) on juli 18. Missing data indicated in grey.

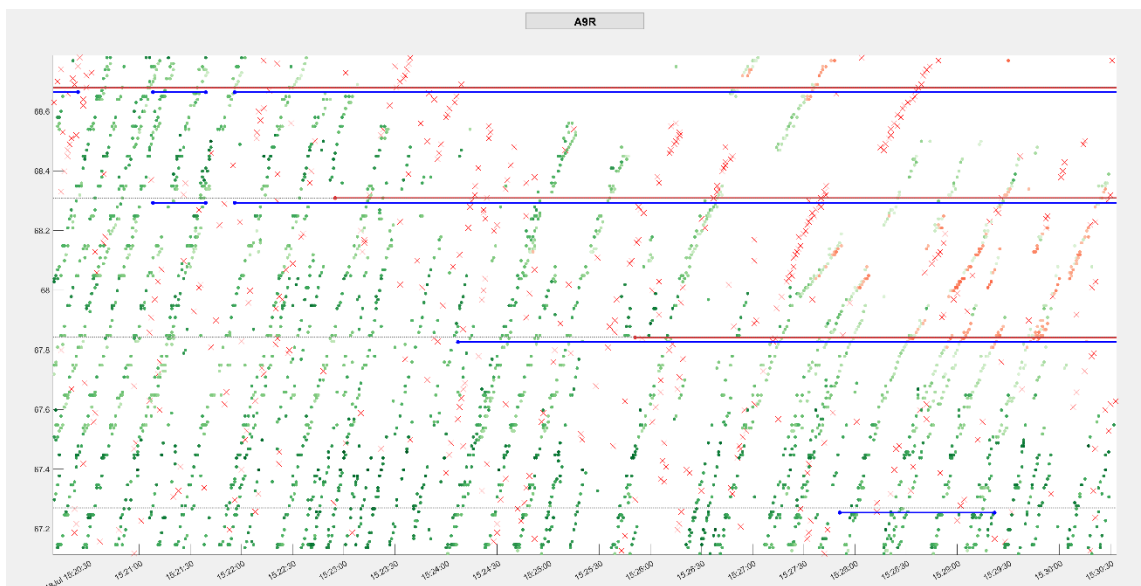


Figure 6: Snapshot of time-space selection equal to Figure 5 with replicated data (including missing data) hidden. Red crosses (x) indicate initial intervals with missing data.

Note that OptaSense speeds are reported with a 4 digits accuracy and are not otherwise restricted to a specific set of values. For example, 1.896.147 unique speed values have been reported on July 18. Therefore the likelihood of the two consecutive values being equal in absence of an underlying cause is negligible.

3.2 Possible explanations for the low data refresh rate

The frequency with which data are refreshed is much lower than is to be expected based on the average traffic flow rate on the A9. Apparently the reported data is a snapshot of an underlying data-model that is updated with a frequency much lower than the rate implied by typical vehicle headways . At this point we can only speculate why. A few possible causes come to mind:

- The acoustic footprint (where “acoustic” refers to *underground* vibrations) of many individual vehicles might be insufficient for detection;
- This applies even more when speeds drop due to congestion;
- The level of acoustic background noise is too high;
- The capability of transmission of vibrations through the ground might vary from location to location;
- Only heavy vehicles are detected;
- Only platoons are detected;
- The cells of the data model are updated sequentially, but due to CPU or data transfer limitations, not all rows can be updated each second.

The low refresh rate was not noticed in earlier versions of the OptaSense system, but this might also be due to the higher aggregation (1 minute averages) of the earlier version of the system.

An alternative and more worrying explanation for the perceived difference in refresh rate might be a sensitivity reduction of the system. Such a change is typically applied to remedy the effect of background noise. It reduces the number of anomalies the final data, but can also suppress relevant signals, like in this case, person cars and slowly moving traffic. This explanation is more worrying, as it implies that the sensitivity parameters have a high impact on the outcomes of the system, while setting these parameters is a somewhat subjective process. If tuning the parameters is a process that has to be repeated for each new site, this would be bad news for the scalability of the system.

3.3 Implication for fiber based AID algorithms

The dataset shown in Figure 6 can be derived from Figure 5 and vice versa.

When constructing an AID algorithm based on the OptaSense speed data, it is probably a good idea to base the algorithm on the data shown in Figure 6. Although from an information point of view Figure 5 and Figure 6 are equivalent, Figure 6 shows a much more parsimonious representation.

In other words the same information is represented with fewer numbers. Using this data as input for an explanatory model obviates the need for the model to remove redundant data and facilitates specifying a model that is as simple as possible while still taking into account all relevant information.

3.4 Patterns in speed innovations

We will define speed data that deviate from the preceding period as "speed innovations". If only speed innovations are shown as in Figure 6 a few patterns emerge.

3.4.1 *Pseudo trajectories*

Firstly, it seems that the locations of the speed innovations are organized in "pseudo trajectories" that propagate in the travel direction.

The obvious question here is how to interpret these trajectories. They could be related to individual trajectories of vehicles with a large acoustic footprint, to vehicle platoons, or they could just appear as an artefact of the algorithm that are used.

To investigate, we inspect a nightly period, hoping to match pseudo trajectories with fcd trajectories. The next interactive reference compares "pseudo trajectories" to "true" FCD trajectories:

https://modelit.nl/compare/do_compare.html?ref=fibernight.png&alt=fcdnight.png

The data were taken from a period during the night where very little traffic occurs and even less FCD trajectories are observed. This facilitates matching separate trajectories. To distinguish OptaSense speed data from FCD data, OptaSense speed innovations are plotted as triangles.

Only 16 FCD trails were recorded in the time interval 03:35-04:25. This makes it possible to verify if individual FCD trajectories match with individual pseudo trajectories. For the trajectories starting 03:36, 3:42 and 03:49 this is convincingly shown.

The mean speeds for these FCD trajectories as computed in the table below are well above the maximum speed for trucks of 80 km/h. This is not very surprising as the FCD data are derived from users of the Flitsmeister app. This app is not aimed at truck drivers. However, it is shown that at least some of the pseudo trajectories match individual passenger cars or light vans. Other (non matched) pseudo trajectories use 12 minutes or more to

traverse from km 52 to km 68, implying a speed of 80 km/h or less. These are likely to originate from trucks.

passage at km 52	passage at km 68	travel time [MM:SS]	mean speed [km/h]
03:36:03	03:45:40	09:37	99.8
03:42:23	03:53:38	09:15	103.8
03:49:19	03:57:34	08:15	116.4

With respect to the initial question, the answer is that at least some of the pseudo trajectories originate from individual vehicles and that, based on the trajectory angles, these vehicles are likely to be trucks.

3.4.2 ***Initial missing data echo pseudo trajectories***

A second finding is that, if two pseudo trajectories are more than 60 seconds apart, the red crosses that mark the beginning of a period with missing data form a trajectory that lags the preceding pseudo trajectory with a delay of about 60 seconds.

Although the underlying algorithm is a black box, it seems that the algorithm has a built in logic that sets the data status for a specific location to "missing" if no changes have been applied to the estimate for a certain period.

3.4.3 ***Pseudo trajectories do not propagate through congestion***

For the third finding we have to view data patterns that were taken from a period during the day:

https://modelit.nl/compare/do_compare.html?ref=fiberday.png&alt=fcdday.png

or zoomed in on a shorter interval:

https://modelit.nl/compare/do_compare.html?ref=fiberday2.png&alt=fcddy2.png

These plots show that the fiber pseudo trajectories usually (but not always!) stop if congestion occurs. See also Figure 7.

3.5 **Third indication for congestion: capped pseudo trajectories**

Figure 7 simultaneously plots FCD data (colored dots) and fiber data (colored triangles). When sufficiently zoomed in, trajectories can be distinguished in the fiber data. When these trajectories reach the congested area they often either dissolve, or continue as a set of datapoints that report a low speed (plotted using a red color). In both cases a so called "endpoint" can be defined as the last datapoint of the pseudo trajectory reporting a sufficiently high speed.

This creates a third possibility to identify congestion. The other two being the observation of low speeds and the prolonged observation of missing data.

The outline of a heuristic method based on this third possibility would be a three step process:

- 1- identify pseudo trajectories;
- 2- identify endpoints of these trajectories were either these trajectories dissolve, or the underlying data indicate speeds below a threshold;
- 3- group endpoints that are sufficiently close by into a congestion frontier.

To transform this outline in a fully functional and robust algorithm quite an elaborate logic is required. As there is not really a theoretic basis that underpins the outline, the only available guideline for creating a working algorithm is observing how well a candidate algorithm reproduces the available FCD or MTM data. A deep learning algorithm might be best suited for this application.

In a follow up project a dedicated team in the Rijkswaterstaat DataLab group, will investigate the derivation of AID's from fiber data using AI techniques. In the DataLab project the source data will not be restricted to speed data, but will also include the "raw" data generated by OptaSense system.

To avoid overlap, the construction of a heuristic based on the outline will not be attempted here.

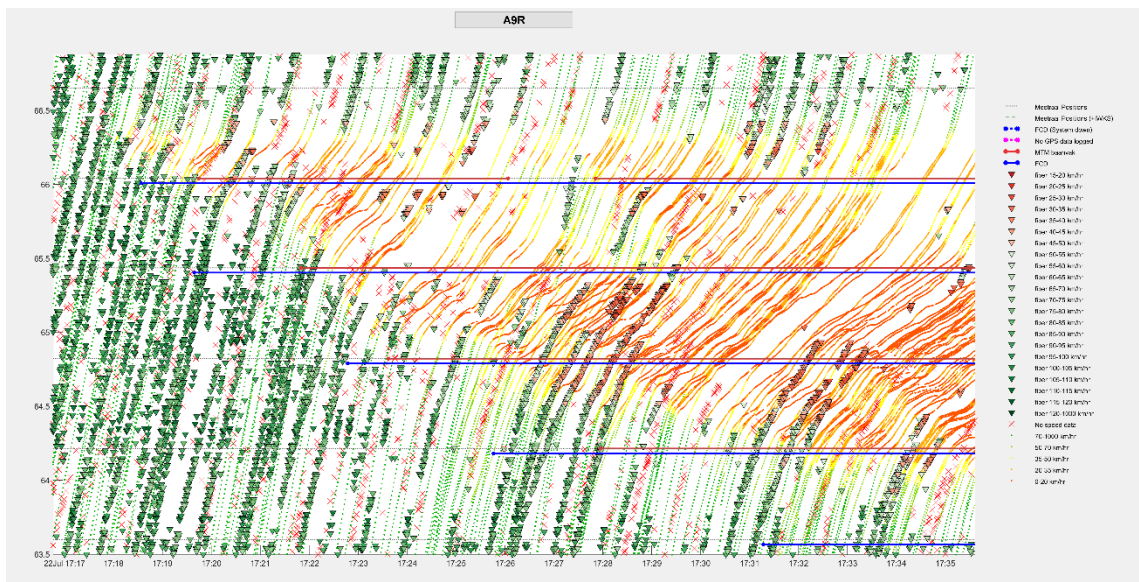


Figure 7: *Pseudo trajectories do not propagate through congestion*

3.6 Detailed inspection – conclusions

Inspecting the OptaSense speed data in more detail shows a high proportion of missing data (about 57%). There is a strong positive correlation between the absence of observed speeds and the occurrence of

congestion, even to the point that the lack of observed speeds is indicative of congestion.

The frequency with which changed data values appear is much less than the 1 second granularity of the data. The time series are therefore referred to as piecewise constant. The update frequency is also much less than the typical vehicle headway to be expected on the A9. The impression is that the system mainly detects heavy traffic and traffic driving at higher speeds.

The only way to remedy this would be to increase the sensitivity of the system, but the current sensitivity setting is already the result of a careful trade off between being able to observe details on the one hand and producing faulty results as result of noise on the other hand. Apparently this trade off allows insufficient slack to allow for a setting that satisfies both requirements at an acceptable level.

When constructing an AID algorithm based on the OptaSense speed data, it is probably a good idea to base the algorithm on the reduced data set that results if all repeated values are ignored.

It is observed that the time instant / longitude pairs where the remaining data appear are located on lines that we refer to as "pseudo trajectories". Some of these pseudo trajectories coincide with observed FCD trajectories, although nightly data had to be inspected to ascertain this.

When pseudo trajectories reach a congested area they either dissolve or contain low-speed data points. Based on this observation a heuristic has been outlined.

This heuristic has not been investigated any further here, but in a follow up project to be carried out by the Rijkswaterstaat DataLab group the usage of AI techniques for deriving AID's will be investigated. This project will not only use OptaSense speed estimates but all available data, including the raw data generated by OptaSense system.

Apart from the possible success that the DataLab team might have with an AI based algorithm the detailed data inspection in this chapter has not been successful in showing the way to a robust AID algorithm based on OptaSense speed data for A9R case.

4 Bayesian Framework for estimating congestion state

4.1 Introduction

In this chapter we investigate an alternative for the congestion code as derived by OptaSense. This work has been done in the period between September 2021 and November 2021.

The objective is to formulate a mathematical model that describes how congestion propagates on the one hand and the relation between congestion and observed speeds on the other hand. This model should then be used for estimating the current state of the system (congested or not).

A Bayesian approach will be used. A Bayesian approach describes, in probabilistic terms, how the state evolves in time and what the relationship between system state and measurements is. Based on this a probability distribution for the system state conditional to all available measurements is derived.

At least in theory a Bayesian approach solves any estimation problem in a systematic way, where the attractive property is that it makes explicit assumptions about uncertainty, while in other methods these assumptions are merely implicit. The estimator should then follow directly from the assumptions. In practice it is not always tractable to derive a Bayesian solution to a problem. If this happens assumptions must be simplified or the Bayesian approach is abandoned altogether.

In this chapter a Bayesian estimator will be derived that generates AID intervals based on OptaSense speed data. The estimator has been applied to OptaSense data up to November 2021 and compared to AID intervals available from the MTM system and the BeMobile FCD system.

4.2 Elements of a Bayesian approach

4.2.1 Some notation

Some notation and notational conventions are introduced below.

Notation

S_t	Traffic state vector at time instant t . $S_t = [s_{1t}, s_{2t}, \dots, s_{Mt}]$ where each element defines the traffic state at a predefined set of locations $1, 2, \dots, M$. The elements of S_t can be 0 (no incident at this location) or 1 (incident at this location).
U_t	Vector with observed speeds at time instant t . $U_t = [u_{1t}, u_{2t}, \dots, u_{Mt}]$. Note that speed observations might not be available for all locations in the set $1, 2, \dots, M$.
$P[S_t U_1, \dots, U_t]$	The conditional probability of traffic state S_t given observations U_1, \dots, U_t .

4.2.2 Elements of a Bayesian approach

A Bayesian approach implements the following elements:

- Time extrapolation
given: $P[S_t | U_1, \dots, U_t]$ (S_t is incident status, 0 or 1)
compute: $P[S_{t+1} | U_1, \dots, U_t]$ (U_t is observed speed)
- Measurement update
given: $P[S_{t+1} | U_1, \dots, U_t]$ and U_{t+1}
compute: $P[S_{t+1} | U_1, \dots, U_{t+1}]$
- Point estimate:
given: $P[S_{t+1} | U_1, \dots, U_{t+1}]$
compute: AID(t+1)

Sections 4.2.3 through 4.2.5 explain these elements at a general theoretical level. Section 4.3.1 through 4.3.3 work out the specifics for the AID estimation case.

4.2.3 Measurement update

The objective of the measurement update is to compute the conditional probability distribution function (PDF) $P[S_t | U_1, \dots, U_t]$ from the prior PDF $P[S_t | U_1, \dots, U_{t-1}]$ and the latest measurement U_t .

In words: for each possible outcome of the incident status S_t we want to compute the probability of occurrence, given all observations available a time instant t .

The input for the measurement update is the *a priori* distribution $P[S_t | U_1, \dots, U_{t-1}]$, the likelihood function $P[U_t | S_t, U_1, \dots, U_{t-1}]$ and the *normalization constant* $P[U_t | U_1, \dots, U_{t-1}]$.

Using this input, the following holds:

$$P[S_t | U_1, \dots, U_t] = P[S_t | U_1, \dots, U_{t-1}] \cdot P[U_t | S_t, U_1, \dots, U_{t-1}] / P[U_t | U_1, \dots, U_{t-1}] \quad (1)$$

The equation above is known as *Bayes rule* and can be checked by replacing each conditional probability $P[a|b]$ by its definition $P[a,b] / P[b]$

Note that only the *a priori* distribution and the likelihood function depend on the state S_t . In many cases the expression $P[U_t | S_t, U_1, \dots, U_{t-1}]$ can be simplified to $P[U_t | S_t]$ because once the state is known, earlier observations do not carry additional information about U_t . The exception is if the observations contain errors that are serially correlated.

The value of $P[U_t | U_1, \dots, U_{t-1}]$ follows from the requirement that the addition of $P[S_t | U_1, \dots, U_t]$ over all admissible values of S_t must be 1, hence the name *normalization constant*.

So with all simplifications in place, the measurement update equation reads like:

$$P[S_t | U_1, \dots, U_t] = C \cdot P[S_t | U_1, \dots, U_{t-1}] \cdot P[U_t | S_t] \quad (2)$$

This equation computes $P[S_t | U_1, \dots, U_t]$ from the inputs $P[S_t | U_1, \dots, U_{t-1}]$ and $P[U_t | S_t]$

4.2.4 **Time extrapolation**

To close the recursion we need a mechanism that computes $P[S_{t+1}|U_1..U_t]$ from $P[S_t|U_1..U_t]$. This step is known as the time extrapolation.

As a point of departure we use that $P[S_{t+1}, S_t|U_1..U_t]$ equals the summation of $P[S_{t+1}, S_t|U_1..U_t]$ over all admissible values of S_t :

$$P[S_{t+1}|U_1, \dots, U_t] = \sum_{S_t} P[S_{t+1}, S_t|U_1, \dots, U_t]$$

Note that by definition:

$$P[S_{t+1}, S_t|U_1, \dots, U_t] = P[S_{t+1} | S_t, U_1, \dots, U_t] \cdot P[S_t|U_1, \dots, U_t]$$

Usually it is safe to assume:

$$P[S_{t+1} | S_t, U_1, \dots, U_t] = P[S_{t+1} | S_t]$$

because any observations prior to $t+1$ carry no information on the state-change that occurs between time instant t and time instant $t+1$.

Combining the three equations above gives:

$$P[S_{t+1}|U_1, \dots, U_t] = \sum_{S_t} P[S_{t+1}|S_t] \cdot P[S_t|U_1, \dots, U_t]$$

4.2.5 **Point estimate**

The objective of the point estimate is to translate the conditional probability density function $P[S_t|U_1..U_t]$ which defines a separate number for each possible outcome into a single estimate. This can be done in different ways, for example by computing the expected value or the value that maximizes the $P[S_t|U_1..U_t]$.

In the current case the typical point estimate should be "congestion" or "no congestion", while $P[S_t|U_1..U_t]$ contains the information that there is an $x\%$ probability of congestion. One way to deal to this is to set the point estimate to "congestion" if and only if $P[S_t|U_1..U_t]$ indicates that the probability of congestion exceeds a (high) threshold such as 97%. The value of the threshold to be used is implied by the percentages of false positives and false negatives that are acceptable.

4.3 **Application of the Bayesian framework to AID**

In order to apply the Bayesian framework to compute AID's from OptaSense speed data we have to create equations for the measurement update, time extrapolation and point estimate that we can actually apply. The subsections below specify these equations. Some of these equations contain parameters that can be used to tune the method.

4.3.1 Measurement update

In this section we will initially consider the AID estimation for one specific location only. For this location the state S_t can either be 1 (incident present at time instant t) or 0 (no incident present at time instant t).

Because of this, the prior distribution $P[S_t|U_1..U_{t-1}]$ is characterized by two probabilities: $P[S_t=0|U_1..U_{t-1}]$ and $P[S_t=1|U_1..U_{t-1}]$.

The objective of the measurement update is to compute two new numbers: $P[S_t=0|U_1..U_t]$ and $P[S_t=1|U_1..U_t]$.

For this we apply Bayes rule as given in equation (1).

$$\begin{aligned} P[S_t=0|U_1..U_t] &= P[S_t=0|U_1..U_{t-1}] \cdot P[U_t|S_t=0, U_1..U_{t-1}] / P[U_t|U_1..U_{t-1}] \\ &\text{and} \\ P[S_t=1|U_1..U_t] &= P[S_t=1|U_1..U_{t-1}] \cdot P[U_t|S_t=1, U_1..U_{t-1}] / P[U_t|U_1..U_{t-1}] \end{aligned} \quad (3)$$

We now argue that:

$$\begin{aligned} P[U_t|S_t=0, U_1..U_{t-1}] &= P[U_t|S_t=0] \\ &\text{and} \\ P[U_t|S_t=1, U_1..U_{t-1}] &= P[U_t|S_t=1] \end{aligned} \quad (4)$$

The argumentation for this is that, once the congestion state S_t is known, the earlier observed speeds are no longer relevant for the current speed U_t . Even if this might not be entirely true it is the only route to tractability as we have no useful mathematical expression for $P[U_t|S_t=0, U_1..U_{t-1}]$ available.

For $P[U_t|S_t=0]$ and $P[U_t|S_t=1]$ we can make some plausible assumptions though. In fact we do not even need the separate values of $P[U_t|S_t=0]$ and $P[U_t|S_t=1]$. We only require the ratio $L[U_t]=P[U_t|S_t=1]/P[U_t|S_t=0]$.

Proof

The sum of the PDF over all admissible values must be 1:

$$P[S_t=0|U_1..U_t] + P[S_t=1|U_1..U_t] = 1 \quad (5)$$

Now combine (3), (4) and (5):

$$\begin{aligned} C \cdot (P[S_t=0|U_1..U_{t-1}] \cdot P[U_t|S_t=0] + P[S_t=1|U_1..U_{t-1}] \cdot P[U_t|S_t=1]) &= 1 \\ \text{with} \\ C &= 1 / P[U_t|U_1..U_{t-1}] \end{aligned} \quad (6)$$

As a result:

$$C = 1 / (P[S_t=0|U_1..U_{t-1}] \cdot P[U_t|S_t=0] + P[S_t=1|U_1..U_{t-1}] \cdot P[U_t|S_t=1]) \quad (7)$$

And:

$$\begin{aligned}
& \frac{P[S_t=1|U_1..U_t]}{P[S_t=1|U_1..U_{t-1}].P[U_t|S_t=1]} \\
& \frac{(P[S_t=0|U_1..U_{t-1}].P[U_t|S_t=0] + P[S_t=1|U_1..U_{t-1}].P[U_t|S_t=1])}{P[S_t=1|U_1..U_{t-1}].P[U_t|S_t=1]} \\
& \frac{P[S_t=1|U_1..U_{t-1}].L[U_t]}{(P[S_t=0|U_1..U_{t-1}] + P[S_t=1|U_1..U_{t-1}].L[U_t])}
\end{aligned} \tag{8}$$

So finally:

$$\begin{aligned}
& \frac{P[S_t=1|U_1..U_t]}{P[S_t=1|U_1..U_{t-1}].L[U_t]} \\
& \frac{P[S_t=1|U_1..U_{t-1}].L[U_t]}{(P[S_t=0|U_1..U_{t-1}] + P[S_t=1|U_1..U_{t-1}].L[U_t])}
\end{aligned} \tag{9}$$

with:

$$L[U_t] = P[U_t|S_t=1] / P[U_t|S_t=0]$$

End of proof

Equation (9) is a remarkably simple update equation that only requires the specification of $L[U_t]$. Although we have no exact specification of $L[U_t]$, this function should be an S curve with its maximum at $U=0$ and its minimum at $U=100$ (or above) and a tipping point somewhere between 50 and 30 km/u. Table 4 shows the values that have been used in the current experiments.

Special care will be needed for specifying $L[NaN]$ where NaN (not a number) indicates that no speed observation is available. Experience learns that missing data are indicative of congestion but can also occurs when traffic is absent.

The equations above all apply to a single location. These equations have to be applied to each location of the transit network.

4.3.2 **Time extrapolation**

Unlike the measurement update, for the time extrapolation we cannot consider a specific location in isolation. This is because congestion tends to propagate in space. The typical propagation speed is -18 km/h but we must take into account propagation speeds between 0 km/h and -36 km/h. Usually the propagation direction of congestion is opposed to the travel direction.

Assuming that time and space have been discretized to a set of time-instants and longitudes. The congestion state at position x and time instant $t+1$ typically depends on the congestion states at time instant t at the adjacent downstream locations, as illustrated in the figure below.

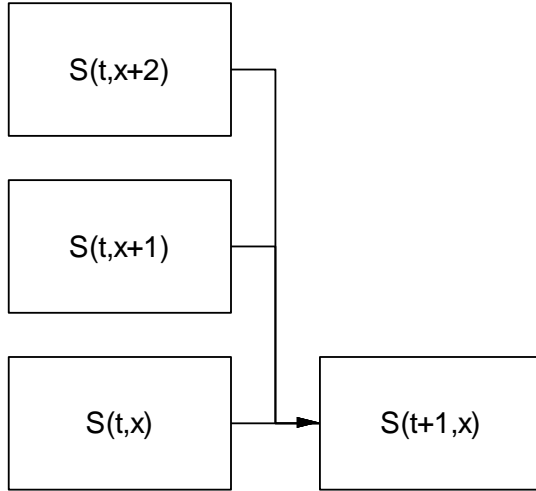


Figure 8: Dependencies implied by the propagation of congestion

How many adjacent locations must be taken into account depends on the spacing between locations x , $x+1$, $x+2$, etc., the time step, and the range of propagation speeds taken into account.

At the start of the time extrapolation $P_x[S_t|U_1..U_t]$, $P_{x+1}[S_t|U_1..U_t]$ and $P_{x+2}[S_t|U_1..U_t]$ should be available.

Next $P_x[S_{t+1}=0|U_1..U_t]$ represents the combined event that no incident propagates from location x , $x+1$ or $x+2$ and no new incident arises between t and $t+1$. All these events have to be true at the same time. Under the assumption that these sub-events occur independently, we can compute $P_x[S_{t+1}=0|U_1..U_t]$ by multiplying the probabilities of all required sub-events.

$$\begin{aligned}
 & P_x[S_{t+1}=0|U_1..U_t] \\
 & = \\
 & \text{Prob}[\text{incident did not propagate from previous period \& no new} \\
 & \quad \text{incident has occurred}] \\
 & = (1 - \text{Prob}[\text{propagation from } x])^* \\
 & \quad (1 - \text{Prob}[\text{propagation from } x+1])^* \\
 & \quad (1 - \text{Prob}[\text{propagation from } x+2])^* \\
 & \quad (1 - \text{Prob}[\text{new incident}])
 \end{aligned} \tag{10}$$

To evaluate this expression we will use:

$$\begin{aligned}
 \text{Prob}[\text{propagation from } x] &= P_x[S_t=1|U_1..U_t] \cdot P_{\text{propagate}}(x,x) \\
 \text{Prob}[\text{propagation from } x+1] &= P_{x+1}[S_t=1|U_1..U_t] \cdot P_{\text{propagate}}(x+1,x) \\
 \text{Prob}[\text{propagation from } x+2] &= P_{x+2}[S_t=1|U_1..U_t] \cdot P_{\text{propagate}}(x+2,x) \\
 \text{Prob}[\text{new incident}] &= \Delta t \cdot P_{\text{incident}}
 \end{aligned} \tag{11}$$

In this expression $P_{\text{propagate}}(x,x)$, $P_{\text{propagate}}(x+1,x)$, $P_{\text{propagate}}(x+2,x)$ and P_{incident} are fixed parameters.

Table 5 shows the parameters that were used in the current experiments. This table defines the propagation probabilities as a function of the

propagation speed. This propagation speed is depends on the time-longitude step ratio. To take into account that the propagation probability decreases if a larger time step is applied an incident fading coefficient is applied. This coefficient that is set at a reduction of 50% for every 600 sec.

P_{incident} expresses the probability that a new incident occurs and is set at 1 incident per 3600 seconds.

4.3.3 Point estimate

To obtain a point estimate, the following recipe could be applied.

- Switch AID false>>true for any cross section with $P[S_{x,t+1}|U_1..U_{t+1}] > 0.98$ for *at least 1* x in "look ahead distance"
- Switch AID true>>false for any cross section with $P[S_{x,t+1}|U_1..U_{t+1}] < 0.97$ for *all* x in "look ahead distance"

4.4 Application

4.4.1 Matlab code

The logic as described in this chapter has been implemented in Matlab code. Two files are involved:

- BayesMethod.m. This is a class definition file that defines the methods "measurementUpdate", "timeExtrapolate" and "pointEstimates", and a number of auxiliary functions;
- OptaSenseAIDgenerator.m. This is a script that loads in the OptaSense speed data 1 day at a time, computes the Bayes probabilities and generates AID intervals for a set of predefined longitudes (the so called MSI positions).

4.4.2 Tuning the parameters

To apply the method a number of parameters are needed. For the measurement update the required parameters are given in Table 4. As argued before, only $L[U]$ is actually needed. $L[U]$ is graphed in Figure 9.

Table 4: Selected parameters for measurement update

Speed range	$P[U 1]$	$P[U 0]$	$L[U]$
NaN	0.9157	0.0811	11.29
0-10	0.0071	0.0001	54.20
10-20	0.0071	0.0001	54.20
20-30	0.0182	0.0003	54.20
30-40	0.0189	0.0003	54.20
40-50	0.0236	0.0005	45.17
50-60	0.0025	0.0011	2.26
60-70	0.0019	0.0042	0.45
70-80	0.0008	0.0188	0.05
80-90	0.0003	0.0738	0.00
90-100	0.0009	0.1937	0.00
100-110	0.0013	0.2817	0.00
110-120	0.0011	0.2338	0.00

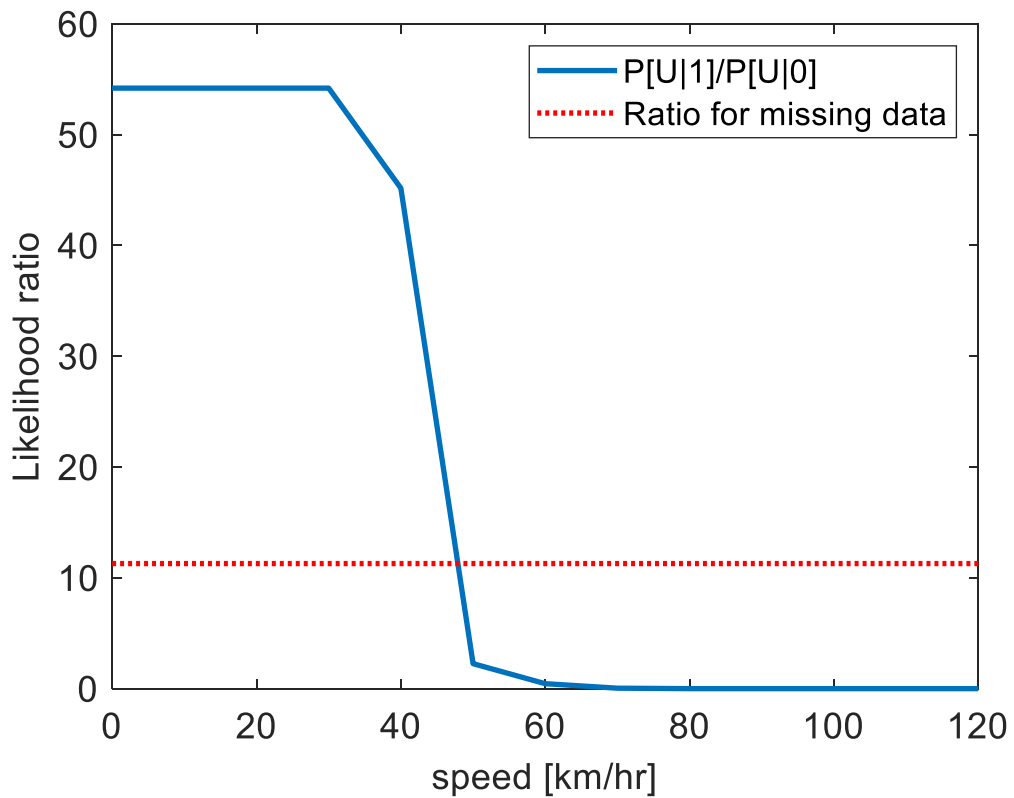


Figure 9: Likelihood ratio $P[U_t|S_t=1]/P[U_t|S_t=0]$ for valid observations (blue) and missing observation (red)

Also the time extrapolation requires a number of parameters. These are given in Table 5 and Table 6.

Table 5: Selected parameters for time extrapolation

Propagation speed from [km/hr]	Propagation speed to [km/hr]	Propagation probability
-36	-24	0,15
-24	-12	0,70
-12	1	0.15

Table 6: Additional parameters for time extrapolation

Parameter	Value	Note
Incident fading coefficient (50%)	600 sec	The propagation probabilities is reduced with a factor $0.5^{(\text{delay}/\text{value})}$
Incident frequency	1/3600 incidents per hour	P_{incident} is approximated as $\text{delay} \cdot \text{value}$ This approximation is only valid for small delays.

4.4.3 **Time range of data**

The Bayesian method as described in this chapter has been developed between September 2021 and November 2021. To verify the method, it has been applied to OptaSense data that were obtained between March 1 2021 and November 21 2021.

At this point the project has been paused awaiting progress in the realization of software that makes the OptaSense data available for online use.

Due to summer holidays and a Covid wave in the autumn (see Table 7) little congestion has occurred in the period March-November 2021.

From a traffic point of view a later period (starting March 2022) would have been more interesting. However a new OptaSense version has been rolled out in the spring of 2022 and this version uses a different granularity for time and space discretization, and a new data format. Also it seems that the properties of the data has been changed as well in the new release, meaning different speed estimates and less congestion warnings. To repeat the verification of the Bayesian method for this new period would require at least redoing the tuning and the experimenting that has been done up to November 2021.

Table 7: *Covid time table 2021, summary based on <https://www.rijksoverheid.nl/onderwerpen/coronavirus-tijdslijn/2021>*

jan-april	traffic is heavily impacted by COVID. People working from home. Evening curfew in January.
may-sept	gradual reduction of impact of COVID.
okt-dec	increasing infections, hard lockdown in December.

It has been decided to stick with the experiments that were done up to November 2021. The results of this are presented here.

4.4.4 **Applied methods**

Two methods have been implemented. Both methods generate AID intervals for a predefined set of cross sections (the MSI positions) on the A9R. These can then be compared to the intervals generated by MTM and BeMobile.

OptaSense method

This method uses the OptaSense congestion indicator as part of departure. An AID is generated if 1 or more locations within the lookahead distance (500 meters) of the cross section have a nonzero congestion status.

Bayes method

This method is identical to the OptaSense method, but with congestion status swapped out for the point estimate of the Bayes probability as described in section 4.3.3.

4.5 Results

4.5.1 In depth analysis of single congestion event

Before presenting results for the full period we analyze the behavior of the OptaSense congestion code and the Bayes method for a congestion event that occurred 5 November 2021. This event is representative for other events on the A9R.

This analysis is done on the basis of the figures listed in Table 8.

Table 8:

Figure	Content of figure
Figure 10	Legend applicable to all subsequent figures
Figure 11	BeMobile FCD data for congestion event on November 5
Figure 12	OptaSense speed data. Missing data are plotted as grey dots
Figure 13	OptaSense speed data with overlay of OptaSense congestion code (blue)
Figure 14	OptaSense speed data with overlay of Bayes probabilities, starting at 97 th percentile
Figure 15	OptaSense speed data with overlay of Bayes probabilities, starting at 98 th percentile
Figure 16	Plot of only the AID intervals

The next plots show screenshots of the fcdview application. This application has already been introduced in earlier chapters. In this chapter the application is used to plot FCD data, OptaSense speed data, OptaSense congestion code and Bayes probabilities.

On top of all plots the AID intervals plotted, see Figure 10 for the legend. These are imported from MTM or BeMobile log files or derived using the method "OptaSense" and "Bayes".

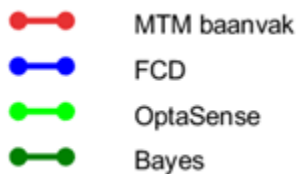


Figure 10: Legend for subsequent figures

Figure 11 and Figure 12 show the raw speed data from FCD versus OptaSense. Note that in the OptaSense plot "grey" means missing data. Both plots show similar patterns, but observed low speeds as shown in the FCD plot appear as missing data the OptaSense plot for the largest part.

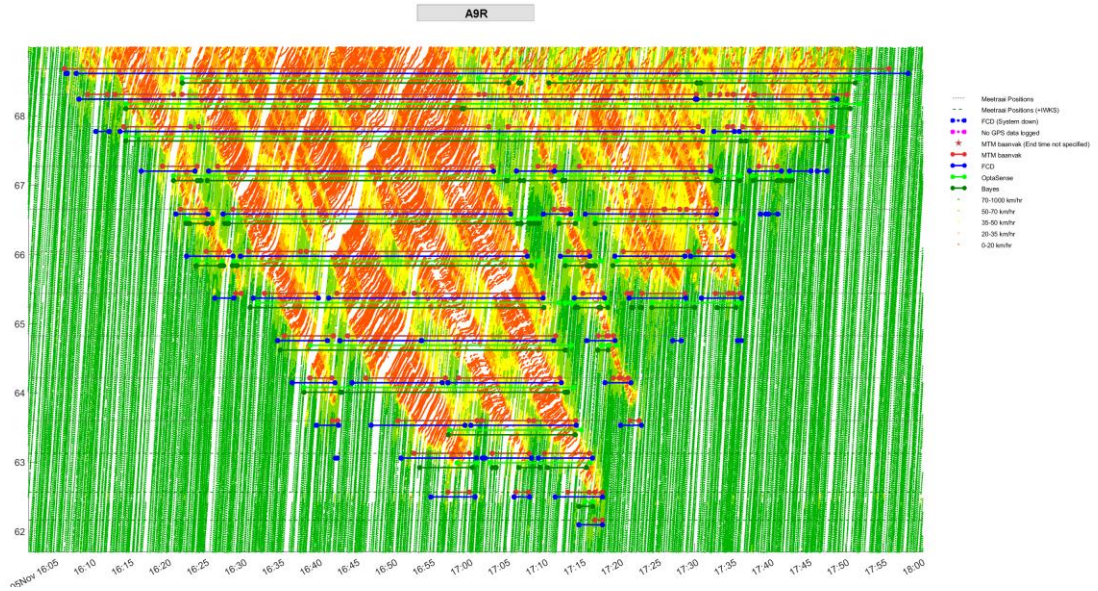


Figure 11: BeMobile FCD data for congestion event on November 5

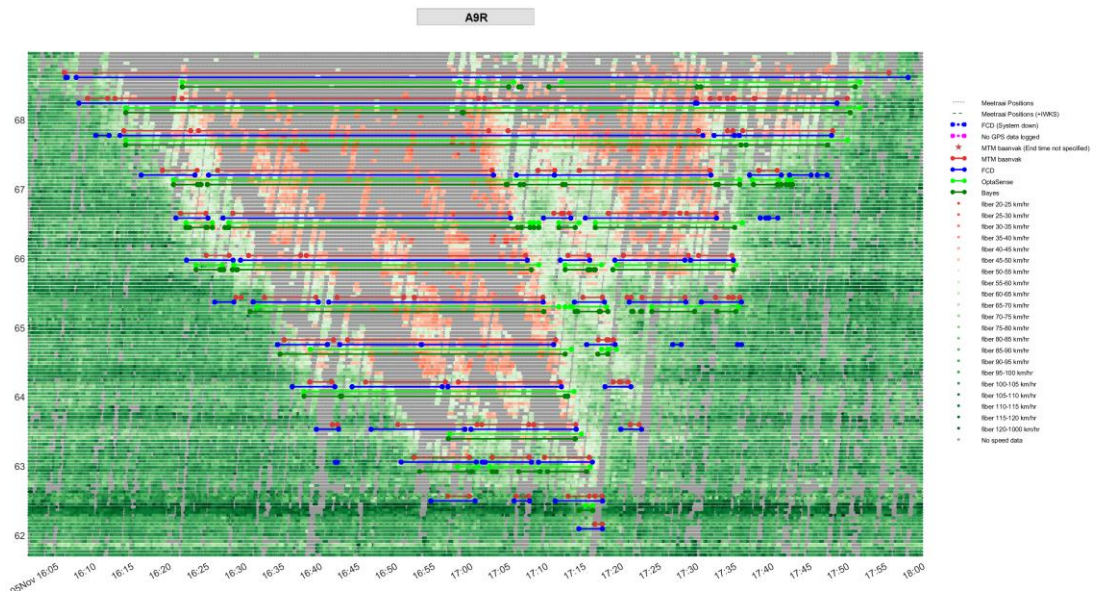


Figure 12: OptaSense speed data. Missing data are plotted as grey dots

Figure 13 shows the OptaSense speed data with an overlay of the OptaSense congestion code in blue. Although the logic behind this congestion code is not known exactly, it appears that the congestion code is triggered when a cluster of low speeds are observed, and then extended to all adjacent patches of missing data. This would at least explain the “square” shape of the identified congestion area and why the method does not initially trigger when missing data occur, but does cover later patches of missing data.

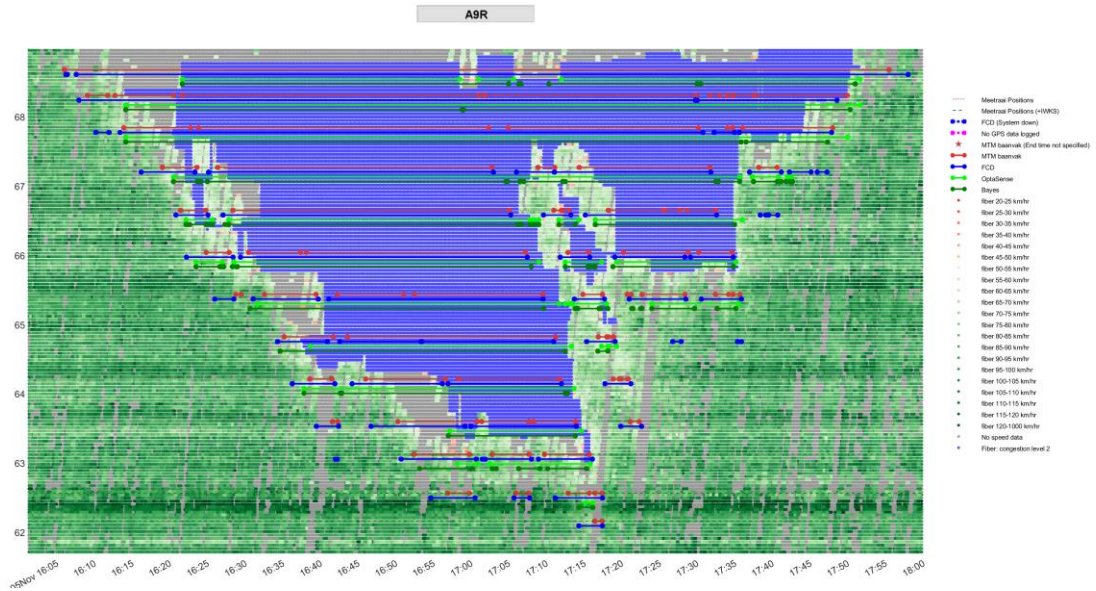


Figure 13: *OptaSense speed data with overlay of OptaSense congestion code (blue)*

Figure 14 and Figure 15 again show the OptaSense speed data but now with an overlay of Bayes probabilities above 0.97 (Figure 14) and 0.98 (Figure 15). There is quite a big difference between the two. Points with the Bayes probability above the 0.97 threshold are much more frequent than points above the 0.98 threshold. Points with a Bayes probability in the range 0.97-0.98 are also frequently found outside the congested area as indicated by the FCD data (see Figure 11).

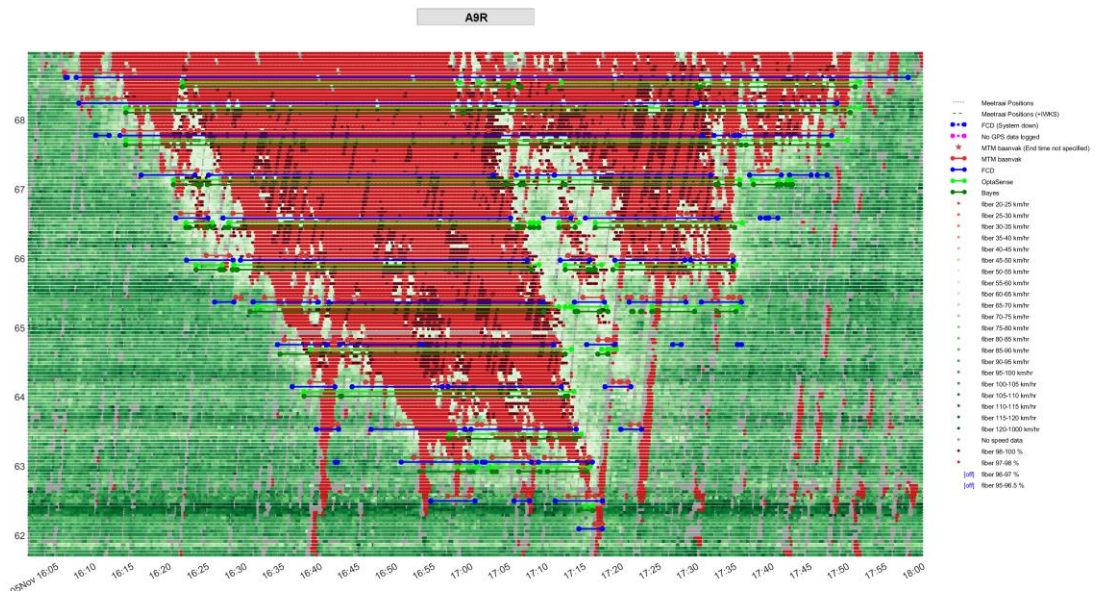


Figure 14: *OptaSense speed data with overlay of Bayes probabilities, starting at 0.97*

Setting the threshold for the Point Estimate at 0.97 would result in an unacceptably high number of false positives, mainly triggered during spells where OptaSense speed data are reported as missing. Setting the threshold at 0.98 (see Figure 15) eliminates these false positives quite efficiently but also fails to identify a large part of the points that are clearly identifiable as

congestion in the FCD dataset. Further tweaking the parameters might improve this but has not been attempted.

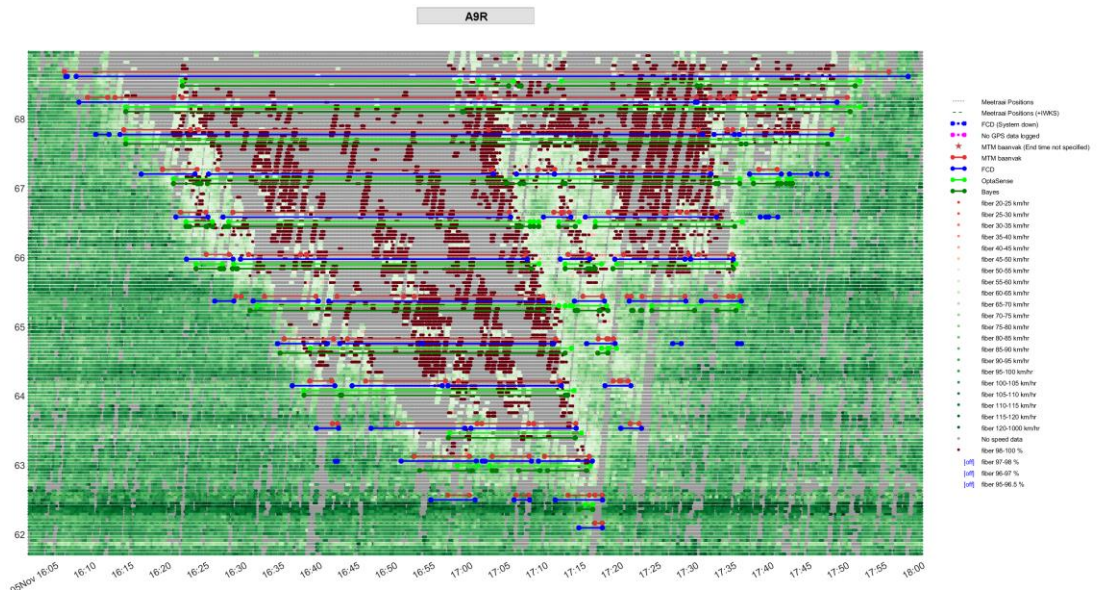


Figure 15: *OptaSense speed data with overlay of Bayes probabilities, starting at 0.98*

AID's are generated if 1 or more locations within the look ahead distance of the AID location have a Bayes probability above 0.98, and maintained until the Bayes probability of all locations in scope are below 0.97.

This logic partly compensates for the relatively high probability of false negatives for separate locations. In other words: even if only a part of the congested locations are identified, there is still a good probability that the AID for the upstream MSI position is triggered.

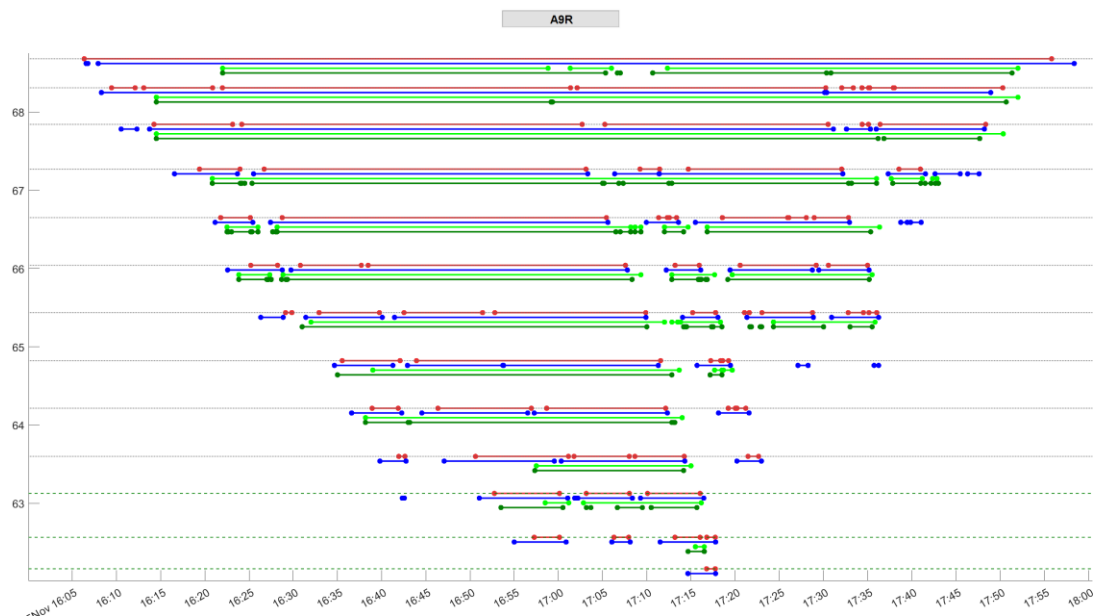


Figure 16: *AID intervals only. Red=MTM, Blue=FCD, Light green=OptaSense congestion code, Dark green=Bayes method*

Figure 16 shows the AID intervals without any overlay. Comparing the light green (based on OptaSense congestion code) and dark green lines (based on Bayes method) learns that the two are very similar. The AID's based on Bayes method have a slightly lower lag than the AID's based on the congestion code.

This finding is based on a single congestion event. In section 4.5.2 a large set of congestion events is inspected.

4.5.2 Full period analysis

Note:

As mentioned the period March-November does not contain many situations with severe congestion. Also no BeMobile FCD data are available for the period in 29 September – 3 November. These data are normally used as a reference, jointly with MTM.

Figure 17 shows an overview of the period 1 September – 21 November. Each marker in this figure indicates the begin or end of an AID period. The dotted grey lines indicate the MSI positions. A separate color is used for each AID method. Figure 10 shows the complete legend for Figure 17 and subsequent figures.

Figure 17 does not show enough detail to compare variants, but it shows that more than other methods the Bayes method appears without confirmation of MTM or FCD, indicating a false positive. This happens especially at positions 52.0, 53.126, and 56.210. Zooming in to these periods shows that these AID have durations below 1 minute. Further tweaking of the Bayes method is required to eliminate these false positives.

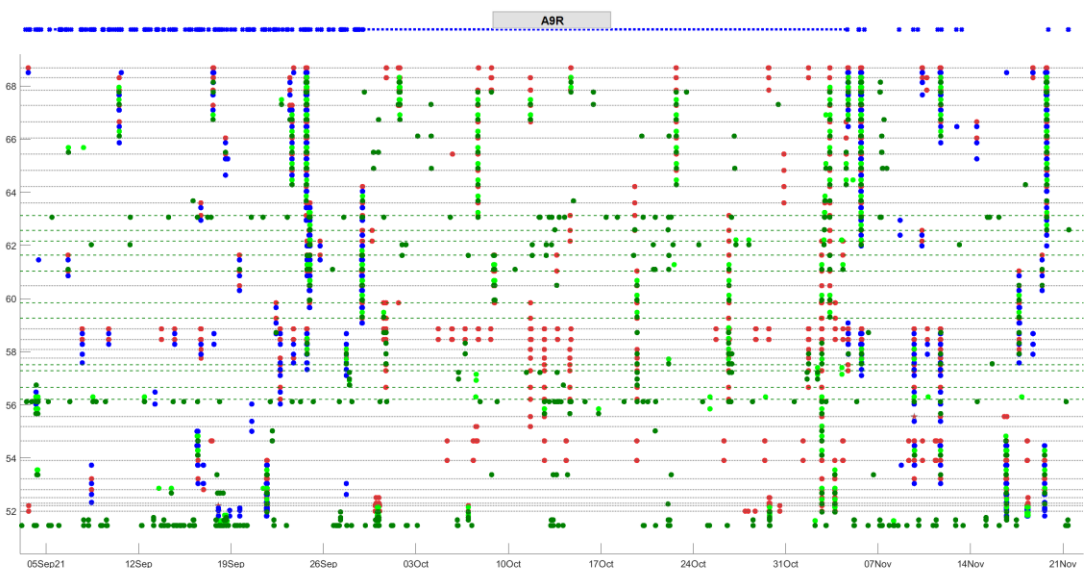


Figure 17: Overview of AID intervals for the interval september-november (legend in Figure 10)

In the study period only a few congestions occur that extend beyond 5 kilometers or 1 hour. The main congestion events are shown in Figure 18, Figure 19 and Figure 20.

At first sight both the OptaSense and the Bayes method overlap the largest part of the MTM and FCD-based AID intervals. Closer inspections shows that at some instances there are considerable delays of 10 minutes or more. On average the Bayes method has a little less lag and a better ability to detect short-lived incidents then the congestion code based method.

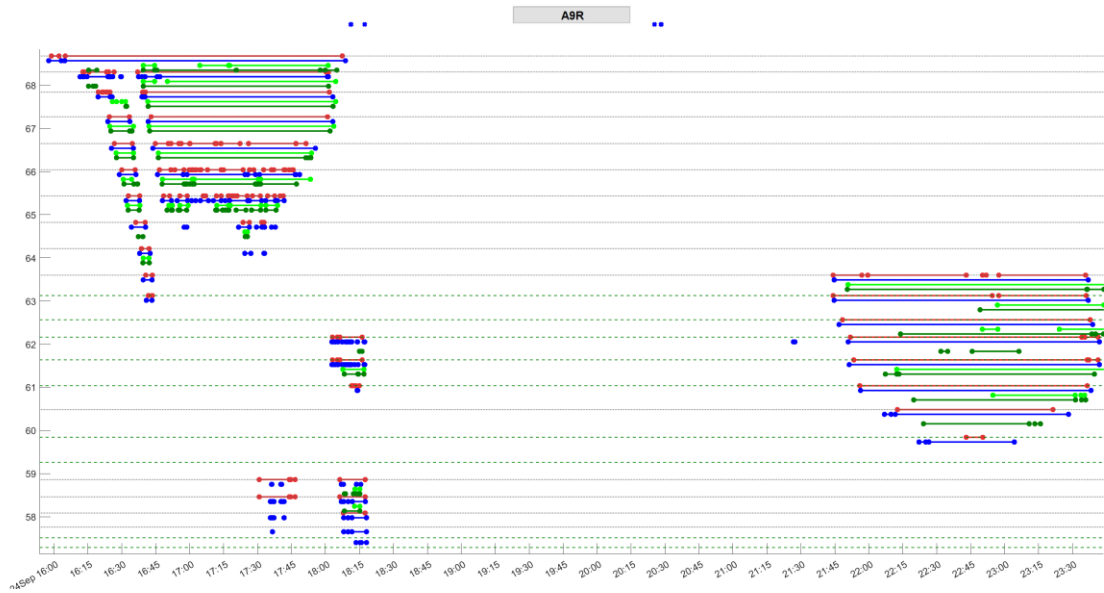


Figure 18: AID intervals on September 24 (legend in Figure 10)

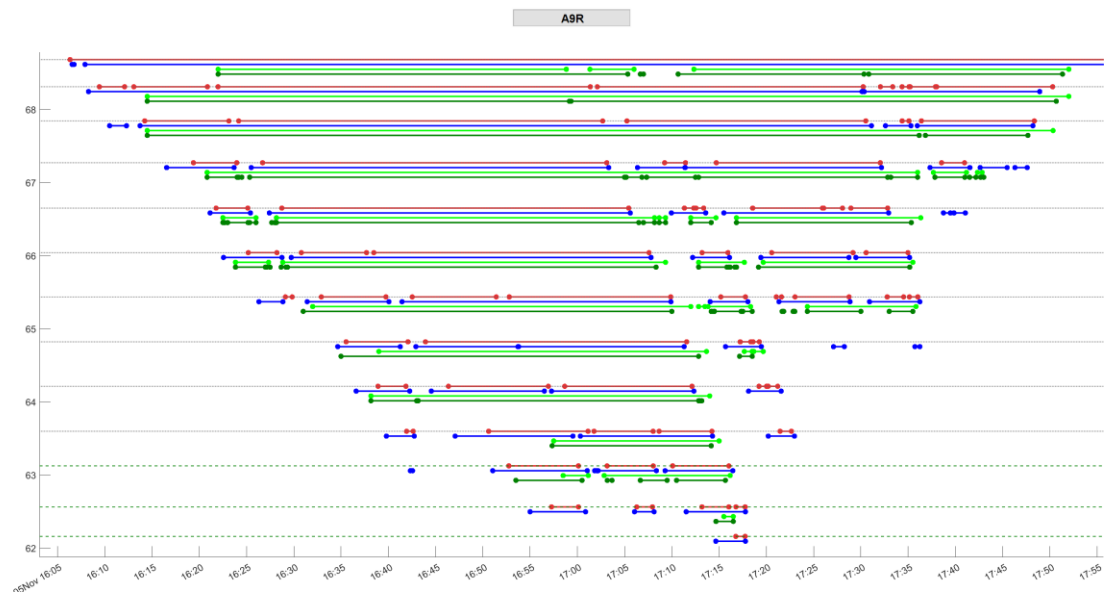


Figure 19: AID intervals on November 5 (legend in Figure 10)

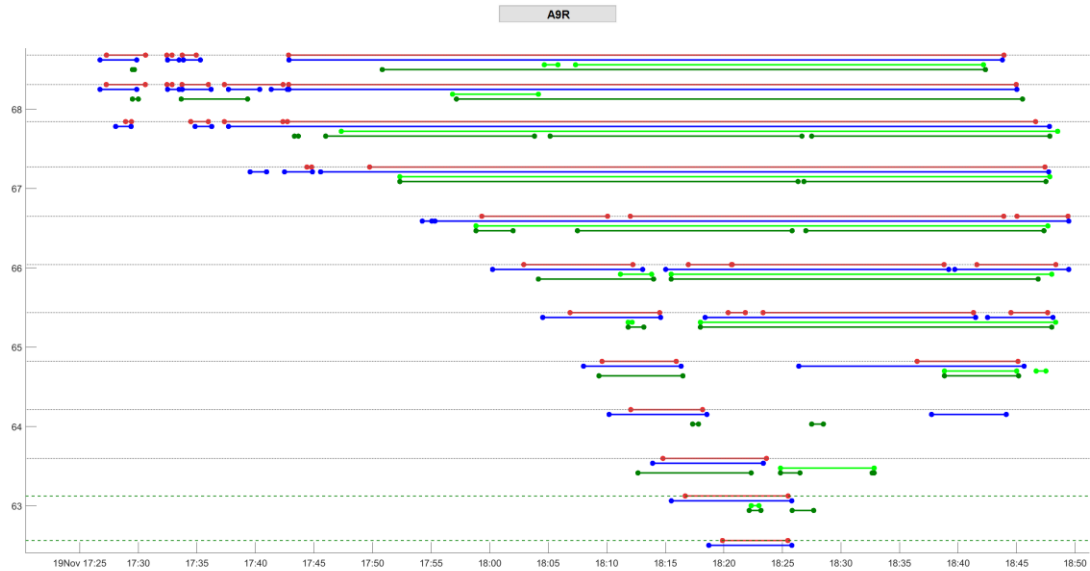


Figure 20: AID intervals on November 19 (legend in Figure 10)

4.5.3 Congestion code comparison before and after system update

Although there is not much material to compare, and only the severe congestion events were considered, the strong impression exists that the OptaSense congestion indicator in the current dataset performs more robust than the congestion code that appears in the period after the system update (see chapter 1).

Examples of this later congestion code can be viewed interactively on <https://modelit.nl/compare/fcdvsfiber.html>. Click the most right column for viewing the congestion code. For convenience we paste the relevant rows of Table 1 from chapter 1:

Table 9: Classification of congestion events

Congestion type	Typical delay	Incident ID's
Heavy	>3 min	06, 08, 20
Very Heavy	>3 min, during >1h	11, 17, 18, 19, 22, 23, 25

The examples can be recalled through the interactive page or by typing the url directly, such as:

https://www.modelit.nl/compare/imagecompare_fiberonly.html?VAR=11

This shows the incident with ID=11. Change the ID to view other examples.

When we consider congestion events of type "Heavy" and "Very Heavy" it appears that in the later version of the system the congestion appears much less frequently. This is especially true for the highest value of the congestion code that was used as point of departure for the current chapter.

4.6 Conclusion

A Bayesian method for estimating AID intervals has been derived, implemented, and applied to OptaSense speed data.

The behavior of the method can be tuned with the following parameters:

Aspect	Tuning option
<i>Propagation</i>	Parameters that describe the probability that congestion propagates to adjacent locations and the probability that a new congestion event is initiated.
<i>Measurement</i>	A likelihood ratio $P[U \text{congested}]/P[U \text{not congested}]$ is defined for each possible outcome U , including the outcome "no data".
<i>AID threshold</i>	Congestion is indicated if the Bayes probability exceeds this value

These parameters make it possible to adapt the method to a wide variety of circumstances. The method has been applied with an initial best guess for the parameters. Compared to the existing OptaSense congestion indicator, the new Bayes method does not offer a spectacular improvement. The resulting AID intervals are more or less comparable.

The main reason to proceed with the new method lies in the possibilities to tune and extend the method while the existing method is a black box. This may be of importance for combining OptaSense data with other data sources.

During the analysis it has also become apparent that the system update that has occurred in the spring of 2022 has had a significant impact on the congestion estimates that OptaSense produces. In the new version reports of congestion are clearly less frequent compared to the earlier version.

There are a few improvements to the Bayes method that should be possible but could not be implemented due to a lack of time.

- At this moment the method still produces too many false positives during prolonged periods of missing data. A mechanism must be implemented to suppress these;
- The method has not been applied to the new version of the OptaSense system. The most visible difference between new and old version is that the time step has been reduced from 10 to 1 second and the spacing from (about) 50 to 10 meters. Maybe as a result of this the data contain many repeated values. Feeding these data directly into a Bayesian method would be suboptimal because it would violate the assumption that the error in each observation is independent from earlier errors. This doesn't hold for repeated values.

5 Comparing the A58 and A9 results

6 Introduction

At first sight the results of the current study seem less optimistic about the usability of the OptaSense solution for AID purposes than the results of an earlier study on the A58 [3].

Although a comparison between the current OptaSense install on the A9 and the earlier install on the A58 was not included in original plan for this study, this chapter will do so anyway.

6.1 Selected conclusions of the A58 study

In 2018 TNO has completed a final report ([3]) about a proof of conduct that was based on a dataset delivered by Cisco who at the time partnered with OptaSense Ltd. OptaSense is also the contract partner for the current project.

The characteristics of the dataset used by TNO are specified below:

<i>Date range</i>	09-02-2017 – 29-08-2017
<i>Study area</i>	A58L km 62-14
<i>Time granularity</i>	60 sec
<i>Space granularity</i>	50m
<i>Data availability</i>	Off-line

Apart from the location, these specifications are identical to the OptaSense system as installed on the A9 originally. This is, before a system update on the A9 system took place that changed the space granularity to 10 meters and the time granularity to 1 second.

TNO has compared this dataset to inductive loop data (ILD) obtained from the A58L by the system that operates the overhead Variable Messages Signs. In the current study this system is referred to as MTM.

Table 10 contains a number of quotes found in the TNO report that are very favorable for the OptaSense solution.

Table 10: *Selected quotes from "Cisco Acoustic Sensing Final Report"*

Page 3	The main conclusion after analyzing all the provided data is that purely on the basis of comparison with existing data, the Cisco Acoustic Sensing system is able to replicate the data provided by the existing RWS ILD system to a high standard.
Page 3	The Cisco Acoustic Sensing system replicates traffic jam information well.
Page 3	All types of shock waves were depicted accurately and the overall picture fits the information currently supplied by the ILD system.

Page 3	It is recommended that future analysis of the Cisco system concentrates on improvements that would take the system beyond the capabilities of the existing ILD system. The current data quality is good enough that for the days where the system is identified as working correctly, and aside from the known limitations (speed and flow), under certain circumstances it is possible that the Cisco system provides just as good or a better replication of the ground truth when compared to the existing ILD system.
Page 11	Visual inspection of the plots at the location level as well as the heat maps (route level) show further that Cisco Acoustic Sensing detectors did not miss any traffic jams throughout the whole study period

6.2 Comparison between A58 and A9 results

The conclusions of the A58 report suggest that the OptaSense system matches or even exceeds the capabilities of the MTM system. Based on, among others, these findings it was decided to install a similar system on the A9 that delivers real time data, and investigate the possibility of deriving real time AID's based on these data.

The current study has investigated the outcomes, first by considering the OptaSense congestion indicator for direct use (chapter 1) and then by inspecting the OptaSense speed data to verify if AID's could be extracted from them (chapter 3). The next two quotes are cited from both chapters:

"Based on visual inspections of the OptaSense congestion indicator jointly with FCD data and loop-based AID's we conclude that in the case of A9R the OptaSense indicator cannot be used as a direct replacement for AID's that are generated from FCD or loop data." (quoted from section 2.8.1)

"Apart from the possible success that the DataLab team might have with an AI based algorithm the detailed data inspection in this chapter has not been successful in showing the way to a robust AID algorithm based on OptaSense speed data for A9R case." (quoted from section 3.6)

There is a stark contrast between these citations and the findings reported in [3].

6.3 Visualizations of the A58 data

As a first step in understanding this contrast the visualizations as used in [3] are copied into this report.

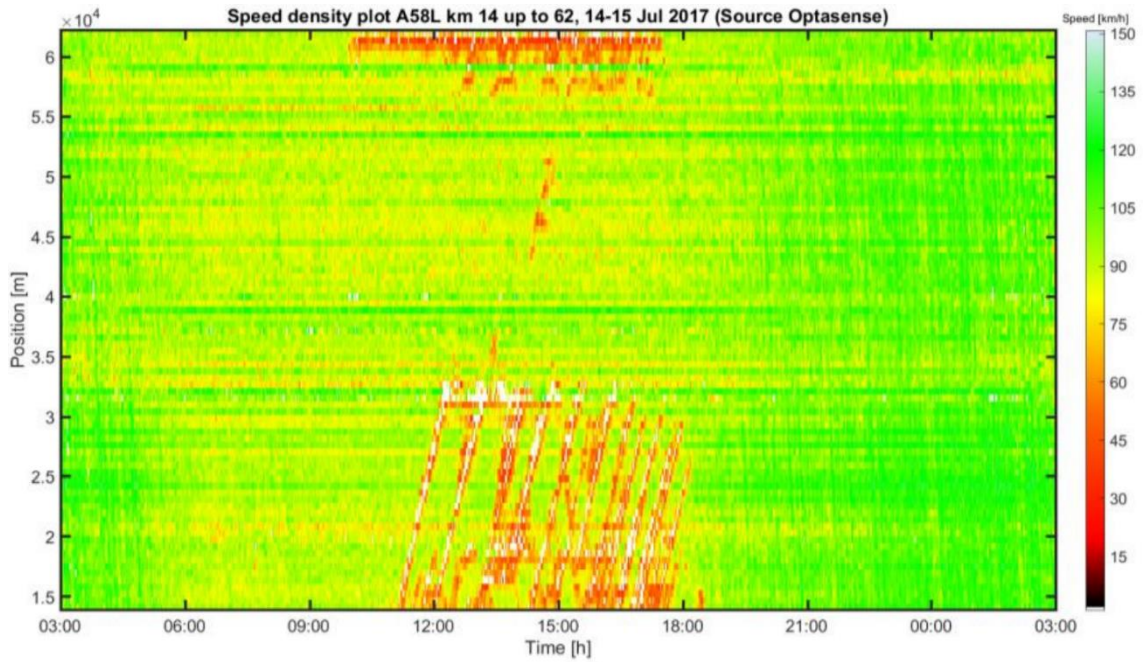


Figure 21: *OptaSense speed data for the A58L (source: [3]). Note that the y-axis shows the road side longitude and that this longitude decreases in the traffic direction, hence km 60 presents the most upstream end of the corridor. Compared to the conventions used in the current reported the graph is "upside-down".*

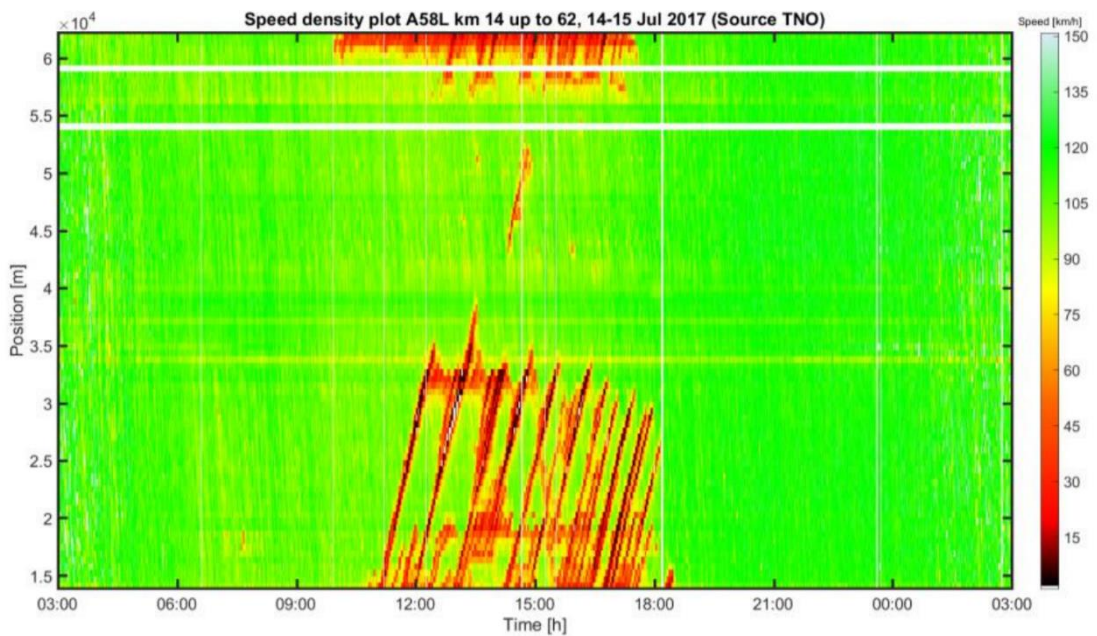


Figure 22: *Corresponding ILS speed data (source: [3]).*

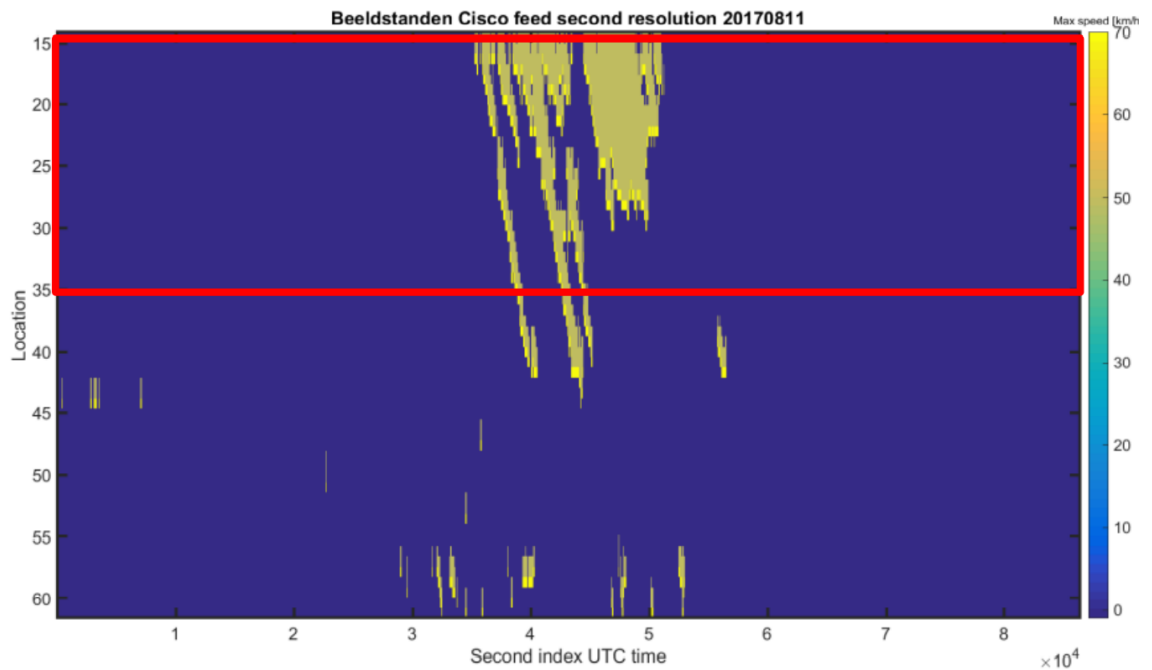


Figure 23: Cisco AID data (source [3]). The accompanying text in [3] does not explain in detail how the AID intervals are derived. Presumably they correspond to the OptaSense congestion indicator, but they could be derived in another way.

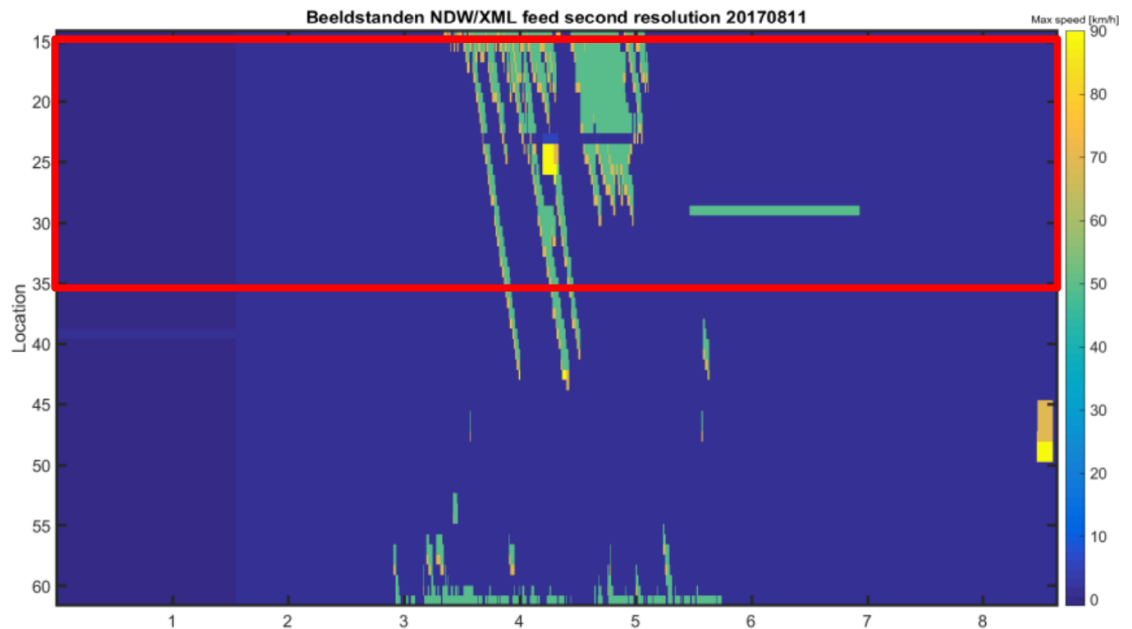


Figure 24: Corresponding MTM AID data (source: [3]). The accompanying explanation in [3] indicates that the AID intervals are derived from the overhead message signs and include both "50" and "70" messages.

6.4 Possible causes for found differences

6.4.1 Possibility 1: difference of scale of the analysis

The visualizations in section 6.3 appear to demonstrate a good correspondence between the OptaSense speed and AID and the ILS based speed and AID. This contradicts the results of the comparisons as presented in chapter 1 and chapter 3 of the current report. Reminder: these comparisons can be accessed through <https://modelit.nl/compare>.

However, the plots shown [3] in are much more aggregate than the plots used in the current study. For example the plots cited in the previous section show 24 hour of data and a stretch of 45 km while the most extensive plot in the current report involves 5600 seconds and 8600 meters. Most of the plots used in the current study focus on an even smaller area. The reason for this is that massive congestion as observed on the A58L did not happen on the A9R during the study period.

Visualizing missing data

One remarkable property of the current dataset is the high proportion of missing data. The current study struggles with interpreting patches of missing data because these could be caused by the presence of congestion or by a lack of traffic. [3] does not mention such a struggle so it seems that missing data are less prominent in the A58 dataset. However, also in the current dataset, the missing data become visible only after zooming in to level that permits noticing intervals smaller than a few minutes wide.

Visualizing AID accuracy

In order for an AID algorithm to be useful in practice it must satisfy quite strict requirements. It is hard quantify these requirements exactly but the following template gives a general idea:

- “The algorithm must respond to a new incident within X seconds”;
- “Once the algorithm responds to an incident, it must predict the upstream location of the incident with a positional accuracy of Y meters”;
- “The proportion of incident detection that in fact turn out to be false positives must be less than Z percent”.

For Dutch motorways typical values for X, Y and Z are X=120 sec, Y=500 meter and Y = 4%. These values correspond to estimated worst case performance of the current MTM system.

Another ratio for these values is that presenting incident warnings that do not meet these criteria would probably confuse road users rather than adequately inform them about upcoming incidents.

Many of the graphs inspected through <https://modelit.nl/compare> do not or not all meet these requirements, hence the conclusions quoted in section 6.2.

Whether or not the time-space graphs in [3] meet the same requirements cannot be verified from the report only as the graphs displayed in the report

lack the level of detail that would be required to do so. A more zoomed in view would be needed for this.

6.4.2 Possibility 2: difference in detection accuracy

The A58L and the A9R both rely on the same detection technique: detecting and interpreting vibrations in a fiber optic cable that is routed alongside the road.

In order for this to work vibrations caused by vehicles must reach the fiber optic cable. It is likely that the rate with which this happens is heavily impacted by factors like:

- the location of the cable;
- the type of subsoil;
- the smoothness of the road surface;
- signal reflections;
- the constructive method used for the cable (for example buried in the soil, cable tray, or routed through a protective casing);
- the presence of noise (for a parallel road or opposed half of the road).

Due to this the performance of the system might vary from site to site. Clues that this is indeed the case are that the system on the A9 produces, as mentioned before, a lot of missing data instances. In [3], missing data seems to play no role of importance.

6.4.3 Possibility 3: Calibration

Each OptaSense system is calibrated before it is taken into operation. In the early stages of the A9 experiment it has been observed that these calibration steps had a big impact on the outcomes that were preliminary at the time. So at least in theory, differences in quality between the A58 and the A9 case might be resolved by an additional calibration effort.

6.4.4 Possibility 4: Changes in the OptaSense system

A third possibility is that since the A58 study the way the raw signals are processed into traffic indicators (speed and congestion indicator) has changed. More specifically, a system update took place during the current project. The most visible effects of this update has been:

- The module that historizes OptaSense data to csv files is no longer available;
- Data for analysis are now be obtained from the real time OptaSense webservice;
- This data has a time resolution of 1 second versus 10 seconds for csv files and a space resolution of 10 meters versus 50 meters for the csv files.

According to OptaSense these changes in resolution have no significant impact on the output data.

This could not be checked immediately as after the update, initially there was no method available to extract data from the system for further

analysis. Only after a few months a method was developed by Modelit that interrogates the OptaSense webservice and logs the data for off-line use.

The data were obtained in this way have been used in chapters 1 and 3. Only when the Bayesian method that was developed in an earlier stage was documented (chapter 4) it was observed that the OptaSense congestion indicator behaves different after the system update. The main difference is that the sensitivity for incidents seems to be reduced. This effect has been illustrated in section 6.5

6.5 Impact of system update during project

In this section the impact of a system update in the spring of 2022 of the OptaSense system is studied. This is done by presenting 5 examples, 3 before the update and two after the update. Each example presents an incident leading to congestion according to MTM and FCD reference data.

When looking at the examples, there is seems that for the examples from data before the update congestion events were adequately covered by the OptaSense congestion status and after the update similar incidents are only partly covered by the OptaSense congestion status.

In our view the examples illustrate that after the update the OptaSense congestion indicator is less sensitive to speed drops than before.

6.5.1 Examples before update

6.5.1.1 Example 1

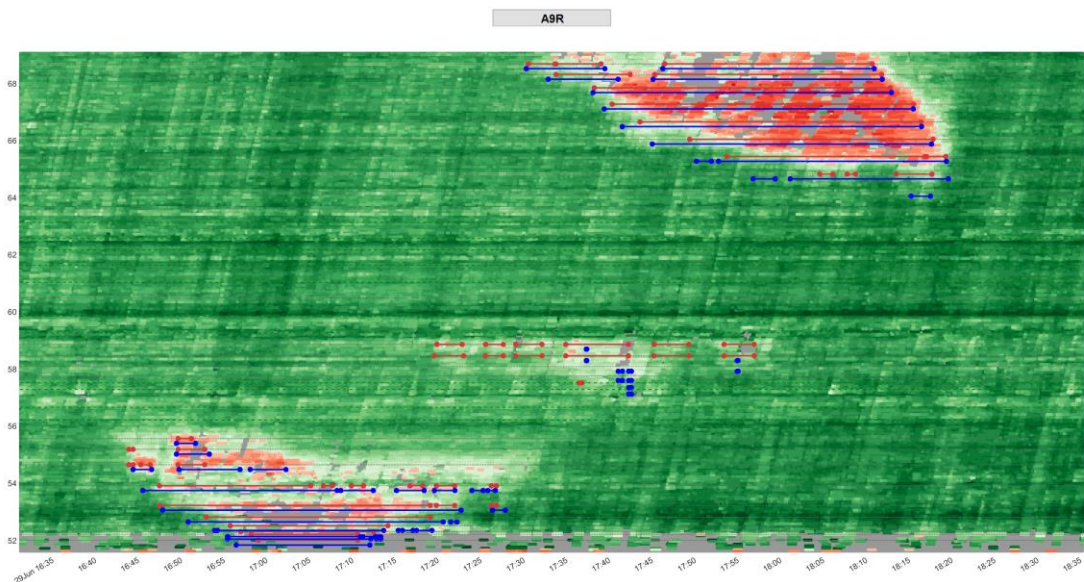


Figure 25: *OptaSense speeds for incident on June 29, 2021. The speed legend is not plotted but is identical to Figure 1 and modelit.nl/compare/spd.png (detailed view). The congestion patches as indicated by MTM and FCD AID's mainly coincide with low speeds and missing data in the OptaSense dataset.*

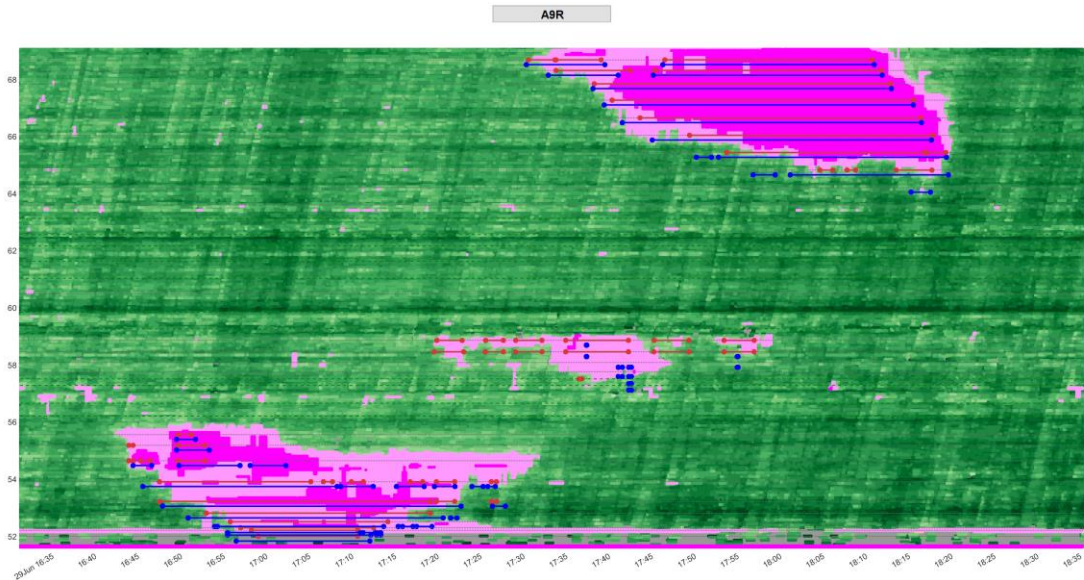


Figure 26: *OptaSense speeds for incident on June 29, 2021 with overlay of OptaSense congestion indicator. Legend: Congestion code 1 (light); Congestion code 2 (severe)*

6.5.1.2 Example 2

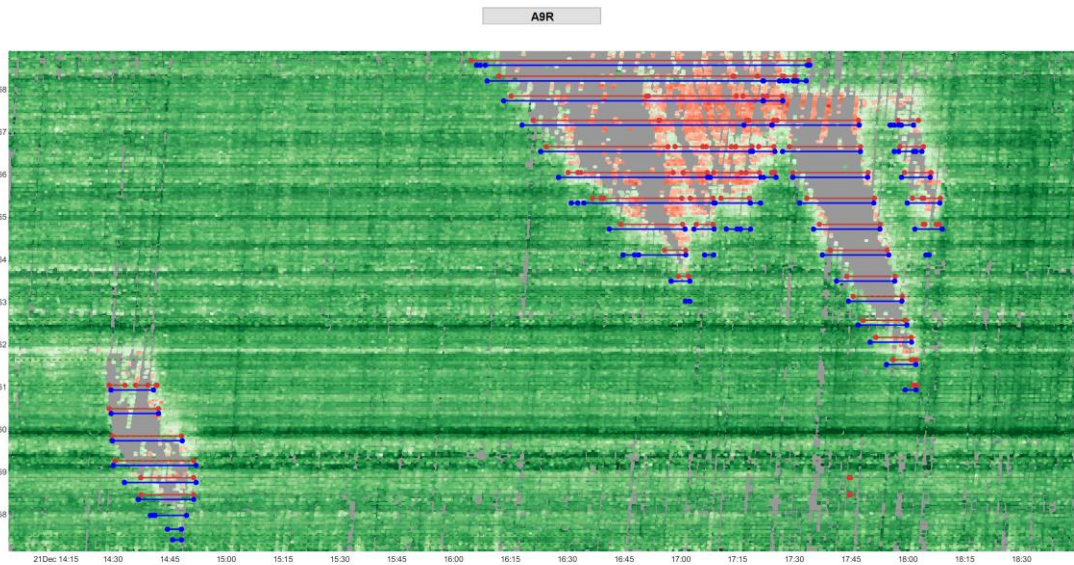


Figure 27: *OptaSense speeds for incident on December 21, 2021. The congestion patches as indicated by MTM and FCD AID's mainly coincide with missing data reports for the OptaSense dataset.*

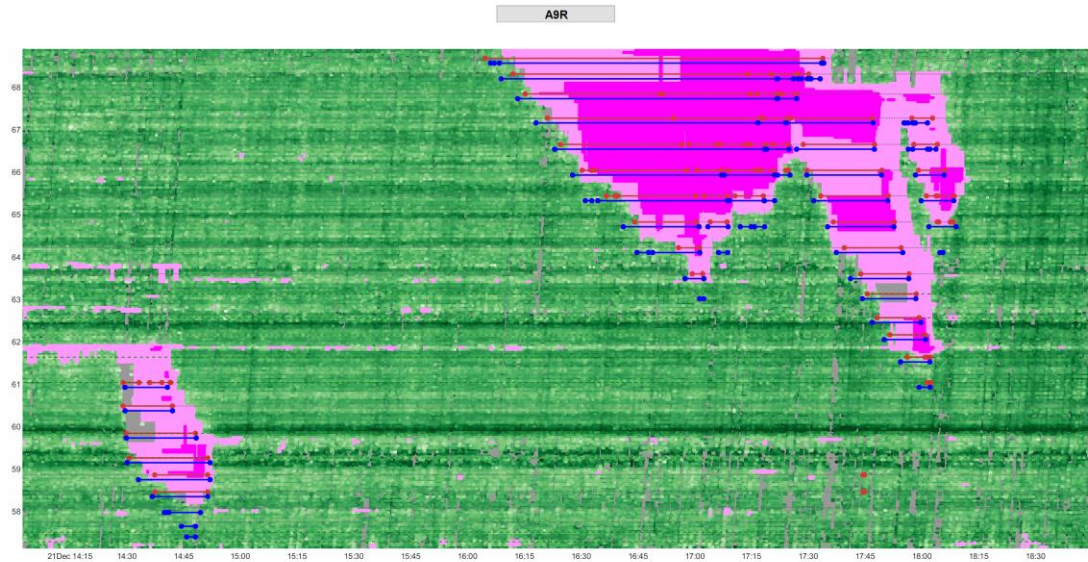


Figure 28: *OptaSense speeds for incident on December 21, 2021 with overlay of OptaSense congestion indicator. Despite observations reporting low speeds are scarce for this incidents, the OptaSense congestion indicator covers large parts of the incident, with congestion code=1 and congestion code 2.*

6.5.1.3 Example 3

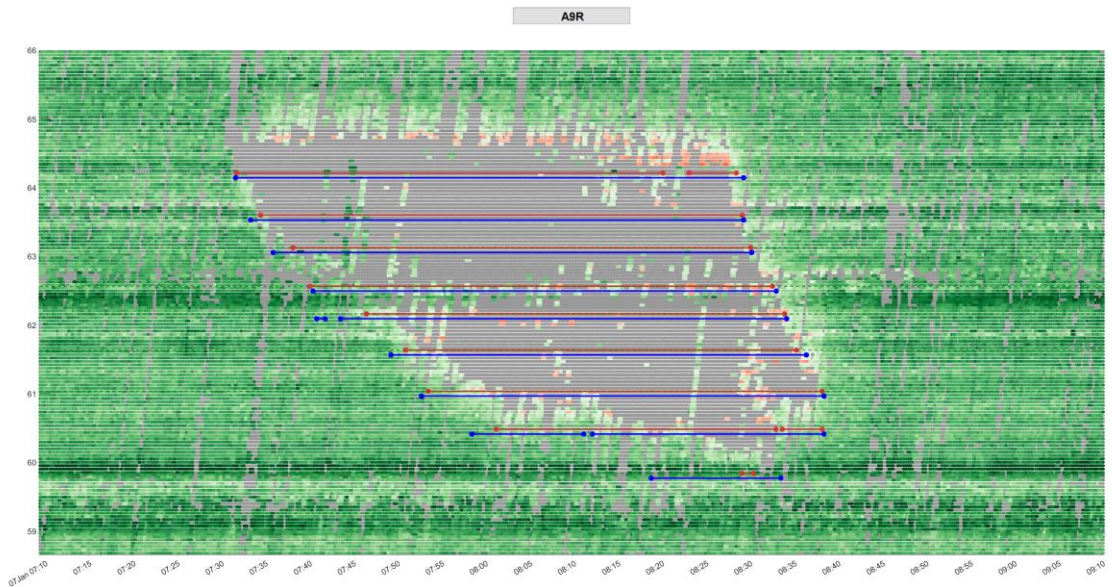


Figure 29: *OptaSense speeds for incident on OptaSense speeds for incident on January 7 2022*

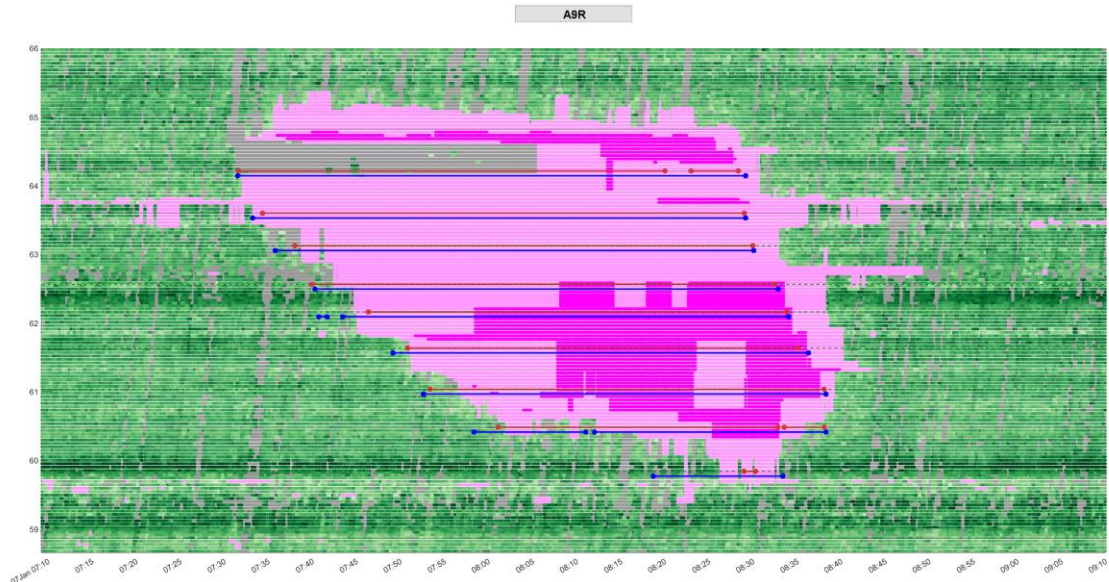


Figure 30: *OptaSense speeds for incident on January 7, 2022 with overlay of OptaSense congestion indicator. Despite observations reporting low speeds are very scarce for this incidents, the OptaSense congestion indicator covers large parts of the incident, although with congestion code=1.*

6.5.2 After system update

6.5.2.1 Example 4

The following example refers to an incident on 6/7/2022. It measures 8.6 km and lasts 5600 seconds, although one can argue that this includes 2 separate incidents. The first incident is shockwave with a width of 13 minutes that propagates upstream over a distance of 8 km. The second incident looks like a local capacity restriction at km 68 that last for 60 minutes but does not propagate more than 3 km.

Measured in occupied area this is the largest incident that is available for analysis after the update.

The second part of the incident is detected by the OptaSense congestion indicator, the first is ignored for the biggest part.

The impact of the incident can also be viewed interactively through:

<https://www.modelit.nl/compare/imagecompare.html?VAR=17>

and:

https://www.modelit.nl/compare/imagecompare_fiberonly.html?VAR=17

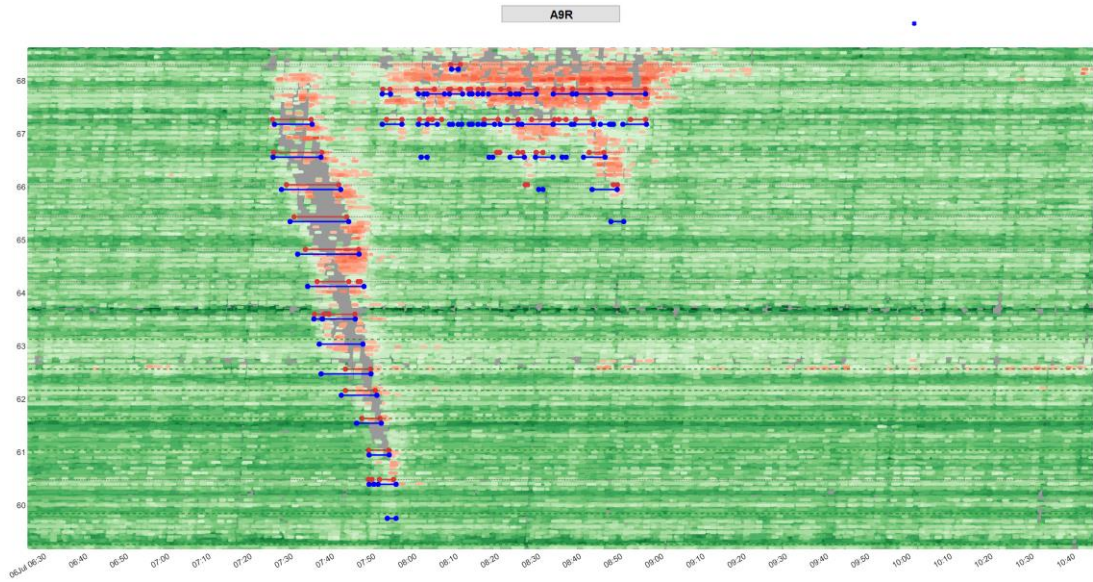


Figure 31: *OptaSense speeds for incident on OptaSense speeds for incident on July 7 2022*

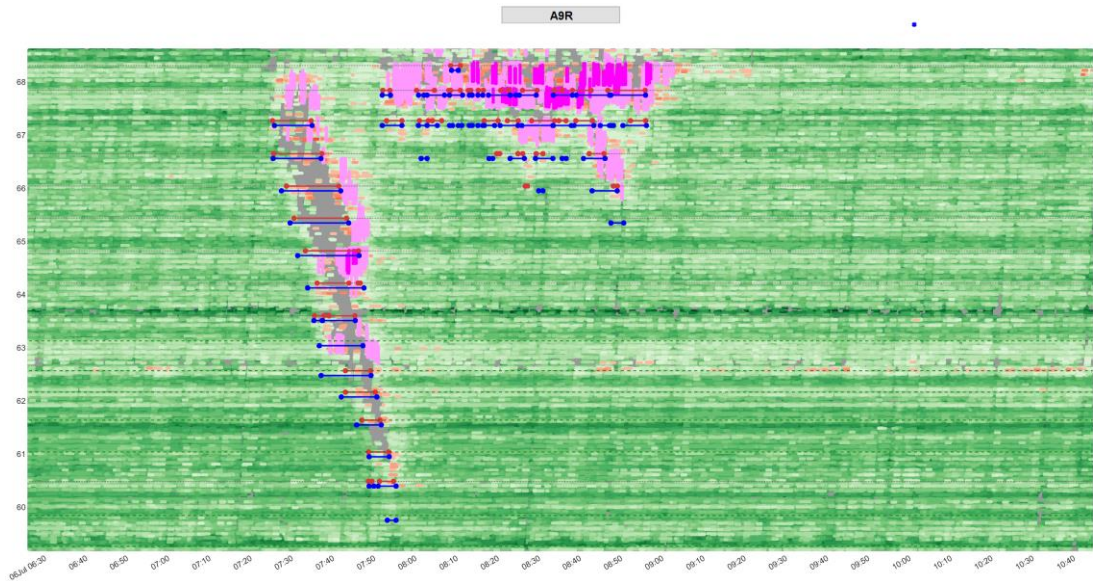


Figure 32: *OptaSense speeds for incident on OptaSense speeds for incident on July 7 2022 with congestion code (mainly status=1) overlaid.*

6.5.2.2 Example 5

The following example refers to an incident that occupies 5400 seconds and 7700 meters. It consists of a shockwave that is 45 minutes wide.

Despite this above average duration of the shockwave the OptaSense congestion status is not adequately triggered.

See:

<https://www.modelit.nl/compare/imagecompare.html?VAR=22>

and:

https://www.modelit.nl/compare/imagecompare_fiberonly.html?VAR=22

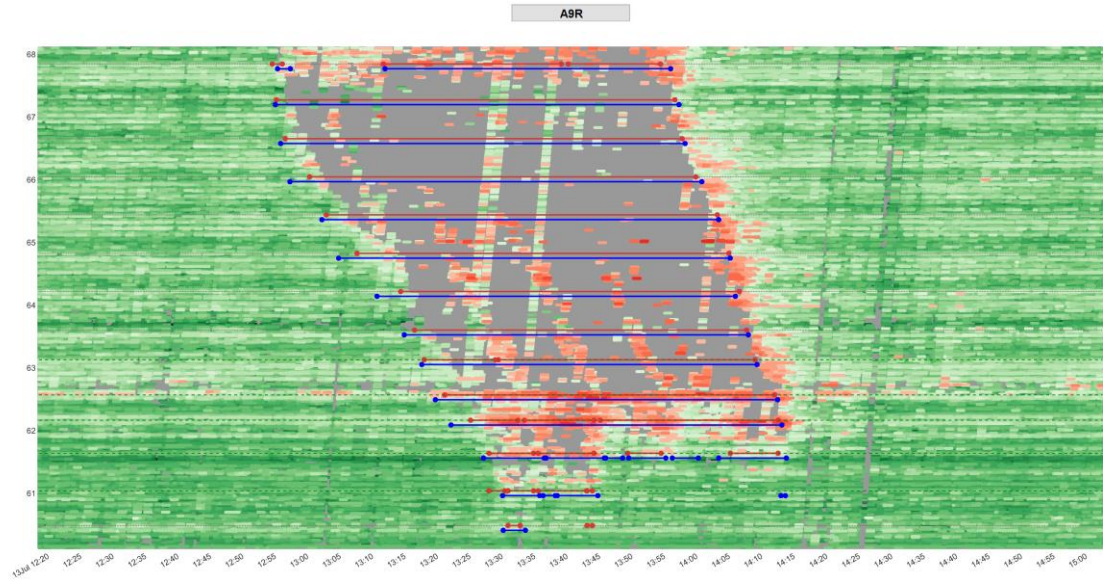


Figure 33: *OptaSense speeds for incident on OptaSense speeds for incident on July 13 2022*

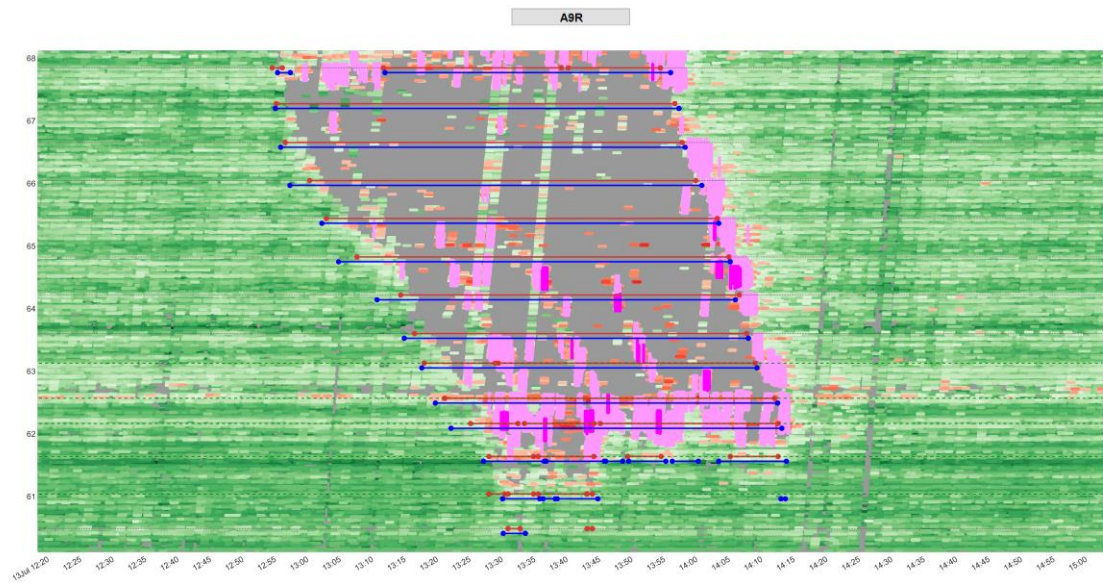


Figure 34: *OptaSense speeds for incident on OptaSense speeds for incident on July 13 2022 with congestion code overlaid.*

6.6 Conclusions

At first sight the results of the current study seem less optimistic about the usability of the OptaSense solution for AID than the results of an earlier study on the A58 [3]. For this there are a number of reasons may or may not be attributable to system changes between the A58 and A9 experiment.

Not attributable to system changes

- In the A58 study (graphical) results have been presented at a much higher aggregation than used in the A9 study. Requirements with respect to positional accuracy and latency imposed in the current study cannot be checked at that aggregated level;
- The incidents analyzed on the A58 have longer length and duration compared to those on the A9. Given the same absolute accuracy, it is to be expected that AID's have a better relative performance on the A58 compared to the A9.

Attributable to system changes

- The OptaSense data used in the A9 study contain more missing periods than those in the A58 study. In part this may be due to the higher level of detail (1sec/10meter versus 60sec/50meter) but especially in congested periods the A9 data display a very high proportion of missing data that seems to be absent in the A58 data;
- During the A9 study a system update has taken place. After the update the sensitivity of the OptaSense congestion indicator has dropped significantly, indicating the algorithm that computes the congestion indicator wasn't scaled properly;
- Possibly (but unproven) the A58 system operated under more favorable physical conditions or was better calibrated.

7 References

- [1] Joe Ciorra, Graham Bruce (2016) OptaSense® Road System Interfaces Tech Note, (REF TDS-14-000096-ITN)
- [2] Modelit (2021), Exploration of OptaSense Road System as a data source for AID algorithms, Report commissioned by Rijkswaterstaat
- [3] Taoufik Bakri, Spencer Milburn (2018) Cisco Acoustic Sensing Final Report, TNO Report, TNO 2018 R10704

8 Appendix: Verifying OptaSense Raw Speed and congestion data

A short look at some of the plots of the data that were harvested from the OptaSense webservice indicates that there is a mismatch between the location of the OptaSense speed and OptaSense congestion indicator.

In this appendix we verify if the data as provided by the OptaSense webservice have been matched with the network correctly.

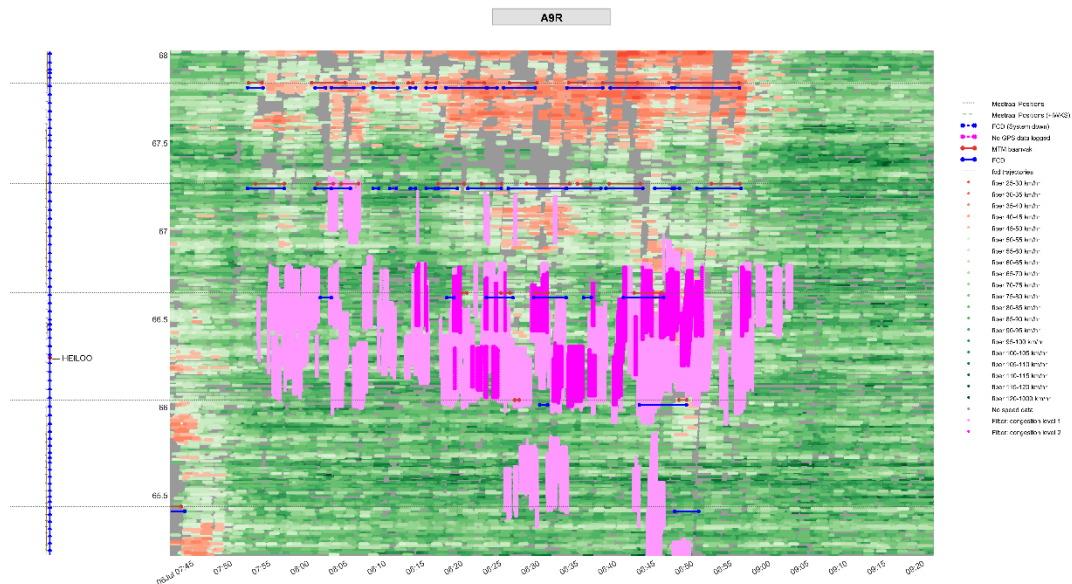


Figure 35 *OptaSense speed (green/red) and OptaSense congestion indicator (purple) shown in single image. A full interactive view is shown at https://modelit.nl/compare/do_compare.html?ref=cng.png&alt=spd.png*

8.1 Location of speed data

First we check if the speeds are plotted at the correct location. We verify this by comparing speeds obtained from OptaSense with speeds obtained from FCD probe data. This is best done by using the interactive view <https://www.modelit.nl/compare/imagecompare.html?VAR=22>. A still is shown below in Figure 37. Comparing both images shows that the location where both images show slow traffic is consistent.

Via the link <https://modelit.nl/compare/fcdvsfiber0.html> a multitude of additional time/space intervals can be inspected (see Figure 36), none of which indicate a mismatch between Fiber and FCD positioning of the speed data points.

Therefore we conclude that the positioning of the speed data points is correct.

FCD vs Fiber speed detection (original congestion specification)

Incident ID	Date	Time	Position (downstream) [km]	Incident duration [sec]	Incident length [km]	FCD vs Fiber	FCD vs Cong.	Fiber vs Cong.
1	20-06-2022	15:46	66.727	1315	5.239	compare	compare	compare
2	20-06-2022	21:50	56.760	422	0.715	compare	compare	compare
3	21-06-2022	10:10	63.894	430	1.828	compare	compare	compare
4	21-06-2022	21:12	59.907	4817	4.529	compare	compare	compare
5	22-06-2022	09:19	64.827	692	2.679	compare	compare	compare
6	22-06-2022	15:27	66.622	1417	3.587	compare	compare	compare
7	23-06-2022	07:07	61.342	611	2.132	compare	compare	compare
8	24-06-2022	17:15	69.180	1777	2.356	compare	compare	compare
9	25-06-2022	11:49	64.696	848	3.301	compare	compare	compare
10	26-06-2022	13:25	66.698	290	0.987	compare	compare	compare
11	27-06-2022	17:22	69.169	5002	5.750	compare	compare	compare
12	27-06-2022	17:29	59.055	336	0.356	compare	compare	compare
13	30-06-2022	13:03	62.101	564	2.832	compare	compare	compare
14	30-06-2022	13:13	66.308	256	0.925	compare	compare	compare
15	03-07-2022	13:23	68.733	2239	0.938	compare	compare	compare
16	05-07-2022	22:19	64.712	769	1.264	compare	compare	compare
17	06-07-2022	07:24	68.660	5616	8.577	compare	compare	compare
18	06-07-2022	16:21	66.671	4767	3.637	compare	compare	compare
19	08-07-2022	15:43	66.627	4828	3.122	compare	compare	compare
20	11-07-2022	16:59	60.005	3594	7.316	compare	compare	compare
21	12-07-2022	17:03	55.359	3634	3.030	compare	compare	compare
22	13-07-2022	12:49	68.416	5365	7.732	compare	compare	compare

Figure 36: Table with shortcuts to "FCD vs Fiber plot". Click 7th column to compare inspect incident speed.

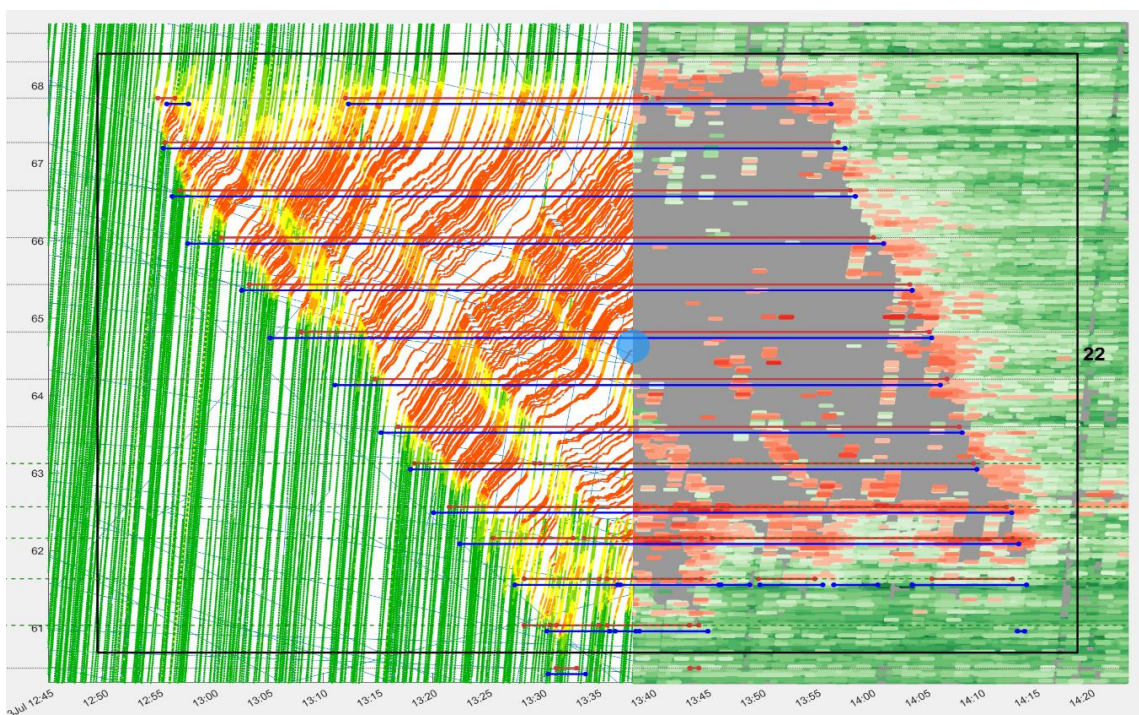
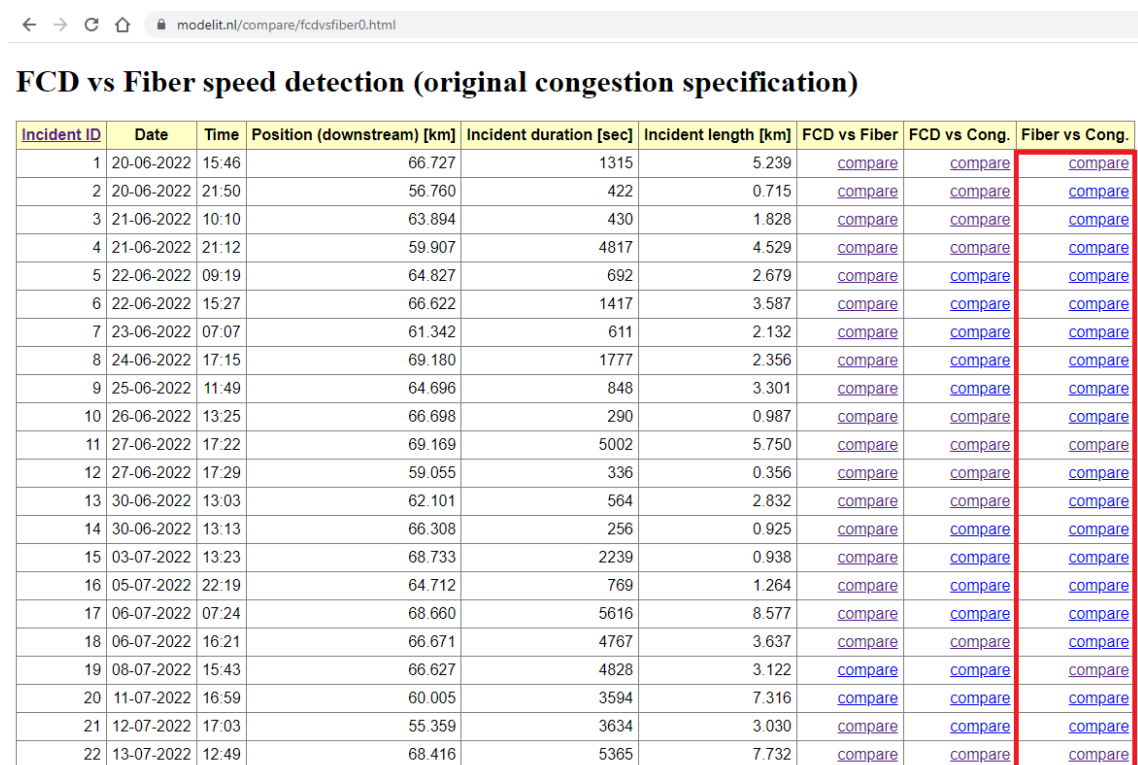


Figure 37: FCD speed (left) vs Fiber speed (right) with moveable divider (middle). See <https://www.modelit.nl/compare/imagecompare.html?VAR=22> for interactive version

8.2 Location of congestion data

The mismatch between the OptaSense speeds and OptaSense congestion location can be inspected, by clicking in the 9th column of the table shown at <https://modelit.nl/compare/>. The difference is most apparent if severe incidents are selected. For example inspect the incident dated 27/06, 17:22.

The link is: <https://www.modelit.nl/compare/imagecompare2.html?VAR=11>



FCD vs Fiber speed detection (original congestion specification)

Incident ID	Date	Time	Position (downstream) [km]	Incident duration [sec]	Incident length [km]	FCD vs Fiber	FCD vs Cong.	Fiber vs Cong.
1	20-06-2022	15:46	66.727	1315	5.239	compare	compare	compare
2	20-06-2022	21:50	56.760	422	0.715	compare	compare	compare
3	21-06-2022	10:10	63.894	430	1.828	compare	compare	compare
4	21-06-2022	21:12	59.907	4817	4.529	compare	compare	compare
5	22-06-2022	09:19	64.827	692	2.679	compare	compare	compare
6	22-06-2022	15:27	66.622	1417	3.587	compare	compare	compare
7	23-06-2022	07:07	61.342	611	2.132	compare	compare	compare
8	24-06-2022	17:15	69.180	1777	2.356	compare	compare	compare
9	25-06-2022	11:49	64.696	848	3.301	compare	compare	compare
10	26-06-2022	13:25	66.698	290	0.987	compare	compare	compare
11	27-06-2022	17:22	69.169	5002	5.750	compare	compare	compare
12	27-06-2022	17:29	59.055	336	0.356	compare	compare	compare
13	30-06-2022	13:03	62.101	564	2.832	compare	compare	compare
14	30-06-2022	13:13	66.308	256	0.925	compare	compare	compare
15	03-07-2022	13:23	68.733	2239	0.938	compare	compare	compare
16	05-07-2022	22:19	64.712	769	1.264	compare	compare	compare
17	06-07-2022	07:24	68.680	5616	8.577	compare	compare	compare
18	06-07-2022	16:21	66.671	4767	3.637	compare	compare	compare
19	08-07-2022	15:43	66.627	4828	3.122	compare	compare	compare
20	11-07-2022	16:59	60.005	3594	7.316	compare	compare	compare
21	12-07-2022	17:03	55.359	3634	3.030	compare	compare	compare
22	13-07-2022	12:49	68.416	5365	7.732	compare	compare	compare

Figure 38: Table with shortcuts to "FCD vs Fiber plot". Click 7th column to compare incident congestion location.

8.3 Drill down to raw data

Now that we have demonstrated that the positioning of the speed data is correct and that the positioning speed and congestion data are inconsistent, we conclude that the position of the congestion shown in Figure 35 is not correct.

For this reason we will now reconstruct the positioning of the congestion, starting with the "raw" messages as harvested from the OptaSense webservice. This a somewhat tedious task that must be done nevertheless, to exclude any errors that might have sneaked in the processing of the data.

8.3.1 Sample

As a sample we focus on the period between 08:13:45 and 08:16:15 local time (subtract 2 hrs. to obtain UTC). This sample is part of the congestion event with "ID=17" (See https://www.modelit.nl/compare/imagecompare0_fiberonly.html?VAR=17). More specifically, we will focus on the even shorter interval 08:14:15 – 08:14:45 which includes a period with "congestion2=true" (indicated a darker shade of purple in Figure 39)

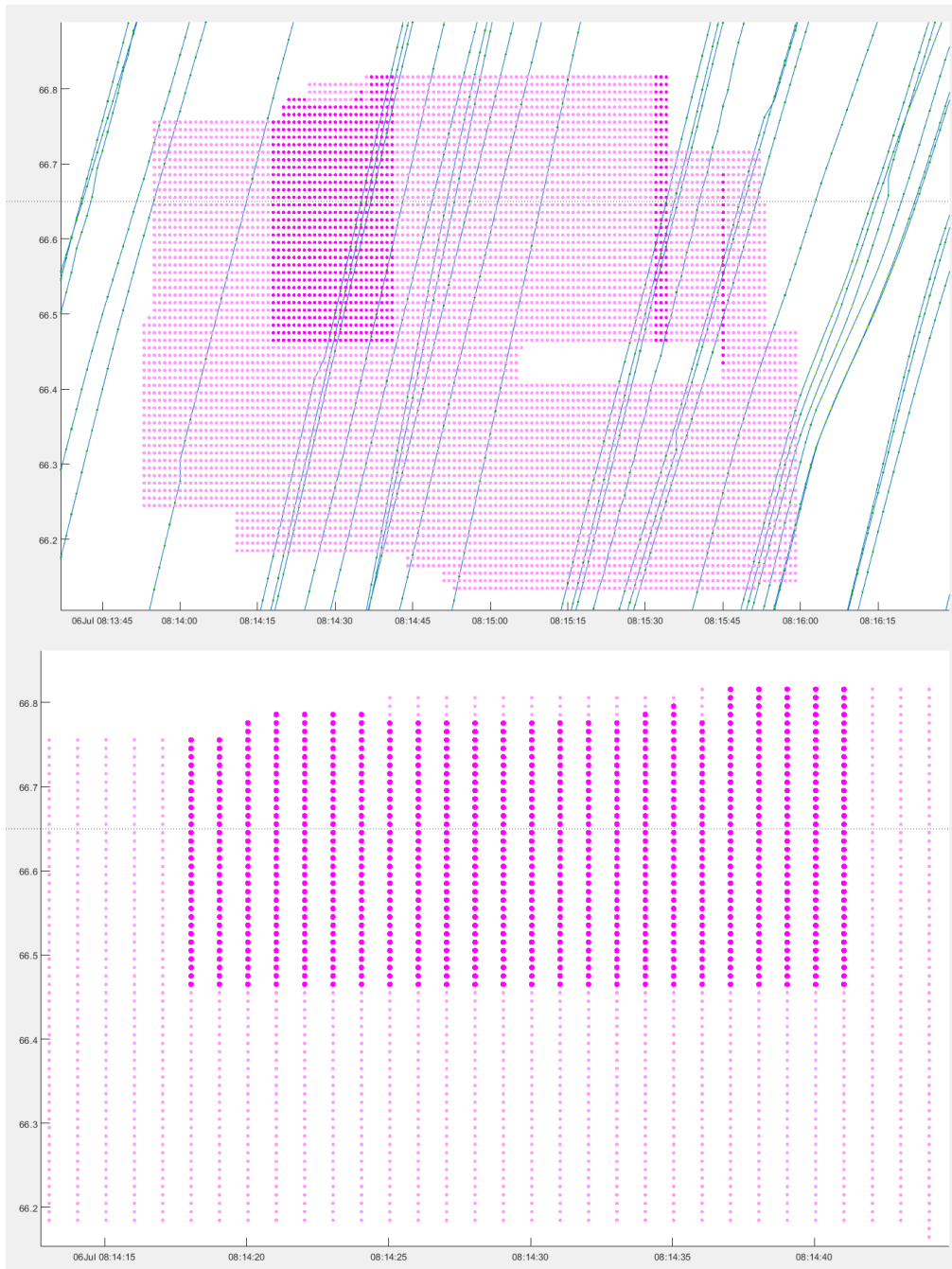


Figure 39: Sample data for inspection

8.3.2 Reading JSON messages

The raw data consist of JSON messages like:

```
[{"carriagewayName":"Northbound","carriagewayID":"A","spanList":[{"startLocation":{"carriagewayName":"Northbound","roadChannel":1492,"fibreID":0,"fibreChannel":1646,"latitude":52.58169736428571,"longitude":4.719188581402598},"endLocation":{"carriagewayName":"Northbound","roadChannel":1521,"fibreID":0,"fibreChannel":1676,"latitude":52.58424392174603,"longitude":4.720110356198413},"startTime":"2022-07-06T06:14:18.043Z","lastUpdateTime":"2022-07-06T06:14:18.043Z"}]},{"carriagewayName":"Southbound","carriagewayID":"B","spanList":[]}]
```

Expanding this message to a data structure yields something like:

Path to parameter	Parameter value
element[1]	Northbound
+----carriagewayName	A
+----carriagewayID	
+----spanList	
+----startLocation	
+----carriagewayName	Northbound
+----roadChannel	1492
+----fibreID	0
+----fibreChannel	1646
+----latitude	52.5817
+----longitude	4.7192
+----endLocation	
+----carriagewayName	Northbound
+----roadChannel	1521
+----fibreID	0
+----fibreChannel	1676
+----latitude	52.5842
+----longitude	4.7201
+----startTime	2022-07-06T06:14:18.043Z
+----lastUpdateTime	2022-07-06T06:14:18.043Z
element[2]	
+----carriagewayName	
+----carriagewayID	Southbound
+----spanList []	B

8.3.3 Tabulating sample data

To get a better overview we will now tabulate the records that relate to the period 08:14:15 – 08:14:45. We are interested in records with carriagewayID=Northbound and in the fields:

Data item	Denote as
spanlist.endLocation.startTime	startTime
spanlist.startLocation.roadChannel	start_roadCh
spanlist.startLocation.longitude	start_long
spanlist.startLocation.latitude	start_lat
spanlist.endLocation.roadChannel	end_roadCh
spanlist.endLocation.longitude	end_long
spanlist.endLocation.latitude	end_lat

In the table below all records with a relevant timestamp for Figure 39 have been collected. The last two columns were not supplied directly by the webserver records, but were computed based on the longitudes and latitudes as shown in the record, using a projection method.

Note that when data are processed in batch mode start_HM and end_HM are read from a lookup table based on roadChannel. There is a fixed mapping between roadChannel en longitude-lattitude coordinates. So in batch mode there is no need to repeat the projection process for each individual datapoint.

startime	start_roadCh	start_long	start_lat	end_roadCh	end_long	end_lat	start_HM	end_HM
2022-07-06T06:14:18.043Z	1492	4.719	52.582	1521	4.720	52.584	66.465	66.756
2022-07-06T06:14:19.041Z	1492	4.719	52.582	1521	4.720	52.584	66.465	66.756
2022-07-06T06:14:20.043Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:21.041Z	1492	4.719	52.582	1524	4.720	52.585	66.465	66.786
2022-07-06T06:14:22.043Z	1492	4.719	52.582	1524	4.720	52.585	66.465	66.786
2022-07-06T06:14:23.041Z	1492	4.719	52.582	1524	4.720	52.585	66.465	66.786
2022-07-06T06:14:24.043Z	1492	4.719	52.582	1524	4.720	52.585	66.465	66.786
2022-07-06T06:14:25.041Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:26.043Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:27.041Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:28.043Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:29.041Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:30.043Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:31.041Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:32.043Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:33.041Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:34.043Z	1492	4.719	52.582	1524	4.720	52.585	66.465	66.786
2022-07-06T06:14:35.041Z	1492	4.719	52.582	1525	4.720	52.585	66.465	66.796
2022-07-06T06:14:36.043Z	1492	4.719	52.582	1523	4.720	52.584	66.465	66.776
2022-07-06T06:14:37.041Z	1492	4.719	52.582	1527	4.720	52.585	66.465	66.816
2022-07-06T06:14:38.043Z	1492	4.719	52.582	1527	4.720	52.585	66.465	66.816
2022-07-06T06:14:39.041Z	1492	4.719	52.582	1527	4.720	52.585	66.465	66.816
2022-07-06T06:14:40.043Z	1492	4.719	52.582	1527	4.720	52.585	66.465	66.816
2022-07-06T06:14:41.041Z	1492	4.719	52.582	1527	4.720	52.585	66.465	66.816

8.3.4 Comparing matched positions

What needs to be done is verifying that the location for start- and end correspond with Figure 39. Looking at this figure, it becomes clear that congestion level "2" occurs approximately from km 66.47 to 66.80. This is in line with start_HM and end_HM as shown in the table.

8.4 Conclusion

The speed- and congestion data as provided by the OptaSense webservice have been mapped to the network and plotted in a correct manner.

8.5 Update 25/07 2022

After submitting the findings presented in this appendix to OptaSense. OptaSense has confirmed our suspicion that the congestion location is mis-reported. To obtain the correct location the data reported through parameter "fibreChannel" must be treated as if it were "roadIndex".

This means the analysis on OptaSense congestion location done until 25/07 must be done again.

The corrected plots can be accessed through the link <https://modelit.nl/compare/fcdvsfiber.html>.

9 Appendix: scripts and tools to specify and publish incident data

This appendix archives the names of the scripts and folders that were used to publish the graphs in the current report and the interactive html pages that the report refers to. This makes it easier to find and re-use these scripts, should it be necessary to reproduce the data and graphs created in the current report at a later date.

9.1 Published data

url	content
https://www.modelit.nl/compare/fcdvsfiber0.html	Compare FCD speeds with Fiber speeds and with original definition of fiber-reported congestion
https://www.modelit.nl/compare/fcdvsfiber.html	Compare FCD speeds with Fiber speeds and with corrected definition of fiber-reported congestion

9.2 Script "recordMTMIncident"

Call	Action
recordMTMIncident(true)	<ul style="list-style-type: none"> Start the application "fcdview" Zoom in to a specific area that demarcates a congestion event Execute the "recordMTMIncident(true)" <p>The indicated congestion event will now be added to the archive "Congestion2022_A9R.mat". The congestion event is based on the current area visible in the "fcdview" application. When done, all congestion events are plotted.</p>
recordMTMIncident	<ul style="list-style-type: none"> Start the application "fcdview" run the command The archive "Congestion2022_A9R.mat" is read and all congestion events in it are plotted in the current fcdview figure
recordMTMIncident ([],'createGraphs')	<ul style="list-style-type: none"> Start the application "fcdview" In the script "recordMTMIncident", update the parameter "TASK" depending on what plot is required (see Table 11) Execute: recordMTMIncident([],'createGraphs')

	For each congestion event recorded earlier a graph will be saved. The content and the storage location of the graph depends on the keyword "TASK".
recordMTMIncident ([], 'createhtml')	<ul style="list-style-type: none"> • Edit the parameter "CREATEHTML" to 1 (for fcdvsfiber0.html) or 2 (for fcdvsfiber.html) • Run the command <p>html pages are created that are ready to be published in the folder Modelit.nl/compare.</p>

Table 11: Possible values for the parameter "TASK"

TASK	SHOWFCD	SHOWFIBER	SHOWCONG	Save png file in subfolder
1	true	false	false	fcd
2	false	true	false	fiber
3	false	true	true	fibercong0 or fibercong (*)

(*) as of 1/8/22 data for this case are recomputed, using the corrected OptaSense congestion location, results are stored in subfolder "fibercong".