# Experimental Simulation for Traffic Wave Dissipation via AI Controlled Vehicles

Andy Lam, Automotive Engineering BEng, B628297
Project Supervisor: Dr. Jingjing Jiang

*Department of Aeronautical and Automotive Engineering,*
*Loughborough University, Leicestershire LE11 3TU, UK*

The advancement of autonomous vehicle technology gives rise to new alternative methods for bulk traffic control. In particular, recent researches have confirmed that a small percentage autonomous vehicles in a traffic stream is capable of influencing the driving behaviour of humans around them. Therefore, it is theoretically possible to maintain highway traffic flow in such a way that stop-and-go wave development is discouraged using existing already technology. In this study, a simulation is produced to provide a quick control algorithm testing platform for future researchers. Modelling after the study conducted in real life by Stern et al. (2018), the simulation involves 10 simulated drivers placed in a single-lane ring to represent a section of a continuous road. The stop-and-go waves produced in the simulation is then validated using empirical data from Stern et al. (2018). Several attempted improvements to the original Stern et al. (2018) controller are tested using the simulation platform. Results demonstrate that modifications made are equally capable in dissipating an existing traffic wave, while also enhancing improvements in traffic throughput by raising the platoon's stable cruising velocity.

# Contents

# 1. Introduction

## 1.1. Motivation

Stop-and-go traffic waves are an experimentally reproducible phenomenon that is a frequent cause of congestions around the world. These waves are generated when traffic density reaches a certain threshold (Yuki Sugiyama, 2008; Shin-ichi Tadaki, 2013), where the driving behaviour of individuals induces instability in the traffic flow. The fluctuation amplifies and become stop-and-go waves as it travels backwards (M.J. Lighthill, 1955). Moreover, lane changing behaviour (Jonghae Suh, 2015) and physical bottlenecks (Jin, 2010) contribute to increasing local traffic density at various points, which encourages fluctuation in vehicle velocity and wave generation. However, these waves are "in principle avoidable" (Raphael E. Stern, 2018) if individual drivers can be influenced to maintain steady and uniform velocity as groups.

A new "slow-in, fast out" or "jam-absorption driving (JAD)" strategy is devised and studied as a possible solution to mitigate congestions by implementing knowledge of kinematic waves into traffic flow (Ryosuke Nishi, 2012; Beaty, 1998; Yohei Taniguchi, 2015). Assuming the congestion is already formed, "slow-in" describes the action to decelerate in advance to avoid becoming a part of the congestion; and "fast-out" describes the increased acceleration following the car ahead without leaving unnecessary gaps to allow drivers behind to accelerate earlier and escape the congestion quickly. This aids in reducing local vehicular density in the congested region which eliminates the continuation of the traffic wave. Using a microscopic model, Taniguchi has proven that an individual driver in JAD behaviour is effective in dissipating small traffic waves without causing a secondary wave (Yohei Taniguchi, 2015).

Currently, highway traffic control focuses on externally imposing variable speed limits (VSL) and variable speed advisories (VSA) to enforce traffic flow similar to JAD behaviour (Albania Nissan, 2011; Markos Papageorgiou A. K., 2002; Andreas Hegyi, 2005). Follow-up studies show limited improvements under enforced VSL while VSA failed to produce a notable positive impact on traffic flow as advisory speeds are often ignored (Markos Papageorgiou E. K., 2008). The cost-effectiveness of such systems remains debatable as these systems rely on expensive infrastructure built along the highway. The fixed location of signs also limits the special resolution of the control system, where smaller, local fluctuations aren't detected or resolved.

The concept of using connected, autonomous vehicles as actuators inserted into the traffic stream is a potential alternative solution in the near future. Studies have proven the capability of connected autonomous vehicles in controlling macroscopic traffic flow properties (Meng Wang, 2016; Maxime Guériau, 2016; Alireza Talebpour, 2016) and inter-vehicle connection allows controller vehicles to instantaneously detect the state of traffic flow and react accordingly (Yibing Wang, 2005; Sébastien Blandin, 2012). In 2018, Stern et al. conducted an experiment using an autonomous vehicle to influence a platoon of 19 human drivers and dissipate a naturally generated traffic wave. The new control strategy tested opens a new door to more effective bulk traffic control strategies and substantially improve traffic throughput and vehicle fuel consumption (Raphael E. Stern, 2018). Following Stern et al. study, this paper aims to provide a basic framework in simulating and testing control algorithms and explores potential improvements to Stern's control strategy.

### 1.2. Problem Statement

This study is inspired by the work of Stern et al. (2018) who have first produced experimental evidence to demonstrate the practical capability of autonomous vehicles controlling human driver behaviour in a closed system. The experiments are conducted by placing 21 human-driven vehicles in a single-lane ring of fixed radius to simulate a continuous road where drivers are instructed to drive "as usual in rush hour traffic". The work offers a new method to efficiently apply JAD behaviour into the traffic flow by influencing the human drivers using a small percentage of autonomous vehicles. However, the experiments require large amounts of physical and labour resources to set up. This will be a challenge for future researchers in developing new control algorithms for the autonomous controller vehicles, whereas a simulated approach allows rapid testing and development of algorithms before live-testing.

To address the issue, this study focuses on applying Stern et al. (2018) control models into a microscopic car-following model and improving its compatibility to simulations, as the controllers are originally designed using dynamics of their test vehicle.

## 2. Experimental Methodology

### 2.1. Base Model Setup

The simulation is a half-scale model of the Stern et al. (2018) experiments using *MATLAB and Simulink*$^{TM}$. It operates as a microscopic, discreet-time model with timestep set to 0.01s. 10 simulated vehicles, modelled as particles are initially stationary and placed evenly apart in a virtual "ring" of 130m circumference. The driver behaviour in each vehicle is simulated by a car-following model, where the model outputs the magnitude of acceleration for every timestep that is translated into velocity and position through integration. Boundary conditions are set universally to keep vehicle velocity or acceleration within reasonable ranges.

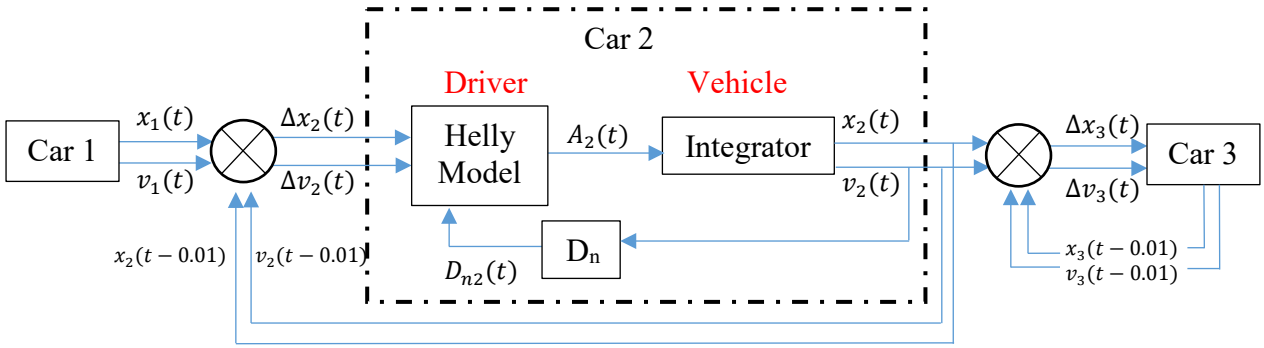



Fig 2.1.1 Simplified block model of simulation setup

### 2.2. Car-Following Model

The Helly model (1959) is chosen for its simplicity to ease computing power required. It is described by a set of linear differential equations. (Constants in A.1)

$$a_i(t) = C_1 \times \Delta v_i(t - T_i) + C_2[\Delta x_i(t - 3.43) - D_{ni}(t - T_i)] \quad (1)$$

$$D_{ni}(t) = 7 + 2v_i(t) \quad (2)$$

The subscript $i$ denotes the car number 1 to 10. The model output $a_i(t)$ is the acceleration decided at time $t$. It relies on external information of distance $\Delta x_i$ and relative velocity $\Delta v_i$ to the vehicle ahead at time $(t\text{-}T_i)$. The constant $T_i$ describes the reaction delay of a human driver and constants $C_{1i}$ and $C_{2i}$ describes the "aggressiveness" of the driver in terms of accelerating and decelerating. $D_{ni}(t)$ is the perceived safe following distance that is relative to vehicle velocity $v_i(t)$ of car $i$ at time $t$. The model follows the basic human logic to drive at the highest

velocity possible until the driver becomes too close to the vehicle ahead. As the modelled vehicles are prone to "crashing" into the vehicle ahead ($\Delta x < 0$), another equation is added to lower the magnitude of fluctuation of acceleration output.

$$A_i(t) = \frac{1}{2}[a_i(t) + \overline{a_i(t_n)}\,] \qquad t_n = \{(t-0.01), (t-0.02) \dots (t-2.5)\} \quad (3)$$

$A_i(t)$ is the final decision for acceleration, taken from the average between $a_i(t)$ and the mean value of $a_i(t)$ over the past 2.5 seconds (250 steps).

### 2.3. Experiment Mechanics

*2.3.1. Simulation without Autonomous Controllers*

Before applying the autonomous vehicle controllers into the test, the model is simulated for 500s. Data from test is then compared to results of Stern et al. (2018) and Sugiyama et al. (2008) to confirm the model designed generates realistic traffic waves. This will provide the baseline for comparison in improvements in further tests.

*2.3.2. Simulation with Autonomous Controllers*

Test setup assumes "Car 1" as the autonomous vehicle where vehicle controllers outlined in *section 3* replaces the "human driver" ("Helly model" block in *Fig 2.1.1*) as the acceleration output when turned on. "Car 1" begins each test under "human" control until traffic fluctuation develops into a stop-and-go wave at 200s (detailed in *section 4.1*). Autonomous control behaviour is turned on when the stop-and-go wave is fully developed at 220s, then turned off at 400s (short test) and 10000s (long test). An additional 100s is simulated after controller off to observe potential lasting controller effects.

Various models are simulated repeatedly with minor changes in settings, controller settings described in *section 3* and results shown in *section 4* are optimum results of each approach. Precise testing procedures as follows:

i. Short Test: Model is simulated for 500s, autonomous controller turned on from fully developed traffic wave at 220s to 400s (explained in *section 4.1 and 4.2*). Data reviewed to confirm controller capability of dissipating traffic wave in short period of time.
ii. Long Test: Model is simulated for 10100s, autonomous controller turned on from 220s to 10000s. Data reviewed to evaluate controller performance in improving traffic flow according to metrics outlined in *section 4.1*.

## 3. Autonomous Vehicle Controllers

The autonomous controllers used in the previously described tests are modelled after Stern et al. (2018) designs. The original design considers the velocity control of an actual vehicle, where the controller output is applied through the ECU as accelerator input. As such, controllers in this present study are broken down into high- and low-level controllers, the former outputs the *command velocity* $v^{cmd}$ as the AI control decision; and the latter outputs the appropriate *acceleration a* to simulate ECU input and vehicle acceleration accordingly (*Fig 3.1*).

The goal is to produce a basic framework to allow future quick testing for control strategies, especially high-level control algorithms. Thus, this study focuses on producing a suitable controller to replace the low-level control used by Stern et al. (2018). Their controller was designed specifically to their test vehicle's properties, which is not suitable for controlling other vehicles, or the simulated vehicle in this study.

In the following sections, *section 3.1* describes in detail on the attempted modelling of the *FollowerStopper* (FS) and *low-level controllers* designed by Stern et al. (2018) along with two proposed replacements for low-level controls and a possible improvement to the *FS controller*. The general concept for the proposed low-level controller replacements is to accurately follow $v^{cmd}$ and avoid sudden fluctuations in acceleration. Therefore, potential wave generation is minimised, which allows the platoon of vehicles to achieve higher velocities without compromising flow stability.

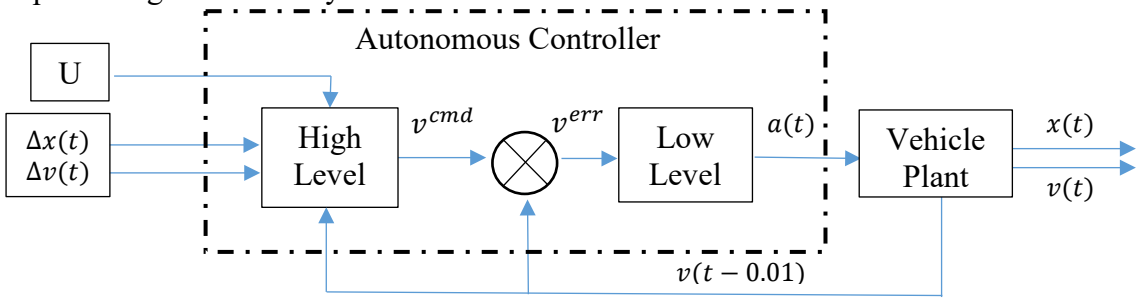



Fig 3.1 General autonomous controller structure

## 3.1. Stern et al. (2018) Original Controller

### *3.1.1. FollowerStopper Controller (FS)*

Taken directly from Stern et al. (2018), the *FS controller* references an externally decided *desired velocity, U* and aims to output $v^{cmd} = U$ as long as safety permits.

$$v^{cmd} = \begin{cases} 0 & if \Delta x \leq \Delta x_1 \\ v\dfrac{\Delta x - \Delta x_1}{\Delta x_2 - \Delta x_1} & if\ \Delta x_1 < \Delta x \leq \Delta x_2 \\ v + (U - v)\dfrac{\Delta x - \Delta x_2}{\Delta x_3 - \Delta x_2} & if\ \Delta x_2 < \Delta x \leq \Delta x_3 \\ U & if\ \Delta x_3 < \Delta x \end{cases} \tag{4}$$

Transition boundaries $\Delta x_{1,2,3}$ are following distances that define following behaviour into 4 components depending on system input of $\Delta x$ and $v$:

- Cruising, $v^{cmd} = U$, whenever it is safe to cruise at $U$
- Stopping, $v^{cmd} = 0$, when controller vehicle becomes too close to vehicle ahead
- Adaptative behaviour 1 and 2, smoother transition between cruising and stopping to avoid excessive deceleration

Transition boundaries in *Eq. 4* are defined as: (Constants in A.2)

$$\Delta x_k = \Delta x_k^0 + \frac{1}{2d_k}(\Delta v)^2 \qquad k = \{1, 2, 3\} \tag{5}$$

$\Delta x_k^0$ is a set of fixed reference distance to vehicle ahead; and *deceleration rates* $d_k$ modify the boundaries based on system input of $\Delta v$. This ensures controller vehicle decelerates earlier when catching up (large negative $\Delta v$ when controller vehicle travels at higher velocity than vehicle ahead).

### *3.1.2. Stern et al (2018) low-level controller*

The low-level controller Stern et al. (2018) used is a multi-mode PID controller with separate gain settings for acceleration and deceleration. The controllers are designed specifically with the test vehicle as a simple first order model based on physical responses to acceleration input. However, the information is not applicable without their test vehicle. Therefore, the main objective of this test focuses on the multi-mode structure. Stern et al. (2018) PID controllers are replaced with transfer functions that produce similar performance characteristics.

$$a(t+1) = \begin{cases} h_1\left(v^{cmd}(t), v(t)\right) & if\ v^{cmd}(t) - v(t) > -0.25 \\ h_2\left(v^{cmd}(t), v(t)\right) & if\ v^{cmd}(t) - v(t) \leq -0.25 \\ 0 & otherwise \end{cases} \quad (6)$$

PID controllers $h_1$ and $h_2$ represents the acceleration and deceleration control respectively, where $h_1$ rises to 1m/s$^2$ and $h_2$ drops to -1m/s$^2$. The controller aims to hover at $v^{cmd}(t)$ -$v(t)$= -0.25m/s using $v^{cmd}(t)$ -$v(t)$ as the error signal, with a specific characteristic of smoother acceleration compared to rapid deceleration when brakes are applied.

### 3.2. Improvements at Low-Level

The following controllers aim to replace Stern et al. (2018) low-level controller to allow *FS controller* output achieve better performance in simulations.

#### *3.2.1. Simple Proportional Controller (P)*

With inspirations taken, a new controller is designed to replace the low-level controller Stern et al. (2018) originally used in their test. The original multi-mode structure is removed as the switching behaviour may become a new source of velocity fluctuation in traffic flow. The new controller is a single PID controller using $v^{cmd}(t)$ -$v(t)$ as error input and outputs $a(t)$. It is expected to perform similarly to Stern et al. (2018) original model, except with smoother velocity output with the switching mechanism eliminated. Initial testing shows smooth, robust control with settings: P=1, I=0, D=0. Therefore, further settings are not tested, and controller is hence named as the *Simple Proportional controller*.

#### *3.2.2. Lookup Table using Transfer Function (Lookup)*

Using a similar logic as the *P controller*, $v^{cmd}(t)$ -$v(t)$ is translated by a hyperbolic tangent function to produce output $a(t)$, bound between ±1m/s. The idea is to reduce velocity fluctuation at larger values of $a(t)$ to prevent initiation of wave development; and more responsive to smaller values of $a(t)$ when minor corrections are required. The controller output should retain the smoothness from the *P controller* but more responsive to *FS controller* command inputs. The transfer function is given as:

$$a(t) = \frac{e^{v^{err}} - e^{-v^{err}}}{e^{v^{err}} + e^{-v^{err}}} \quad (7)$$

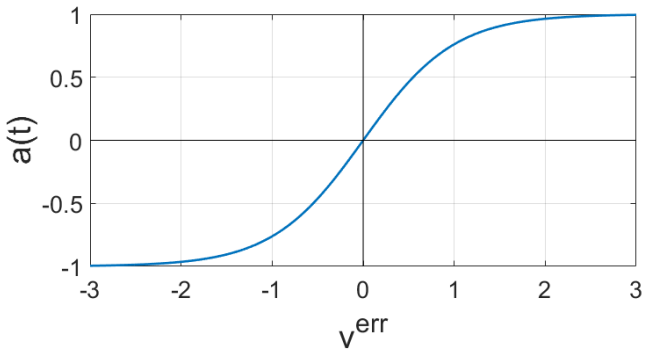



Fig 3.2.2.1 Graphical representation of *Eq.9*

### 3.3. Improvements at High-Level

*3.3.1. FS Controller with Internalised U*

As explained in *section 3.1.1*, the *FS controller* relies on an external operator for the input of *U*. A possible extension to the controller is made to internalise the decision of *U* which simplifies the task of the operator into only deciding whether controller function is switched on depending on traffic condition. The assumption is that the *FS controller* is capable to dissipate the existing traffic wave when *U*=2.5m/s (observed in *Fig 4.3.1.1(a)*). Then, the cruising velocity of the controller vehicle can be slowly increased to a certain limit of stability (explained in *section 4.3*), which is set to 3.55m/s for this study.

$$U(t) = \begin{cases} 2.5 & if \partial a \leq -0.2 \\ f(t) & if -0.2 < \partial a \leq 0.1 \\ 2.5 & if\ 0.1 < \partial a \end{cases} \quad (8)$$

$$f(t) = \begin{cases} U_{old} + 0.00025 & if\ U_{old} < 3 \\ U_{old} + 0.00005 & if\ 3 < U_{old} \leq 3.2 \\ U_{old} + 0.0000006 & if\ 3.4 < U_{old} \leq 3.55 \\ U_{old} & if\ 3.55 < U_{old} \end{cases} \quad (9)$$

$U_{old}$ represents the *U* output of the previous timestep. A new value $\partial a$ is measured as the derivative of acceleration over the past second to detect the occurrence of a slowdown manoeuvre and its magnitude. *f(t)* represents the incremental command under smooth traffic conditions, where the acceleration is gradually reduced as velocity becomes closer to the stability limit.

With the considerations in mind, *U* output will continuously increase in a mild gradient when traffic flow is steady, and stay constant when the stability limit is reached. The output resets to *U*=2.5m/s when large positive and negative values of $\Delta a$ indicate fluctuation in traffic flow.

## 4. Results

### 4.1. Definition and Calculation of Metrics

To effectively compare controller performances, it is important to define the metrics of comparison. For each timestep, data of position in ring *x*, vehicle velocity *v*, distance and relative velocity to vehicle ahead $\Delta x, \Delta v$, and acceleration *a* for each vehicle is recorded. Platoon mean velocity is then computed to represent throughput of the system and velocity standard deviation indicates the magnitude of instability in the traffic flow.

The instantaneous mean velocity and velocity standard deviation of all vehicles is given as:

$$\bar{v} = \frac{1}{10ab} \sum_{i=1}^{10} v_i(t) \quad (10)$$

$$\sigma = \sqrt{\left[ \frac{1}{10} \sum_{i=1}^{10} (v_i(\mathrm{t}) - \bar{v}(\mathrm{t}))^2 \right]} \quad (11)$$

And over a certain time interval *ab*:

$$\overline{v_{ab}} = \frac{1}{10ab} \sum_{t=0.01}^{ab} \bar{v}(t) \quad (12)$$

$$\sigma = \frac{1}{ab-1}\sum_{t=0.01}^{ab}\sqrt{\left[\frac{1}{10}\sum_{i=1}^{10}(v_i(\mathrm{t}) - \bar{v}(\mathrm{t}))^2\right]} \tag{13}$$

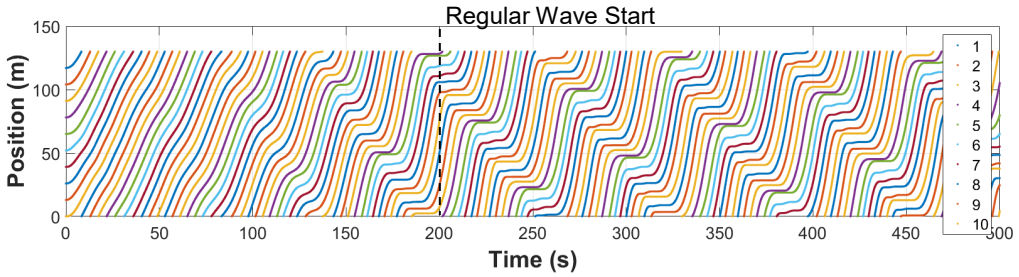
*4.1.1. Defining the Stop-and-go Traffic Wave*

With instantaneous mean velocity and velocity standard deviation being computed for every timestep, the traffic is considered to be under severe wave condition (i.e. stop-and-go behaviour observed) when velocity standard deviation is over 1m/s. The threshold is used to define whether traffic flow is desirable. It is significantly lower than the Stern et al. (2018) definition of 2.5m/s as the simulation does not include noise and other inaccuracies from human driving behaviour and difference of vehicles' physical properties.

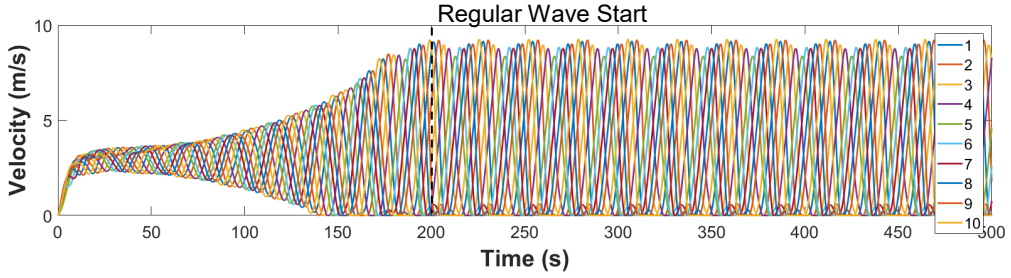
## 4.2. Model without Autonomous Controller

The base model with all 10 vehicles set as "human" drivers is simulated for 500s to validate model as a realistic testing platform. As seen in trajectory and velocity plots in *Fig. 4.2.1*, the traffic system behaves similarly to results from Stern et al. (2018) and Sugiyama et al. (2008). Initially, vehicles are evenly spaced, stationary in the ring and set off to catch up to the vehicle ahead. In absence of external interference, traffic flow immediately experiences fluctuations solely due to interactions with other vehicles. It continues to develop into a recurring stop-and-go wave, where complete stops are first observed at 140s and repeating patterns start at 200s.

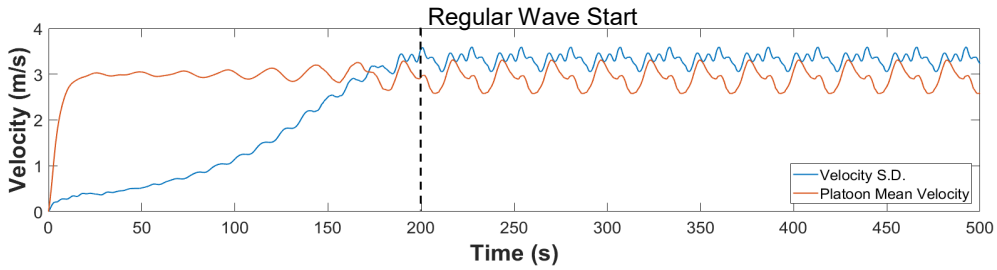
Calculated metrics measured over recurring wave condition are compared to empirical data from Stern et al. (2018) experiment in *Table 4.2.2*. The platoon mean velocity in traffic wave from simulation model is approximately half of empirical data due to properties of the car-following model employed. The Helly model (1959) used describes general highway driving behaviour and predicts the occurrence of traffic waves. However, unlike real life drivers who will tailgate the vehicle ahead, the car-following model maintains a minimal perceived safe distance $D_n$ of 7m even when stationary, as given in *Eq. 2*. Thus, the congested section occupies more space in the ring and reduces distance and time for drivers to move in free flow conditions, resulting in vastly lower mean velocity. On the other hand, velocity standard deviation from simulation model matches empirical data. This indicates that the magnitude of velocity fluctuation in simulation is realistic and the stop-and-go wave simulated behaves as real waves observed by Stern et al. (2018).




(a) Trajectories of all vehicles




(b) Velocity of all vehicles




(c) Velocity std. dev. and mean velocity

Figure 4.2.1 Plots for base model simulation (500s)

**Table 4.2.2 Results comparison to Stern et al. (2018) empirical data**

*Test model data measured between 200-400s, empirical data taken from "wave start" section of experiment A results from Stern et al. (2018).*

| Result | Mean Velocity (m/s) | Velocity Std. Dev. (m/s) |
|---|---|---|
| Base Model Simulation | 2.91 | 3.33 |
| Stern et al. (2018) | 6.28 | 3.31 |

## 4.3. Models with Autonomous Controllers

For *sections 4.3.1-3*, the various low-level controllers are commanded by the *FS controller* to produce a comparison of performances of the low-level controllers. Desired velocity, *U* is given from an external input using a lookup table described in *Table 4.3.1*. The control strategy aims to start at low velocity to stabilise traffic flow, then gradually increase cruising velocity to improve throughput. The gradient of acceleration is reduced in steps to prevent controller vehicle safety manoeuvres due to coming too close to vehicle ahead. For long tests, gradient between 400s to 10000s is set to extremely low value to find upper velocity limit that allows stable flow. Pursuing higher *U* from then results in controller vehicle periodically decelerating to maintain safe distance, which generates a new traffic wave.

**Table 4.3.1 Desired velocity *U* values and break points of lookup table**

*Controllers switched on at 220s for all tests.*

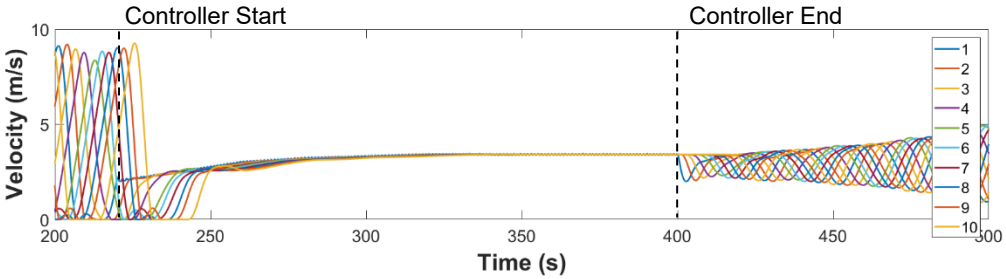
| | Desired Velocity (m/s) | |
|---|---|---|
| Time (s) | Short Test (500s) | Long Test (10100s) |
| 220 | 2 | 2 |
| 260 | 3.0 | 3.0 |
| 320 | 3.4 | 3.4 |
| 400 | 3.4 | 3.4 |
| 10000 | - | 4.0 |

### *4.3.1. Stern et al. (2018) PID Controller*

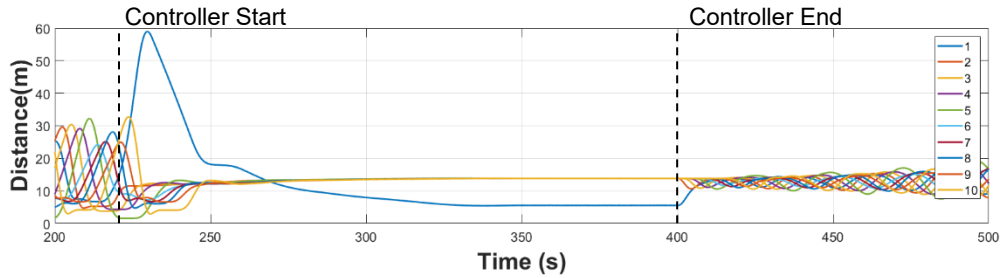
The original design by Stern et al. (2018) is capable of immediately dissipating the existing traffic wave when switched on at 220s. Initially, by cruising at a significantly lower speed than the platoon mean velocity, the controller vehicle managed enforce the "slow-in" strategy (Ryosuke Nishi, 2012; Yohei Taniguchi, 2015) to all vehicles behind the controller vehicle, resulting in rapid dissipation of traffic wave in the short test, as seen in *Fig 4.3.1.1(a)*. The initial concern for the predicted oscillating behaviour by the controller is observed, but the time period and amplitude of oscillation is extremely small that the problem is negligible when applied to a vehicle on the road.

The controller also exhibits satisfying performance in the long test in gradually increasing the stable cruising velocity of traffic flow. As seen in *Fig 4.3.1.*2, peak mean cruising velocity reaches 3.43m/s at 670s, producing a before controller vehicle slows down for safety and generates a new traffic wave. The new wave is expected following Stern et al. (2018) finding, this represents the velocity upper limit of flow stability. As desired velocity $U$ is not set to plateau when the stability limit is found, the controller vehicle repeatedly attempts to cruise at a higher velocity, resulting in excessive acceleration before being forced to slow down when it catches up to the vehicle ahead. The entire platoon of vehicles behind the controller vehicle mirrors the behaviour, causing the platoon mean velocity and velocity standard deviation to exhibit periodic wave behaviour as the secondary wave is caused solely by the controller vehicle's behaviour.

However, as seen in *Figure 4.3.1.1(b),* as cruising velocity increases, the controller vehicle keeps a much smaller distance from vehicle ahead compared to other vehicles in the system. The problem becomes amplified in the long test as car 1 becomes as close as 5.2m from car 10 at peak cruising velocity, compared to platoon average of 13.9m. This may cause concerns when real human drivers (or "car 10" in simulation) are being tailgated at a close distance, which may result in unexpectable human responses.




(a) Velocity of all vehicles




(b) Distance to vehicle ahead of all vehicles

Fig. 4.3.1.1 Velocity and distance plots in *Stern et al.(2018) PID controller* short test (500s)

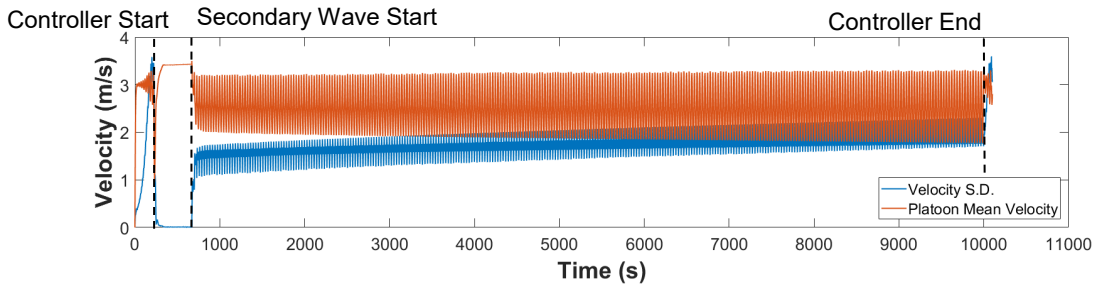



Fig. 4.3.1.2 Velocity std. dev. and mean velocity in *Stern et al.(2018) PID controller* long test (10100s)

**Table 4.3.1.3 Velocity std. dev. and mean velocity in *Stern et al.(2018) PID controller* long test**

| Interval | Time (s) | Interval mean velocity (m/s) | Interval velocity std. dev. (m/s) |
|---|---|---|---|
| Uncontrolled Wave | 200-220 | 2.91 | 3.31 |
| Short Test Period | 220-400 | 3.03 | 0.27 |
| Long Test (stable) | 400-670 | 3.42 | 0.01 |
| Long Test (unstable) | 670-10000 | 2.52 | 1.74 |

*4.3.2. Simple Proportional Controller (P)*

As expected, the *P Controller* performed similarly as the *Stern et al. (2018) PID controller* in the short test (as seen in *Fig 4.3.2.1)* as changes made are minor. The main differences observed are found in results in the long tests. The absence of high frequency oscillation behaviour following the removal of multi-mode mechanism allowed maximum cruising velocity to reach 3.57m/s at 3312s. The accuracy of control output from *P Controller* provides a smoother and more stable control behaviour. Results as seen in *Fig 4.3.2.2 and Table 4.3.2.3* shows that platoon mean velocity and velocity standard deviation when cruising over stable limit to be better than before controller implementation and results from *Stern et al(2018) PID controller*. Thus, a transition period between 3312s to 8000s is observed as traffic flow becomes unstable yet not breaking the threshold for severe wave conditions. The problem of the controller vehicle tailgating the vehicle ahead at a very close distance persists, as the gap reduces to 2.63m at peak velocity in *P controller* test.

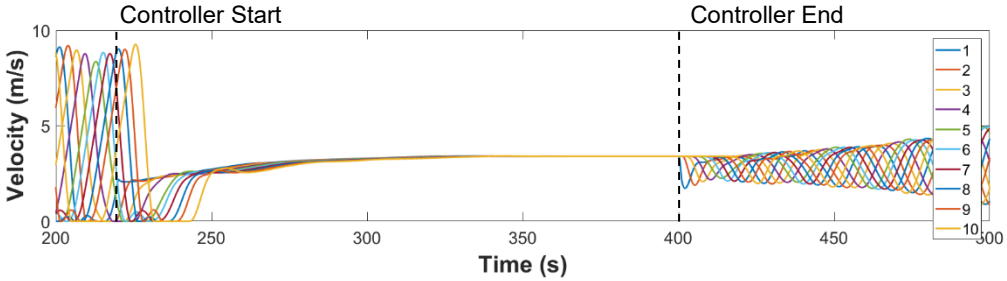



Fig. 4.3.2.1 Velocity plot of all vehicles in *P controller* short test (500s)

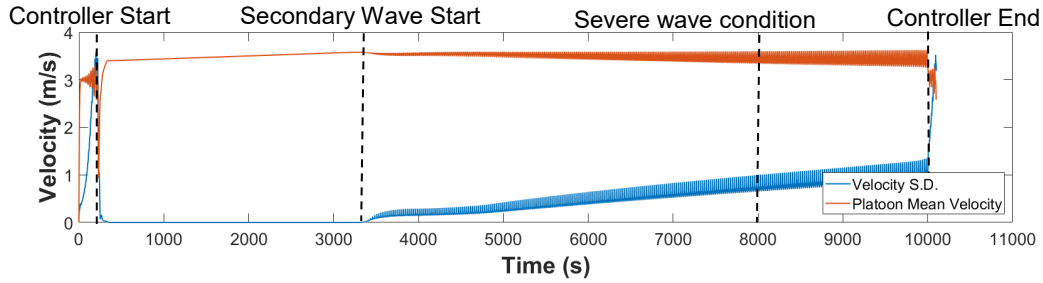



Fig. 4.3.2.2 Velocity std. dev. and mean velocity in *P controller* long test (10100s)

**Table 4.3.2.3 Velocity std. dev. and mean velocity in *P controller* long test**

| Interval | Time (s) | Interval mean velocity (m/s) | Interval velocity std. dev. (m/s) |
|---|---|---|---|
| Uncontrolled Wave | 200-220 | 2.91 | 3.31 |
| Short Test Period | 220-400 | 3.02 | 0.27 |
| Long Test (stable) | 400-3312 | 3.49 | 0.00 |
| Long Test (unstable) | 3312-8000 | 3.51 | 0.38 |
| Long Test (severe waves) | 8000-10000 | 3.44 | 0.88 |

*4.3.3. Lookup Table using Transfer Function (Lookup)*

Comparing *Fig. 4.3.3.1-2* to *Fig. 4.3.2.1-2*, the velocity plots show that the *Lookup controller* performed exactly the same as the *P controller* before reaching the peak cruising velocity of 3.57m/s at 3312s. A notable observation seen in *Fig 4.3.3.2* is that the traffic flow enters severe wave condition much quicker compared to results the *P controller*. It is unknown whether the phenomenon is due to the acceleration limits of ±1m/s$^2$ or the amplified output at smaller *a(t)* output from *Eq. 9*.

The results would indicate the extension is unnecessary as it does not produce any improvements to the *P controller*.

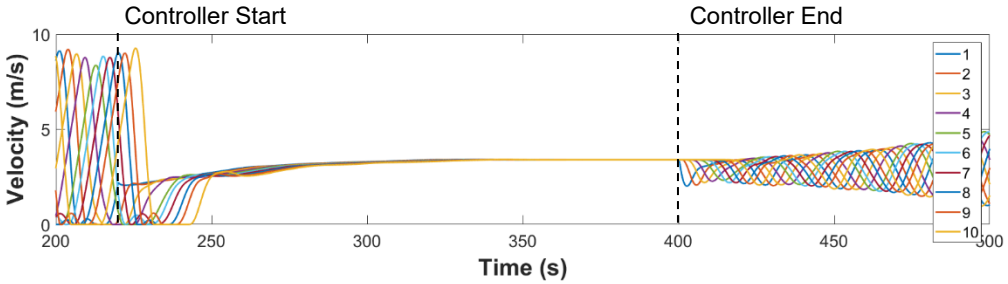



Fig. 4.3.3.1 Velocity plot of all vehicles in *Lookup controller* short test (500s)

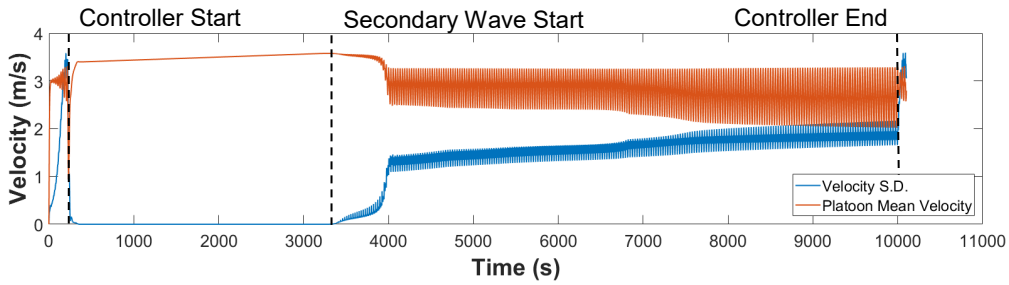



Fig. 4.3.3.2 Velocity std. dev. and mean velocity in *Lookup controller* long test (10100s)

**Table 4.3.3.3 Velocity std. dev. and mean velocity in *Lookup controller* long test**

| Interval | Time (s) | Interval mean velocity (m/s) | Interval velocity std. dev. (m/s) |
|---|---|---|---|
| Uncontrolled Wave | 200-220 | 2.91 | 3.31 |
| Short Test Period | 220-400 | 3.02 | 0.27 |
| Long Test (stable) | 400-3312 | 3.49 | 0.00 |
| Long Test (unstable) | 3312-10000 | 2.85 | 1.48 |

*4.3.4. High-Level extension with internalised desired velocity U (Int. U extension)*

From previous sections, the *P controller* exhibits the best performance between the low-level controllers tested. Thus, the experimental setup uses the *P controller* low-level with the *Int. U extension* added to the *FS* high-level control to produce the best attainable controller performance. As seen in *Fig. 4.3.4.1*, the extension does not affect actual performance of the controller's capability in dissipating the traffic wave. The principle of implementing the extension is to enable the controller to regulate desired velocity *U* and subsequently the platoon mean velocity without the help of external operators. As seen in *Fig. 4.3.4.2*, the plot in *(a)* shows results of the controller with *Int. U extension* operating exactly as *Eq.7* describes and the maximum *U* output is set to 3.55m/s to avoid platoon mean velocity exceeding the stability limit; whereas in *(b)*, the maximum *U* output is set to 3.6m/s to allow mean velocity to surpass the stability limit, and demonstrate the controller's ability to recover quickly from random events of instability. Hence, instability due to pursuing higher velocities is not observed in this test, where traffic flow remains stable for the majority of duration when controller is on.

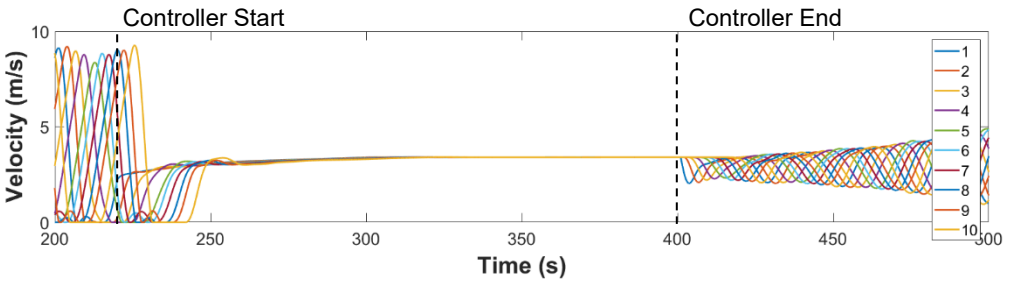
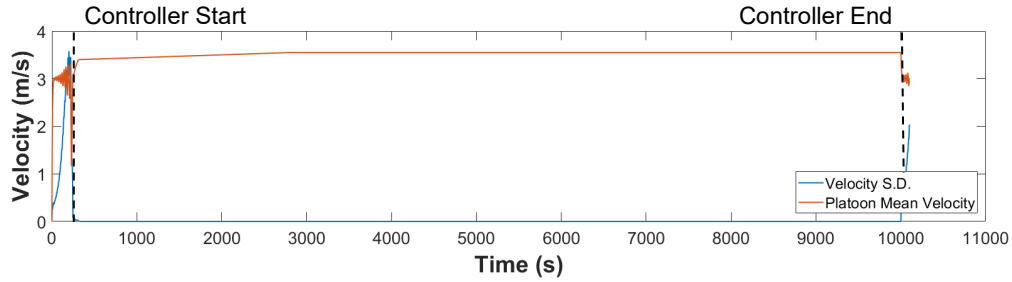
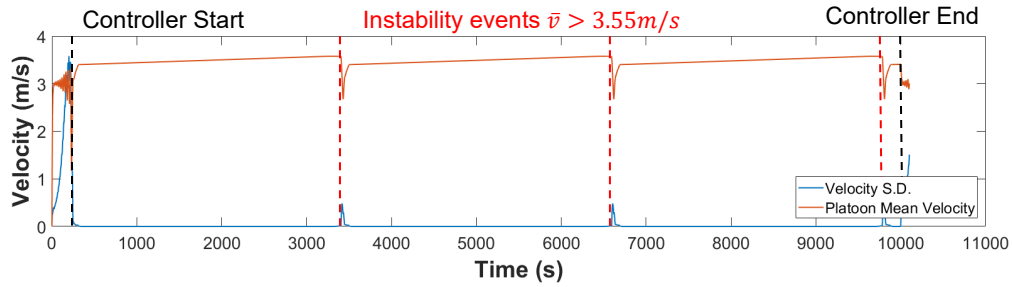



Fig. 4.3.4.1 Velocity plot of all vehicles in *Int. U extension* short test (500s)




(a) Maximum $U$ output set to 3.55 m/s




(b) Maximum $U$ output set to 3.6 m/s

Fig 4.3.4.2 Velocity std. dev. and mean velocity in *Int. U extension* long tests (10100s)

**Table 4.3.4.3 Velocity std. dev. and mean velocity in *Int. U extension* long test**

| Interval | Time (s) | Interval mean velocity (m/s) | Interval velocity std. dev. (m/s) |
|---|---|---|---|
| Uncontrolled Wave | 200-220 | 2.91 | 3.31 |
| Short Test Period | 220-400 | 3.14 | 0.27 |
| Long Test (stable) | 400-10000 | 3.53 | 0.00 |

## 5. Conclusion

In this paper, an experimental simulation using car-following models is produced to aid future development of traffic control algorithms to be implemented into autonomous vehicles. The simulation is validated by comparing Stern et al. (2018) empirical data and results from the simulated model before autonomous controller elements are implemented. The Stern et al. (2018) control model along with several modifications are tested in an attempt to improve its compatibility to the simulation environment. An attempt is also made to improve the functionality of the modified model by reducing its reliance on external operators.

The test results (*Table 5.1*) show better improvements in both velocity standard deviation and platoon mean velocity. It is partially (potentially wholly) due to the absence of various noise sources encountered in real life tests. Regardless, the various modifications made to the controller are able to demonstrate their ability to further improve traffic flow stability, which allows higher cruising velocities compared to the simulated Stern et al.(2018) controller.

However, the simulation produced in this study only represents a very basic model of traffic control simulation, substantial improvements have to be made to accurately simulate real highway traffic conditions for practical controller testing. As observed in *section 4.2*, the Helly model (1959) chosen leaves an unrealistic gap to the vehicle ahead when stationary and only references the vehicle directly ahead. Car-following models proposed more recently, such as the *Intelligent Driver Model* (Martin Treiber, 2017) and the *Parsimonious Car-Following Model* (Jorge A. Laval, 2014) are more capable in simulating real driver behaviours in congestions by implementing dynamic minimal spacing based on vehicular density. Implementing lane-changing behaviour such as the *MOBIL* model (Arne Kesting, 2014) allows simulated drivers to reference multiple vehicles and subsequent lane-changing actions will also significantly improve realism of simulation model. A new challenge under multi-lane conditions is that velocity differences and large gaps results in lane changes, which is a known cause for traffic wave occurrences (Jin, 2010). New macro-level control strategies using multiple connected controller vehicle are required, where it must accommodate a certain level of lane-changing behaviour to allow individual drivers to enter and leave the highway, but not too much as to produce significant congestions.

The immense potential of autonomous vehicles opens new doors to traffic control strategies as driver behaviour is directly influenced by controllers immersed in the traffic flow. The hardware required to create autonomous vehicle control systems are readily available. New manufactured vehicles equipped with inter-vehicular communication and locationing technology and adaptive cruise control systems can form intelligent control networks in selected areas. The only remaining hurdle ahead is the development of control algorithms to be implemented into existing technology.

**Table 5.1 Velocity std. dev. and mean velocity improvements of all tests conducted, inc. Stern et al. (2018) original test results**

*All % improvements compared to "Uncontrolled Wave" interval values*

| | Interval velocity std. dev. (%) | | Interval mean velocity (%) | | Peak mean velocity (m/s) |
|---|---|---|---|---|---|
| | Short | Long | Short | Long | |
| Stern et al. (2018) original result* | - | -80.8 | - | +14.1 | - |
| Simulated original controller | -91.8 | -99.7 | +4.0 | +12.9 | 3.42 |
| P controller[+] | -91.8 | -100 | +3.8 | +19.9 | 3.49 |
| Lookup controller | -91.8 | -100 | +3.8 | +19.9 | 3.49 |
| Int. U extension | -91.8 | -100 | +7.9 | +21.3 | 3.53 |

**Stern et al. (2018) didn't split tests into short and long tests, results shown as "long test" result, value of peak interval velocity also irrelevant for comparison to simulation result thus not included*

[+] *Results for P controller taken from stable interval (Table 4.3.2.3)*

# Appendix A

## A.1 Constants set for vehicles in base simulation, according to *Eq.1*

*All constants randomly generated. $C_1$ and T from normal distribution according to mean and std. dev. below, $C_2$ from uniform distribution between 10-20% of corresponding $C_1$. Values are unchanged throughout tests for fairness in experiments.*

| Constants | $C_1$ | $C_2$ | T |
|---|---|---|---|
| Mean | 0.5975 | - | 0.945 |
| Std. Dev. | 0.0800 | - | 0.15 |
| Car 1 | 0.6686 | 0.0936 | 82 |
| Car 2 | 0.5057 | 0.0708 | 96 |
| Car 3 | 0.5120 | 0.0922 | 76 |
| Car 4 | 0.5327 | 0.0959 | 78 |
| Car 5 | 0.3620 | 0.0434 | 94 |
| Car 6 | 0.7126 | 0.1069 | 117 |
| Car 7 | 0.6235 | 0.0873 | 83 |
| Car 8 | 0.5371 | 0.0913 | 100 |
| Car 9 | 0.7071 | 0.1202 | 91 |
| Car 10 | 0.4606 | 0.0829 | 111 |

## A.2 Constants set for FS Controller

*Values according to Stern et al. (2018)*

| k | Distance from leading vehicle (m) $\Delta x_k^0$ | Deceleration rate (m/s$^2$) $d_k$ |
|---|---|---|
| 1 | 4.5 | 1.5 |
| 2 | 5.25 | 1.0 |
| 3 | 6.0 | 0.5 |

# 6. References

Albania Nissan, H. N. (2011). Evaluation of the Impact of Advisory Variable Speed Limits on Motorway Capacity and Level of Service.

Alireza Talebpour, H. S. (2016). Influence of connected and autonomous vehicles on traffic flow stability and throughput.

Andreas Hegyi, B. D. (2005). Model predictive control for optimal coordination of ramp metering and variable speed limits.

Arne Kesting, M. T. (2014). General Lane-Changing Model MOBIL for Car-Following Models.

Beaty, W. (1998). *TRAFFIC "EXPERIMENTS"AND A CURE FOR WAVES & JAMS*. Retrieved from http://www.amasci.com/amateur/traffic/trafexp.html

Helly, W. (1959). Simulation of Bottlenecks in Single Lane Traffic Flow.

Jin, W.-L. (2010). A Kinematic Wave Theory of Lane-Changing Traffic Flow.

Jonghae Suh, H. Y. (2015). An empirical study on the traffic state evolution and stop-and-go traffic development on freeways.

Jorge A. Laval, C. S. (2014). A parsimonious model for the formation of oscillations in car-following models.

M.J. Lighthill, G. W. (1955). On Kinematic Waves Part II, A Theory of Traffic Flow on Long Crowded Roads.

Markos Papageorgiou, A. K. (2002). Freeway Ramp Metering: An Overview.

Markos Papageorgiou, E. K. (2008). Effects of Variable Speed Limits on Motorway Traffic Flow.

Martin Treiber, A. K. (2017). The Intelligent Driver Model with stoichasticity - New insights into traffic flow oscillations.

Maxime Guériau, R. B.-E. (2016). How to assess the benefits of connected vehicles? A simulation framework for the design of cooperative traffic management strategies.

Meng Wang, W. D. (2016). Cooperative Car-Following Control: Distributed Algorithm and Impact on Moving Jam Features.

Raphael E. Stern, S. C. (2018). Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments.

Ryosuke Nishi, A. T. (2012). Theory of jam-absorption driving.

Sébastien Blandin, A. C. (2012). On sequential data assimilation for scalar macroscopic traffic flow models.

Shin-ichi Tadaki, M. K. (2013). Phase transition in traffic jam experiment on a circuit.

Yibing Wang, M. P. (2005). Real-time freeway traffic state estimation based on extended Kalman filter: a general approach.

Yohei Taniguchi, R. N. (2015). Jam-absorption driving with a car-following model.

Yuki Sugiyama, M. F.-i. (2008). Traffic Jams Without Bottlenecks - Experimental Evidence for the Physical Mechanism of the Formation of a Jam.

Aeronautical and Automotive Engineering: Project Meeting Log

| | |
|---|---|
| Meeting….. Date: 15/02/2019<br>Review of actions from meeting…..:<br>- Review on initial implementation of FS controller | Agreed actions for next meeting:<br>- Fine tuning on FS controller<br>- Build PI controller<br><br>Agreed date/time of next meeting:<br>When progress |
| Signed Student ….[signature]…….. | Supervisor ……Tingjing Tiang…… |
| Meeting….. Date: 25/02/2019<br>Review of actions from meeting…..:<br>- PI controller mathematically impossible to use<br>- FS controller can be pushed even more ~~boosted~~ when ran gradually | Agreed actions for next meeting:<br>- Push FS controller even more over longer test periods<br>- Rebuild PI controller<br><br>Agreed date/time of next meeting:<br>04/03/2019 2pm |
| Signed Student …[signature]…….. | Supervisor ……Tingjing Tiang…… |
| Meeting…..Date: 04/03/2019<br>Review of actions from meeting…..:<br>- FS controller hard limit found<br>- Original PI controller scraped | Agreed actions for next meeting:<br>- Rework low level controls<br>- Focus on improvement of FS controller<br><br>Agreed date/time of next meeting:<br>12/03/2019 10:30am |
| Signed Student …[signature]…….. | Supervisor ……Tingjing Tiang…… |

Aeronautical and Automotive Engineering: Project Meeting Log

| | |
|---|---|
| Meeting….. Date: 12/03/2019<br>Review of actions from meeting…..:<br>- Replaced low level acceleration switch to PID controller<br>- Improved ~~the~~ speed limit | Agreed actions for next meeting:<br>- Replace PID controller to math function lookup table<br>- Potentially internalise generation of U<br><br>Agreed date/time of next meeting: 15/03/2019 4:15pm |
| Signed Student ML | Supervisor Jingjing Jiang |
| Meeting….. Date:<br>Review of actions from meeting…..: | Agreed actions for next meeting:<br><br>Agreed date/time of next meeting: |
| Signed Student …….. | Supervisor …… |
| Meeting…..Date:<br>Review of actions from meeting…..: | Agreed actions for next meeting:<br><br>Agreed date/time of next meeting: |
| Signed Student …….. | Supervisor …… |