

# Intersection Control for Automated Vehicles

Edward Lambert

June 7, 2021

## Abstract

## 1 Introduction

The capacity and safety of an intersection can be considered in terms of the number and type of conflict points which it contains. When every entrance to the intersection is connected to every exit by a smooth arc, a conflict point exists wherever two arcs intersect. They can be classified as head-on, following, crossing, diverging or merging. In many studies these points are assumed to be fixed, and only speed of the vehicles which controls the rate of progress along the arcs is varied to minimise delay.

With the development of path planning around obstacles on-the-fly, it becomes possible to consider what arrangement of conflict points is best given the traffic at a particular instant in the near future.

## 2 Simulation

Platooning with speed choice by a centralized controller was implemented with a vehicle to intersection messaging scheme. The full site is divided into zones, each one containing a single intersection. Each AGV in the fleet has a copy of the roadmap which is static. The fleet controller interfaces with the warehouse management system to get the next material transfer job, consisting of a pick location and a drop location. All jobs are assumed to be of unit size and each AGV has a capacity of one unit. With these assumptions, a straightforward policy is to assign the next job to the AGV nearest to the pick location - first-come-first-served scheduling. When an AGV receives a new job, it finds the shortest path through the roadmap using the Floyd-Warshall algorithm [?]. Next it must send its planned path to the intersection controller for the zone it currently occupies. The intersection controller stores the plan and current position of every AGV approaching the conflict point of the intersection. Every time it receives a new plan it must recalculate the approach speed for every approaching AGV to minimize total travel time without collision. This will

happen every time an AGV enters the zone from somewhere else, or an AGV within the zone is assigned a new job.

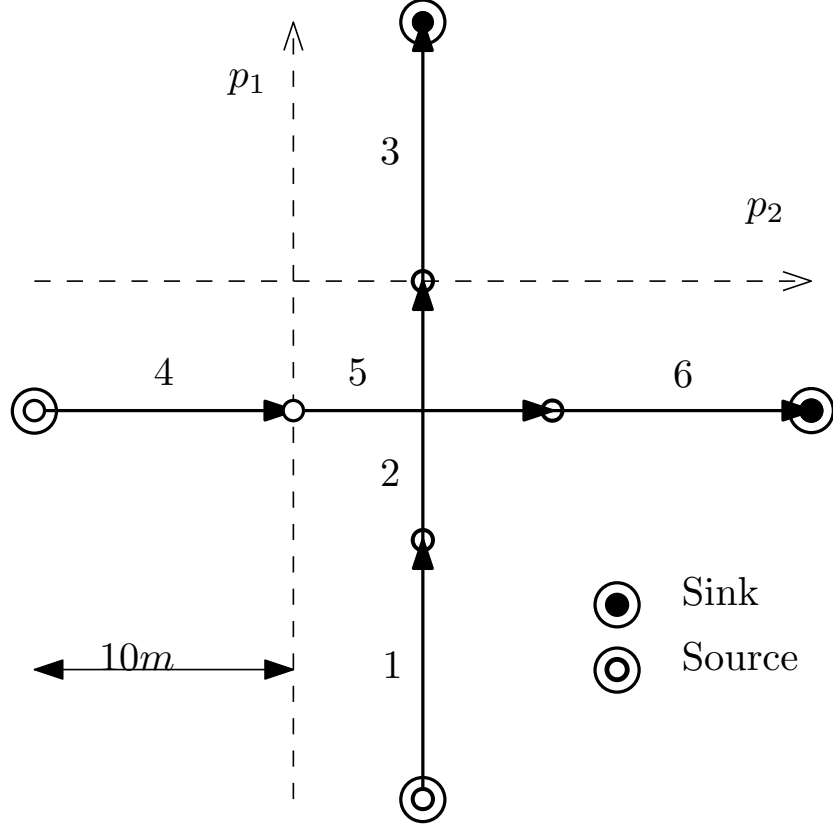


Figure 1: Intersection layout with two conflicting routes.

The intersection controller was implemented based on [?]. The surrounding lanes are first discretized into segments. The intersection shown in Figure 1 is divided into six segments, each of length 10 meters. The critical segments are the two that cross in the center. There are two routes defined, one starting on the left and traveling to the right and the other starting at the bottom and traveling up. One AGV takes route 1 and the other takes route 2. If they both travel at maximum speed they will collide in the center.

The dynamic model for each AGV assumes they are able to exactly follow the path, and attempt to reach the target speed for each segment subject to a limited rate of acceleration of  $am/s^2$ .

The ApproachPlan message sent by the AGV contains a sequence of segments which it intends to traverse, along with its current distance along the first one. The flow of messages is shown in Figure 2. The SpeedList sent by the intersection controller contains the optimal speed for every segment in the

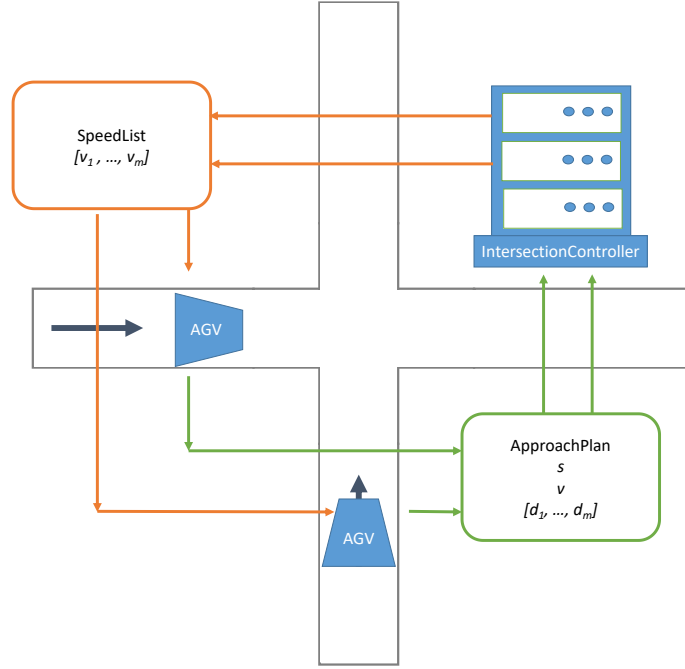


Figure 2: Messages exchanged by participants approaching intersection.

plan. The speeds can be found with the nonlinear program in Equation 9.

## 2.1 Motor Dynamic and Electrical Model

The simulated AGV are based on a small payload 100kg total mass, with a maximum speed of 10m/s and peak acceleration of  $5\text{m/s}^2$ , allowing them to stop safely within 10 metres. The intersection controllers use a constant acceleration model to calculate the time and space deadlines they pass to the AGV. To make the simulation test worthwhile a simulation model with slightly more complexity was used to evaluate the performance.

The brushless DC electric motor used in [?] has suitable properties to propel a 100kg battery powered vehicle. A DC motor can be modelled by the steady state equivalent circuit shown in Figure 3 which is well known, for example see [?].

This simple circuit leads to the relationship between current and internal resistance given by Equation 1.

$$V_{cc} = E_D + I_a R_a \quad (1)$$

It should be noted that the brushless DC motor is sometimes grouped with AC machines such as the induction motor due to its varying excitation [?]. The

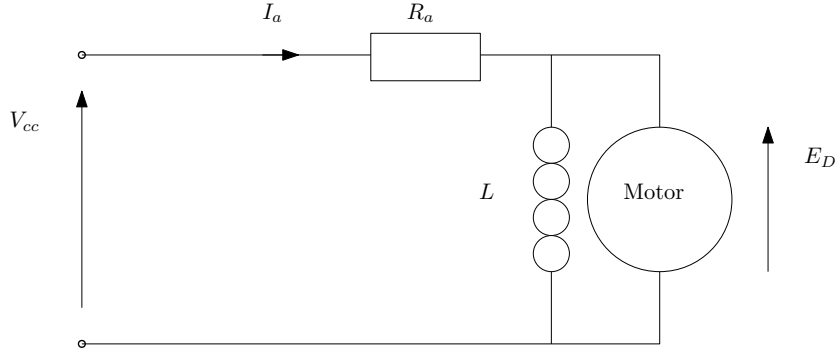


Figure 3: Steady state equivalent circuit for a DC motor.

Table 1: Motor parameters used in simulation.

$k_v$ [rpm/v]	6
$k_T$ [Nm/A]	1.53
$P_{mech}@375\text{rpm}$ [kW]	3.6
$P_{elec}@375\text{rpm}$ [kW]	6.37
$*R_a$ [Ohms]	1.0

coils are energised in order to keep the magnetic field at 90 degrees to the field generated by the permanent magnets mounted on the rotor.

The field strength of the magnets, the number of poles and the number turns of the armature coils can be captured in the motor constant  $k_T$  relating torque to armature current.

$$\tau = k_T I_a \quad (2)$$

$$E_D = \frac{rpm}{k_e} \quad (3)$$

There are numerous loss sources in an electric motor such as winding resistance, flux leakage, eddy currents in the core and so on [?]. By using real-world measured mechanical power output and electrical input, an equivalent winding resistance  $R_a$  for the simple model can be found. The parameters are shown in Table 1.

## 2.2 Air Resistance

In addition to the resistive losses in the motor windings losses due to air resistance were also modelled. AGVs typically operate at low speed 3m/s, hence the unaerodynamic shape of many models. However, the target speed of 10m/s is quite high around 30 miles per hour and drag could become significant.

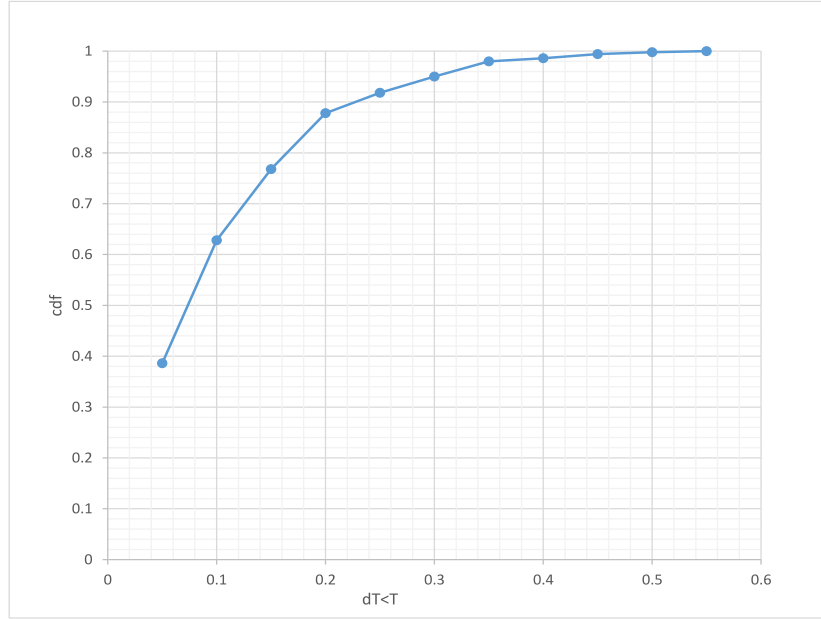


Figure 4: Cumulative Distribution Function of arrivals following a Poisson distribution.

The drag coefficient  $C_{drag}=1$  for a cuboid shape was used, taken from [?]. The frontal area  $A$  of  $1\text{m}^2$  is consistent with a box shape for maximum load carrying capacity. The density of air at room temperature is close to  $1\text{kg}/\text{m}^3$ .

$$F_D = C_{drag}\rho v^2 A \quad (4)$$

## 2.3 Arrival Distribution

Previous work in road traffic modelling has used a variety of point distributions to model arrivals. A good summary is given in [?] or the chapter on Microsimulation in [?].

### 2.3.1 Learning from Road Transport Models

One option is to assume arrivals at a point are completely independent of each other, but occur at an average rate for the time of day which has been measured by inductive loop placed in the road. These assumptions lead to a Poisson distribution such as that shown in Figure 4. This can be generated computationally by drawing a sequence of numbers from a uniform distribution and applying Equation ?? to compute the time delta until the next arrival. To validate the arrivals drawn in this way, we check the log plot of the frequency against time

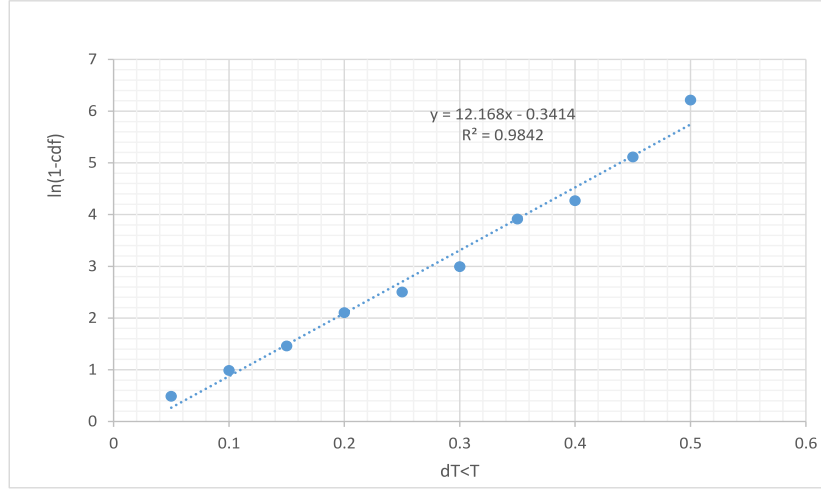


Figure 5: Log Cumulative Distribution Function of arrivals with linear fit.

delta is linear and the median is equal to the specified rate as shown in Figure 7.

The assumption that arrivals are independent does not hold if traffic density is high. This is because vehicles slow down to keep a safe distance from the one in front of them. The simplest modification to the poisson distribution to account for this is to discard time deltas which violate the specified safe headway, leading to the shifted exponential distribution. A scatter plot of times from this distribution is shown in Figure 6. Now the arrivals will always be realistic in the sense vehicle will not overlap/arrive unsafely but the variance will be smaller. The two parameters the average arrival rate  $\lambda$  and the minimum safe headway  $\tau$  together control the variance. This is because the median of the distribution must be  $\lambda$ , so if  $1/\tau$  is close to  $\lambda$  the variance must be very small.

### 2.3.2 Unique Features of AGV Traffic

The arrival distribution corresponds to the properties of AGV traffic satisfying the following assumptions:

- More than one AGV is permitted to enter the same link if they are travelling in the same direction.
- Entry to links in the roadmap is denied if the number already present on the link reaches a fixed capacity
- Car-following behaviour within the link is governed by the frontal safety system on board each AGV. This provides an outer (warning) zone which, if tripped, the AGV will slow to  $w$ m/s at  $a_w$ m/s and an inner zone (emergency), which if tripped the vehicle will come to a complete stop at  $a_e$ m/s, where  $|a_e| \leq |a_w|$

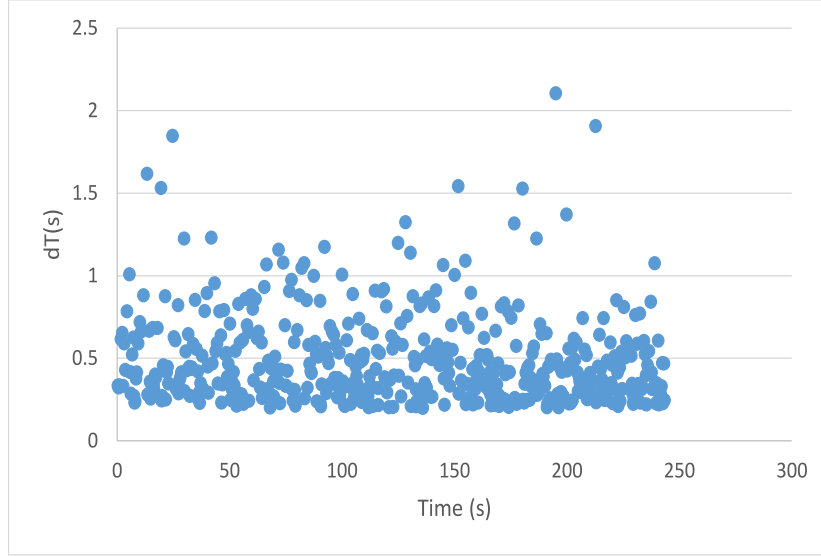


Figure 6: Scatter plot of 500 arrival time deltas drawn from a shifted exponential distribution with a minimum headway of 0.2 seconds.

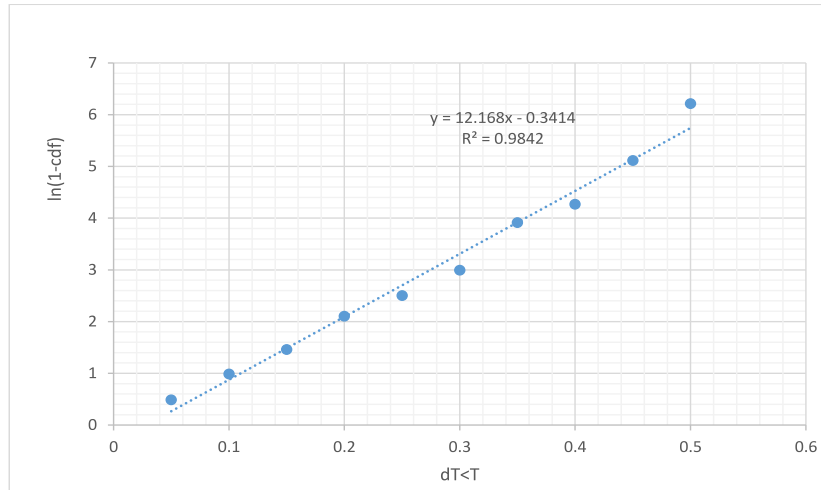


Figure 7: Log Cumulative Distribution Function of arrivals with linear fit.

- The capacity of a link is the number of AGVs which would fit without the inner front safety zone of each being tripped by the one in front. This can be calculated for  $n$  AGVs each with safety zone length  $d_e$  as  $n_c = n d_w$
- Every AGV is informed whether it is able to proceed before it reaches the end of its current link. The AGV will keep slowing down to stop at the end, until it is informed there is space on the next link, at which point it will accelerate to full speed.

According to these assumptions, a suitable model is the shifted exponential, with a minimum headway based on the size of the AGV safety zone.

### 2.3.3 Arrival Variation Based on Approach Lane Occupancy

Another consideration is how the traffic on the approach lanes should feed back into the arrival times. Based on the counting semaphore for each link assumption, this can be modelled with two queues at the entryway. The lane queue and the busy queue. If the occupancy of the lane is less than  $n_c$ , new arrivals are added to the lane queue at full speed, in the simulation interval exceeding their point arrival time, at a position they would have reached by the simulation time. If the occupancy of the lane is more than  $n_c$  new arrivals are added to the back of the busy queue. At the next time step where  $n < n_c$ , the AGV at the front of the busy queue is moved onto the lane. The arrival speed is reduced according to acceleration rate  $a_W$  based on the time between the arrival and the time step where it can proceed. For longer time periods the new arrival will have zero speed. All arrivals which were not at the front of the busy queue will arrive at the warning speed  $v_w=0.3\text{m/s}$ .

## 2.4 Division of Responsibility Between Intersection and AGV Controllers

With different signal-free intersection control approaches, the functionality carried out centrally by the fixed-server compared to that by the mobile AGV varies significantly. Of particular interest are the interfaces used by Digani et al [?] where the intersection provides average speeds for discrete segments, and the one used by Levin and Rey [?] where the reservation message sent by the intersection contains the arrival time at the conflict point.

The arrival time at the conflict point is similar to a widely used waypoint interface. As the time is the important quantity for ensuring collisions are avoided this interface seems more sensible. However, on its own it is not sufficient to ensure FIFO behaviour of AGVs approaching in the same lane. Many studies assume an external mechanism for car-following based on local sensors such as the Intelligent Driver Model in [?].

Without a car-following or platooning approach locally on the AGV, simple methods to manage cross traffic can easily fail to maintain FIFO ordering. This problem is often encountered in traffic simulation with cellular models and there



are various techniques to avoid it without generating a trajectory for each vehicle [?].

[?] and [?] do not mention any local car following behaviour and seek to avoid all collisions by the centralised application of constraints. This makes it difficult to compare them to a semaphore based intersection controller, unless car following is used in combination with semaphore access control.

### 3 Methods For Zone-Based Intersection

#### 3.1 Intersection Controller Objective

The objective is to minimize  $J_T$  the total travel time for all vehicles. It is convenient for exposition to optimize over the inverse of speed of each segment  $\phi_k = 1/v_k$ . Vehicle  $i$  submits a plan containing  $m_i$  segments before the conflict and  $n_i$  segments in conflict. The control model is based on the average speed of each approaching AGV over each segment. This is to simplify the description of the intersection controller, and assist analysis. More sophisticated motion models could take the place of Equation 10 and Equation 12 to create a similar type of problem with a convex travel time objective and non-convex constraints. The parameters for one vehicle can be collected in the vector  $\phi_i$  as shown in Equation 5

$$\phi_i = [\phi_1, \dots, \phi_{(m_i+n_i)}]^T \quad (5)$$

The parameters for  $p$  vehicles each traversing  $(m_i + n_i)$  segments are assembled into a vector as in Equation 6

$$\phi = [\phi_1^T, \dots, \phi_p^T]^T \quad (6)$$

Similarly, the length of each segment in plan  $i$  can be arranged into a vector

$$\mathbf{d}_i = [d_1, \dots, d_{(m_i+n_i)}]^T \quad (7)$$

and collected for  $p$  vehicles into a vector as in Equation 8.

$$\mathbf{d} = [\mathbf{d}_1^T, \dots, \mathbf{d}_p^T]^T \quad (8)$$

This leads to the minimum travel time objective in Equation 9.

$$\begin{aligned} \min_{\phi} J_T &= \mathbf{d}^T \phi \\ \text{subject to} & \\ \phi_{max} &> \phi > \phi_{min} \\ \phi^T \mathbf{H}_{i,j} \phi &> \mathbf{0} \quad \forall i, j \in [1, p] \quad \text{with } j > i \end{aligned} \quad (9)$$

The condition  $j > i$  in Equation 9 indicates that the number of constraints varies with the number of vehicles  $p$  as  $\frac{p(p-1)}{2}$ . This corresponds to one constraint between each pair of approaching AGVs.

### 3.2 Intersection Controller Timing Constraints

By definition, each intersection has a single conflict zone, the union of all segments which intersect there. This makes it possible to express the constraint that vehicles do not collide in terms of time. Vehicle  $i$  arrives at the first conflicted segment  $\omega_i^{min}$  and departs from the last at  $\omega_i^{max}$ . The following three subsections set out three alternative ways of expressing the collision avoidance constraints which have been evaluated. The arrival time is given by Equation 10. Considering average speeds, the departure time  $\omega_i^{max}$  is also linear, this is given by Equation 12.

$$\omega_i^{min} = \sum_{k=1}^{m_i} \mathbf{d}_i[k] \phi_i[k] = \mathbf{e}^T \phi_i \quad (10)$$

where

$$\mathbf{e}[k] = \begin{cases} \mathbf{d}_i[k] & \forall k < m_i \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

and  $m_i$  is the number of segments on the path of vehicle  $i$  before arrival at the conflicted segment.

$$\omega_i^{max} = \omega_i^{min} + \sum_{i=1}^{n_i} \mathbf{d}_i[k] \phi_i[k] = \mathbf{f}^T \phi_i \quad (12)$$

where

$$\mathbf{f}[k] = \begin{cases} \mathbf{d}_i[k] & \forall k < m_i + n_i \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

and  $n_i$  is the number of segments on the path of vehicle  $i$  which are conflicted. Note that Equations 10 and 12 only depend on the  $\phi_i$  of vehicle  $i$ .

#### 3.2.1 Linear FIFO Constraints

If the order in which the AGV cross the conflict zone is fixed to be First-In-First-Out, the timing constraint is linear. For a pair where the leader enters the conflict at  $t_i$  and takes  $\Delta t_i$  seconds to cross at maximum speed the condition for entry time of the follower  $t_{i+1}$  given by Equation 14.

$$-t_i + t_{i+1} \leq \Delta t_i \quad (14)$$

There is one constraint between each adjacent pair so  $p - 1$  constraints total for  $p$  vehicles. These can be expressed in the form  $\mathbf{A}_{ub} \phi \leq \mathbf{b}_{ub}$  as in Equation 15. This is correct for two AGV arranged in distance order, each traversing one approach and one conflict segment.

$$\begin{bmatrix} -d_1 & d_2 & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & -d_{p-1} & d_p \end{bmatrix} \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_p \end{bmatrix} \leq \begin{bmatrix} \Delta t_1 \\ \vdots \\ \Delta t_{p-1} \end{bmatrix} \quad (15)$$

The timing constraint between each pair of vehicles can be expressed with a modulus operator as in Equation 16.

$$|\alpha_i - \alpha_j| > \beta_i + \beta_j \quad (16)$$

Here

$$\alpha_i = \omega_i^{max} + \omega_i^{min} \quad (17)$$

represents the midpoint of the time vehicle  $i$  occupies the conflicted segment and

$$\beta_i = \omega_i^{max} - \omega_i^{min} \quad (18)$$

represents the range of the time either side of the midpoint, both scaled by a factor of two.

In matrix form

$$\alpha_i = \mathbf{f}^T \phi_i + \mathbf{e}^T \phi_i = \mathbf{1}_i^T \mathbf{A} \phi_i \quad (19)$$

with  $\mathbf{A} = \text{diag}(\mathbf{f} + \mathbf{e})$

$$\beta_i = \mathbf{f}^T \phi_i - \mathbf{e}^T \phi_i = \mathbf{1}_i^T \mathbf{B} \phi_i \quad (20)$$

with  $\mathbf{B} = \text{diag}(\mathbf{f} - \mathbf{e})$

The resulting linear program (with parameters  $\in \mathbb{R}$ ) has  $p - 1$  constraints as each AGV is only constrained by the preceding one.

### 3.2.2 Quadratic Constraints

Another way to treat the modulus operator in Equation 16, without forcing any particular arrival order is to square both sides as to give the expression in Equation 21.

$$\alpha_i^2 - \alpha_j^2 - 2\alpha_i\alpha_j - (\beta_i^2 + \beta_j^2 + 2\beta_i\beta_j) > 0 \quad (21)$$

Collecting terms by subscript gives

$$(\alpha_i^2 - \beta_i^2) - (\alpha_j^2 + \beta_j^2) - 2(\alpha_i\alpha_j + \beta_i\beta_j) > 0 \quad (22)$$

The matrix  $\mathbf{\Lambda}_{ij}$  captures the constraints between a pair of vehicles and always contains four sub-matrices as shown in in Equation 23. It is compatible with  $\phi_{i,j} = [\phi_i^T, \phi_j^T]^T$ , containing only the relevant speeds for vehicles  $i$  and  $j$ .

$$\mathbf{\Lambda}_{ij} = \begin{bmatrix} \mathbf{\Lambda}_{ij}^{ii} & \mathbf{\Lambda}_{ij}^{ij} \\ \mathbf{\Lambda}_{ij}^{ji} & \mathbf{\Lambda}_{ij}^{jj} \end{bmatrix} \quad (23)$$

Expanding

$$\begin{aligned} [\phi_i^T, \phi_j^T] \begin{bmatrix} \mathbf{\Lambda}_{ij}^{ii} & \mathbf{\Lambda}_{ij}^{ij} \\ \mathbf{\Lambda}_{ij}^{ji} & \mathbf{\Lambda}_{ij}^{jj} \end{bmatrix} \begin{bmatrix} \phi_i \\ \phi_j \end{bmatrix} \\ = \phi_i^T \mathbf{\Lambda}_{ij}^{ii} \phi_i + \phi_j^T \mathbf{\Lambda}_{ij}^{jj} \phi_j + \phi_i^T \mathbf{\Lambda}_{ij}^{ij} \phi_j + \phi_j^T \mathbf{\Lambda}_{ij}^{ji} \phi_i \end{aligned} \quad (24)$$

makes it possible to compare terms with the scalar expression in Equation 22. This leads to the following expressions for the submatrices in  $\Lambda$  in terms of  $\alpha_i = \mathbf{1}_T \mathbf{A}_i \phi_i$  and  $\beta_i = \mathbf{1}_T \mathbf{B}_i \phi_i$

$$\Lambda_{ij}^{ii} = (\mathbf{A}_i - \mathbf{B}_i) \mathbf{1}_i \mathbf{1}_i^T (\mathbf{A}_i - \mathbf{B}_i) \quad (25)$$

$$\Lambda_{ij}^{jj} = -(\mathbf{A}_j + \mathbf{B}_j) \mathbf{1}_j \mathbf{1}_j^T (\mathbf{A}_j + \mathbf{B}_j) \quad (26)$$

$$\Lambda_{ij}^{ij} + \Lambda_{ij}^{ijT} = -2(\mathbf{A}_j + \mathbf{B}_j) \mathbf{1}_j \mathbf{1}_i^T (\mathbf{A}_i + \mathbf{B}_i) \quad (27)$$

For more than two vehicles this can be arranged into a block diagonal matrix  $\mathbf{H}_{ij}$  which is compatible with the input parameters, but still only represents the constraints between a pair.

Expressed in this way it is clear the constraints are quadratic and it is trivial to differentiate twice to find the Hessian is the stack of constraint matrices  $[\mathbf{H}_{ij}, \dots]$ . The objective is certainly convex as it is linear but the constraints may not be. If the Hessian of the constraints is positive semi-definite then they are convex and interior point methods will either find the global optimum or prove that there is no feasible solution [?]. The Hessian depends on the parameters of the roadmap, the number of approaching vehicles and their distance from the conflict.

### 3.2.3 Mixed Integer Constraints

A third way of treating the timing constraint in Equation 16, also without forcing any particular arrival order involves splitting each constraint into two based on the sign of  $(\alpha_i - \alpha_j)$  as shown in equation 28. Again, this is expressed in terms of the midpoint  $\alpha_i, \alpha_j$  and extent  $\beta_i, \beta_j$  of the time when vehicle  $i$  and vehicle  $j$  occupy the segment on their own path which passes through the conflict point,

$$\begin{aligned} \alpha_i - \alpha_j &> \beta_i + \beta_j && \text{if } \alpha_i > \alpha_j \\ \alpha_i - \alpha_j &< -(\beta_i + \beta_j) && \text{otherwise} \end{aligned} \quad (28)$$

where  $\alpha_i, \alpha_j, \beta_i, \beta_j > 0$

In order to apply these OR constraints, an additional integer parameter  $b_k$  can be introduced for each pair of AGV, along with an arbitrary large number  $M$  as shown in Equation 29.

$$\begin{aligned} \alpha_i - \alpha_j + b_{i,j} M &> \beta_i + \beta_j \\ \alpha_i - \alpha_j - (1 - b_{i,j}) M &< -(\beta_i + \beta_j) \\ \text{where } b_{i,j} &\in [0, 1] \\ M &>> \alpha_i, \alpha_j, \beta_i, \beta_j \end{aligned} \quad (29)$$

Now the problem is combinatorial rather than convex and appropriate methods must be used. These may be based on exhaustive search such as Branch-and-Bound, or solving a sequence of convex relaxations of the original problem [?]. Combinatorial problems quickly become intractable for large numbers of variables, but in this case the underlying problem of arrival order is combinatorial, so exhaustive methods are needed to find the global minimum.

It is possible to further assume every AGV travels at maximum speed once it reaches the conflict, simplifying equation Equations 17 and 18 with a constant  $p_i = g_i\phi_{min}$ .

$$\alpha_i = 2\omega_i^{min} + p_i \quad (30)$$

$$\beta_i = p_i \quad (31)$$

where  $g_i$  is the length of the conflicted segments in the plan of vehicle  $i$ . In this case Equation 29 can be restated

$$\begin{aligned} \omega_i^{min} - \omega_j^{min} + b_{i,j}M &> p_j \\ \omega_i^{min} - \omega_j^{min} - (1 - b_{i,j})M &< -p_i \\ \text{where } b_{i,j} &\in [0, 1] \end{aligned} \quad (32)$$

## 4 Methods For Conflict Point Intersection

For a intersection between roads with multiple lanes, it makes sense to plot the centreline of each lane, and the arc of each turning motion between lanes. Any point where two of these arcs cross is called a conflict point. By controlling arrival time at a conflict point, collisions can be avoided, while vehicles that do not pass the same conflict point can both proceed [?].

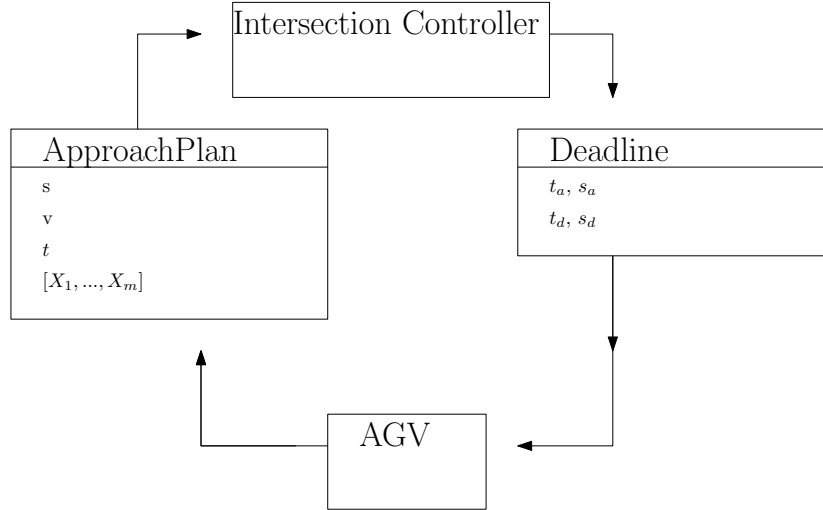


Figure 8: Key messages of the “Plan/Deadline” interface governing access to a conflict point intersection.

A controller based on a common heuristic for controlling access to a shared resource, the binary semaphore, is included for comparison with the optimal methods. Similar approaches are widespread in AGV control [?] and [?].

Table 2: Parameters bounds used to generate test scenarios.

Parameter	High Bound	Low Bound
$T_C$ [s]	0.5	0.1
$\frac{1}{\lambda} = \Delta T_a$ [s]	10	2

	$\lambda_1$	$\lambda_2$	$f$
HLHT	0.5	0.5	2
HLMT	0.1	0.5	2
HLLT	0.1	0.1	2
LLHT	0.5	0.5	10
LLMT	0.1	0.5	10
LLLTT	0.1	0.1	10

Table 3: Parameters for test scenarios. All units  $s^{-1}$ . Each scenario is identified with the first two characters relating to the latency between periodic messages from the intersection controller where High Latency is 500ms and Low Latency is 100ms and the second two relating to the arrival rate, where High Traffic has  $\lambda=10$  arrivals per second on both approaches, Low Traffic has  $\lambda=2$  arrivals per second on both, and Mixed Traffic has one lane with  $\lambda_1=10$  and the other with  $\lambda_2=2$ . For example High Latency, High Traffic becomes HLHT.

#### 4.1 Semaphore Approach

To implement the “plan/deadline” interface, a binary semaphore for each conflict point was needed. The semaphore controller does not compute the entry and exit times of the conflict as it has no model of the vehicle dynamics. It simply raises the semaphore, sending a “full speed ahead” message to the closest AGV to the intersection and a position deadline to all others.

For this reason the deadline message contains the position at which the AGV must stop, without any timing. The AGV must stop at the given position until it receives a ‘proceed’ message from the intersection controller.

### 5 Performance Comparison

Both conflict point representation and the zone representation are identical if the intersection has exactly one conflict. On this basis the semaphore approach and two versions of minimum crossing time optimal control, one with FIFO linear constraints and the other with quadratic constraints were compared on a simulated intersection comprising two lanes which cross in the middle. The approach distance is fixed at 15 metres.

The controllers will be tested with traffic levels and control latencies intended to bound the likely values of these parameters, shown in Table 2.

## 6 Simulation Results

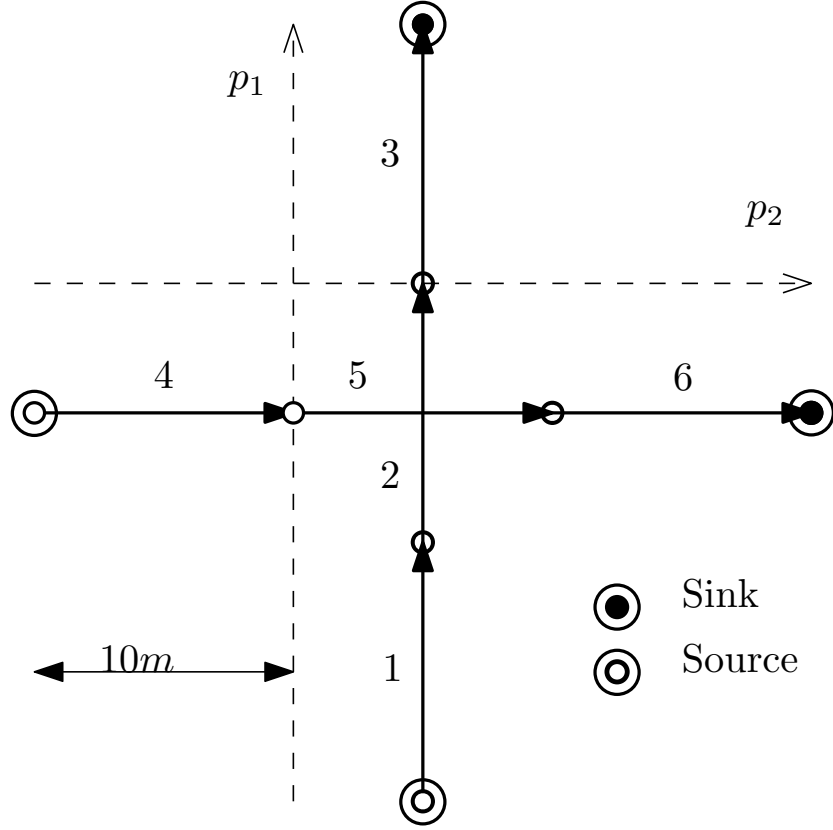


Figure 9: Intersection layout with two conflicting routes.

The different approaches to intersection control were evaluated on a simulation of a simple intersection, comprised of two 30m lanes which cross in the middle as shown in Figure 9. There are two entrances to the map, one at the start of each lane. By varying the arrival rate  $\lambda$  and the update frequency  $f$ , six scenarios were created with the parameters shown in Table 6.

### 6.1 Trajectory Comparison with Fixed Arrival Pattern, 30 Second Run

The distance-time plot for the semaphore controller is shown in Figure 10. The distance axis is measured from the start of the submitted plan for the AGV. All plans start 15 metres from the conflict point on both lanes. AGV travelling in the x-direction are shown with 'x' markers, while those travelling in the y-direction are shown with triangle markers. The simulation did not proceed

	$\lambda_1$	$\lambda_2$	$f$
HLHT	0.5	0.5	2
HLMT	0.1	0.5	2
HLLT	0.1	0.1	2
LLHT	0.5	0.5	10
LLMT	0.1	0.5	10
LLLT	0.1	0.1	10

Table 4: Parameters for test scenarios. All units  $\text{s}^{-1}$ . Each scenario is identified with the first two characters relating to the latency between periodic messages from the intersection controller where High Latency is 500ms and Low Latency is 100ms and the second two relating to the arrival rate, where High Traffic has  $\lambda=10$  arrivals per second on both approaches, Low Traffic has  $\lambda=2$  arrivals per second on both, and Mixed Traffic has one lane with  $\lambda_1=10$  and the other with  $\lambda_2=2$ . For example High Latency, High Traffic becomes HLHT

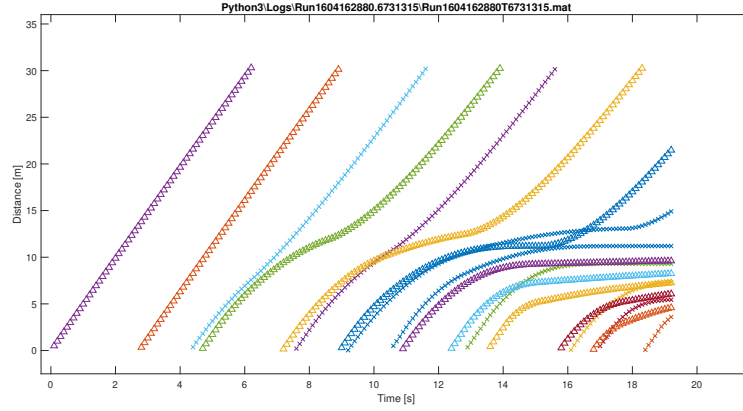


Figure 10: Position time series for the semaphore heuristic controller in scenario 1.



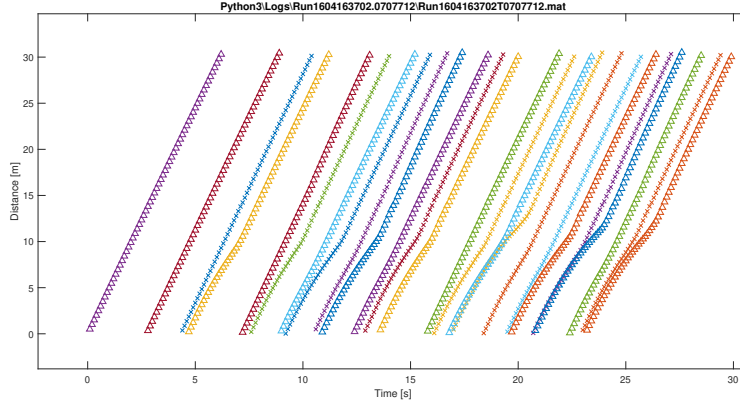


Figure 11: The position time series for the FIFO Optimal Controller in scenario 1.

beyond 20 seconds because of a collision with a new arrival.

The distance-time plot for the FIFO Optimal Controller is shown in Figure 11. It was able to proceed for the full 30 seconds without any collision as the approach lane did not fill up. This only shows that the throughput of the FIFO optimal control is higher than semaphore method, and is sufficient to meet the demand of  $\lambda = 0.5$  on each lane, without a queue forming.

In these two runs the arrival sources were not linked to the occupancy of the approach lane, so if the approach lane fills up, and average speed drops, so the new arrivals can appear right on top of vehicle at the back of the queue, leading to a collision which the intersection controller is unable to prevent. This is not a realistic crash situation because in a real site approaching vehicles will detect the back of queue with their front safety sensor and stop in time.

In Figure 11 it can be seen that 24 vehicles pass through the intersection in 30 seconds. This is very close to the upper limit of one vehicle per second determined by on the arrival rate.

## 6.2 Trajectory Comparison with Dynamic Arrival Pattern, 30 Departures Run

To understand more about the performance of the controllers the arrival pattern was linked to the approach lane occupancy. Rather than deal with the complexity of speed reductions due to safety sensors, the model was based on a roadmap reservation system which are widely used in centralized or decentralized form. The capacity of the approach lane (10 metres long) was set to 1 AGV. If the static arrival pattern suggested an arrival while the lane was full, it entered an arrival queue. At the next simulation time step where the approach lane had capacity, if there was any vehicles in the queue, the first would be introduced

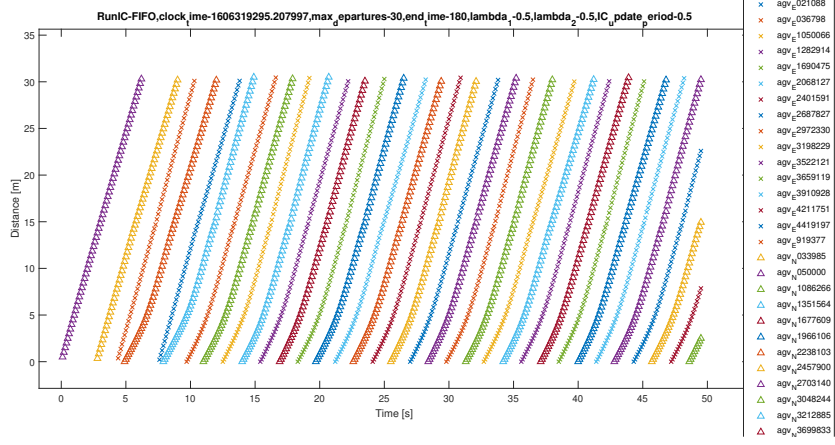


Figure 12: The position time series for the FIFO Optimal Controller in the High Traffic High Latency Scenario

to the simulation at the start of the approach lane with zero speed (as opposed to the random arrivals, which enter the lane at full speed).

With arrival lane capacity of one AGV, 30 AGVs were able to cross the intersection in 82.3 seconds with the semaphore control. With FIFO optimal control this was reduced to 49.6 seconds. This is a reduction of 39.7%, entirely due to centralized speed choice, as both used FIFO ordering. The centralized optimal FIFO controller is able to predict when the conflict will become unoccupied, so following vehicles don't need to slow down as much as with Semaphore control.

The free flow time to cross 30 metres at maximum speed of 5 metres per second is six seconds. The mean travel time across the Semaphore controlled intersection was 309.1, indicating a delay of  $309.1/30 - 6 = 4.30$  seconds. The delay due to the FIFO Controlled intersection was  $195.1/30 - 6 = 0.50$ s. This dramatic reduction might be expected given the objective for the FIFO optimal method was to minimise total travel time.

### 6.3 Energy Consumption

The use of an approach speeds minimising total travel time increases the energy consumption per vehicle as shown in Table 6.3. The extra information about the departure time of leading vehicles should lead to reduced energy expense slowing down. As total energy usage (and therefore usage per vehicle) actually increase, and air resistance was not considered, it suggests the higher average speeds reduce the efficiency of the motor.

## References

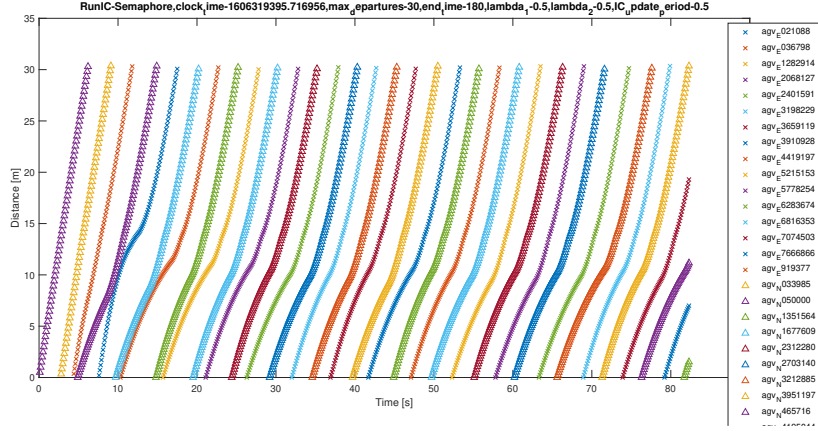


Figure 13: The position time series for the Semaphore Controller in the High Traffic High Latency Scenario.

	Semaphore	FIFO Optimal
Total Electrical Energy (MJ)	542.310	641.170
Total Mechanical Energy (MJ)	61.549	68.448
Completion Time (s)	82.3	49.6
Total Travel Time (s)	309.1	195.1

Table 5: Energy and Completion Time Results

$t_{\text{time}}=1606319295.207997, \max_d \text{epartures}=30, \text{end\_time}=180, \lambda_1=0.5, \lambda_2=0.1$

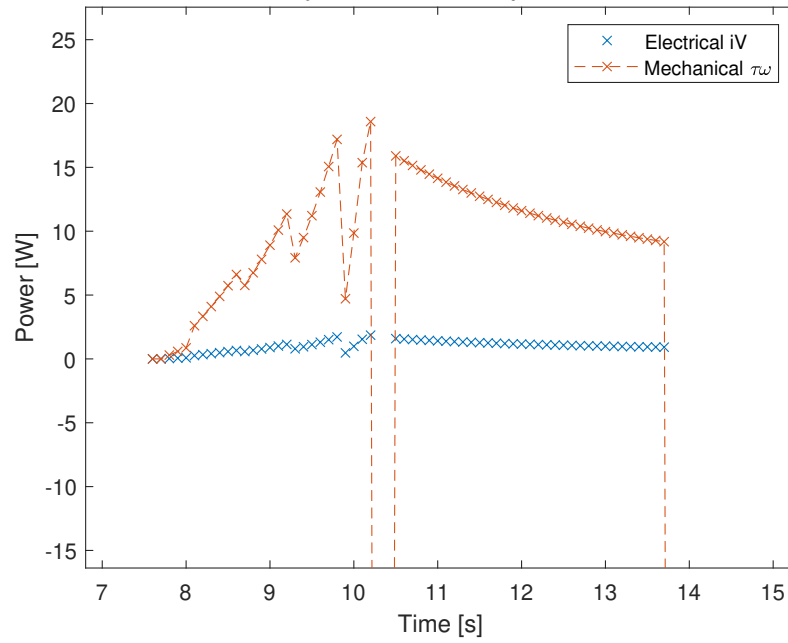


Figure 14: The power dissipated over time for one AGV under FIFO control.

,max<sub>d</sub>epartures-30,end<sub>t</sub>ime-18

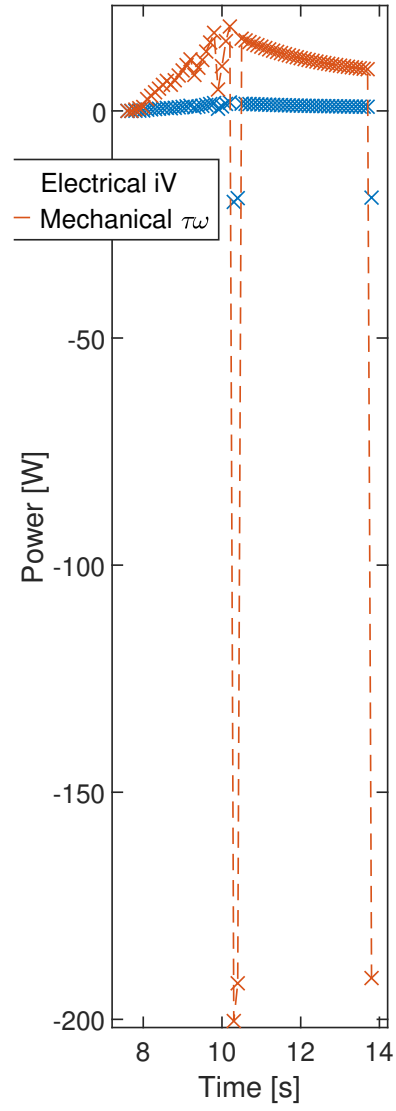


Figure 15: Spurious data points of the power dissipated over time for one AGV under FIFO control.

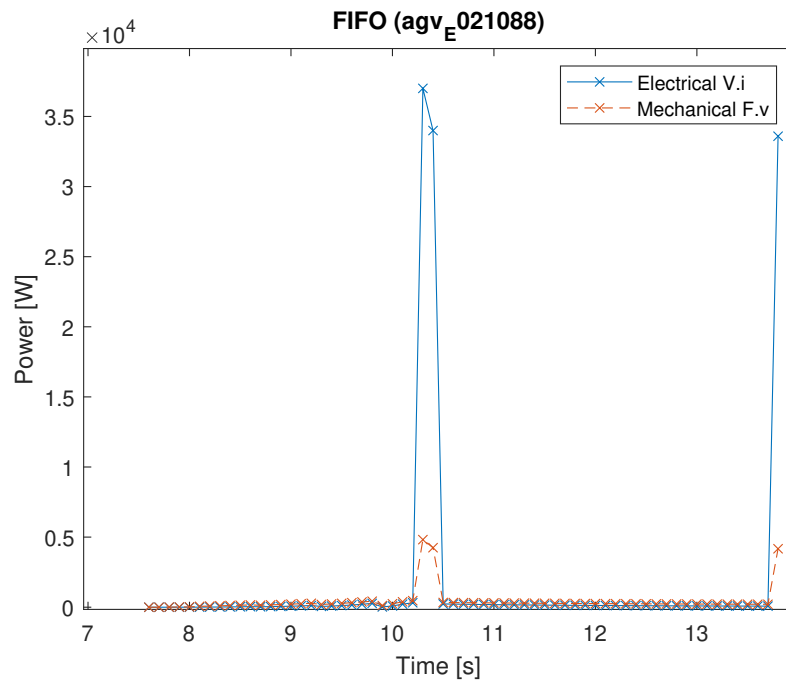


Figure 16: The electric power was recalculated from the logged value of armature current  $i^2R$  with a resistance of  $R=0.001\omega$ . The mechanical power was recalculated using the  $\text{abs}(\text{motive\_force} \times \text{velocity})$ .

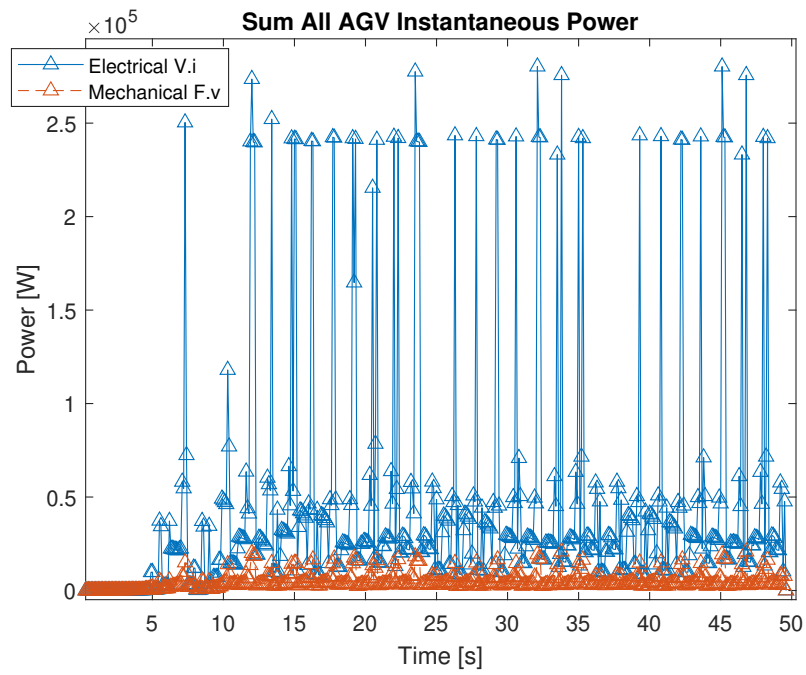


Figure 17: The power dissipated over time for all AGVs in total under FIFO control.

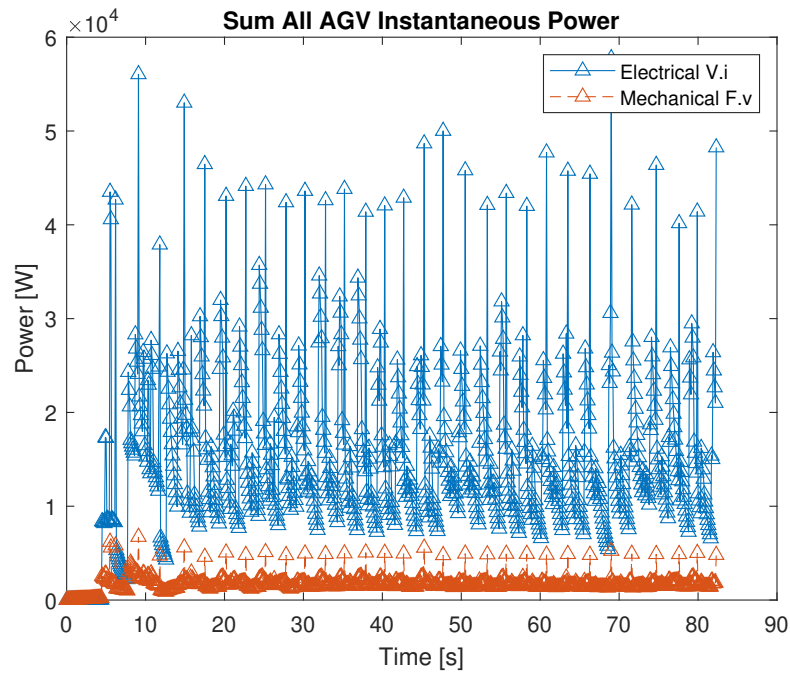


Figure 18: The power dissipated over time for all AGVs in total under Semaphore control.