ADAPTIVE MESSAGING BASED ON AOI FOR CONGESTION CONTROL IN VANETS

BY

JORDI MARIAS I PARELLA

Submitted in partial fulfillment of the
requirements for the Master's Degree of
Advanced Telecommunications Technologies (Universitat Politecnica de Catalunya)
in
Department of Electrical and Computer Engineering
in the Graduate College of the
Illinois Institute of Technology

Approved ____________________
Advisor

Approved ____________________
Co-Advisor

Chicago, Illinois
December 2021

ACKNOWLEDGMENT

The work of Jordi Marias I Parella at Illinois Institute of Technology was supported in part by the National Science Foundation under grants CNS-2008092

I would like to dedicate this thesis to my flatmates in Chicago: Atticus, Cristóbal, Hernán and Raeh. Whom I had the pleasure of meeting during my stay in the US.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Page

ABSTRACT

Vehicular Ad-Hoc Network (VANET) are mostly used to support safety applications within mobility environments. But the nature of such communications, where the networks are highly dynamic, with messages usually broadcasted and without any acknowledgements or prior knowledge of who will receive a sent packet; makes these networks easy to get congested. Especially in urban environments, where it's easy to find large amounts of vehicles in a relatively small area. This project makes use of the Age of Information (AoI) theory and metrics to design a new Cooperative Awareness Message (CAM) dissemination algorithm which automatically handles the frequency of sending messages adjusting itself to the congestion. Proving that, using this AoI-aware algorithm, there is a better performance than the standardized solution.

CHAPTER 1

INTRODUCTION

The Basic Safety Applications (Cooperative Awareness in Europe) are the 101 of the VANET use cases. They mainly rely on the periodical sending of the position by each element involved in the mobility environment to the close surroundings. That way, each vehicle, has an extended sight of its neighbouring agents. Thanks to that, dangerous behaviours and problematic trajectories can be spotted to prevent accidents or any other kind of dangerous circumstance that may arise.

Despite we can rarely find a commercial vehicle with the expectation to be prepared for such communications, these sorts of applications have been under development and study for a long time. To the point that the main standardizing agencies have already issued versions of their own standards: The WAVE protocol stack issued by the IEEE and SAE (otherwise called the US standard) and the ETSI C-ITS standard issued by the European Telecommunications Standards Institute (ETSI) (otherwise called the European Standard).

Nevertheless, there are a lot of challenges to overcome to get these technologies to hit the commercial ground as optimized as possible. The principal one is intrinsic to VANET communications: the congestion of the environment due to the fact that, by nature, you never know the number of nodes that are sending messages at the same place.

Also, another great challenge is the capacity to assess the performance of such networks. Because its goals are completely different from other, let's say, more conventional ones. For example, the point of IP/TCP/conventional wifi or mobile networks would be to get as great throughput as possible, with the same protocols but using UDP the goal would be to get lower delay and minimization of packet loss; while the point of VANETs will just be to keep the awareness of the trajectory of the surrounding vehicles as accurate as possible. And is at that point where Age of Information

(AoI) fits perfectly with the Basic Safety use cases of the standard. If we look at the problem of Cooperative Awareness with the perspective of a sensor, in that case, a GNSS device; that sends its gathered measurement through a network, VANET, to a monitor, surrounding vehicle. We can see that the point is not to get a high throughput or low delay, although having a good score in these two is also preferred; but to keep the "monitor" (surrounding vehicle) with the freshest information possible. Here is where AoI defines itself as the metric that regards how old is the freshest information kept by a monitor [22].

In that line, the present project defines its main goals as the improvement of Basic Safety Message (BSM)/Cooperative Awareness Message (CAM) sending schedule, to lower; as much as possible, the channel load though using the AoI metrics and theoretical approach. This will help us demonstrate that with the right perspective (and metric) there can be achieved similar performance, but with less channel load.

Despite the telecom markets developing at a quick, accelerating pace. The truth is that mobility-related industries (like the automobile) take a longer time to adopt new technology. Usually, people change their phone every 18 months and their car every 7 years; which gives us a sense of how much time it takes to renew all cars around compared to phones. This explains why, despite the kickoff moment for vehicular technologies was in the early 2010s, we haven't still seen any broad implementation of that technology. However, all the major institutions have already started their work on it.

1.1 Intelligent Transportation Systems

The concept of Intelligent Transportation Systems, in some sense or another, has been around nearly since the beginning of modern-day transportation. From the first 19th century trains, whose primitive telephone lines were deployed along its tracks and also provided coordination between the different stations to avoid collisions; to the development of the first traffic lights (when the technological developments allowed to). To today, where the huge availability of computer processing and communication power our society have allows us to imagine and plan to deploy a nearly autonomous, self-driving, fully aware mobility environment. In that line, being strict on the definition, the idea of Intelligent Transport systems comes from a 2010 European Directive that says: “Intelligent Transport Systems (ITS) are advanced applications which without embodying intelligence as such aim to provide innovative services relating to different modes of transport and traffic management and enable various users to be better informed and make safer, more coordinated and ‘smarter’ use of transport networks”.

But our interest is on the side of networking. In that sense, ITS means the capability of all of the entities involved in a mobility environment of communicating between themselves in order to provide noticeable improvements in areas like autonomy, safety, fluidity and reliability. To do so, there must be considered a wireless environment where all the moving parts of the system are capable of “talking” to one another. The problem to solve appears when it’s noticed that such kinds of communications are done in an environment that has nothing in common with other more traditional communication networks.

The fact that a vehicular environment is highly dynamic poses a great challenge. The moving parts of the system can appear and disappear from the area of concern of a specific node. I.e. in an urban environment, the cars take different directions ergo they enter and leave the sight of a particular car, which means that the communications are pretty much unstable. Usually, a vehicle rarely knows who will be the receiver of a sent message. And also, rarely it will be capable of predicting how many vehicles will be trying to communicate with him. This forced the networking engineers to open a new branch of research and technology development. The branch of VANETs, which are the main concern for the current project.

1.2 VANETs

VANET stands for Vehicular ad hoc networks, which, is a subset of what during the 90s was called a wireless ad hoc network (WANET). Whose general meaning, as the name says, is an ad hoc network where the communications are spontaneous between the different nodes involved[20]. The reason why the term VANET currently is more known is that a vehicular environment is a paradigmatic case where to use such a network. If we take for example a case where a car relies on such technology while moving, we can quickly see that most of the interactions done by it are completely spontaneous; and the car will never know how many nodes are in its area of influence. This means that VANETs are designed to overcome the challenges explained in the section above.

1.2.1 Challenges of VANETs. VANETs are by definition problematic. The highly dynamic environment for which they are designed makes them by definition have a higher degree of instability than other conventional static wireless networks.
In a mobility environment, every entity (vehicle or infrastructure) is moving and changing its behaviour. Which makes it impossible, or highly inefficient, to rely on centralized Access Points. On top of the normal challenges that an ad-hoc network poses, there can't be known which nodes will receive your messages (which users are within your area of concern) or from which entities your node will receive information from. Thing, that forces the communications to be generally broadcasted. Adding this extra layer of difficulty prevents you from acknowledging if your frames have been properly received or not. Making VANETs generally unreliable and with poor robustness. And finally, like if we hadn't had enough, the system must be designed to deal with a huge amount of nodes. Urban mobility environments have entities in

the order of magnitude of hundreds if not thousands. Which makes the handling of congestion a clear priority.

1.3 Use Case and Motivation

The current project use case regards the so-called safety applications for VANETs. More specifically the Basic Safety Service or what is also called the Cooperative Awareness Service[13]. This use case relies on the constant sending of position and characteristics by a road user to his peers. Doing this, we are given an extra layer of awareness of the position of all the entities involved in a vehicular environment. Which is information that can be used in a huge variety of applications. The most obvious use is collision avoidance, knowing the trajectory of all of the vehicles can give us incredible capabilities of avoiding dangerous behaviours.

As it can clearly be seen, the need for having accurate and fresh position and trajectory awareness of neighbouring road users requires reliability and robustness which is the main weak point in the nature of VANETs. Along with that, we know that VANETs can get to a congested medium pretty quickly. If they are compared to other domestic local wireless networks, like our home Wifi WLAN, they usually just have a small number of connected nodes. But within a vehicular environment the nodes communicating at the same time, especially in an urban environment, can grow really quickly. which means that is easy to congest the medium.

The current approaches of handling the congestion are quite complex and disregard completely the relevance of the information carried by the packets they are restricting. The main pillar of the current project, AoI, gives us the necessary tools to deal with this problem.

CHAPTER 2

STATE OF THE ART AND RELATED WORK

2.1 Intelligent Transport System Protocol Stacks

When we are talking about ITS protocol stacks we are referring to the VANET specific standardized protocols which are designed to deal with all of the challenges that such a type of network poses. We can find two vehicular communication stacks fighting to become the world norm: The “US” stack (standardized by the IEEE and the SAE) and the “European” stack (standardized by the ETSI). Despite both proposals sharing nearly all of the working principles and the ideas behind the use cases for which are designed, to the point that some of the standardized messages are just a copy-paste with a changed name; the reality is that don’t relate to each other and are not compatible. The author of the current thesis comes from a European country and has more experience with the European proposal. And for timing reasons, the current project has been developed with these standards in mind. Although all of the concepts and algorithms can relate to both stacks. To give an interesting background to the state of the art when it comes to the congestion control in VANETs, which is completely transversal in multiple layers of the protocols stack I’m going to give a small glance over both stacks.

2.1.1 IEEE stack. The US stack, which has the support of all of the major American car manufacturers along with the majority of US hardware designers, has been generally standardized by the IEEE in collaboration with the SAE. After expert committees exposed the US congress to issue a directive reserving part of the radio spectrum to VANET communications and established the incentives for big telecom and car manufacturing companies to start collaborating towards the design and deployment of such technologies. Now, about 20 years later, the 5.9 GHz frequency band is reserved for such communications. And the IEEE issued a WiFi (Access

Layer) protocol designed for VANETs which uses CSMA as MAC [9]. Which, despite it has been widely proved that has a huge room for improvement, especially in dealing with highly congested vehicular networks [19]. Is still considered an acceptable solution because the channel allocation MAC standard gives acceptable reliability [10]. As an alternative on the access layer, the mobile standards have started issuing improvements to the existing IEEE 802.11p and now, the more promising technology for medium access is the so-called C-V2X. In its "ad-hoc" VANET-useful working mode (called C-V2X Mode 4) uses SC-FDMA as MAC and have better availability of resources [1].

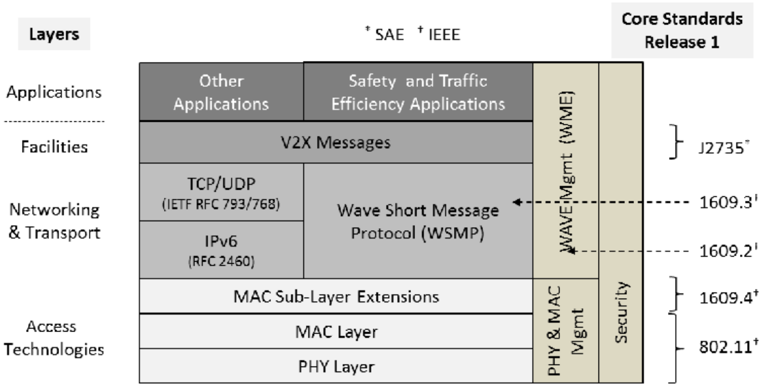

Figure 2.1. IEEE and SAE protocol stack

On the network and transport layers, the IEEE standardized the WAVE protocol. Which covers most of the use cases regarding VANETs by giving the capability of sending bandwidth-efficient single-hop broadcast messages called WSMP [11]. Finally, the Facilities layer, which is the layer of concern for the current project, standardizes the dissemination and type of information messages that should be sent in a VANET in order to cover generalized well-known use cases for vehicular communications. The institution that standardized all of these messages is the SAE and the dictionary of software services and messages are regarded as Dedicated Short Range

Communications (DSRC) [16]. The service of interest in the current project is the Basic Safety Message (BSM) which, according to the standard is constantly emitted by all of the nodes involved in a vehicular environment; revealing their trajectory. That way, an extra layer of vision is given to the vehicles so they can anticipate and prevent possible accidents, or misbehaviour on the road.

2.1.2 ETSI's stack. The so-called "European" standard came after the "US" one. And, in some sense, is a copy-pasted stack with added features and improvements and the names changed. On the MAC layer, the technologies used are exactly the same as the IEEE standard. To the point that today, the continent still using IEEE 802.11p the most, is still Europe. Although the colliding celerity of the innovations constantly appearing on the telecommunications market contrasted with the slow pace of the mobility market. Forces all of the car manufacturers to take extra caution in taking a stand for any particular technology. Despite some conferences are still discussing the battle between C-V2X and 802.11p, with the probable improvements. The mobile networks technology and the constant releases issued by the 3GPP have the potential to leave obsolete most of the current standards in a matter of a few years.

On the network layer, the ETSI proposes the GeoNetworking protocol. Which takes the basic "single-hop" ideas of the WAVE protocol and extends its capabilities to other multicast and unicast capabilities within the environment of VANETs. Also adds extra information to the headers of each message [6]. On the transport layer, it standardizes the Basic Transport Protocol (BTP), which could be quickly explained in a sense of saying that is an even simpler UDP. There is no acknowledgement and no handshake of any kind, adapting perfectly to the nature of VANET communications.

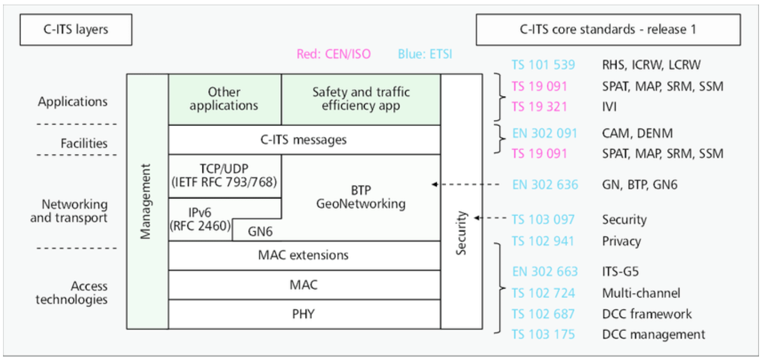

Figure 2.2. ETSI protocol stack

Finally, the Facilities layer, literally copies most of the DSRC dictionary of services and messages, adding an extra header (called the ITS header) [8].

2.2 ETSI's Decentralized Congestion Control (DCC)

The ETSI's Decentralized Congestion Control (DCC) standardizes a set of algorithms that runs in nearly all layers of the protocol stack. This set of the algorithm run without any coordination with other nodes in the VANET. The Figure 2.3, shows where all the DCC algorithms operate across the protocol stack. The point of this transversality is that nearly every layer is capable of checking if there is any possible restriction.

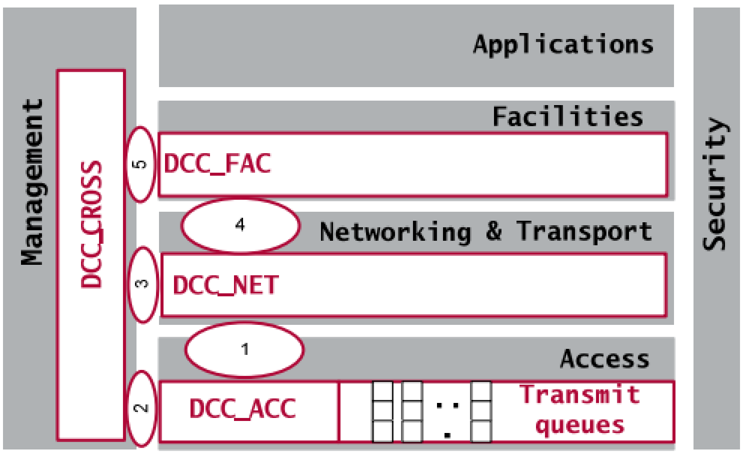

Figure 2.3. Decentralized Congestion Control (DCC) across all protocol layers

As said before, the DCC runs completely without any awareness or coordination with peer nodes within the vehicular network. To assess the network and restrict the sending of new messages the DCC combines two approaches on the MAC layer, that later are communicated to the upper layers.
To do so can apply multiple ways of controlling the network load. By which, when changing those parameters is capable of improving the Channel Busy Ratio (CBR) which is ratio between the time the channel is sensed as busy and the total observation time.

- **Transmit Power Control (TPC)**

  The node is capable of changing the output power of the sent messages. If the channel is congested can reduce the energy so the farther away nodes don't get the messages and their CBR improves.

- **Transmit Rate Control (TRC)**

  The node is capable of changing the time between two consecutive packets. If the time is small, the sending frequency gets higher. By adapting this parameter, we can reduce the congestion.

- **Transmit Datarate Control (TDC)**

  This control capability regards the lower layers wireless data rate options. to decrease the time on which a message is being sent, this could be adapted.

Those parameters are then adapted using two parallel approaches:

- **Reactive Approach** The reactive approach restricts or relaxes the control of the parameters described above depending on the values measured by the CBR.
- **Adaptive Approach** The adaptive approach runs every 200ms and, by applying some measurements to the present and past CBRs is capable of calculating a parameter that represents the maximum fraction of time that an ITS-S is capable of transmitting on a wireless medium for a given interval. With that, then it models the control capabilities commented above.

2.3 ETSI's Cooperative Awareness Basic Service

The reason this section exists is because the AoI based algorithm proposed on the current project relies on the simplification of VANET's congestion control; along with the improvement of the mechanism that adapts the sending frequency of Cooperative Awareness Message (CAM)s, which are the messages that constantly are revealing the trajectory of the sending entity.

According to the standard, the Cooperative Awareness Basic Service is the vehicular communications service that makes sure that all of the road users and roadside infrastructure are informed about each other's position, dynamics and attributes [5]. This service allows for the appliance of several VANET's use cases like "Longitudinal Collision Risk Warning (LCRW)[4]" or "Intersection Collision Risk Warning (ICRW)[7]"; thus is mandatory for any node in a vehicular network to provide that service.

The point of interest that poses to this project is that the rules specifying when to emit a new trajectory (CAM) are quite simple. And most of the time lead to unnecessary updates. Which provides a huge margin of improvement if we try to get a better sampling frequency adapting method. The fields that can be found in a CAM revealing a given trajectory are the following ones:

- Position (latitude, longitude)
- Speed
- Acceleration
- Heading
- YawRate
- Vehicle Parameters

And the service forces us to send a message with the following rules:

- The CAM sending interval must be great than 100ms
- The CAM generation interval must be lower then 1 second
- Depending on the DCC finding the medium congested a CAM is sent when:
    - Current heading and last send heading exceed 4º
    - Current position and last send position exceed 4m
    - Current speed and last send speed exceed 0,5 m/s

2.4 Age of Information

Traditional network assessing metrics, like throughput, delay or packet loss are the gold standard when it comes to checking the performance of most networking systems. But, despite that, there is mostly forgotten the end-use of a network when examining them. This created the need, now that cutting edge telecom technologies like 5G pursue the development of applications that require huge low latency communications; to start getting new perspectives on information freshness. And here is where the Age of information gets relevant. The main pillar of this new approach (that also generates metrics) is the timeliness of the information that a node receives from other nodes. In other words, it provides a completely new approach to measuring the staleness of the stored updates on a receiving entity. So, if a diagram like the following one is considered:

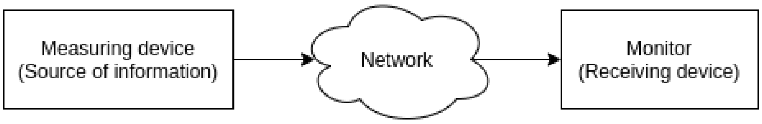

Figure 2.4. Diagram showing the Source to Monitor flow of data in a low latency network

Where there is a source of the information, that has to keep a monitor updated with the freshest information possible, from the measurements that the device is constantly doing. AoI can be described in a fairly simple way: Let's imagine a measurement generated at time u. Then, if such update is received at time t, we say that, at the receiving moment, it has an "age" (AoI) of $t-u$, obviously understanding that always $t > u$. In that sense, the AoI is a timely function where there is always considered the last update measurement and the current time

$$\Delta(t) = t - u(t)$$

To get a more accurate sense of how this function use to look there can be plotted a simple AoI through time measurement:

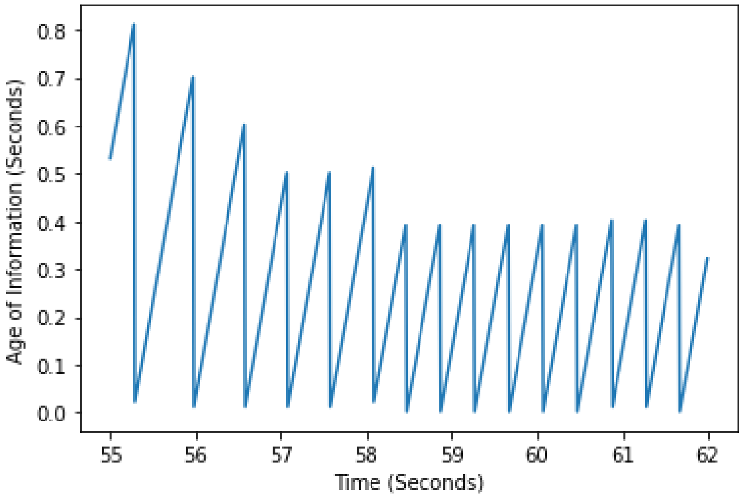

Figure 2.5. Example evolution of a constant measurement of the AoI

By analyzing Figure 2.5. Considering the x-axis the time and the y axis the evolution of the Age of Information of the updates received in a monitor: there can be seen a saw shape graph. Where the staleness of the information increases linearly with time and decreases in a slope because a new update has arrived on the monitor. Also, despite it might be quite difficult to see, the AoI never reaches 0. The reason behind this is the delay that the network poses when an update goes through it. Also, by watching the plot, there can clearly be seen that the information never gets older than 0.8 seconds, and usually gets refreshed before getting older than half a second. The great point of AoI is that not only focuses on the end purpose of a network. That, in the mentioned case, is to keep as fresh information as possible at the monitor side; but integrates all of the traditional metrics within. The local minimas are nothing more than the value of the delay added by the network. If all of the slopes are counted, one can figure out how many packets have reached the destination, and with just knowing the size of each packet the throughput can be quickly computed.

The integration of all of these "assessments" of a network in one graph, forces us to focus on certain aspects of the plotted function to get numeric metrics, that then are useful to compare the performance of different systems. So the following sections describe the best transformations that can be done to such a function to get exactly the "meaningful" desired metric.

2.5 Average AoI

Usually, when trying to get a single value out of a numeric series, the first transformation that comes to mind is the mean value. If the particular case is considered, there should be taken into account that we are averaging the value through the time of the sawtooth function. If expressed mathematically the equation would be the following one:

$$\langle \Delta \rangle_\tau = \frac{1}{\tau} \int_0^\tau \Delta \left(t \right) dt$$

Where τ is the time window on which such value is computed. And the $\Delta \left(t \right)$ is the Age of Information function.

If we give proper meaning to the numeric value acquired by averaging the AoI through time. There can be said that this value is useful if the application on which the data is used requires to have a data freshness on average below the computed number. This also, has incredibly interesting applications coming from the fact that is computed in a specific time window, opening the possibility to create a system that constantly computes the average. And, when the value increases, an anomaly on the behaviour of the network or the source is detected.

For the current project, despite we will use this metric to compare the performance of all of the prosed algorithms with the basic standard, the truth is that is not our main interest.

2.6 Peak AoI

Usually, when designing low-latency cyber-physical systems, the main concern is the worst-case scenario. How worse the latency can get when transmitting using a certain network; so, depending on the application preventive measures can be taken to avoid these possible problems. And here is when the Average peak AoI gets all the relevance. It averages all of the values reached when peaking, so we get a good glance at the maximum staleness peak.

In other words, gets the values of all of the local maximums and averages them all within a period of time. If the idea is placed into a formula it would look like this:

$$\Delta^{(p)} = \frac{1}{N}\sum_{i}^{N} p_i$$

Where p_i is a local maximum found in the interesting time period τ(In that case it could be the same timespan as the Average AoI).

This transformation, and the metric measured with it, is the main pillar to assess the performance of a system like the VANET Basic Safety service. This happens, because when designing such a system, where all the vehicles reveal their position. The interest comes from finding the worst-case scenarios. And this metric provides a value describing how old the information gets in the system. So, for example, if there are two cars periodically emitting their positions through a VANET. And there is known that this VANET has an Average peak AoI of 5, is known that the system will be designed to be robust in case of 5 seconds of staleness in data[21].

In the current project this metric, along with the following explained one is the main pillar to prove the enhancement of the standard through AoI.

2.7 Nonlinear Age Functions

Despite plain AoI is perfect for assessing a few communication systems. Sometimes it can fall into a meaningless metric that doesn't describe the performance of a given network if the final application is considered. One example would be when we are transmitting, to a monitor, the temperature of a certain place. If the temperature never changes, but the measuring device keeps updating the monitor with completely irrelevant updates. The AoI metrics described in the past two sections will mark that this system is performing better than a system that awaits a new relevant temperature change to update the monitor. What makes this flaw on the plain AoI assessment really important to be amended is that if the proper treatment to the metrics is not given, a lot of networking resources might be uselessly used with irrelevant data packets.

To give more meaning to the AoI metrics, any transformation can be applied to the original function, so the linear parts of the plot adapt to our interest.

Continuing with the example of the temperature, when it doesn't change and the source doesn't update the monitor the data gets old, but not stale. Here is where according to our interests we have to apply a transformation to the function to measure that: the staleness of the information (where the main input into this transformation is how old the information is). In the current example, there could be taken into account how different is the new measured temperatures with the last one sent; and how old is the last sent packet.

$$\Delta_p\left(t\right) = p\left(\Delta\left(t\right)\right)$$

2.8 Related work

As explained in sections before, the main pillar of the current project is the appliance of the AoI theory to the congestion control in VANET's. Now that we have already explained the theoretical background to understand such kinds of networks, the standards, and the challenge they pose; let's talk about the similar work already done by other researchers.

There is related work on every layer of the stack. Thus, the congestion control concerns the behaviour of the whole system. With special emphasis on the access layer, which is the most obvious one to do research about. Because it handles the direct queues of all of the messages that have to be emitted through the physical medium.

If there is taken a look at the AoI survey paper [22]. The first practical background that points out is the appliance of AoI in different queuing systems. The first non-obvious characteristic that has to be taken into account is that there is a tradeoff between the delay of a given system and the average AoI [18]. Meaning that there is a sweet spot to be found and also explains that an improvement of the delay of a given network doesn't necessarily mean better performance when it comes to the end goal.

Another great way to improve the AoI performance of a VANET is to make the Access Layer aware of the performance is having in that sense. This approach is carried out at [3]; along with Reinforcement Learning. Which is the one that penalizes the behaviour of an AI every time it starts performing worse. And clearly shows good signs for improvement.

On the other side, there can be found other tradeoffs. Like the one between throughput and AoI. Where, when both parameters are relevant, there must be optimized a middle-ground where both perform under an acceptable level [12].

When it comes to improving the congestion, using AoI there can also be proven that the MAC CSMA (which is the one used in the IEEE 802.11p) underperforms other distributed schemes like SC-FDMA which is the one used by the C-V2X [14].

CHAPTER 3

METHODOLOGY

3.1 AoI and VANETs

The main pillar of the present project research is the usage of the AoI theory to be able to propose and run a new algorithm controlling the sending of positions within the VANET environment. The theoretical framework of AoI is perfect for the problem being solved because focuses exactly on what is of our interest: The position and trajectory of all the elements involved in an ITS environment. In that sense; is a far better metric, against, for example; the classical “delay”. Because is more relevant to a safety-oriented VANET application to have the current trajectory: than the time it takes to refresh that trajectory.

3.2 Giving meaning to AoI

If there are compared two data refreshing systems, without losses or queuing issues: one that sends double the messages the other. The obvious result when it comes to AoI metrics is that the one that sends more messages (with fresher information) more frequently is the one that will get better AoI. This would mean that the main goal of this research: which is based on reducing the channel load and keeping the information from staleness is impossible. However, the main point of AoI is that you can give meaning to the metric and assess it by transforming it through a penalty function. Giving meaning to AoI allows us to, not also assess the freshness of the received data; but also the relevance. The perfect example would be when a car in a VANET environment stops in front of traffic lights. Using the current standard it should keep sending messages revealing its position. But the position is always the same; which means that the new information is irrelevant to the surrounding cars. If the last example is analyzed with the perspective of simple AoI, there would be seen that a constantly-refreshing car performs better than a car that prefers to remain silent until changing its position. However, when transforming AoI with a penalty function that takes into account the relevance of the sent data. In that case, the predictability of the trajectory of the node. There can be seen that both systems perform similarly; with the difference that the latter one sends fewer packets and congests less the channel.

3.3 The Naive Algorithm

As stated in the previous section, the point of using AoI-driven systems is to adapt the AoI measurement to one that has meaning for the end use case. In that case, we are talking about Cooperative Awareness; a system that makes sure every road traffic user gets informed about each other's position. To have a reliable awareness, there must be ensured that the positions calculated from the interpolation of trajectories between one update and the other are accurate. To do so, we know that the freshness of data stored within a node depends on how much the trajectory of other road users has differed from the last sent CAM.

This means that the non-linear increasing function that gives meaning to plain AoI is the one that measures how much the calculated position with the interpolation of the last revealed CAM and the real position differ. If we consider $\vec{p} = (p_x, p_y, v_x, v_y)$, the current real position of a node; $\vec{p_i} = (p_{ix}, p_{iy}, v_{ix}, v_{iy})$ the last sent update and t_{AoI} the Age of Information of $\vec{p_i}$. Then the penalty AoI could have the following form:

$$p\left(t_{AoI}, \vec{p}, \vec{p_i}\right) = ||\vec{p}, I\left(t_{AoI}, \vec{p_i}\right)||$$

Where I is the transformation that computes the predicted trajectory based on $\vec{p_i}$ at time t_{AoI}. I in that sense can be any trajectory predicting function.

3.4 Trajectory prediction

The I transformation could have multiple forms; and in most if cases with huge impact to the outcome of the present project. The approach there has been taken in the current project is the usage of simple physics. Although, if there had been more time, more complex and improved methods like Kalman Filtering or the usage of Deep Learning; it could have been great (See future work). So the basic physics formula used for predicting the current position considering the last known one is the following one:

$$p = p_i + v_i t + \frac{1}{2}at^2$$

Which is simple high school level, quickly to compute math.

3.5 Computation of the Penalty function

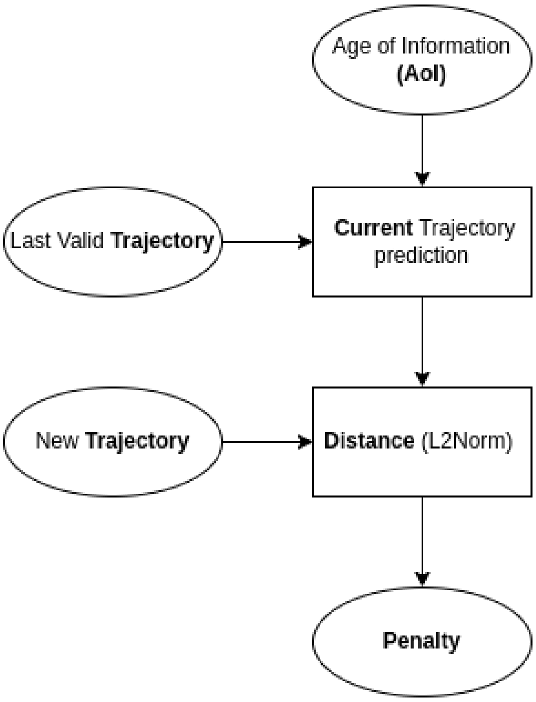

Figure 3.1. Penalty function computation

3.6 Naive algorithm implementation

To explain the implementation of the algorithm and the logic behind it, we will use the case of a simple node A transmitting to a simple node B. Due to the nature of VANETs, where there is never a known static network and the environment is in constant change, there is no possibility of acknowledgements. So, all of the transmitting policy frequency of the algorithm runs on the transmitting node (In that case, node A). We also have to take into account the type of data we are handling here: positional data gathered from a GNSS device, with the known limitation of providing at most one measurement every 100ms. As explained in past sections, the penalty function that gives meaning to the AoI is the distance (L2Norm) between the measured and predicted trajectory (position and speed). Also, the predicted trajectory is done through simple physics. On the sending node, the node receives a new measurement every 100ms and compares the "real" position with a self-predicted one. Which, is based on the last sent location. That way, node A is capable of, with imperfections checking how well the receiving nodes, like B, will predict its own position. Node B, on the other side, will constantly provide to other VANET-related services the predicted positions of the surrounding vehicles using the same predicting algorithm as A. Doing this, we can ensure that the difference between the self-predicted position at node A and the predicted one at B is the same.

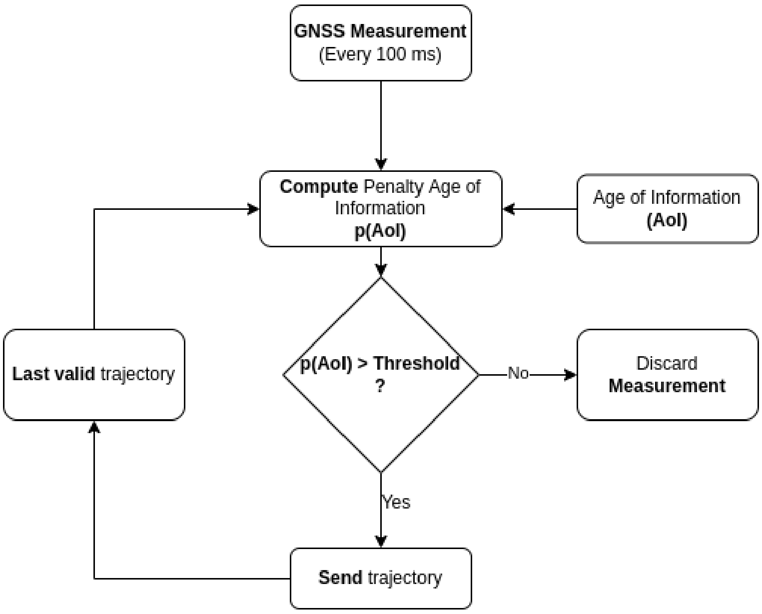

Figure 3.2. Implementation of the naive algorithm

The effects of using this approach are easy to understand. The difference in comparing the last measured location and the self predicted one works as a predictor of staleness of the last sent position which is also a way to check the relevance of information.

3.7 Finding the optimal threshold

The nature of VANET forces us to compute all the AoI related metrics after running the simulation. The receiving node doesn't have direct access to the ground truth positions gathered by the sending node. This means that just have to believe the receiving positions and constantly predict surrounding vehicles trajectories based on last received updates. This means that there is a tradeoff between the number of sent packets (which in some sense depends completely on how low the threshold is) and the congestion when it comes to the AoI performance. If there is a lot of congestion and the threshold is low the AoI will indicate a bad performance while, if there is congestion and the threshold is within the appropriate level; the AoI will reach optimal performance.

3.8 Symmetric Algorithm

Departing from the last proposed algorithm, there should be recognized that have one structural flaw. It isn't capable of detecting the channel congestion directly and just relies on the previously made tests to prove that the channel almost never gets congested with the solution. And it has its point, VANETs don't have any capabilities for acknowledgements by nature. All of the communications are ad-hoc and broadcast. This means that usually, one node doesn't know how many nodes will receive the packets that is emitting. And also the main standard only relies on the DCC to have little control over the congestion. But one of the points of this project is not just the improvement of the information transmission performance and efficiency also is to apply the AoI theory to get a simplification.

The perfect extension for this problem is to consider that the network is symmetrical. And all of the nodes are revealing their positions. If that is assumed, we know that if two nodes receive packets from another; then they can both calculate the past penalty AoI of each other and get a value to know how congested the medium is. To understand how this approach is developed, we will start with the simplest example. Two nodes (A and B), send each other their position using the new proposed algorithm. Every time A gets an update from B. A computes the old "meaningful" AoI of the information sent by B. To do so, gets the last packet (before that new one) received from B; and computes the predicted trajectory for the newly received update.

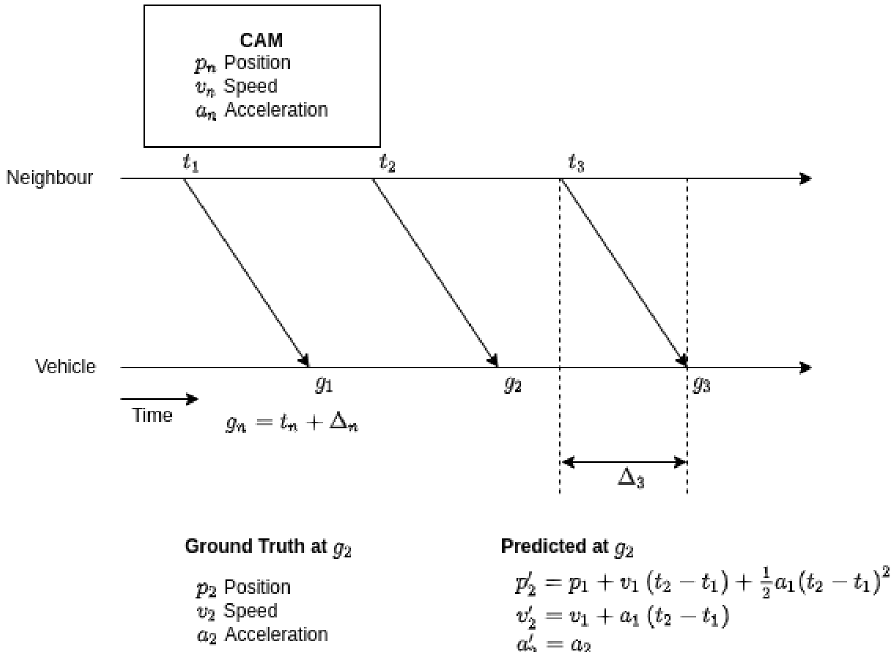

Figure 3.3. Computation of neighbours AoI

To clarify it, Figure 3.3 shows how the ground truth is gathered to compute the other nodes $p(AoI)$. To understand the figure, there must be pointed out that Δ_n is the delay of update n, t_n the sending time and g_n the receiving time. The vehicle computes the $p(AoI)$ of the neighbour when it receives an update that allows him to check how precise has the trajectory prediction is at the vehicle side while the neighbour was measuring it. To put it in more technical terms. At time g_2 the vehicle computes the penalty AoI at time t_2 by comparing the prediction done by the vehicle at time t_2 using the update 1, received at time g_1.

Going back to the main analogy of node A and node B. That way A knows the penalty AoI of the information gathered by B at the moment B gathered it. At the same time, B does exactly the same with all of the updates received from A. If the medium is congested or there is a problem in transmission, this will be reflected on the calculations of both nodes.

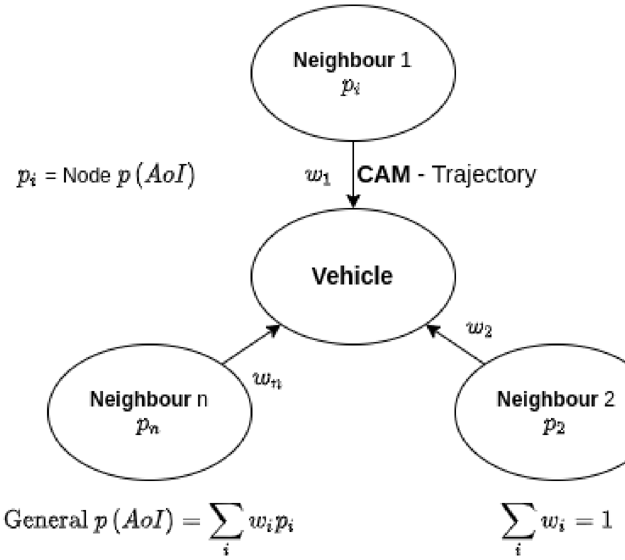

Figure 3.4. Network Symmetry and Weights computation

The second challenge this new approach poses is how to compute the penalty AoI of the medium when there are more nodes. And also, rank the relevance of each node to the penalty AoI. For example, the further any node is from our node, the less relevant the “meaningful” AoI is. On the other side, when more time has passed since the last received update; the more relevant is the penalty AoI of the known mode.

The approach used to solve this problem is to weigh the relevance of a known node penalty AoI (Like it can be seen in Figure 3.4); so the final “general” penalty AoI can be expressed the following way:

$$p\left(AoI\right) = \sum_i w_i p_i$$

Where p_i is the last penalty calculated and the w_i is the normalized weight of the node i. So:

$$\sum_i w_i = 1$$

That way, we can design a calculation to give higher weights to more relevant nodes. And lower ones to the ones are not interesting for us. To do so, as explained above, we consider the distance and the time since the last update. If the distance is higher the interest is lower while the exact opposite happens for the time. If there has been a lot of time since the last update, this is a bad signal and the penalty should be more relevant in the overall computation.

To compute the contribution of distance to the weight; the following formula is used:

$$z_i = \frac{D - d_i}{D}$$

Where D is the distance to the node that is further away and d_i is the distance of the given node. Obviously, this value is not computed if there is no more than one node spotted, and in the case of one node, the default value is 1.

The way the relevance of time is computed is different. Usually, short time periods shouldn't be a matter of great concern. But, at the same time, if a node escapes the communications range the contribution of time to the overall penalty should not the lineal. That's the reason why the sigmoidal function is used to get the time contribution to the weight with a value between 0 and 1:

$$w_{ti} = \frac{1}{1 + e^{-\frac{1}{5}(t-15)}}$$

where t is the time (in seconds) since the last received update. Finally, the weight without normalization w_i is computed just by multiplying the penalization of time and distance:

$$w_i = z_i w_{ti}$$

3.9 Adaptive Algorithm Implementation

The implementation of the adaptive algorithm is done as an extension of the first, naive, algorithm. If the first one was running on a “sending” thread. Where every time a new position from the GNSS device is measured, the whole system decides if this information is relevant or not. The current adapting approach activates itself every time a new CAM arrives. With different words, runs on the “receiving node”. Every time a new CAM from a neighbouring node arrives; its $p\left(AoI\right)$ is computed (following the way described above).

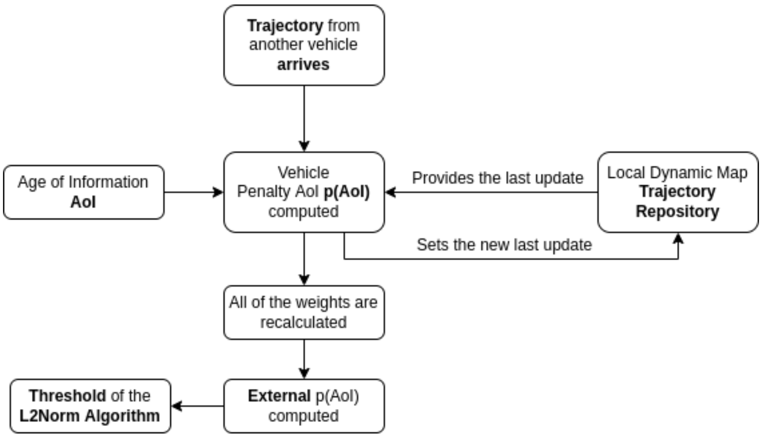

Figure 3.5. Adaptive threshold

As can be seen in Figure 3.5, then, all the weights of all the nodes are recomputed. Taking into account the new parameters that this new neighbour position brings. With all of the measured and relevant positions, the general external $p\left(AoI\right)$ is computed. which is fed to the first algorithm presented in this project as the threshold(with a value slightly lower than the one computed, so the algorithm always tries to improve the quality of the information).

CHAPTER 4

EXPERIMENTS

As explained in the sections above, the current project experimentation compares the AoI-based metrics performance between the ETSI CAM dissemination standard and the proposed AoI-driven algorithm. To do so, there are used the simulation capabilities provided by the OMNeT++ and SUMO simulators; aligned with the Veins/Artery framework. To prove better performance than the standard is important to remark the fact that we are using the "meaningful" penalty function that indicates how stale the currently predicted data (based on the last update) is. As explained in sections above this is done through the calculation of the trajectory. If basic AoI is measured, in non-congested environments, the better performing system is always the one that updates more frequently. But in highly congested ones, then there is a sweet spot to be found in the tradeoff of high updating frequency and a highly congested environment. That's the reason why multiple AoI and non-AoI based metrics will be used to prove that our proposed algorithms perform better despite updating less frequently

4.1 Environment: Illinois Institute of Technology (IIT)

The map over which the simulated vehicles move is the Illinois Institute of Technology (IIT) main campus. Which, curiously, provides the perfect description of what would be a regular neighbourhood of a regular American city. The fact that there are large streets with traffic lights at the end, allow us to see how the predicting part of the algorithm reacts to different vehicle behaviour, i. e. car stopped in front of traffic lights, car changing lane, car keeping the speed...

4.2 Tool: Simulation of Urban MObility (SUMO)

SUMO is an open source traffic simulation [2]. Its main purpose is the experimentation regarding mobility environments well design. There can be checked if a set of roads design has acceptable traffic fluidity. But for the current project, we are using the simulator as a way to create realistic cars trajectories and behaviours. Which then has a direct effect on the VANET behaviour.
In the simulating pipeline, SUMO supports the network simulator by providing all of the nodes behaviour inside the system. Then the network simulator uses this behaviour to simulate how the packets are transmitted around the environment.

4.3 Tool: OMNeT++

OMNeT++ is an open sourced discrete event simulator. Which by "de facto" has become the most used open source network simulator. Its flexibility and support for multiple frameworks using research state standards has allowed this project to completely simulate the whole ETSI C-ITS protocol stack (from the IEEE 802.11p on the MAC layer to the Cooperative Awareness in the facilities layer) with huge reliability. Even the power and deterioration of a sent signal are computed, along with any possible collision.

4.4 Tool: Vehicular Frameworks Veins and Artery

The Vehicular simulation frameworks allow the combination of SUMO and OMNeT++ tools to combine themselves to be able to simulate all of the complexities of a VANET. Both frameworks are the same with the small change that Veins implements the IEEE standard and Artery implements the ETSI one. For the current project, the framework used is the Artery.

4.5 Results

The results for the current project have been collected by the simulation of the described environment (the IIT campus) with a lightly loaded environment, just two cars communicating with each other. And a congested one, with 200 cars communicating with each other. The output from the simulations has been the moment where the packets were sent by a node and a list of all the received packets from other nodes. With this raw data then, all the AoI regarded metrics can be computed after the whole simulation has been completed. And they can be aggregated by creating an average of how well all of the cars perform.

The tow more relevant metrics are the "Average Prediction Distance $p\left(AoI\right)$" and the "Average Peak Prediction Distance $peakp\left(AoI\right)$. The first one tells us how accurate and fresh the information hold by a receiving node is all the time. And the second one tells us how stale use to get the information before receiving a new update.

As it can be noted both metrics are of great relevance, especially on the current use case where we need to ensure freshness and accuracy along with being able to say with certainty how inaccurate the information can get, so the errors can be bounded.

4.6 Two Cars scenario

	ETSI CAM Standard	**L2Norm Penalty AoI**	**Symmetric Congestion Control**
Average AoI (ms)	405,89	428.30	444,83
Average Peak AoI (ms)	430,07	443.25	549.3
Average Prediction Distance p(AoI)	3.045	2.956	3.01
Average Peak Prediction Distance (peak p(AoI))	6.19	6.05	5.8
Average Packets Sent per Minute	180	140	135
Packet Received Ratio	98.16%	99.08%	98.61%

Table 4.1. Results of the 2 cars scenario

In Table 4.1 there can be seen the results obtained in the 2 cars scenario. The main point that has to be commented is that we are obtaining similar (slightly better) Average Prediction Distances sending substantially fewer packets than the current standard. And, even in the case of the adaptive algorithm otherwise called "Symmetric Congestion Control" there can clearly be seen how minimizes the Peak Prediction Distance whilst reducing the number of packets sent at the expense of having slightly worse performance on the average prediction distance.

The 2 cars run is relevant because it proves that the proposed naive algorithm performs better than the standardized solution even in a lightly loaded environment. The fact that the AoI aware solution, by default, provides a better sampling method means that the algorithm can work in all possible scenarios, and there will be better results.

4.7 Two hundred Cars scenario

4.7.1 With Decentralized Congestion Control (DCC). In Table 4.2, there can be seen the results of the simulation regarding the highly congested environment. The reason behind it has been run with the DCC, which is the Congestion Control of the standard is because the solution proposed, not just only can work stand-alone it can work altogether with the standard.

	ETSI CAM Standard	**L2Norm Penalty AoI**	**Symmetric Congestion Control**
Average AoI (ms)	252.57	405.48	434.77
Average Peak AoI (ms)	291.03	419.22	440.21
Average Prediction Distance p(AoI)	1.7	1.68	1,75
Average Peak Prediction Distance (peak p(AoI))	3.54	3.15	3.26
Average Packets Sent per Minute	210	148	138
Packet Received Ratio	90,15%	98,65%	98,86%

Table 4.2. Results of the 200 cars scenario (with DCC)

Basically, the results are quite the same as the 2 cars scenario. This means that the algorithm gets robust against congestion. Both in the naive case and the adaptive case. On average, both the adaptive and the naive perform quite similar. With the difference that the adaptive is capable of getting a little better performance by sending fewer messages and compromising a little bit the peak penalty AoI.

	ETSI CAM Standard	**L2Norm Penalty AoI**	**Symmetric Congestion Control**
Average AoI (ms)	252.57	394.47	419.68
Average Peak AoI (ms)	291.03	405.87	432.70
Average Prediction Distance p(AoI)	1.7	1.67	1.69
Average Peak Prediction Distance (peak p(AoI))	3.54	3.21	3.19
Average Packets Sent per Minute	210	152	143
Packet Received Ratio	90.15%	96.12%	95.19%

Table 4.3. Results of the 200 cars scenario (without DCC)

4.7.2 Without Decentralized Congestion Control (DCC). Finally, in table 4.3, there can be seen the results of both algorithms without the DCC. The fact of not having the DCC affects the sense that more packets are lost. This means means that the adaptive algorithm reacts by sending slightly more messages. But in general terms, the performance is basically similar. This means that this proves the fact that the current's project approach is capable of not just beating the standard solution; but also replacing it.

CHAPTER 5

CONCLUSIONS

The Age of Information theoretical approach to VANETs, and more specifically the congestion-related challenges, open a path towards a great simplification along with an improvement of such challenges.

The results shown in this project prove that giving meaning to AoI using a penalty function based on the right predictability of a sent trajectory through VANETs allow to drastically reduce the number of updates sent with complete similar(slightly better) performance. Or, to give the analogue result, with the same number of packets sent, the AoI-aware approach is capable of performing much better.

The ETSI CAM dissemination algorithm is clearly inefficient. And yet there is a huge room for improvement using the current AoI approach. Both proposed algorithms reduce substantially the number of packets sent per minute while having similar "meaningful" penalty AoI, which measures how stale the information gets on average on a receiving node. And peak penalty AoI measures how stale the information is on a receiving node when a new update is received.

In other terms, there could clearly be proven that the AoI approach to the problem is nearly the perfect one, which gives us the tools to get the needed perspective.

CHAPTER 6

FUTURE WORK

The current project has done the first steps in the path of creating a simplified, AoI-aware approach to the CAM dissemination algorithm. Due to the lack of time and resources, further extensions and slightly more fine-tuned parameters of the proposed algorithm haven't been able to be developed.

But the truth is that there is huge room for improvement. In the penalty function $p\left(AoI\right)$, which uses a fairly simple trajectory prediction. In the computation of the adaptive threshold; which can use different penalty functions, different weights or even different normalizing functions(i.e. now we are using the Sigmoidal function to normalize the staleness of a given packet) for the weight regarded parameters.

Needless to say, the combination of the current approach with improvements in all of the congestion aware layers of the V2X protocol stack can provide the perfect combination to ensure a nearly perfect optimization of the usage of the network resources.

6.1 Improvement on the trajectory prediction

The current trajectory prediction is quite simple and naive. Although the laws of physics (and in the current case we are dealing with a simple physics problem) apply completely. The truth is that they are not quite good when it comes to predicting the trajectory of a vehicle driven through a road. The improvement of the trajectory prediction would mean a lower $p\left(AoI\right)$ because the relevance and durability of each update would be longer. Which, as is obvious, would lead to fewer packets sent and less network congestion.

6.1.1 Kalman Filtering. The Kalman Filtering Algorithm has been developed to be used as a tool to identify the hidden state of a lineal dynamic system. With the extra feature that is quite robust against white noise inaccuracies. In that sense, such a filter is the perfect tool to improve the prediction of positions. [15]

6.1.2 Machine Learning. Machine learning, with all of the derived technologies that come from it (deep learning, reinforcement learning, random forests...) has proven to be a great tool for pattern detection. This, in essence, that's what we are trying to do when we are talking about vehicle position prediction. The trajectory that a car may follow in a mobility environment (like a highway, or a simple intersection) is clearly better described taking into account general patterns than simple physics. [17]

APPENDIX A

AGE OF INFORMATION GRAPHICS

A.1 ETSI C-ITS Standard

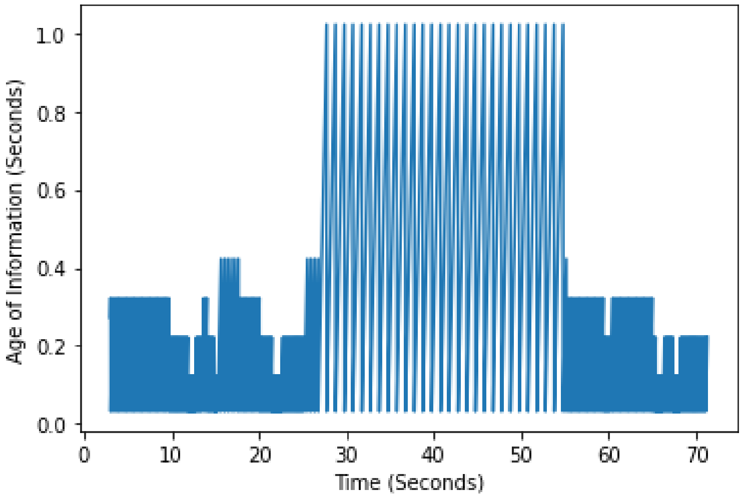

Figure A.1. ETSI C-ITS Standard AoI

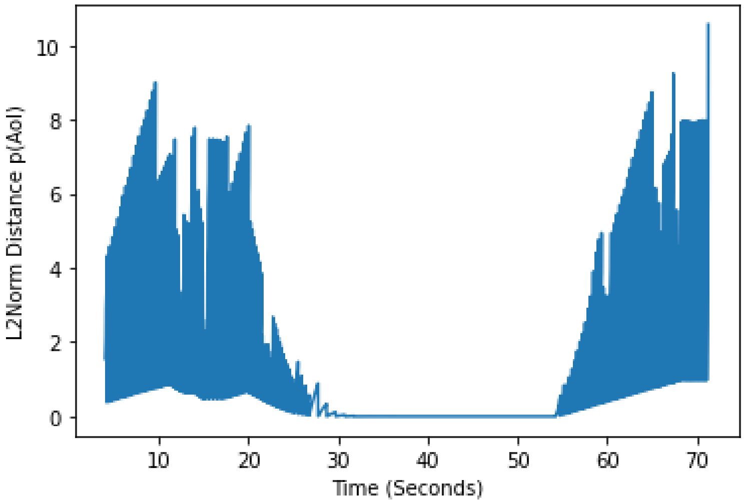

Figure A.2. ETSI C-ITS Standard Penalty AoI

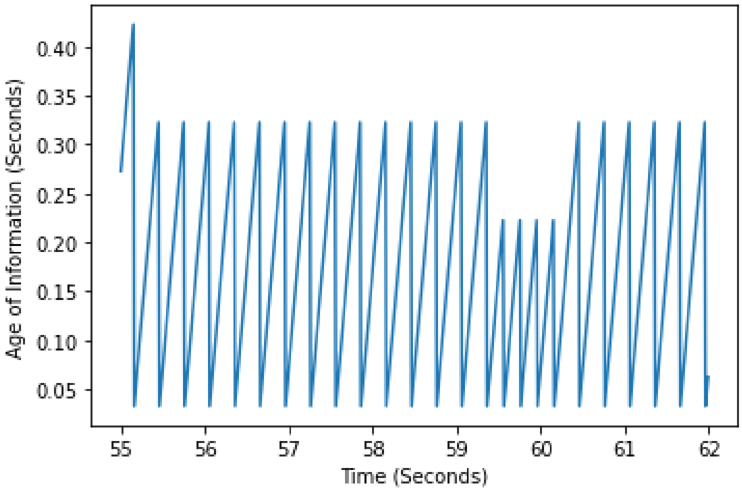

Figure A.3. ETSI C-ITS Standard AoI (Resolution)

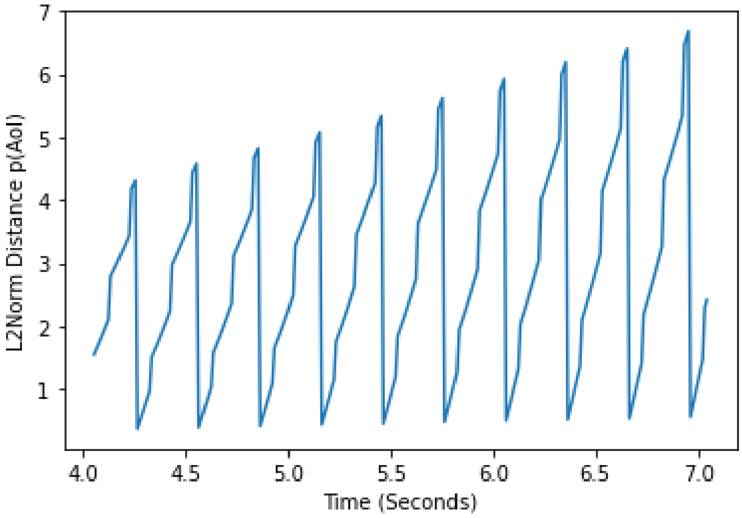

Figure A.4. ETSI C-ITS Standard Penalty AoI (Resolution)

A.2 Naive Algorithm Graphics

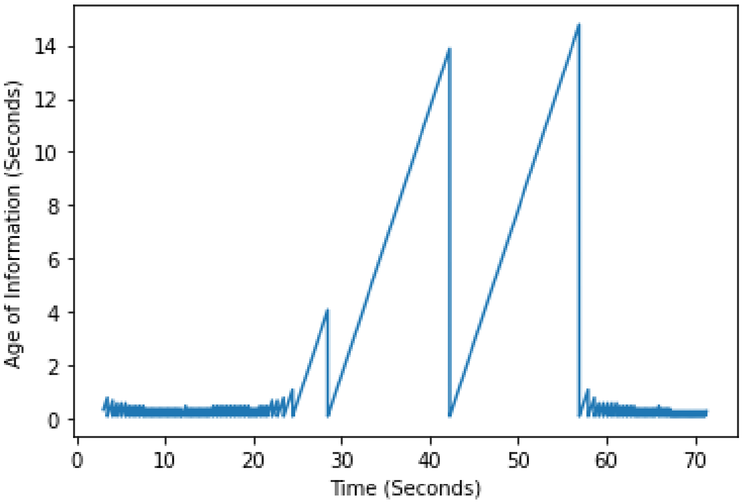

Figure A.5. Naive Algorithm AoI

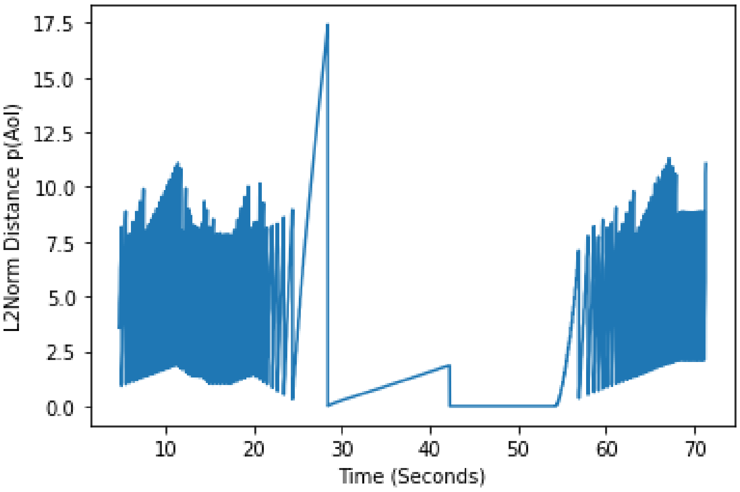

Figure A.6. Naive Algorithm Penalty AoI

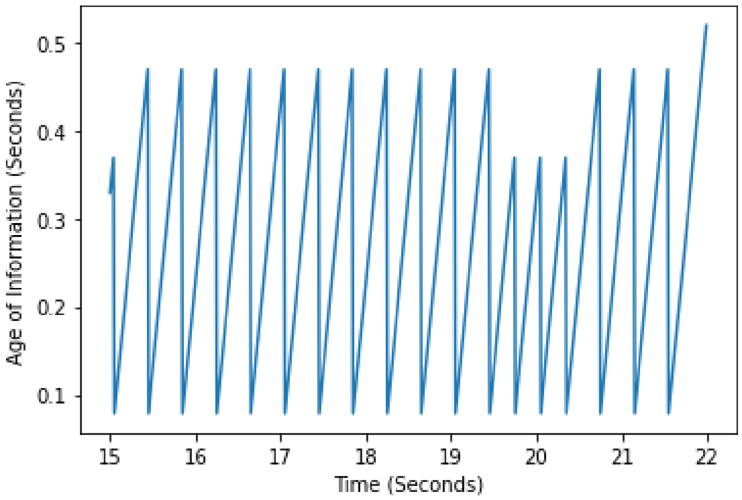

Figure A.7. Naive Algorithm AoI (Resolution)

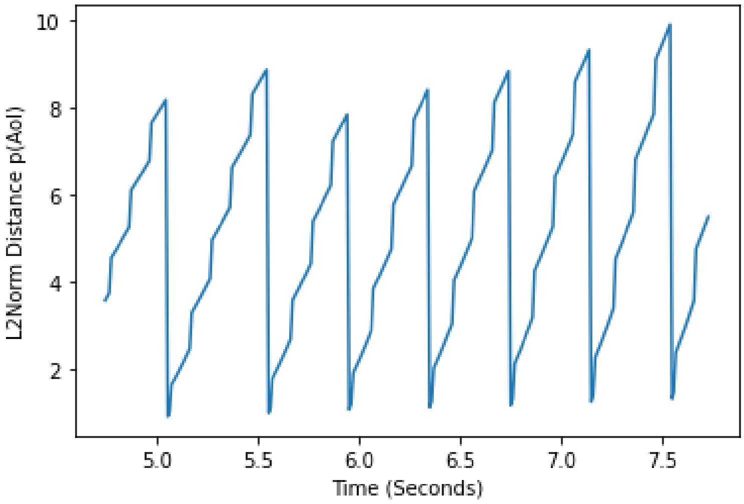

Figure A.8. Naive Algorithm Penalty AoI (Resolution)

A.3 Adaptive Algorithm

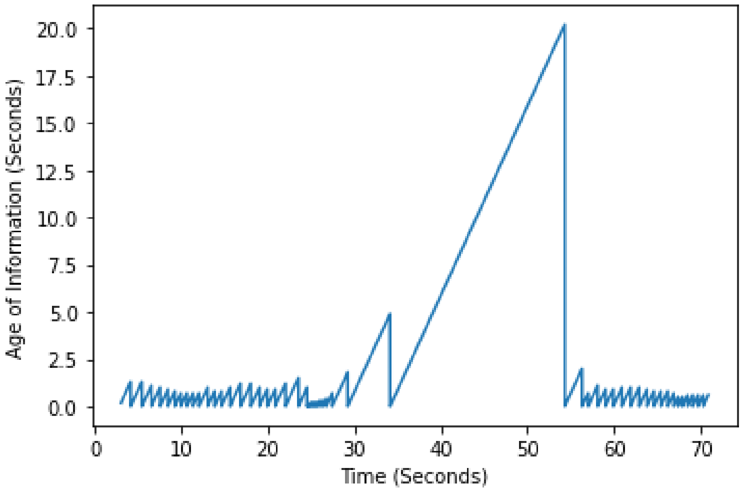

Figure A.9. Adaptive Algorithm AoI

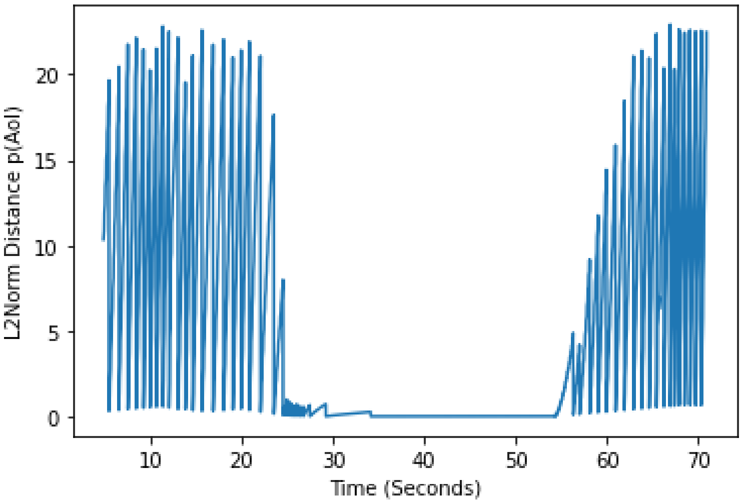

Figure A.10. Adaptive Algorithm Penalty AoI

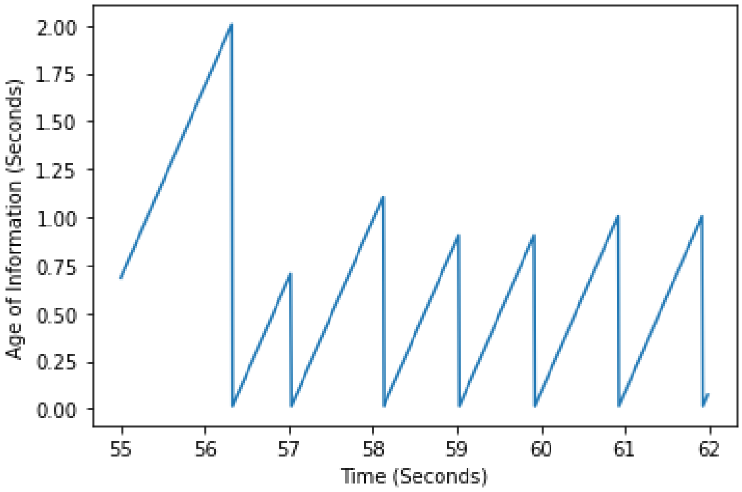

Figure A.11. Adaptive Algorithm AoI (Resolution)

Acronyms

AoI Age of Information. iv, vii, viii, 2, 5, 11, 13–21, 23–27, 30, 32, 33, 36, 37, 40–45

BSM Basic Safety Message. 2, 8

C-ITS Cooperative-Intelligent Transport Systems. v, vii, 1, 31, 40, 41

CAM Cooperative Awareness Message. viii, 2, 11, 12, 21, 29, 30, 36, 37

DCC Decentralized Congestion Control. iv, v, vii, 9, 10, 12, 25, 34, 35

DSRC Dedicated Short Range Communications. 7–9

ETSI European Telecommunications Standards Institute. iv, v, vii, 1, 6, 8, 9, 11, 30, 31, 40, 41

ICRW Intersection Collision Risk Warning. 11

IEEE Institute of Electrical and Electronics Engineers. iv, vii, 1, 6–8, 19, 31

IIT Illinois Institute of Technology. v, 30, 32

ITS Intelligent Transport Systems. 3, 6, 9, 20

ITS-S Intelligent Transport Systems - Station. 11

LCRW Longitudinal Collision Risk Warning. 11

OMNeT++ Objective Modular Network Testbed in C++. v, 30, 31

SAE Society of Automotive Engineers. vii, 6, 7

VANET Vehicular Ad-Hoc Network. iv, viii, 1, 2, 4–9, 11, 16, 18, 20, 23–25, 31, 36

BIBLIOGRAPHY

[1] A. Bazzi, G. Cecchini, A. Zanella, and B. M. Masini, *Study of the impact of phy and mac parameters in 3gpp c-v2v mode 4*, IEEE Access, 6 (2018), p. 71685–71698.

[2] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, *Sumo–simulation of urban mobility: an overview*, in Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation, ThinkMind, 2011.

[3] X. Chen, C. Wu, T. Chen, H. Zhang, Z. Liu, Y. Zhang, and M. Bennis, *Age of information-aware radio resource management in vehicular networks: A proactive deep reinforcement learning perspective*, 2019.

[4] ETSI (European Telecommunications Standards Institute), *Intelligent Transport Systems (ITS); V2X Applications; Part 3: Longitudinal Collision Risk Warning (LCRW) application requirements specification*, 11 2013. V1.1.1.

[5] ——, *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, 11 2014. V1.3.2.

[6] ——, *Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 1: Media-Independent Functionality*, 08 2017. V1.3.1.

[7] ——, *Intelligent Transport Systems (ITS); V2X Applications; Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification*, 06 2018. V1.1.1.

[8] ——, *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Facilities layer protocols and communication requirements for infrastructure services*, 08 2018. V1.2.1.

[9] IEEE (Institute of Electrical and Electronics Engineers), *IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments*, 06 2010.

[10] ——, *IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Multi-Channel Operation*, 01 2016.

[11] ——, *IEEE Standard for Wireless Access in Vehicular Environments (WAVE)– Networking Services*, 12 2020.

[12] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, *Minimizing age of information in vehicular networks*, in 2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, 2011, pp. 350–358.

[13] V. Kumar, S. Mishra, N. Chand, et al., *Applications of vanets: present & future*, Communications and Network, 5 (2013), p. 12.

[14] A. Maatouk, M. Assaad, and A. Ephremides, *Minimizing the age of information in a csma environment*, 2019 International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT), (2019).

[15] S. Romaniuk and Z. Gosiewski, *Kalman filter realization for orientation and position estimation on dedicated processor*, Acta Mechanica et Automatica, 8 (2014).

[16] SAE (Society of Automotive Engineers), *Dedicated Short Range Communications (DSRC)*, 07 2009.

[17] L. Su and L. Li, *Trajectory prediction based on machine learning*, IOP Conference Series: Materials Science and Engineering, 790 (2020), p. 012032.

[18] R. Talak and E. Modiano, *Age-delay tradeoffs in queueing systems*, 2019.

[19] B. Toghi, M. Saifuddin, H. N. Mahjoub, M. O. Mughal, Y. P. Fallah, J. Rao, and S. Das, *Multiple access in cellular v2x: Performance analysis in highly congested vehicular networks*, 2018.

[20] C. Toh, *Ad Hoc Mobile Wireless Networks: Protocols and Systems*, Pearson Education, 2001.

[21] E. Uysal, O. Kaya, S. Baghaee, and H. B. Beytur, *Age of information in practice*, 2021.

[22] R. D. Yates, Y. Sun, D. R. B. I. au2, S. K. Kaul, E. Modiano, and S. Ulukus, *Age of information: An introduction and survey*, 2020.