VANET

Safety

Efficiency

traffic/roads condition

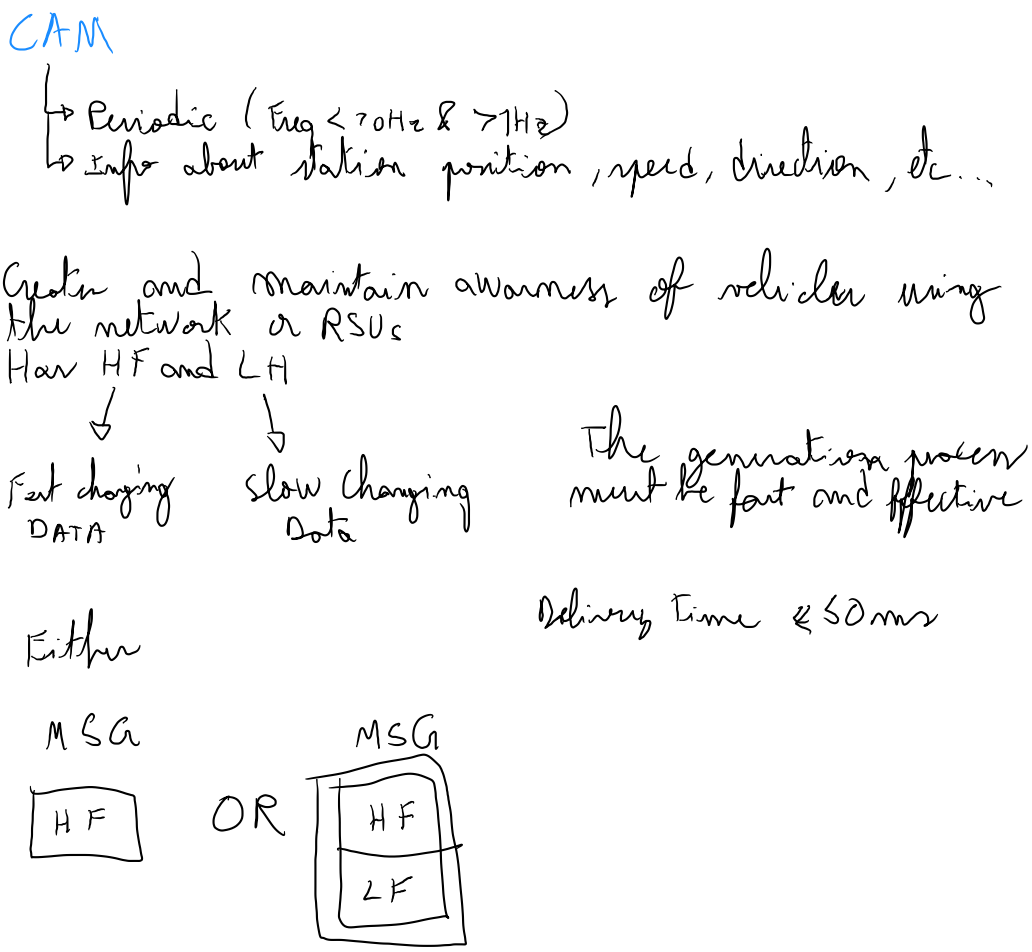
Local information

Vehicular Networks

V2V /V2I

OBUs to perform communications, routing and application

Messages:



DENM

- ASYN
- (i) about an event and the station

Create and maintain awarness of events that occurred in the road.
DENMs are voted based on events that happen and have a validity P, meaning that said DENM msnage is no longer up-to-date.
When an event ends then is a DENM termination msg to announce said termination

VAM

Periodic msg between stations to create and maintain awarness of VRUs
Can distinguish between types of VRUs...

CPM

Periodic messages between STATIONS to broadcast their perception of the environment based by 1 or + sensors.

SPAT (Signal Phase and Timing)

Defines an interface for 2-way communication between traffic control device and mobile

devices. Specifies current movement state of each active phase (Safety, Mobility, Environment) as well as state of lanes and any priorities. Uses **MAP**

MAP

Message that describes the layout of the intersection.

MCM

Describes the planned manoeuver + alternatives. Upon detection of a needed maneuver, thin messages are rent periodically (1 Hz to 10 Hz). These messages can suggest advices for specific vehicles

ITS-G5

- Designed for V2V & V2I
- Based on 802.11a , possesses CSMA/CA
- Adopted for low-latency
- Low Delay

Challenges:

- Safety communications rely on P broadcasts of messages that contain ① about the vehicle (position, speed,...)
- the periodicity of the messages are chosen to meet latency and accuracy requirement.
- There will be channel congestion in dense environments.
- Lack of Ack/handshakes
- No QoS insurance.

C-V2X

- Based on 3GPP release 14
- Bigger delay than ITS-G5 (BAD)
- Bigger time for con in range (BAD)
- Bigger transfer rate 150 Mbs (Good)
- 2 new interfaces:
    - → PC5 for V2V, V2I
    - → Uu for V2N using legacy LTE
- 2 Transmission modes:
    1- Direct communication, low latency for V2V, V2I & V2P. No network needed.
    2- Network communication for complementary services, wer the mobile operator spectrum.

V2V via PC5 (Direct) in built upon LTE device-to-device with enhancements for high speed, high density, low latency & better synchronization.

V2N via Uu leverages exiting LTE network to have a wide area and to broadcast V2X messages from the servers to vehicles and others

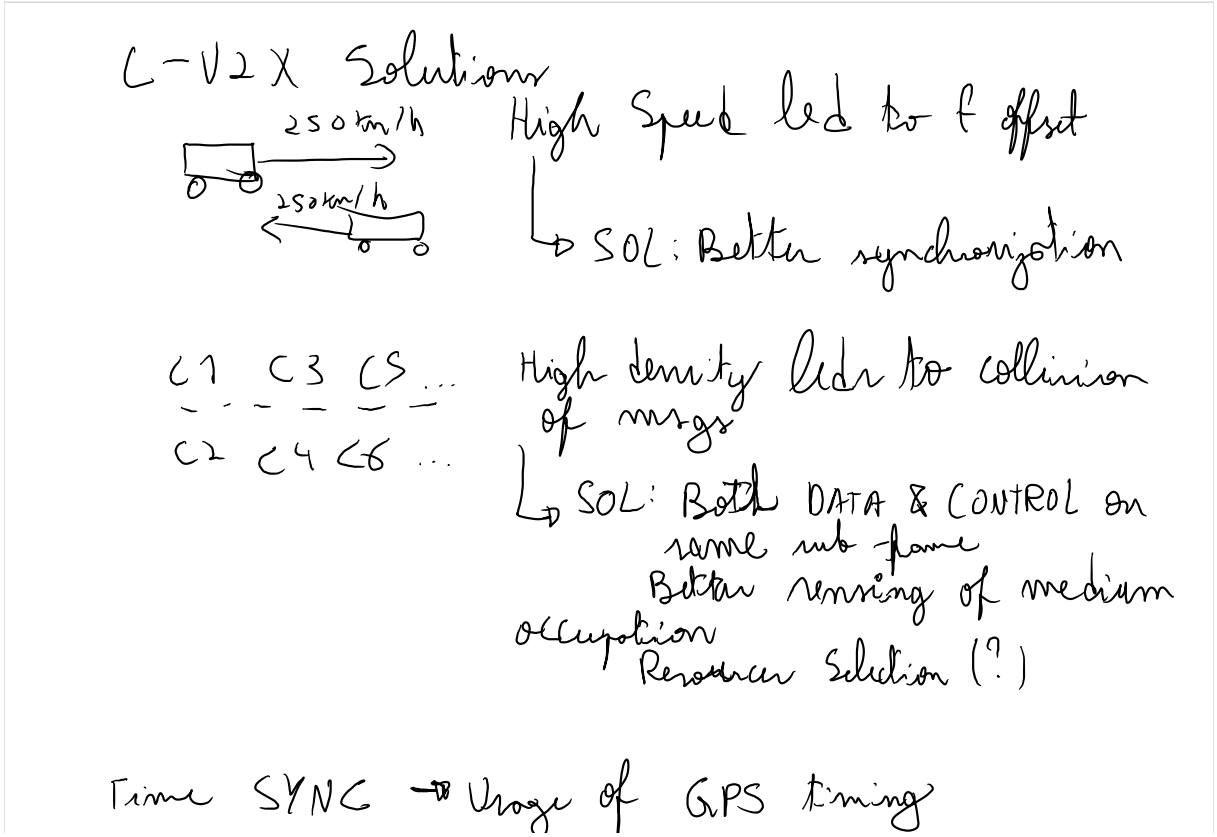
In coverage We use Uu & PC5, and out of coverge only PC5 become we don't have infrastructure.

ITS-G5 vs C-V2X
ITS-G5 opera em um espectro dedicado (5,9 GHz) para aplicações de Sistemas de Transporte Inteligentes (ITS), enquanto o C-V2X utiliza espectro celular licenciado.

ITS-G5 não requer uma infraestrutura de rede centralizada, enquanto o C-V2X utiliza a infraestrutura celular existente.

O C-V2X pode aproveitar as tecnologias celulares em evolução, como o 5G, garantindo compatibilidade e integração potencial com avanços futuros.

O ITS-G5 é amplamente implantado na Europa e no Japão, enquanto o C-V2X está ganhando suporte global e sendo considerado um possível padrão global.



Time SYNC → Usage of GPS timing

QoS

AD-HOC characteristic:

- Router are instable and not always available
- High bit error (lost to congestion/noise)
- Unpreditable (difficult to estimate RTT, BW and time-out)
- Longer the distance worst performance
- Packets compete for the channel

Why TCP bad in ad-hoc?

Misinterpretation of route failures: In ad-hoc nodes can move or links become unstable, route failures can occur. TCP thinks this packet loss is become of congestion and activates its congestion control mechanism reducing the sending bit rate Therefore TCP's congestion algorithm may reduce the sending rate unnecessarily leading to a reduce throughput and performance. .

Misinterpretation of wireless errors: Same as above but about the medium (packet collision, signal interference...).

Delay Spikes: Makes it harder to estimate RTT, time-out,.... With the TCP's fast retransmission being based on these calculations it can happen situation when a retransmission in done without the need.

It's inefficient due to the high loss of retransmission packets

TCP-Cubic:

Uses a function of time, which is related to the last congestion occurrence

QUIC:

Uses UDP and the handshake combines TLS and transport parameters. Exchange of information via streams which are ordered seq. of bytes. It has 2 levels of data flow control (stream and connection).

- RTT estimation through the time a packet is sent and its Ack.
- Packet loss is detected through lack of Ack
- Congestion controll in independent of RTT and applied upon detection of packet loss.

It was designed to minimize head-of-line blocking (delays, latency, reduced throughput) and void delays by retransmission timeouts in TCP. It achieves this by having independent streams within a connection.

TCP variants try to improve performance by estimating the available BW and/or exploiting buffering capabilities.

TCP-Vegas:

It estimates congestion based on the rate of change of RTT rather than packet loss. It uses m algorithm to adjust its sending rate. It seeks to maintain a stable and optimal RTT, reducing unnecessary pocket loss and improving throughput. It can detect congestion earlier without occurring packet loss.

TCP-BuS:

- Uses explicit route failure notifications to detect route failures
- Maker use of special messages much or localized query and reply to find a path.
- when a route failure occurs the intermediate nodes buffer the pending packets, avoiding timeouts and consequently reducing retransmissions, BUT requires a

special routing protocol became of the usage of intermediary nodes.

QoS in UDP:

IntServ (app specifies QoS and traffic)

→ It's hard to estimate available resources,
because of the shared medium and the dynamic BW (mobility of nodes)

→ Hard to make reservations because of the
high motility of nodes (changing position and altering the reservations made, etc...)

→ Resource reservation requires that the
route where the resources have been reservated 'stays still'.

DiffServ (App chooses a class of service)

→ the flows can come from every node (in a normal setup the border routers would be the 'entry'), so it's hard to do admission control.

→ Because the BW variates and routar change in hard to maintain assurances.

QoS Routing:

In essential for QoS. It allows for a source node to know the BW and QoS available to a destination node and it's path, BUT would require for QoS metrics to be present in routing

for AODV:

→ Add new fields to messages

→ Changes to the routing table

→ Only nodes that meet the QoS requirements com rebroadcast the RREQ

In the event of loosing QoS params, the node originates a message to all depending nodes. Thin list of QoS may be caused in the increase of a node workload.

QoS BATMAN

TQ = TQ (B to A) x TQ (A to B) = 80% x 100% = 80%

TQ = TQ (C to B) x TQ (B to C)

= 90% x 80% = 72%

QoS OLSR

Multipoint relay have the best QoS

QoS = available BW x PoNOS x Link Weight

MEC, 5G & 6G

MEC:

It aims to bring the cloud close to the edge and open it for 3rd party applications.
Can provide services to enhance apps, it allows for service/applications instantiation on demand at the best location possible and thus providing lower latency.

Mobile Edge Apps: These apps run in a virtualized environment, and normally they consume services although they can also provide functionalities or services to other apps. They have specific rules and requirements about routing,
necessary resources (RAM, CPU,...), QoS and so on (like a normal application). If supported these apps con be moved around, based on the type of application, mobility of the users using the app..., to another host.

Mobile Edge Platform: Provides an environment where apps can discover, consume, advertise and offer its services. It in a collection of funcionalities required to run a MEC application on a particular virtualization infrastructure and enable them. The MEC platform con also provide services itself. It provides persistant storage and DNS.

Orchestrator: It is responsible for maintaining an venal view of the MEC system such an available resources, available MEC services and topology. It is also responsible for the on-boarding of apps, checking its rules requirements and operator policies, and selecting the appropriate MEC host for the app instantiation, taking into account latency, available resources, available suriar, etc.... It controlls app instantiation, termination and relocation (if supported).

Future: MEC will enable the widespread of emerging technologies like Iot, 5G and IA. It will pride support for ultra-low latency and real-time data processing for applications at the network edge while also enduring who Syrenento and having faster response times and better performance. The integration of MEC with cloud computing will enable distributed and collaborative computing environments by providing a seamless connectivity across divine deview and networks.

Security

Modification Attacks: An intruder announces better routes to be inserted into the network.

→ Redirection by changing seq. number or hop count

→ DOS with the announced routes

Impersonation: Spoofs MAC addr and forms loops

Fabrication: Traffic generation to disturb the network

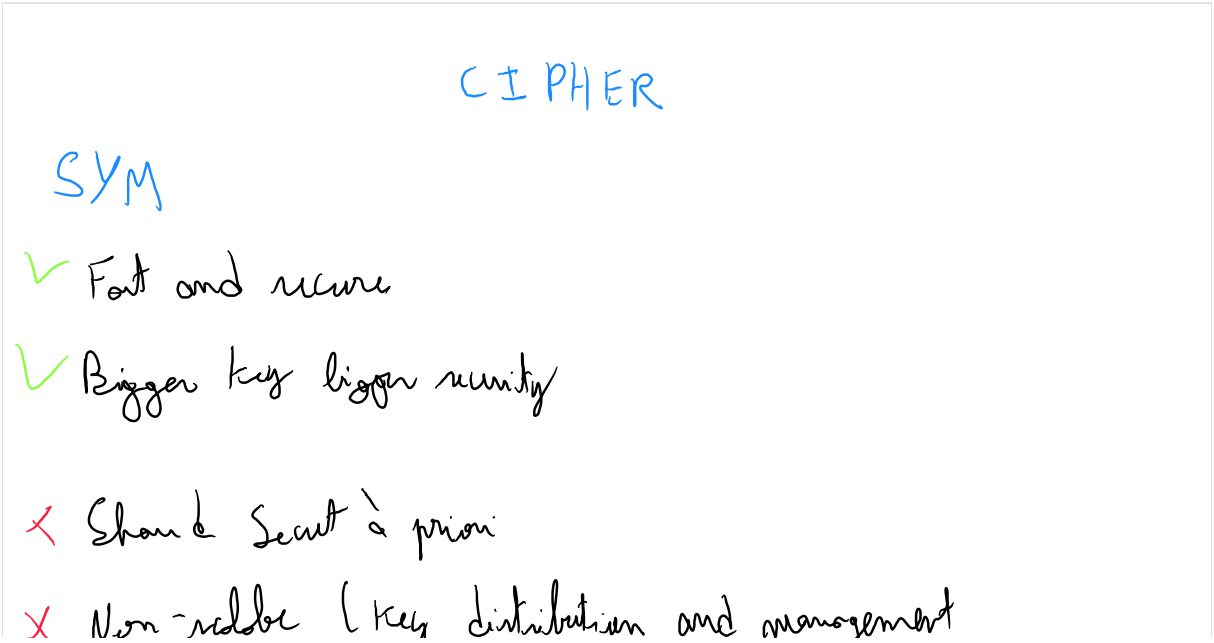
→ False route error messages

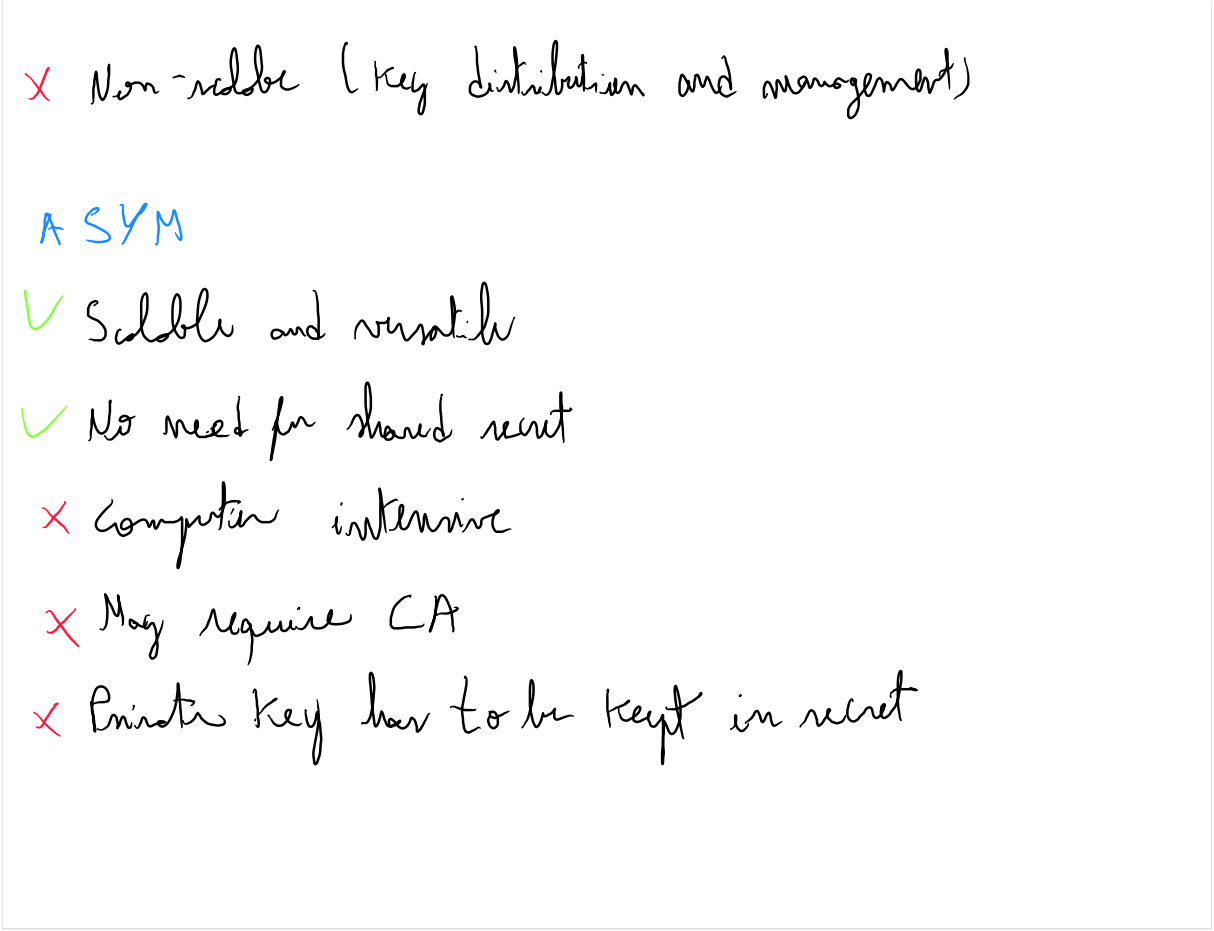
→ corrupt routing table I having fols routes in there)

→ Overflow of routing table

→ Replay Attack (old advertisements to a node)

→ Black Hole





Challenges:

- Vulnerable Node
- High mobility network
- No infrastructure for key distribution

SOPKM:

- Users issues certificates on personal connection
- Certificate Repository
- Expired Certificate Repo
- Each certificate has a validity period, who issued it and binds public key with a node.

Update Repo:
STEP O: Each user creates its own public/ private key pair
STEP 1: Issuing of certificates
STEP 2: Certificate Exchange
STEP 3: Node constructs its repo/graph
STEP 3.5: keeps constructing repo and requesting certificates.

In-depth STEP 2:

- Multicast graph to 1 hop neighbors
- Don't send cert, but hashed of VID
- Upon reception the node sends their hash values of Certs Non/updated
- the node keeps the values of the certs that it doesn't have.

Reputation based:
Routing and communication through high repo nodes. It protects the traffic from bad nodes and minimizer interaction with them. This reputation ambition needs to be a decentralized approach, so each node needs to monitor the behaviour of his neighbour nodes. Repo information his exchanged periodically among neighbours, plus network interfaces of nodes need to listen of other to or from communication.