## Title: Implement a congestion control algorithm for VANETs

**1. Problem Statement**

**2. Methodology**

**3. Implementation**

**4. Algorithm**

**5. Comparative Analysis**

1. **Problem Statement**

Congestion in VANET affects the performance, delays/drops in exchanging the emergency messages in particular when road accidents and high demand messages are delivered, causes unpredictable routing patterns and degrade the performance of the network. The traffic jams always lead to delays, fuel wastage and monetary loss therefore it is very important to have a centralized/de-centralized congestion control algorithm in place to avoid it. Hence Quality of service (QoS) is one of the most important things we need to handle. One of the most important components under QoS is Congestion Control. Congestion Control can be broadly taken care under three main sections:

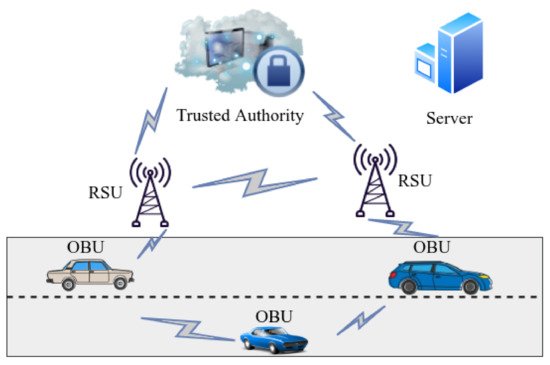
1. broadcast power control,
2. packet broadcast frequency control
3. packet period.
   1. **About**

VANETs help in improving roadway protection and traffic efficiency leading to reduced road dangers for drivers, customers and ramblers.

There are multiple components which are involved when dealing with VANETS. Some of them are:

* Vehicle-to-Vehicle (V2V) – Vehicle to Vehicle Communication which takes place between On-Board Units (OBUs)
* Vehicle-to-Infrastructure (V2I) – occur amongst Road Side Units (RSUs) and OBUs

The communication diagram is shown below:



1. **Methodology**

This section details various congestion control techniques and there

* 1. **Types of Congestion control**

Congestion control can be broadly characterized under two sections can be broadly characterized under two sections:

1. Open loop Congestion Policy: This deals with all the policies which are pre-emptive steps to avoid congestion i.e., before the congestion has actually taken place. Commonly used techniques under it are
   1. Acknowledgement Policy
   2. Discarding Policy
   3. Retransmission Policy
2. Closed loop Congestion Policy: This deals with steps for dealing with congestion once it has occurred. Some of the techniques under this are:
   1. Back pressure
   2. Implicit signaling
   3. Explicit Signaling
   4. **Proposed Solution**

One of the main reasons for congestion is the ***Lack of central coordination*** and **due to inefficiency of the algorithm** like frequent broadcast messaging. The basic idea proposed in this model is that a centralized server with required information can be processed with the algorithm, available static info (bandwidth of the region, local maps and distances between different locations & etc) and take an action to avoid the congestion Or alert the nodes on congestion and take an alternative path. And a Localized decision mechanism for local decision making like *computing power transmission* and etc.

* 1. **Reasons of Congestion**
* Traffic congestion due to accident.
* Rate of transmission of packets.
* The high density of nodes.
* High volume information sharing over the limited bandwidth.
* Topological dynamics of the VANet Or Rapid topology changes.

1. **Implementation**
   1. **Parameters to consider in the algorithm**
2. Node density

Given any point in time we can identify the number of nodes based on the beacon MAC header. Density can be determined based on the region and number of vehicles in the driven state.

*Solution:* increase /decreasing the bandwidth also can lead to better utilization of the network.

1. The oscillation between high power and low power transmissions.

Power transmission can be computed based on the algorithm the number of beacons that failed /beacons sent)\*100

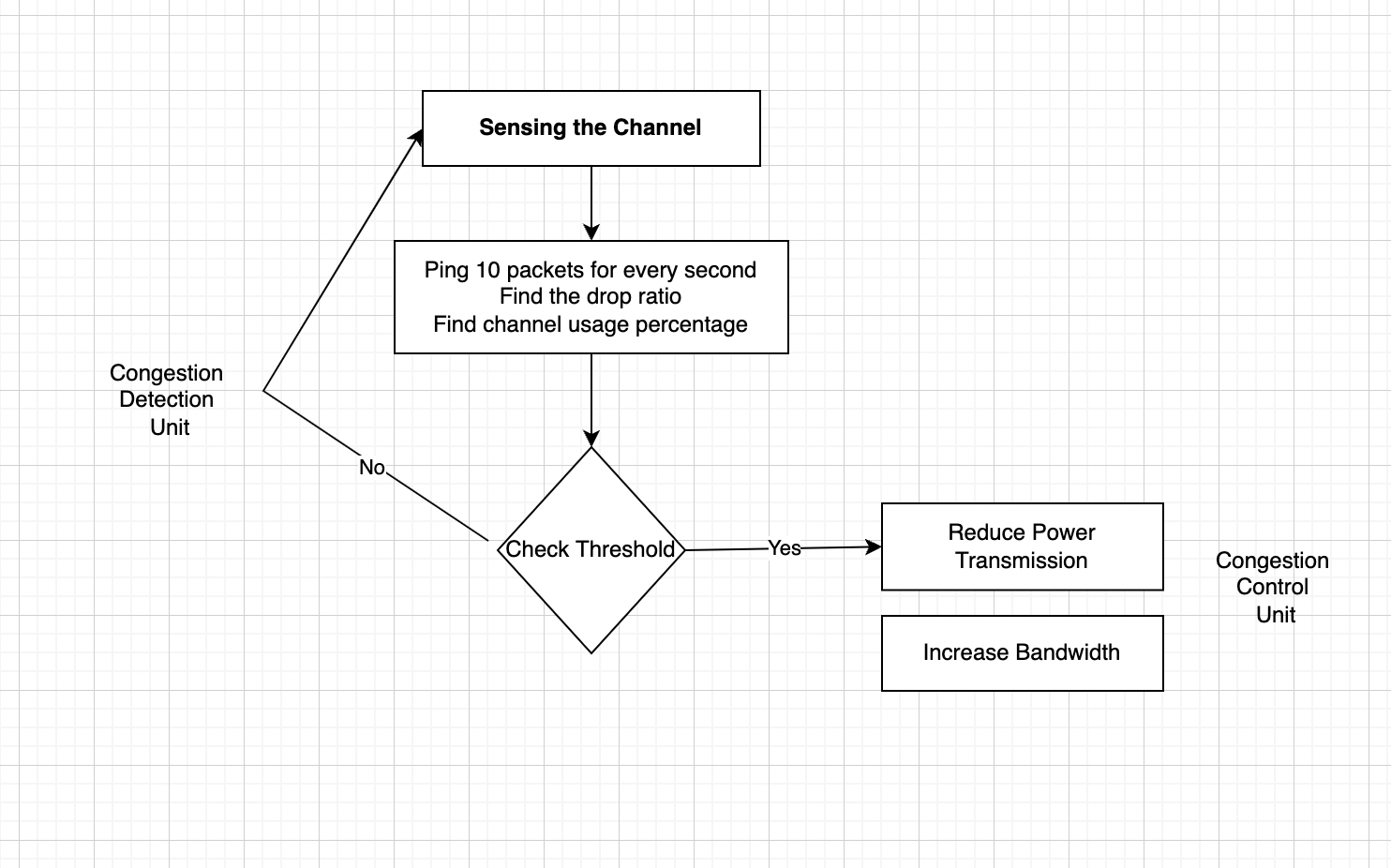
1. single high power transmission can lead to congestion
2. same power transmission at different nodes.
3. Power transmission can be different at different nodes and should be oscillated based on the info like distance to travel, speed of travel and etc.
4. Centralized and localized data congestion control

Using machine learning clustering. k-means clustering algorithm.

1. Localized decision making

For certain decisions like computing the power transmission based on existing congestion, channel busy usage, reducing the rate of beacons transmission or sending only emergency messages on the busy channel.

1. Overall congestion percentage
   1. **Flowchart**



1. **Algorithm**

The proposed procedure works with approaches with one working on server level while another working in a localized way at the RSU level. Below algorithm takes into account both Congestion detection and control.

* 1. **Centralized or Server Level Algorithm**

load\_static\_info() <- region info, bandwidth and history of information, fetch meta-heuristics

compute\_vehicle\_density():

status := NO\_DENSITY

// this can be get based

               set current\_vehicle\_count :=  get\_current\_vehicle\_count():on beacon and MAC header information and maintaining the count with TTL.

set max\_vehicle\_count: // this is based on region

if(current\_vehicle\_count >= max\_vehicle\_count):

status := DENSE

elif(current\_vehicle\_count >= max\_vehicle\_count/2):

              status := MODERATE

if(status := DENSE):

//Increase bandwidth and reduce power transmission

* 1. **Decentralized or Local level Algorithm**

compute\_power\_transmission():

// based on the past data and TTL

set cogention\_identified := get\_congestion\_identified()

// 10 packets sent to neighbouring vehicles to identify the failure rate every 1 second

set beacon\_failure\_rate := get\_beacon\_failure\_rate()

//compute decrease power transmission

// shouldn’t be <minimum power transmission to ensure safety message are to be delivered e

if(cogention\_identified && time <= TTL):

if(not current\_power\_trasmission <= min\_power\_transmission)

else

compute based on and beacon\_failure\_rate