Traffic Vehicles Communication Modeling Using Petri Net

Amal Bentaher, Yasser Hassan
Faculty of Science, Alexandria University
Alexandria, Egypt

Abstract— The number of traffic accident is greatly increasing around the world with the increase in the number of vehicles. Vehicle-to-vehicle (V2V) communications is an important technology that can help in avoiding crashes and improving driving safety. Cooperative driving systems aim to transmit messages from one vehicle to another to avoid accidents. This paper presents a model using V2V communications as a discrete event system. The proposed model can, efficiently, control and exchange messages between cars using V2V communications based on Petri net (PN), and authentication of message broadcasts by verifying all received messages required ID of cars. The proposed algorithm can be able to automatically updates either traffic information stored in cars or/and global traffic conditions, and helps to transmit a vehicle's location.

Keywords-component; Petri ne;, Vehicle-to-vehicle communications; Traffic; Message.

I. INTRODUCTION

Road traffic injuries (RTIs) are a major cause of global mortality and morbidity, killing approximately 1.3 million people and injuring 20 to 50 million each year. Egypt show a road traffic fatality rate of 42 deaths per 100,000 population—one of the highest in the Eastern Mediterranean Region[1], with more than 12,000 deaths and 154,000 injuries in 2007 (World Health Organization, WHO)[2].

The major causes of car accidents are weather conditions, roadway status and mechanical failures, but the dominating factor is the inborn reactive timelimitation of the drivers[3]. Those accidents could be reduced by implementation of safety information networks which combines vehicle-to-vehicle (V2V) communication and vehicle-to-infrastructure (V2I) communication. Safety information networks have been developed in the information technology field and are being implemented to solve some problems[4].

Cooperative vehicular communication system is one of the Intelligent Traffic System (ITS) technologies used to accomplish high levels of safety measurement and efficiency through the continuous exchange of information between vehicles (V2V communications) and between vehicles and infrastructure nodes (V2I communications)[5].

Petri net is a graphical tool used to represent the flow of activities in complex systems. The Petri net is particularly suited to represent, in a natural way, logical interactions among parts or activities in a system than other popular techniques of graphical system representation[6].

The proposal of this paper is to present a technique that can be able to control, efficiently, the exchange of messages using V2V communications with Petri Net (PN). This paper is organized as follows: First, we present basic concepts and definitions about Petri nets. Then,we explain the mechanisms used in theproposed system. Finally, we showthe simulation and results of our proposed model.

II. PETRI NET BASICS

Petri net is one of the several mathematical modeling languages which is used for the description of discrete distributed systems[7]. A Petri net is a directed graph in the most basic form that has three types of components: transitions, places, and directed arcs is shown at Fig.1. Transitions which signified by bars are discrete events that may occur, places which signified by circles are conditions, while directed arcs which signified by arrows describe the relationship between the transitions, and the smaller solid circle inside the hollow circle is called a token, which represents the conditions.

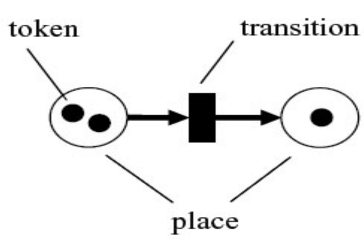


Figure 1.A simple graphical Petri net.

Petri nets are a promising tool to be used in different areas for the studying concurrent, asynchronous, distributed, parallel, nondeterministic, and/or stochastic systems[8].

A Petri net is marked by a five-tuple[9], N= (*P, T, I, O,* M_0) where:

- $P = \{p_1, p_2, ..., p_n\}$ is a finite set of places, n ≥ 0,
- $T = \{t_1, t_2, ..., t_m\}$ is a finite set of transitions, m ≥ 0that P ∩ T = ∅
- I is the transition input relation which represented by means ofarcs directed from places to transitions.
- O is the transition output relation which represented by means of arcs directed from transitions to places.
- M = $\{m_0, m_1, ...\}$ is the marking.

The initial marking is denoted by M_0, and the Petri net is marked if a marking function can be assigned to it. The state of a Petri net is defined by its marking. The dynamic part of a Petri net involves the change in markings over time[10].

III. COMMUNICATION MODEL USING PETRI NET

The intelligent vehicle system is important to be constructed by exchanging emergency-related information between any vehicles, such as urgency stop, traffic accident, and obstacles[11]. This model assuming a maximum communication range of 200 meters, the road has two directions and two lanes per direction, and a maximum speed limited to 55-80 km/hour for all types of vehicles. When two cars were connecting, they will exchange information, in a period of a time (three simulation seconds to communicate). The beginning architecture of the proposed system as shown inFig.2.The road has sets of moving vehicles. The goal is to exchange informationbetween these vehicles to enable each vehicle to control its traffic information. When the vehicles move along the road, they might enter the transmission range ofsome vehicles and exit that of others.In the proposed model, each vehicle stores records of information about itself and some vehicles that connected with it.

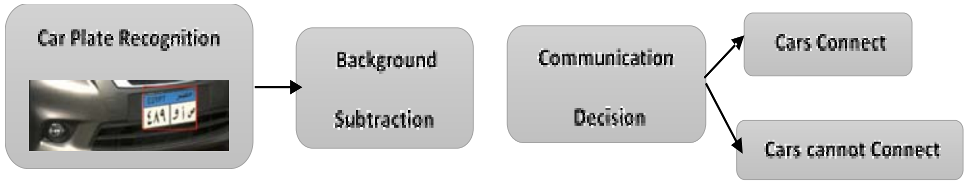


Figure 2.The beginning architecture of model communication.

Each record of a vehicle consists of the following fields:

- Identification (ID):vehicle identification number that is unique number given to each vehicle.
- Position: the location of a vehicle in a range of communication.
- Speed: the current speed of a vehicle that it wants to connect.
- Type: example (car, van, bus, truck).

Every time a message is received, the transmitted vehicles entry in the neighbor records is updated to avoid outdated information.

A. Detection of Cars Plate

Automatic Number Plate Recognition (ANPR) is a highly accurate system capable of capturing images for vehicles and identifying their plate number. ANPR systems consist of high-speed cameras with infra-red (IR) filters to find a vehicle plate number.

Our model recognizes the plate number of a neighbor vehicle by taking static snapshots captured by the vehicle's camera. Next steps will be as the following algorithm[12]:

- Edge detection: The Sobel edge detector is used to find the rectangle shape of the plate using dilation and erosion.
- 2D filter:
  2D median filter mask 5×5 is used to filter and smooth the eroded image.
- Segmentation:
  It is applied to separate and cut the characters and numbers of Egyptian plate with fixed size. Fig.3 shows steps of a typical LP system.
- Recognition:
  In this step, the database of characters and numbers were saved and correlated with characters and numbers of a vehicle to find LPR of this vehicle.

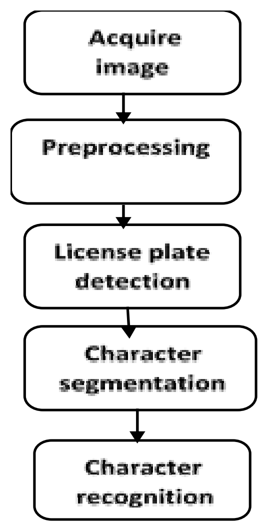


Figure 3.Steps of a typical LP system

B. The Request/ Reply Model

The request/reply model supposes two vehicles, (A) and (B), where vehicle (A) requests to connect with vehicle (B), (see Fig. 4). The following steps explains the communication between these two vehicles based on Petri nets:

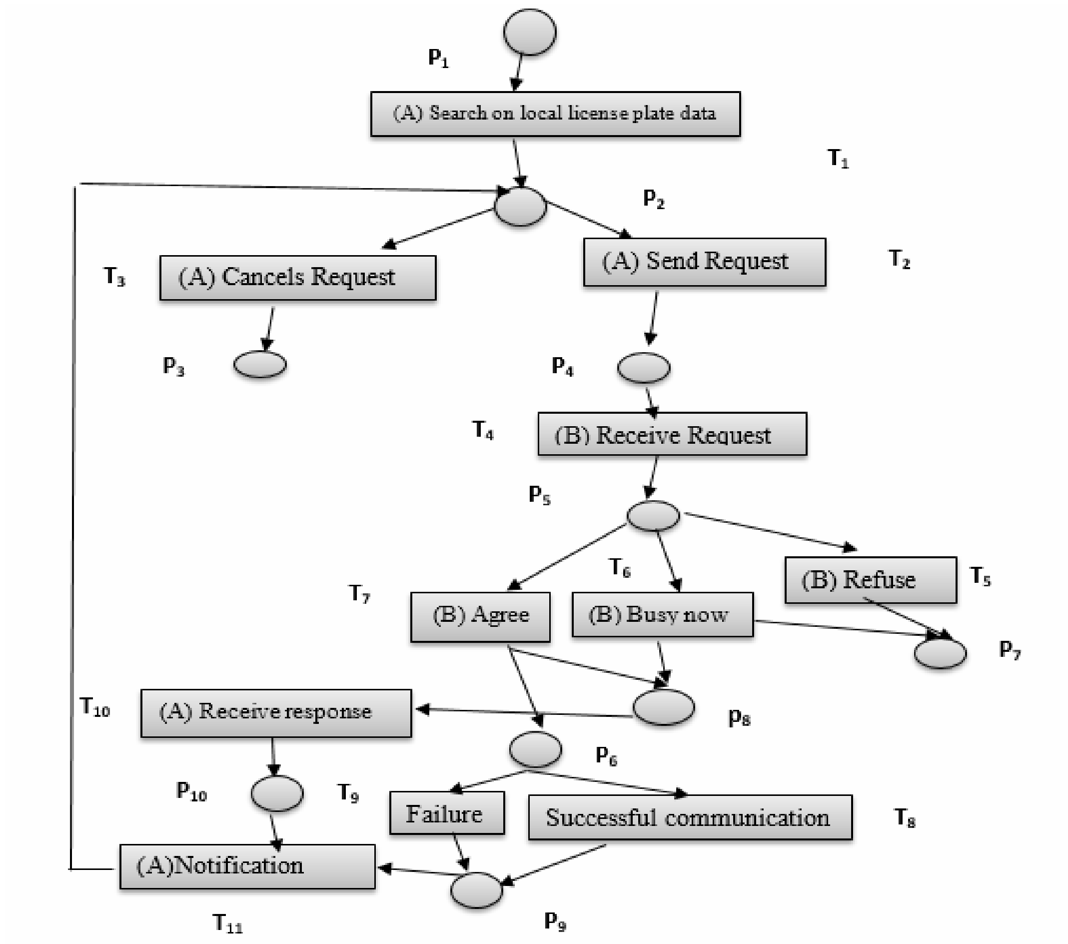


Figure 4.The model of V2V based on Petri net.

1. Vehicle (A) searches on local data (database inside the vehicle contains ID of cars connected with this vehicle before) about license plate numberof vehicle (B). If license plate numberof vehicle (B) is unknown or known in local database of vehicle (A) then the transition T_1 is fired.

2. If vehicle (A) sends connection request with vehicle (B) transition T_2 is fired, otherwise, vehicle (A) cancels the request, and transition T_3 is fired.

3. Vehicle (B) receives a request from vehicle (A) that asks for connection (transition T_4 is fired), then it has three cases:

   a. ***Refuse*:** vehicle (B) refuses the request for connection (transition T_5 is fired). This refusal can be either because it does not have the skills to connect or because the driver does not want to, so the request fails and the information is stored inthe local database of vehicle (B).

b. ***Busy now*:** vehicle (B) sends a message to vehicle (A) to refuse this offer this time but may connect another time (transition T_6 is fired). The connection failed and added information to local database of vehicle (B).

c. ***Agree*:** vehicle (B) accepts the connection with vehicle (A), so the information transmission starts immediately (transition T_7 is fired).

c.1Successful communication: vehicle (B) transmits information to vehicle (A) and this connection is successful (transition T_8 is fired). This successful communication requires both vehicles to keep an appropriate distance in a distinct time. If the distance, or time, exceeded the limits, the connection will fail.

c.2 Vehicle (B) failed to connect (transition T_9 is fired).

d. Vehicle (A) receives notification (transition T_{11} is fired) about connection that was succeeded or failed, and this information will be added to local database of vehicle (A).

e. Vehicle (A) receives response (transition T_{10} is fired) about connection that is agree or busy now.

IV. THE DETAIL OF COMMUNICATION PROCESS BETWEEN VEHICLES

This proposed connected model using Petri net is shown in Fig. 5(a):

- The number of tokens marked in place P_1 defines the license plate numberof the vehicle that can be connected.
- The communication between two vehicles was only available when they stayed within the communication range.
- The simulation result of proposed model after modeling communication is illustrated in stages in Figure 4.
- In Fig. 5(a), for P_1 and $T_{1,}$ vehicle (A) searches for the license plate numberof vehicle (B) in the local traffic license plate data.
- In Fig. 5(b), P_2 and $T_{3,}$ vehicle (A) cancels the connection with vehicle (B).
- In Fig. 5(c), at P_2 and $T_{2,}$ vehicle (A) requests the connection with vehicle (B).

- Fig. 5(d) has three cases:

1) P_5 and T_7,vehicle (B) agrees to connect vehicle (A).
2) P_5 and T_6, vehicle (B) rejects the connection for this time.

3) P_5 and T_5, vehicle (B) refuses the

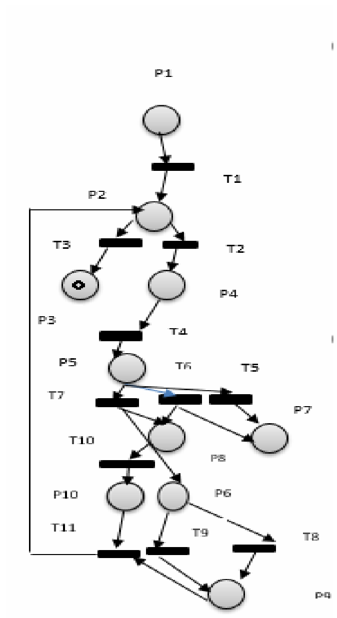
Place	Content	Transition	Content
P_1	Sets license plate number of vehicle B	**T_1**	Vehicle A searches on local data base license plate number data
P_2	license plate number of vehicle B is known	**T_2**	Vehicle A requests connect with vehicle B
P_3	Stops the connection	**T_3**	Vehicle A cancel connect with vehicle B
P_4	Saving plate number of vehicle B in local database of vehicle A	**T_4**	Vehicle B receives request connection with vehicle A
P_5	Vehicle A sends message connection to vehicle B	**T_5**	Vehicle B refuses connect with vehicle A
P_6	Vehicle B begins connection with vehicle A	**T_6**	Vehicle B refuses connect with vehicle A period time.
P_7	The refusal the connection saving in vehicle B in local database	**T_7**	Vehicle B accepts connect with vehicle A
P_8	Acceptation of offer	**T_8**	The connection is successful
P_9	The connection results are saving	**T_9**	The connection is failure
P_{10}	Vehicle A saves a message of response	**T_{10}**	Vehicle A receive response.
		T_{11}	Vehicle A receive notification

connection.

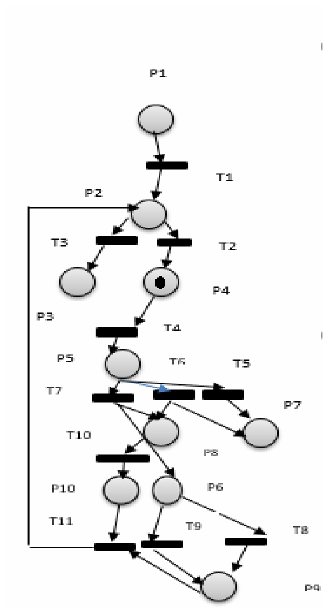
The following status about each role of places and transitions (Fig. 5) as shown asTable.1.

TABLE I. CONTENT OF PLACES AND TRANSITION OF COMMUNICATION BETWEEN VEHICLES.

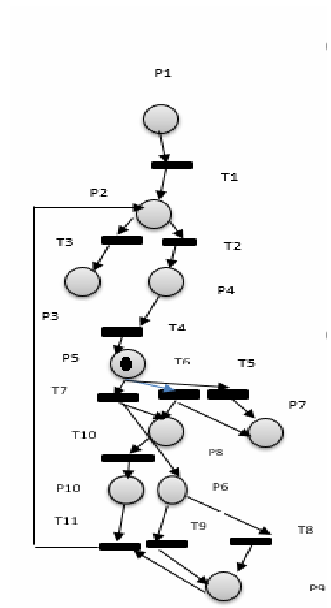
In Fig.5(e), at P_8 and T_{10}, vehicle (B) sends a message to vehicle (A) to accept the connection request (in case agree) or delay connection (in case busy now). In Fig. 5(f), at P_6,vehicle (B) connects with vehicle (A) so this connection in T_8 is successful, otherwise, in T_9 the connection failed. Finally, in Fig. 5(g) at P_9 and T_{11}, vehicle (B) sends a notification to vehicle (A) about connection.


P1
T1
P2
T3
T2
P4
P3
T4
P5
T6
T5
T7
P7
T10
P8
P10
P6
T11
T9
T8
P9

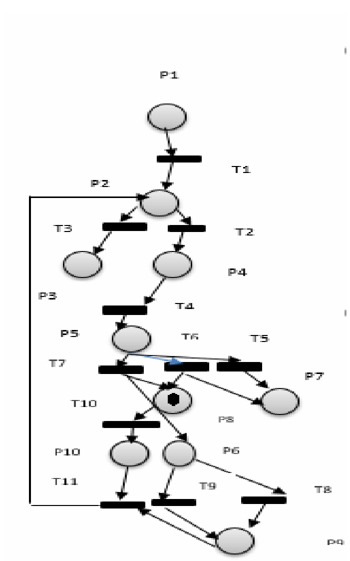
(b)

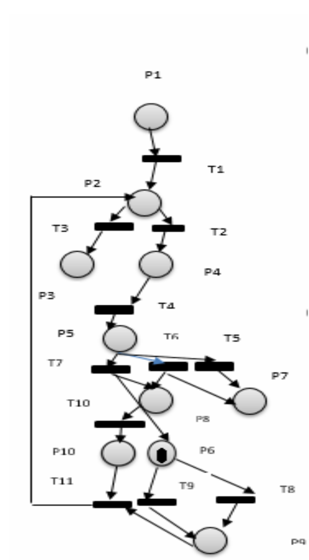

P1
T1
P2
T3
T2
P4
P3
T4
P5
T6
T5
T7
P7
T10
P8
P10
P6
T11
T9
T8
P9

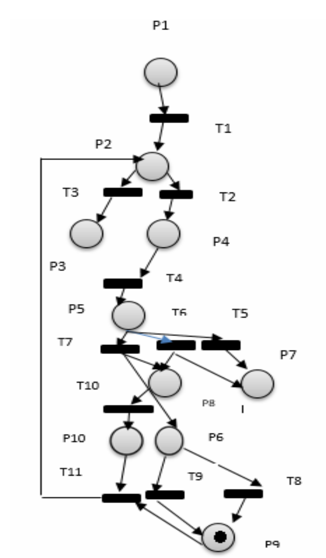
(c)


P1
T1
P2
T3
T2
P4
P3
T4
P5
T6
T5
T7
P7
T10
P8
P10
P6
T11
T9
T8
P9

(d)


P1
T1
P2
T3
T2
P4
P3
T4
P5
T6
T5
T7
P7
T10
P8
P10
P6
T11
T9
T8
P9


P1
T1
P2
T3
T2
P4
P3
T4
P5
T6
T5
T7
P7
T10
P8
P10
P6
T11
T9
T8
P9


P1
T1
P2
T3
T2
P4
P3
T4
P5
T6
T5
T7
P7
T10
P8
P10
P6
T11
T9
T8
P9

(e) (f) (g)

Figure 5: Petri net steps modeling of the communicating vehicles: (a) Initial marking (b) Cancel request (c) Send request (d) Receive request (e) Beginning communication (f) Successfulcommunication or failure (g) Send notification.

GPenSIM is a tool for modeling and simulation of discrete event dynamic systems (DEDS). It was developed by Reggie Davidrajuh in 2010[13], but it lacked graphical user interface (GUI) for creating the initial Petri net model. This model can be written in MATLAB language, allowing seamless integration with other toolboxes that also available on the MATLAB platform.

This section describes simulations with GPenSIM. The first step is to create the definition files for vehicle to vehicle communication, it has number of places, number of transition, number of arcs to connect between elements. All arcs are defined with weight equal to 1.

```
Function [PN_name, set_of_places, set_of_trans, set_of_arcs] = Vehicle to vehicle_pdf()
PN_name= 'Vehicle to vehicle Petri Net' ;
set_of_places=
{'p1','p2','p3','p4','p5','p6','p7','p8','p9','p10'};

set_of_trans={'t1','t2','t3','t4','t5','t6','t7','t8','t9','t10','t11'};

set_of_arcs={'p1', 't1',1,'t1',
'p2',1 ,'p2',
't2',1,'p2','t3',1,'t3',
'p3',1,'t2',p4',1,'p4','t4',1,'t4','p5',1,'p5','t5',1,'p5','t6',
1,'p5','t7',1,'t5','p8',1,'t6','p8',1,'t6','p7',1,'t7','p6',1,'t7','p8',1,'p6','t8',1,'p6','t9',
1,'p8','t10',1,'t10','p10',1,'p10','t11',1,'t8','p9',1,'t9','p9',1,'p9','t11',1, 't11', 'p2',1};
```

Second step, after creating petri net model, simulation can be done.

- pns = pnstruct(' Vehicle to vehicle_pdf');

The model has an initial marks on the places and the firing times of the transition to be assigned and defined by packet dynamic-info.initial-markings and packet dynamic-info.firing-times respectively.

- dyn.m_0 = {'p1',1};
- dyn.ft = {'t1',2,'t2',5,'t3',6,'t4',8,'t5',10,'t6',12,'t7',15,'t8',17,'t9',20,'t10',22,'t11',25};

These transitions, with firing times, pass this packet to the next function, gpensim. Finally, simulated by the function " gpensim" .

- Sim-Results = gpensim (png);

The simulation results are shown by applying the function of print state.

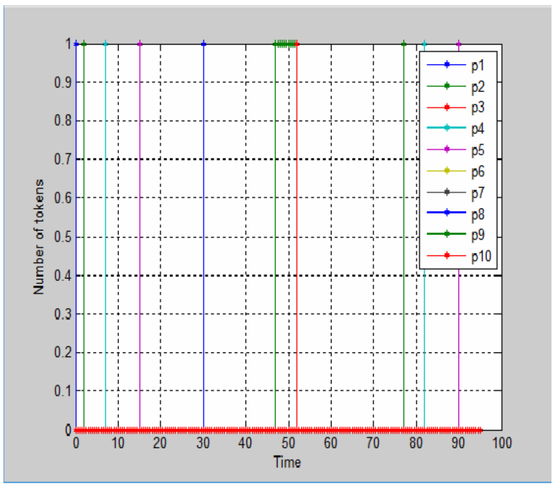
- Prnss (Sim_Results);

The results are drawn by using the function plot.

- plotp(Sim_Results,{'p1','p2','p3','p4','p5','p6','p7','p8','p9','p10'});

The simulation results are given below: State: 0

Initial Markings:



At time: 2 Enabled transitions are: t2, t3
At time: 2 Firing transitions are: t2

Time: 7
State: 2
Fired Transition: t2
Current State:

p1 p2 p3 p4 p5 p6 p7 p8 p9 p10

0 0 0 1 0 0 0 0 0 0

At time: 4 Enabled transitions are: t4
At time: 4 Firing transitions are: t4

[PRINT OUT DETAILS FOR STEPS 3 - 8 ARE DELETED FOR BREVITY …]

Time: 90
State: 9
Fired Transition: t4
Current State:

p1 p2p3 p4 p5 p6 p7 p8 p9 p10

0 0 0 0 1 0 0 0 0 0

Enabled transitions are: t10, t4

Firing transitions are: t10, t11, t4, t7, t9

We show also at Fig.6 the resulted table graph where thehorizontal axis represents the time (t) at the simulation andthe vertical axis represents the correctly tokens detectedsituations of places position.

Figure 6: The relation between the time (t) of simulation and the correctly tokens detected situations of places position.

CONCLUSION

Vehicle to vehicle communication has become an active area of research that offers important information about traffic accidents, road conditions, speed limits, and weather…, etc. Petri Net is a powerful tool for modeling and simulating of discrete event systems. GPenSIM is a Petri Net simulator which was initially developed as a tool of work in discrete simulation. In this paper, we proposed a technique based on controlled efficient exchange of messages between vehicles by applying Petri net for modeling and simulating the vehicle to vehicle communications as a new approach.

References

[1] P. Puvanachandra *et al.*, “Road Traffic Injuries and Data Systems in Egypt: Addressing the Challenges,” *Traffic Inj. Prev.*, vol. 13, no. sup1, pp. 44–56, 2012.

[2] WHO, “Global status report on road safety,” *World Heal. Organ.*, vol. 15, no. 4, pp. 286–286, 2009.

[3] C. Chen, N. Lü, L. Liu, Q. Q. Pei, and X. J. Li, “Critical safe distance design to improve driving safety based on vehicle-to-vehicle communications,” *J. Cent. South Univ.*, vol. 20, no. 11, pp. 3334–3344, 2013.

[4] H. Cho and B. Kim, “Study on Cooperative Intersection Collision Detection System Based on Vehicle-to-Vehicle Communication,” vol. 58, pp. 121–124, 2014.

[5] R. Bauza, J. Gozalvez, and J. Sanchez-Soriano, “Road traffic congestion detection through cooperative Vehicle-to-Vehicle communications,” *Proc. - Conf. Local Comput. Networks, LCN*, pp. 606–612, 2010.

[6] R. A. Ganiyu, E. O. Omidiora, S. O. Olabiyisi, and O. O. Okediran, “The Application of Petri Nets to the Modelling and Analysis of a Producer-Consumer System,” vol. 1, no. September, 2011.

[7] W. BRAUER and W. REISIG, “Carl adam Petri and ‘Petri nets,’” *Fundam. Concepts Comput. Sci.*, vol. 3, no. 5, pp. 129–139, 2009.

[8] T. Murata, “Petri Nets□: Properties , Analysis and Appl k a t ions,” vol. 77, no. 4, pp. 541–580, 1989.

[9] J. Wang, “Petri Nets for Dynamic Event-Driven System Modeling,” *Transition*, no. 4, pp. 1–17, 2006.

[10] S. Pujari and S. Mukhopadhyay, “Petri Net: A Tool for Modeling and Analyze Multi-agent Oriented Systems,” *Int. J. Intell. Syst. Appl.*, vol. 4, no. 10, pp. 103–112, 2012.

[11] S. Yu and G. Cho, “A Selective Flooding Methods for Propagating Emergency Messages in Vehicle Safety Communication,” in *International Conference on Hybrid Information Technology*, 2006, pp. 556–561.

[12] M. A. Massoud, M. Sabee, M. Gergais, and R. Bakhit, “Automated new license plate recognition in Egypt,” *Alexandria Eng. J.*, vol. 52, no. 3, pp. 319–326, 2013.

[13] R. Davidrajuh, “:,” in *Petri Nets Application*, no. Pawel Pawlewski (Ed.), 2010, pp. 527–540.