Towards Modeling Road Tunnels: A Petri Nets based Approach

Reggie Davidrajuh, Joel Fabian Joseph

Electrical Engineering and Computer Science
University of Stavanger, Stavanger, Norway.
Email: Reggie.Davidrajuh@uis.no; joeljoseph49@gmail.com

Abstract **- This paper aims to develop a mathematical model using Petri nets to simulate the traffic flow inside the road tunnel. First, a new modular Petri net theory is used; the modeling approach shown in this paper for modeling a specific road tunnel can also be applied for the other tunnels (reusable modules). Second, since the traffic inside road tunnels is intrinsically continuous, the roads are segmented to make the processes discrete; the segments become the basic building block of road tunnel models. Third, the modular Petri net theory and the modular Petri net model are implemented using the software GPenSIM to model and analyze real-life road tunnels.**

Keywords- Road Tunnels; Modular Models; Petri Nets; Modular Petri Nets; GPenSIM.

I. INTRODUCTION

Several factors have to be considered during the construction of a road tunnel. One crucial factor is the security inside the tunnels since accident inside the tunnel is more severe than accidents outside. Another critical factor is the driver's well-being, as the environment inside the tunnels is much darker and cramped. So it is crucial to improve the well-being of drivers, prevent accidents, and reduce the traffic queue. Using a simulation tool to simulate the traffic flow inside the tunnel would help determine how to construct a tunnel to solve issues to prevent accidents and eliminate the possibility of a traffic queue [1]

This paper starts with a compact literature review (section II) on modeling, simulation, and analysis of road tunnels. This paper uses modular Petri nets for modeling road tunnels. Hence, in section III, the focus is given to the newest modular Petri net theory, proposed by the author of this paper [2] [3]. Section IV uses a modular approach for modeling road tunnels, starting with dissecting a single lane into segments. These segments become the basic building block of a road tunnel model. Finally, section V (discussion) summarizes the approach presented in this paper for developing models of other more-complex road tunnels.

II. SHORT LITERATURE REVIEW ON ROAD TUNNELS

The literature study reveals some mathematical models for traffic flow inside road tunnels. Ref. [4] created a queuing model; this queuing model is based on traffic counts and models traffic flow behavior as a function of relevant determinants. Ref. [5] created a crash-prediction model using road tunnel data from 2006-2009 of a single tube tunnel with unidirectional traffic.

The literature study also reveals works creating Petri nets- based models. For example, [6] proposed an extension for triangular Batch Petri net, which is a hybrid Petri net, for better applicability for urban and road traffic. Another work, [7] applied a fluid stochastic Petri net, which again is also a hybrid Petri net. This work modeled a car safety controller in a road tunnel to have vehicles communicate to maintain a certain distance between each other etc.

III. PETRI NETS

This section presents the formal definitions of Petri nets.

A. P/T (Place-Transition) Petri Nets

Petri net formalism consists of many classes of Petri nets. The simplest and original one is the P/T (Place-Transition) Petri nets. Fig.1 presents the definition for P/T Petri nets.

The P/T Petri Net is defined as a four-tuple:

$$PTN = (P, T, A, M_0),$$

where:

P is a finite set of places, $P = \{p_1, p_2, \dots, p_{n_p}\}$.
T is a finite set of transitions, $T = \{t_1, t_2, \dots, t_{n_t}\}$.
$P \cap T = \emptyset$.
A is the set of arcs (from places to transitions and from transitions to places).
$A \subseteq (P \times T) \cup (T \times P)$.
The default arc weight W of a_{ij} ($a_{ij} \in A$, an arc going from p_i to t_j or from t_i to p_j) is one, unless noted otherwise.
M is the row vector of markings (tokens) on the set of places.
$M = \left[M(p_1), M(p_2), \dots, M(p_{n_p})\right] \in N^{n_p}$,
M_0 is the initial marking.
Due to the markings, a $PTN = (P, T, A, M)$, is also called **a marked P/T Petri Net**.

Figure 1. P/T Petri nets.

B. Modular Petri Nets

Refs. [2] and [3] propose the newest modular Petri Nets. The modular Petri nets proposed in these two works are implemented in the General-purpose Petri Net Simulator (GPenSIM) [8] [9]. Hence, real-life systems can be modeled, simulated, and analyzed with GPenSIM as modular Petri Net models.

The next subsection presents a summary of the newest Modular Petri Nets.

C. Modular Petri Nets = Petri Modules + Inter-Modular Connectors

A modular Petri net consists of one more Petri modules and zero or more Inter-Modular connectors (IMC). Fig.2 presents the formal definition for the modular Petri nets.

A Modular Petri Net is defined as a two-tuple:

$$MPN = (M, C)$$

where:

$M = \sum_{i=1}^{m} \Phi_i$ (one or more Petri Modules)

$C = \sum_{j=0}^{n} \Psi_j$ (zero or more Inter-Modular Connectors)

Figure 2. Modular Petri nets.

Fig.3 and 4 define Inter-Modular connector (IMC) and Petri module, respectively.

Formal Definition of Inter-modular Connector An Inter-modular Connector (IMC) is defined as a four-tuple:

$$\Psi = (P_\Psi, T_\Psi, \Lambda_\Psi, M_{\Psi 0})$$

where,

- $P_\Psi \subseteq P$: P_Ψ is the set of places in the IMC (known as the IM-places). $\forall p \in P_\Psi$,
  - $\bullet p \in (T_{OP} \cup T_\Psi \cup \emptyset)$. (input transitions of IM places are either the output ports of modules, IM transitions of this IMC, or none)
  - $p \bullet \in (T_{IP} \cup T_\Psi \cup \emptyset)$. (output transitions of IM places are either the input ports of modules, IM transitions of this IMC, or none)
    This means, IM places cannot have direct connections with local transitions, or IM transitions of other IMCs.
- $\forall p \in P_\Psi, \quad \forall i \quad p \notin P_{\Phi_i}$ (an IM-place cannot be a local place of any Petri module).
- $T_\Psi \subseteq T$: T_Ψ is the transitions of the IMC (known as the IM-transitions). $\forall t \in T_\Psi$,
  - $\bullet t \in (P_\Psi \cup \emptyset)$. (input places of IM-transitions are either the IM-places of this IMC, or none (cold start))
  - $t \bullet \in (P_\Psi \cup \emptyset)$. (output places of IM-transitions either the IM-places of this IMC, or none (sink))
- $\forall t \in T_\Psi, \quad \forall i \quad t \notin T_{\Phi_i}$ (an IM-transition cannot be a transition of any Petri module).
- $\Lambda_\Psi \subseteq (P_\Psi \times (T_\Psi \cup T_{IP})) \cup ((T_\Psi \cup T_{OP}) \times P_\Psi)$: where $a_{ij} \in \Lambda_\Psi$ is known as the IMC arcs.
- $M_{\Psi 0} = [M(p_\Psi)]$ is the initial markings in the IM-places.

Figure 3. Inter-Modular Connector.

Formal Definition of Petri Module A Petri Module is defined as a six-tuple:

$$\Phi = (P_{L\Phi}, T_{IP\Phi}, T_{L\Phi}, T_{OP\Phi}, \Lambda_\Phi, M_{\Phi 0}),$$

where,

- $T_{IP\Phi} \subseteq T$: $T_{IP\Phi}$ is known as the input ports of the module.
- $T_{L\Phi} \subseteq T$: $T_{L\Phi}$ is known as the local transitions of the module.
- $T_{OP\Phi} \subseteq T$: $T_{OP\Phi}$ is known as the output ports of the module.
- $T_{IP\Phi}$, $T_{L\Phi}$, and $T_{OP\Phi}$, are all mutually exclusive: $T_{IP\Phi} \cap T_{L\Phi} = T_{L\Phi} \cap T_{OP\Phi} = T_{OP\Phi} \cap T_{IP\Phi} = \emptyset$.
- $T_\Phi = T_{IP\Phi} \cup T_{L\Phi} \cup T_{OP\Phi}$ (the transitions of the module).
- $P_{L\Phi} \subseteq P$ is known as the set of local places of the module. Since a module has only local places, $P_\Phi \equiv P_{L\Phi}$.
- $\forall p \in P_{L\Phi}$,
  - $\bullet p \in (T_\Phi \cup \emptyset)$. Input transitions of local places are either the transitions of the module or none (none means a local place can be a source, without any input transition).
  - $p \bullet \in (T_\Phi \cup \emptyset)$. Output transitions of local places are either the transitions of the module or none (none means a local place can be a sink, without any output transition).
    This means, local places cannot have direct connections with external transitions.
- $\forall t \in T_{L\Phi}$,
  - $\bullet t \in (P_{L\Phi} \cup \emptyset)$. Input places of local transitions are either the local places or none (none here means that a local transition can be a cold start (a source), without any input places).
  - $t \bullet \in (P_{L\Phi} \cup \emptyset)$. Output places of local transitions either the local places or none (none means the local transition is a sink, without any output places).
- $\forall t \in T_{IP\Phi}$
  - $\bullet t \in (P_{L\Phi} \cup P_{IM} \cup \emptyset)$. (input places of input ports can be local places or places in inter-modular connectors or can be even an empty set)
  - $t \bullet \in (P_{L\Phi} \cup \emptyset)$. (output places of input ports can only be local places, or empty set)
- $\forall t \in T_{OP\Phi}$
  - $\bullet t \in (P_{L\Phi} \cup \emptyset)$. (input places of output ports can be local places or an empty set)
  - $t \bullet \in (P_{L\Phi} \cup P_{IM} \cup \emptyset)$. (output places of output ports can be local places or places in inter-modular connectors or empty set.
- $\Lambda_\Phi \subseteq (P_L \times T_\Phi) \cup (T_\Phi \times P_L)$: where $a_{ij} \in \Lambda_\Phi$ is known as the internal arcs of the module.
- $M_{\Phi 0} = [M(p_L)]$ is the initial markings in the local places.

Figure 4. Petri Module.

IV. BASIC ASSUMPTIONS

Several assumptions had to be defined for simplifying the model development [1]:

- **Blind vehicle**: The first assumption is that the driver of a vehicle in the tunnel cannot see other vehicles, except the one immediately in front of it or behind it. The "blind vehicle" assumption is necessary so that the movement of a vehicle can be realized with a transition.
- **Constant Speed**: In a real-life situation, the vehicle speed varies over time and has a speed approximate to the speed limit in the tunnel. However, for model simplification, it is assumed that the vehicles have a constant speed, and the speed will match the speed limit unless something else is specified.
- **Constant Acceleration**: Inside the tunnel, the speed limit might change. When a vehicle starts changing the speed to match the speed limit, it must accelerate. Of course, the vehicle's acceleration is not the same for all vehicles, but again, for simplicity, we assume that the vehicles accelerate with a constant acceleration.
- **Trigger upon discovery**: This assumption is about the immediate reaction. For example, when a vehicle sees a speed limit sign, it may start changing the speed before passing it or after. In this assumption, however, we assume that a vehicle will react at the spot of the speed limit sign.
- **Global event**: This assumption dictates the existence of an overall (global) event that monitors all the activities inside the tunnel. For example, there is a monitoring system with all the necessary sensors that can understand the specific status of the individual vehicles. Also, from these data, the system can estimate (or forecast) the overall status of the tunnel.

V. MODEL OF A ROAD TUNNEL

Traffic in a road tunnel is fundamentally a continuous process. Hence, we need to discretize the continuous process so that Petri nets can be used as the model. This paper suggests taking a single lane inside a tunnel and dissect it into a number of zones of equal length. These zones are named as **segments**. For example, a lane of one kilometer could be dissected into ten segments each of 100 meters long. For example, fig.7 shows three consecutive segments, segment-*i* to segment-*i + 2*.

A. Segment

After segmentizing a lane into segments, each segment can be modeled as a Petri module. By this segmentizing, a single lane becomes a series of connected segments (Petri modules).

Characteristics of segments:

- Segments are of equal length. And they are fixed along the length of the lane.
- A number of vehicles are kept within a segment. When a vehicle has travelled through a segment, it will be passed to the next segment. For example, in fig.8, vehicle C and F are passing from one segment into another segment.
- All vehicles within a segment posses the same speed limit, which can not be exceeded by any vehicle ("all drivers are good drivers").
- Within a segment, the drivers are semi-blind as they only monitor the vehicle before them.

B. Multi-Lane Road

As described before, a lane consists of a number of equal lengthed segments. If a one-way unidirectional road possesses multiple lanes, it is easy to model it with multiple series of segments. For example, fig.5 shows a double-lane one-way road, and its segmentized version is shown in fig.9.

Characteristics of multi-lane one-way roads:

- A vehicle can move from one lane into another only at the end of a (sending) segment. And this vehicle will be placed in the beginning of the receiving segment. For example, in fig.9, vehicle X is moving from lane-A into lane-B, from segment-*i* to segment-*i+1*. Whereas, vehicle Y is moving from lane-B into lane-A, from segment-*i+1* to segment-*i+2*.

Figure 5. Double-lane one-way road tunnel.

C. Petri Module representing a Segment

Fig.6 shows a Petri module that represents a segment. In this Petri module:

- tAi_input: This transition functions as the input port of the module. tAi input receives vehicles from previous segment in the same lane, and crossing vehicles from other two neighboring lanes.

- pAi: This internal place keeps the vehicles within the segment.
- tAi: this internal transition is the main element of a segment. This transition updates the travelled length and speed of all the vehicles within the segment.
- tAi_output: This transition functions as the output port of the module. tAi_output sends vehicles to the following segment in the same lane, and crossing vehicles to the other two neighboring lanes.

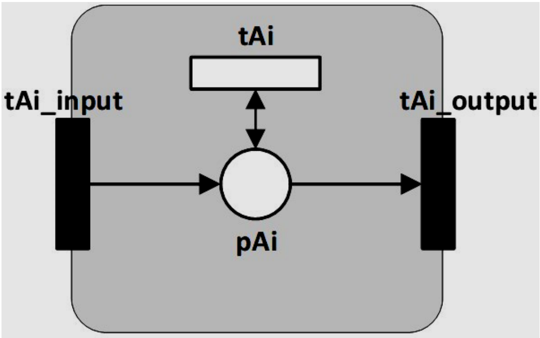

Figure 6. Petri module representing a segment.

D. Modeling One-way Multi-lane Road Tunnel

Fig.10 shows how a model of a double-lane one-way road tunnel (shown in fig.5 and 8) can be developed using Petri modules for segments as the building blocks. Fig.10 shows only two segments, A*i* in lane-A and B*i* in lane-B. Place pAh_Ai is one of the IMC that connects the preceding segment on the same lane A*h* with A*i*. Similarly, pAi_Aj connects A*i* with succeeding A*j*. pAh_Ai takes vehicles on lane-A from A*h* and hands them over to A*j*. However, pAh_Ai also accepts "crossing" vehicles, as pAh_Ai accepts crossing vehicles from lane-B from B*h* and hands them over to A*i*. The other IMC places pBh_Bi, pAi_Aj, and pBi_Bj function similar to pAh_Ai.

VI. DISCUSSION AND CONCLUSIONS

The idea presented in fig.10 for a double-lane one-way road tunnel can be extended for developing a model Modeling for a multi-lane bidirectional road tunnel. This paper does not provide any implementation details as it is still an ongoing research project. However, a case study on a simple one-lane unidirectional road tunnel is available that includes simulation and analysis of the results [1]. The interested reader is encouraged to study this document for GPenSIM coding.

The model becomes complex if the road tunnel possesses some traffic signals inside. In this case, the Petri module for traffic signal control has to be embedded into the model. Also, for real-life traffic in a road tunnel, the assumptions (or most of them) stated in section-IV may be seen as over-simplistic. However, we have to sacrifice some of the details in the real-life system to obtain a workable model.

This paper uses the newest modular Petri net theory, devised by the author of this paper [2] [3]. The modular Petri net model possesses all the necessary characteristics of modular models, such as extensible, comprehensible, and robust. Due to space and time limitations, this paper does not show the other properties of modular model development, such as independent development of the modules & testing and the analysis of the modules independently.

REFERENCES

[1] J. F. Joseph, "Developing Models of Road Tunnels with Petri Nets," Master's thesis, University of Stavanger, Norway, 2020.
[2] R. Davidrajuh, "A new modular Petri net for modeling large discrete-event systems: A proposal based on the literature study," Computers, vol. 8, no. 4, p. 83, 2019.
[3] R. Davidrajuh, "Design of a new modular petri nets," in Petri Nets for Modeling of Large Discrete Systems. Springer, 2021, pp. 95–105.
[4] N. Vandaele, T. Van Woensel, and A. Verbruggen, "A queueing based traffic flow model," Transportation Research Part D: Transport and Environment, vol. 5, no. 2, pp. 121–135, 2000.
[5] C. Caliendo, M. L. De Guglielmo, and M. Guida, "A crash-prediction model for road tunnels," Accident Analysis & Prevention, vol. 55, pp. 107–115, 2013.
[6] Y. Riouali, L. Benhlima, and S. Bah, "Petri net extension for traffic road modelling," in 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA). IEEE, 2016, pp. 1–6.
[7] A. Bobbio, M. Gribaudo, and A. Horvath, "Modelling a car safety controller in road tunnels using hybrid petri nets," in 2006 IEEE Intelligent Transportation Systems Conference. IEEE, 2006, pp. 1436– 1441.
[8] R. Davidrajuh, Modeling Discrete-Event Systems with GPenSIM. Cham: Springer International Publishing, 2018.
[9] GPenSIM, "General-purpose Petri net simulator," http://www.davidrajuh.net/gpensim, Tech. Rep., 2019, Accessed on 20 July 2020.

Figure 7. Three consecutive segments.

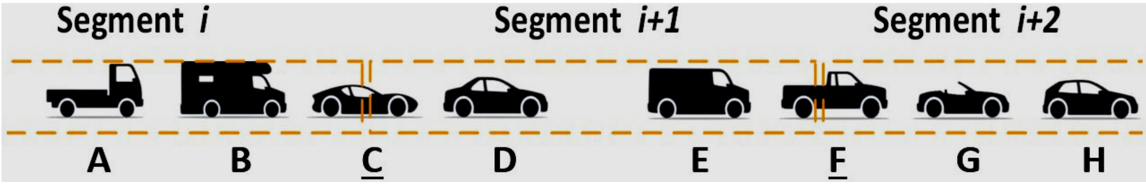

Figure 8. Passing between segments.

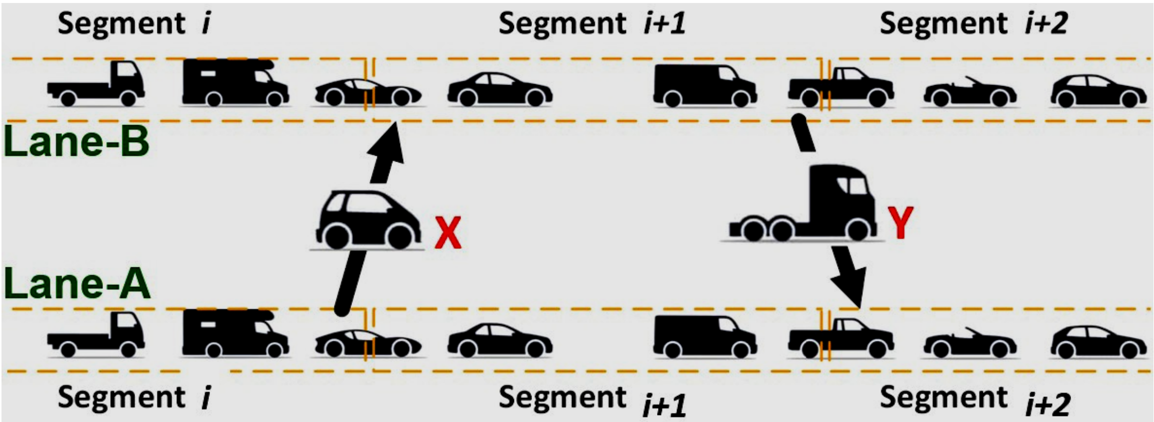

Figure 9. Segmentized Double-lane one-way road tunnel.

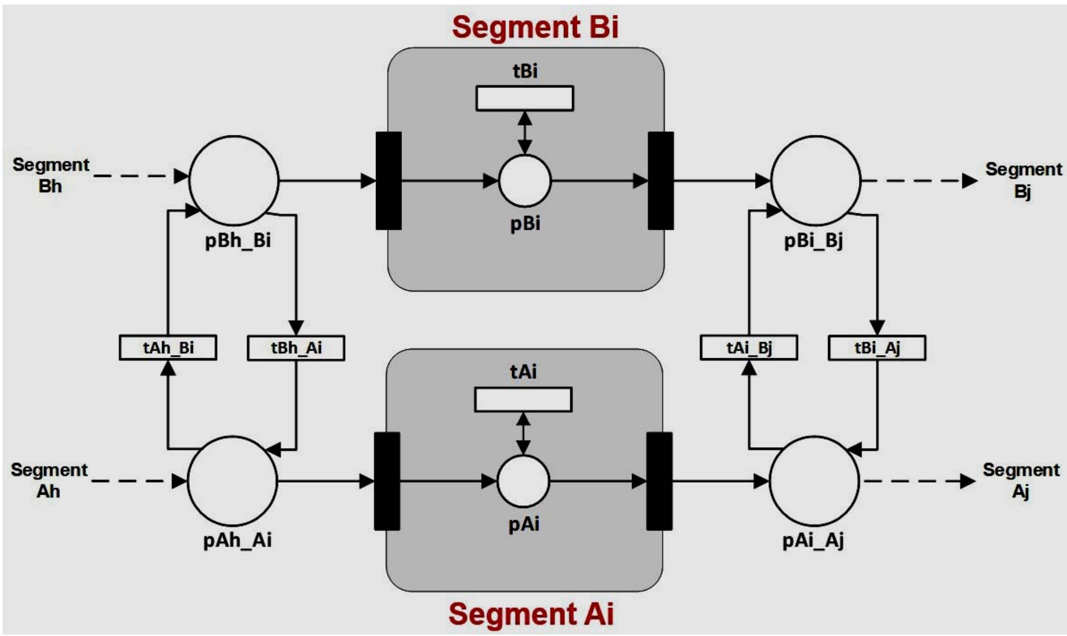

Figure 10. Segments in action tunnel.