

# Analysis of a road/tramway intersection by the ORIS tool

Laura Carnevali, Alessandro Fantechi, Gloria Gori, Enrico Vicario

Department of Information Engineering, University of Florence, Florence, Italy  
{laura.carnevali,alessandro.fantechi,gloria.gori,enrico.vicario}@unifi.it

**Abstract.** Intelligent Transportation Systems for urban mobility aim at the grand objective of reducing environmental impact and minimize urban congestion, also integrating different mobility modes and solutions. However, the different transportation modalities may end in a conflict due to physical constraints concerned with the urban structure itself: an example is the case of intersection between a public road and a tramway right-of-way, where traffic lights priority given to trams may trigger road congestion, while an intense car traffic can impact on trams' performance. These situations can be anticipated and avoided by accurately modeling and analyzing the possible congestion events. Typically, modeling tools provide simulation facilities, by which various scenarios can be played to understand the response of the intersection to different traffic loads. While supporting early verification of design choices, simulation encounters difficulties in the evaluation of rare events. Only modeling techniques and tools that support the analysis of the complete space of possible scenarios are able to find out such rare events. In this work, we present an analytical approach to model and evaluate a critical intersection for the Florence tramway, where frequent traffic blocks used to happen. Specifically, we exploit the ORIS tool to evaluate the probability of a traffic block, leveraging regenerative transient analysis based on the method of stochastic state classes to analyze a model of the intersection specified through Stochastic Time Petri Nets (STPNs). The reported experience shows that the frequency of tram rides impacts on the road congestion, and hence compensating measures (such as synchronizing the passage of trams in opposite directions on the road crossing) should be considered.

**Keywords:** Intelligent Transportation Systems, Transportation Modeling, Integrated traffic model, Stochastic state classes, Markov Regenerative Processes.

## 1 Introduction

By the year 2030, urban mobility will have changed due to sociodemographic evolution, urbanization, increase of the energy costs, implementation of environmental regulations, and further diffusion of Information and Communication Technology (ICT) applications. The demand for public and collective modes of transport will increase considerably. Part of the answer will come from the public

transport that will evolve as an integrated combination of buses, cars, metros, tramways and trains [13,1]. In general, right-of-way (ROW) is the defining characteristic of public transportation modes and we can list three ROW types:

1. *Exclusive*: Transit vehicles operate on fully separated and physically protected ROW. Tunnels, elevated structures, or at-grade tracks are such examples. This ROW type offers very high capacity, speed, reliability and safety. All heavy rail transit systems, like the Metrorail of the Washington Metropolitan Area Transit Authority, belong to this category.
2. *Semi-Exclusive*: Transit ways are longitudinally separated from other traffic, such as private vehicles and pedestrians. Light rail transit (LRT) systems, like the Florence tramway in Italy, are mostly built according to this ROW type.
3. *Fully-Shared*: Transit vehicles share ROW with other traffic, for examples buses, taxi and cars. This ROW type requires the least infrastructure investment, but operations are relatively unreliable due to roadway congestion.

Exclusive ROW needs major investment, thus often semi-exclusive or fully-shared modes are chosen. The drawback of this choice is that the different transportation modalities may end in a conflict due to physical constraints concerned with the urban structure itself. For example, this is the case of an intersection between a public road and a tramway right-of-way, where traffic lights priority given to trams may trigger road congestion, while an intense car traffic can impact on trams' performance. These situations can be anticipated and avoided by accurately modeling and analyzing the possible congestion events. Typically, modeling tools provide simulation facilities, by which various scenarios can be played to understand the response of the intersection to different traffic loads. Simulation techniques are used to support early verification of design choices, but can analyze a limited, yet high, number of different scenarios, and encounter difficulties in the evaluation of rare events. Only modeling techniques and tools that support the analysis of the complete space of possible scenarios are able to find out such rare events [7,4].

In this work, we present an analytical approach to model and evaluate a critical intersection for the Florence tramway, where frequent traffic blocks used to happen. This work has been funded by Fondazione Cassa di Risparmio di Firenze, with the kind help of GEST<sup>1</sup>, the company running the Florence tramway, in providing important data on which to base the study.

Figure 1 shows the route of line 1, which has been put in service in 2010 and links Santa Maria Novella central station to Scandicci (Florence suburbs). This line has overall good performance, with trams running regularly from the end of the line in Scandicci to almost the other end in the city center, but there is a consistent source of delay just a few meters short of the last scheduled stop, near Santa Maria Novella train station [10]. The root cause for these issues is the Diacceto-Alamanni intersection, where both via Iacopo da Diacceto, a street with dedicated tracks for tramways, and via Luigi Alamanni, a street for

---

<sup>1</sup> <https://www.ratpdev.com/en/references/italy-florence-tramway>

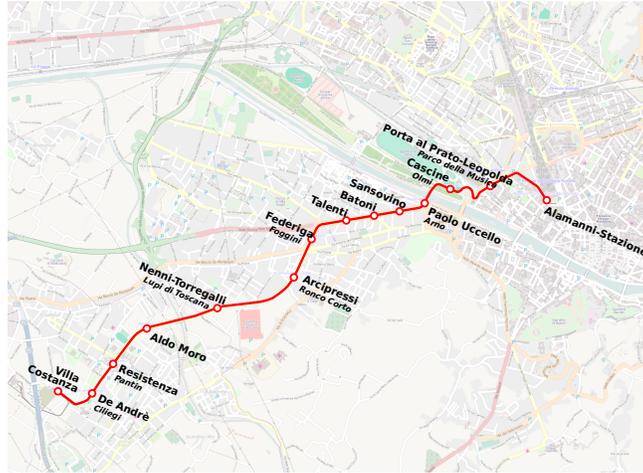


Fig. 1: Map of tram route from Villa Costanza (Scandicci) to Alamanni-Stazione (Santa Maria Novella station). The route is 7720 meters long with 14 tram stops.

private transport, head to Santa Maria Novella train station.<sup>2</sup> An aerial view of this intersection is shown in Figure 2. The darker stripe that crosses the tracks represents the (unidirectional) private traffic flow from Alamanni street that is the source of the analyzed conflict.



Fig. 2: Aerial view of the Diacceto-Alamanni intersection.

Taking this intersection as a case study, we exploit the ORIS tool to evaluate the probability of a traffic block, leveraging regenerative transient analysis based

<sup>2</sup> Actually, the construction works of the new tramway lines (due to be opened soon) have consistently changed the geometry of the Diacceto-Alamanni intersection, partially removing the car traffic. Anyway, the analysis presented in this work refers to a relevant scenario, typical of intersections between a public road and a tramway right-of-way, which will occur more frequently in Florence as new tramway lines will be built.

on the method of stochastic state classes to analyze a model of the intersection specified through Stochastic Time Petri Nets (STPNs). Note that ORIS supports the analysis of models with multiple concurrent temporal parameters associated with a general (i.e., non-exponential) distribution. In particular, the model of the Diaceto Alamanni intersection includes temporal parameters associated with a deterministic value (e.g., tram interleaving period), a uniform distribution (e.g., tram delay time), and an exponential distribution (e.g., private vehicles arrival rate). The reported experience shows that the frequency of tram rides impacts on the road congestion, and hence compensating measures (such as synchronizing the passage of trams in opposite directions on the road crossing) should be considered.

The remainder of the paper is structured as follows. Section 2 summarises related works. Section 3 provides a short introduction to STPNs, the method of stochastic state classes, and the ORIS tool. Section 4 presents the realized model and Section 5 the obtained results. Finally, Section 6 concludes the paper.

## 2 Related Works

Earliest research on integrated control for traffic management at network level can be traced back to the 1970s. The first railway timetables were planned based on the experience and knowledge of dispatchers in resolving train conflicts [17]. This manual scheduling practice proved its low efficiency with the increase of traffic congestion and exacerbated train delays.

An integrated policy for priority signals at intersections is required, given that trams operate in a *semi-exclusive* ROW environment. In the literature, we can find two different streams of studies: the first aiming at optimizing tram schedules without considering their effects on other traffic flows; the second aiming at manipulating the tram schedule so that trams always clear the intersection during green phases, thus reducing influences on other traffic flows. In [27], the tradeoffs between tram travel times and roadway traffic delays are explored. Literature counts several works applying different simulation techniques. Microscopic models, i.e., models in which each vehicle is modeled by itself as a particle, can be divided according to the representation of road structure in greater detail. In the continuous road model group, a base structure of road space is modeled as a continuous one dimensional (1D) link. The behavior of car agents is often implemented by applying car-following theories [26,28,35]. In the cell-type road model group, road space is discretized by homogeneous cells in which the behavior of car agents is expressed using transition rules such as cellular automata [19,29]. In a queuing model group, road networks are modeled as queuing networks [15,2]. Most commercial microscopic traffic simulators employ the continuous road model. In addition, several researchers have proposed simulation frameworks for mixed traffic of two or more models. For example, Yang et al. [32] proposed a framework for pedestrian road crossing behavior in Chinese cities in which they determined the criterion used by pedestrians to decide whether to start crossing a road after considering vehicle flows. Meanwhile,

Zeng et al. [34] modeled pedestrian-vehicles interactions at crosswalks in order to minimize pedestrian-vehicle collisions.

Dobler and Lämmel [12] integrated multi-modal simulation modules to the existing framework of MATSim, a large scale traffic simulation framework based on the queuing model [8]. Their integration approach was based on locally replacing simple queue structures with continuous 2D space at sections with higher traffic flows. The behavior rules of agents in the 2D space are based on the social force model (SFM). Krajzewicz et al. [20] introduced pedestrian and bicycle agent models into SUMO, which is a widely used traffic simulator belonging to the continuous road model group [21]. Finally Fujii et al. [14] introduced an agent-based framework for mixed-traffic of cars, pedestrians and trams by using the simulator MATES [33]. To our knowledge, there is no work that leverages analytic, non simulative, techniques for the analysis of traffic models.

### 3 Background

In this section, we provide some background on STPNs (Sect. 3.1), the method of stochastic state classes (Sect. 3.2), and the ORIS tool (Sect. 3.3).

#### 3.1 Stochastic Time Petri Nets

An STPN is a tuple  $\langle P, T, A^-, A^+, A, m_0, F, W, E, U \rangle$  where:  $P$  is the set of places;  $T$  is the set of transitions;  $A^- \subseteq P \times T$ ,  $A^+ \subseteq T \times P$  and  $A \subseteq P \times T$  are the sets of precondition, postcondition, and inhibitor arcs, respectively;  $m_0 : P \rightarrow \mathbb{N}$  is the initial marking;  $F : T \rightarrow [0, 1]^{[EFT_t, LFT_t]}$  associates each transition  $t$  with a Cumulative Distribution Function (CDF)  $F(t) : [EFT_t, LFT_t] \rightarrow [0, 1]$ , where  $EFT_t \in \mathbb{Q}_{\geq 0}$  and  $LFT_t \in \mathbb{Q}_{\geq 0} \cup \{\infty\}$  are the *earliest* and *latest firing time*, respectively;  $W : T \rightarrow \mathbb{R}_{>0}$  associates each transition with a weight;  $E$  and  $U$  associate each transition  $t$  with an enabling function  $E(t) : \mathbb{N}^P \rightarrow \{true, false\}$  and an update function  $U(t) : \mathbb{N}^P \rightarrow \mathbb{N}^P$ , which associate each marking with a boolean value and a new marking, respectively.

A place  $p$  is an *input*, an *output*, or an *inhibitor* place for a transition  $t$  if  $\langle p, t \rangle \in A^-$ ,  $\langle t, p \rangle \in A^+$ , and  $\langle p, t \rangle \in A$ , respectively. A transition  $t$  is *immediate* (IMM) if  $EFT_t = LFT_t = 0$  and *timed* otherwise; a timed transition  $t$  is *exponential* (EXP) if  $F_t(x) = 1 - e^{-\lambda x}$  over  $[0, \infty]$  with  $\lambda \in \mathbb{R}_{>0}$ , and *general* (GEN) otherwise; a general transition  $t$  is *deterministic* (DET) if  $EFT_t = LFT_t > 0$  and *distributed* otherwise; for each distributed transition  $t$ , we assume that  $F_t$  is the integral function of a Probability Density Function (PDF)  $f_t$ , i.e.,  $F_t(x) = \int_0^x f_t(y) dy$ . IMM, EXP, GEN, and DET transitions are represented by thick white, thick gray, thick black, or thin black bars, respectively.

The state of an STPN is a pair  $\langle m, \tau \rangle$ , where  $m$  is a marking and  $\tau : T \rightarrow \mathbb{R}_{\geq 0}$  associates each transition with a time-to-fire. A transition is *enabled* by a marking if each of its input places contains at least one token, none of its inhibitor places contains any token, and its enabling function evaluates to true;

an enabled transition  $t$  is *firable* in a state if its time-to-fire is equal to zero. The next transition  $t$  to fire in a state  $s = \langle m, \tau \rangle$  is selected among the set of firable transitions  $T_{f,s}$  with probability  $W(t) / \sum_{t_i \in T_{f,s}} W(t_i)$ . When  $t$  fires,  $s$  is replaced with  $s' = \langle m', \tau' \rangle$ , where:

- $m'$  is derived from  $m$  by: removing a token from each input place of  $t$ , which yields an intermediate marking  $m_{\text{tmp}}$ ; adding a token to each output place of  $t$ , which yields a second intermediate marking  $m'_{\text{tmp}}$ ; and, applying the update function  $U(t)$  to  $m'_{\text{tmp}}$ ;
- $\tau'$  is derived from  $\tau$  by: *i*) reducing the time-to-fire of each *persistent* transition (i.e., enabled by  $m$ ,  $m_{\text{tmp}}$  and  $m'$ ) by the time elapsed in  $s$ ; *ii*) sampling the time-to-fire of each *newly-enabled* transition  $t_n$  (i.e., enabled by  $m'$  but not by  $m_{\text{tmp}}$ ) according to  $F_{t_n}$ ; and, *iii*) removing the time-to-fire of each *disabled* transition (i.e., enabled by  $m$  but not by  $m'$ ).

### 3.2 The method of stochastic state classes

The method of stochastic state classes [31,18] permits the analysis of STPNs with multiple concurrent GEN transitions. Given a sequence of firings, a *stochastic state class* encodes the marking and the joint PDF of the times-to-fire of the enabled transitions and the absolute elapsed time  $\tau_{\text{age}}$ . Starting from an initial stochastic state class, the *transient tree* of stochastic state classes that can be reached within a time  $t_{\text{max}}$  is enumerated, enabling derivation of continuous-time transient probabilities of markings (*forward transient analysis*), i.e.,  $p_m(t) := P\{M(t) = m\} \forall 0 \leq t \leq t_{\text{max}}, \forall m \in \mathcal{M}$ , where  $M(t)$  is the *marking process* describing the marking  $M(t)$  of an STPN for each time  $t \geq 0$  and  $\mathcal{M}$  is the set of reachable markings.

If the STPN always reaches within a bounded number of firings a *regeneration*, i.e., a state satisfying the Markov condition, its marking process is a Markov Regenerative Process (MRP) [9], and its analysis can be performed enumerating stochastic state classes between any two regenerations. This results in a set of trees that permits to compute a local and a global kernel characterizing the MRP behavior, enabling evaluation of transient marking probabilities through the numerical solution of Markov renewal equations (*regenerative transient analysis*). Trees also permit to compute conditional probabilities of the Discrete Time Markov Chain (DTMC) embedded at regenerations and the expected time spent in any marking after each occurrence of any regeneration [22], supporting derivation of steady-state marking probabilities according to the Markov renewal theory (*regenerative steady-state analysis*).

While stochastic state classes support quantitative evaluation of an STPN model, the set  $\Omega$  of behaviors of the STPN can be identified with simpler and more consolidated means through non-deterministic analysis of the underlying TPN model. In this case, the state space is covered through the method of *state classes* [30,11], each made of a marking and a joint support for  $\tau_{\text{age}}$  and the times-to-fire of the enabled transitions. In this approach, enumeration of state classes starting from an initial marking provides a representation for the continuous set

of executions of an STPN, enabling verification of qualitative properties of the model, e.g., guarantee, with certainty, that a marking cannot be reached within a given time bound (*non-deterministic transient analysis*).

### 3.3 ORIS overview

ORIS [5]<sup>3</sup> is a software tool for qualitative verification and quantitative evaluation of reactive timed systems. ORIS supports modeling and evaluation of stochastic systems governed by timers (e.g., interleaving or service times, arrival rate, timeouts) with general probability density functions (PDFs). The tool adopts Stochastic Time Petri Nets (STPNs) as a graphical formalism to specify stochastic systems, and it efficiently implements the method of stochastic state classes, including regenerative transient, regenerative steady-state and non-deterministic analysis.

The software architecture of ORIS decouples the graphical editor from the underlying analysis engines. Given the many variants of Petri net features, ORIS was developed with extensibility in mind: new features can be defined by implementing specific interfaces, so that they can be introduced in the graphical editor and made available to the analysis engines. In turn, analysis engines implement a specific interface that allows them to cooperate with the graphical interface, i.e., to collect analysis options from the user, to start/stop analysis runs, to record and display analysis logs, and to show time series and tabular results. The available analysis engines include:

**Non-deterministic Analysis**, to produce a compact representation of the dense set of timed states that can be reached by the model. Non-deterministic analysis based on the theory of Difference Bound Matrix (DBM) supports the identification of the boundaries of the space of feasible timed behaviors [6]. The state space is displayed as a directed graph, where edges represent transition firings while nodes are *state classes* [30] comprising a marking and a DBM zone of timer values. This analysis is useful to debug STPNs models and ensure that their state space  $M$  is finite.

**Transient and Regenerative Analysis**, to compute transient probabilities in Generalized Semi-Markov Processes (GSMPs) and Markov Regenerative Processes (MRPs), respectively. These methods evaluate trees where edges are labeled with transitions and their firing probabilities, while nodes are *stochastic state classes* [18] comprising a marking, the PDF of timers, and their support (a DBM zone). For a given *time limit*  $T$ , the enumeration proceeds until the tree covers the transition firings of the STPN by time  $T$  with probability greater than  $1 - \epsilon$ , where  $\epsilon > 0$  is an *error* term. While standard transient analysis enumerates a single, very large tree of events, regenerative analysis avoids the enumeration of repeated subtrees rooted in the same *regeneration point* (where all general timers are reset or have been enabled for a deterministic time). A *time step*  $\Delta t$  is used to select equispaced time points where transient probabilities are

---

<sup>3</sup> ORIS is available for download at the webpage <https://www.oris-tool.org/>

evaluated (directly or by solving Markov renewal equations).

**Regenerative Steady-State Analysis**, to compute steady-state probabilities in MRPs (and thus Semi-Markov Processes (SMPs) and Continuous Time Markov Chains (CTMCs)) with irreducible state space. This method uses trees of stochastic state classes between regeneration points to compute steady-state probabilities of markings: expected sojourn times in each tree are combined with the steady-state probability of regenerations at their roots [22]. As for transient analysis, this method can be applied to STPNs allowing multiple general timers enabled in each state.

**Transient Analysis under Enabling Restriction**, to compute transient probabilities in MRPs that allow at most one general transition enabled in each state [16].

ORIS engines support instantaneous (transient or steady-state) and cumulative (transient) rewards. A *reward* is a real-valued function of markings  $r : M \rightarrow \mathbb{R}$  that is evaluated by substituting place names with the number of contained tokens in order to compute the instantaneous expected reward  $I_r(t) = \sum_{i \in M} r(i)p_i(t)$  at each time  $t$ , its steady-state value  $\bar{I}_r = \lim_{t \rightarrow \infty} I_r(t) = \sum_{i \in M} r(i)\bar{p}_i$  or its cumulative value over time  $C_r(t) = \int_0^t I_r(t)dt$ . In addition, the user can specify a *stop condition*, i.e., a Boolean predicate on markings such as  $(p_0 == 1) \& \& (p_1 == 1)$ , that is used to halt the STPN. This feature can be used to compute first-passage probabilities [18] or reach-avoid objectives equivalent to bounded until operators [25].

## 4 Diacceto-Alamanni: an STPN model

In this section, we describe the STPN model of the Diacceto-Alamanni intersection. Figure 3 shows the model which is composed of the following two submodels:

- tramway submodel (blue box);
- private traffic submodel (red box).

### 4.1 Tramway submodel

The portion of the tramway submodel in the dotted blue box represents the direction from Santa Maria Novella train station (Alamanni-Stazione), while the one in the dashed blue box represents the opposite direction. GEST provided the interleaving period of trams, which is equal to 220 s; the transition *period*, which models tram departures, fires a new token periodically and is enabled with continuity until place *KO* receives a token. Places *p0* and *p1* represent a tramway departing from Alamanni-Stazione and Villa Costanza, respectively. Transitions *delayFromSmn* and *delayFromScndc* represent the delays cumulated by the two trams, respectively; note that 120 s is an upper bound on the maximum delay

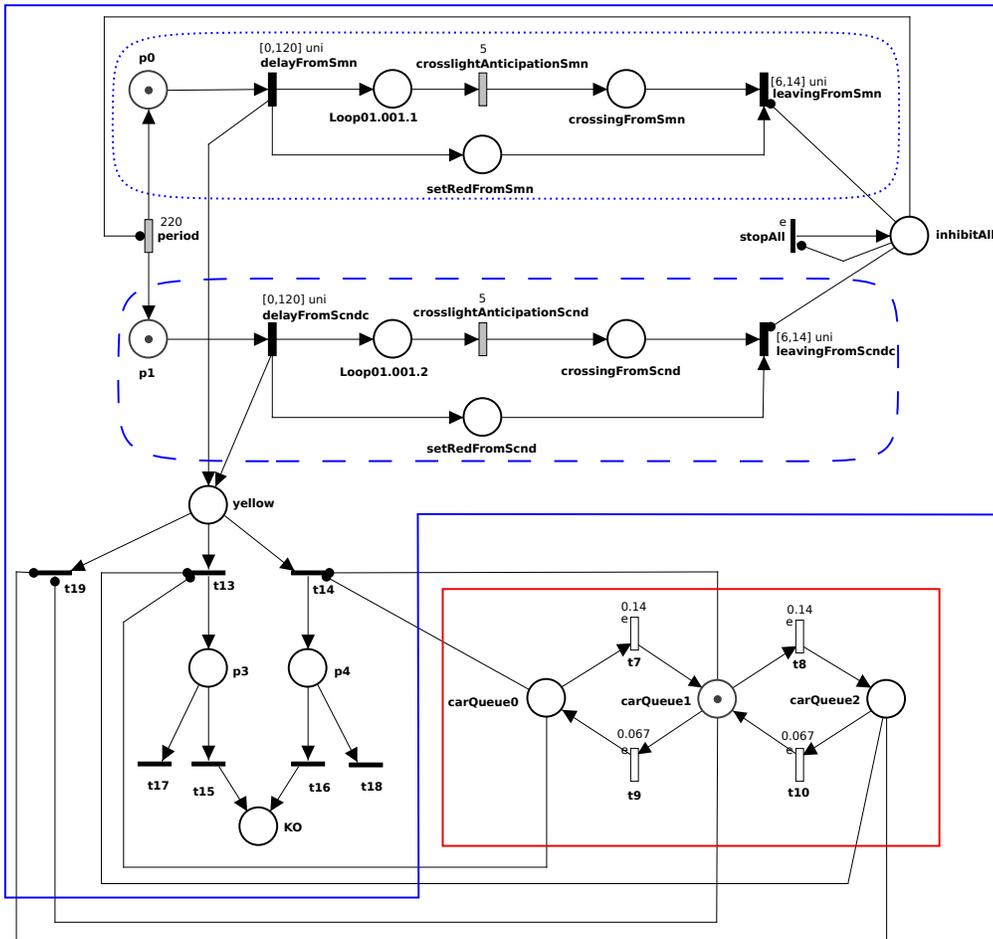


Fig. 3: Intersection model. The tramway submodel is highlighted by the blue box, the private traffic queue submodel is highlighted by the red one. Transitions associated with an enabling function are marked by a label "e".

observed in the available data set and, given that data are few and their distribution is unknown, this parameter is modeled using a uniform distribution [3].

When the tramway is approaching the intersection, dedicated wayside systems (i.e., two loops placed under the railway tracks) are activated (places *Loop01.001.1* and *Loop01.001.2*) and the corresponding traffic lights are set to red (places *setRedFromSmn* and *setRedFromScnd*). The traffic lights are in fact set to red 5 s before the arrival of the tram at the intersection; this parameter has been provided by GEST and is modeled by the DET transitions *crosslightAnticipationSmn* and *crosslightAnticipationScnd*. Places *crossingFromSmn* and *crossingFromScnd* represent the arrival of the tram at the intersection, while transitions *leavingFromSmn* and *leavingFromScndc* account for the time needed to free the intersection. Specifically, the minimum and the maximum time needed to free the intersection are set equal to 6 s and 14 s, respectively, based on the fact that in the data set provided by GEST this temporal parameter has mean value nearly equal to 10 s and a standard deviation approximately equal to 4 s. Also in this case, given that available data are few, this parameter is modeled by a uniform distribution over the interval [6, 14] [3].

## 4.2 Private transport submodel

We model private traffic as a birth-death process with three levels of traffic congestion: specifically, places *carQueue0*, *carQueue1*, and *carQueue2* model the condition of low, moderate, and high volume of traffic, respectively. Since we lack data on car traffic in Florence, we assume that the average traffic density is approximately 1000 cars per hour, which is a typical value for a high traffic flow on a single lane [24], and we consider the case that the arrival/departure of two cars increases/decreases the traffic congestion level, respectively, and that the time needed to occupy the intersection is nearly half the time needed to leave it. According to this, the EXP transitions *t7* and *t8* have rate equal to  $0.14 \text{ s}^{-1}$ , while the EXP transitions *t9* and *t10* have rate equal to  $0.067 \text{ s}^{-1}$ .

Intuitively, the number of cars in the queue increases when the private traffic light is set to red and decreases otherwise. In order to model this behavior, transitions *t7* and *t8* are associated with an enabling function that evaluates to true when at least one token is present in place *setRedFromSmn* or in place *setRedFromScnd* (i.e.  $\text{setRedFromSmn} + \text{setRedFromScnd} > 0$ ). Conversely, transitions *t9* and *t10* are associated with an enabling function that evaluates to true when no token is present in places *setRedFromSmn* and *setRedFromScnd* (i.e.,  $\text{setRedFromSmn} + \text{setRedFromScnd} == 0$ ).

## 4.3 Interaction between the tramway submodel and the private transport submodel

Road congestion may cause cars to stand for a while on the tracks after the private traffic light has turned to red, thus blocking trams. Place *yellow* models the private traffic light set to yellow, while place *KO* actually models the case that a tram ride is blocked by private vehicles on the lane. When place *KO*

receives a token, transition *stopAll* becomes enabled (given that it is associated with an enabling function  $KO > 0$ ) and fires, depositing a token in place *inhibitAll*. This finally disables transitions *period*, *leavingFromSmn*, and *leavingFromScndc*, due to the inhibitor arcs from *inhibitAll* to each of these transitions.

Transitions *t13* through *t19* model the possibility that a tram ride is blocked by private vehicles stopping on the tracks. If the traffic congestion level is low (i.e.,  $carQueue0 > 0$ ), the tram runs regularly and transition *t19* is enabled, so that no token is deposited in place *KO*. If traffic congestion increases to a moderate level ( $carQueue1 > 0$ ) or to a high level ( $carQueue2 > 0$ ), transition *t13* or transition *t14* becomes enabled and fires, respectively. In the former case ( $p3 > 0$ ), transitions *t15* and *t17* fire with probability 0.3 and 0.7, respectively, given that they have weight equal to 30 and 70, respectively; in the latter case, transitions *t16* and *t18* fire with probability 0.4 and 0.6, respectively, given that they have weight equal to 40 and 60, respectively. In doing so, the probability of a traffic block is 0.3 and 0.4 in the case of moderate and high traffic congestion, respectively. These parameters have been estimated from tram delays observed in the data set provided by GEST.

## 5 Analysis and Results

In this section, we report the results obtained from the analysis of the model of Sect. 4. In all the experiments, we performed regenerative transient analysis of the model through the ORIS tool using the following parameters:

- *Time limit*  $T = 7200$  s (corresponding to 2 h);
- *Time step*  $\Delta t = 20$  s.
- *Error*  $\epsilon = 0.01$ ;

The first experiment has been performed with average traffic density equal to 1000 cars per hour (i.e., the EXP transitions *t7* and *t8* have rate equal to  $0.14\text{ s}^{-1}$ , and the EXP transitions *t9* and *t10* have rate equal to  $0.067\text{ s}^{-1}$ , as shown in Figure 3) and crosslight anticipation equal to 5 s (i.e., the value of the DET transitions *crosslightAnticipationSmn* and *crosslightAnticipationScnd* is 5 s, as also shown in Figure 3). Figure 4 shows the probability of the private traffic queue status in a time interval of 2 h, obtained computing the instantaneous rewards “*carQueue0* > 0”, “*carQueue1* > 0”, and “*carQueue2* > 0”. Due to the high value of average traffic density, the car queue tends to be filled quite rapidly. As we can see, the reward “*carQueue2* > 0” (high traffic volume) tends to 1 s, while the rewards “*carQueue0* > 0” (low traffic volume) and “*carQueue1* > 0” (moderate traffic volume) tend to 0 s (note that the sum of tokens in places *carQueue0*, *carQueue1*, and *carQueue2* is 1 s).

Figure 5 shows the KO probability for different values of the crosslight anticipation parameter, obtained computing the instantaneous reward “*KO* > 0”. We observe that the probability of reaching the KO state increases every 220 s for all the displayed curves, due to periodic tram departures. We also note that the probability of reaching the KO state increases when the crosslight anticipation is

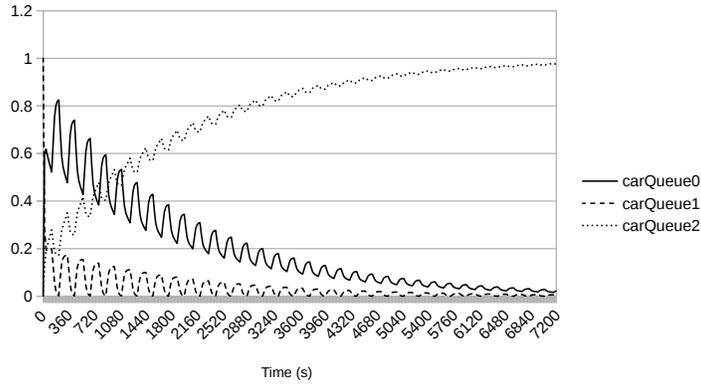


Fig. 4: Transient probability of the traffic queue status.

higher: intuitively, when the anticipation time increases, the time during which private traffic should flow away from the intersection decreases, thus degrading the queue status and consequently increasing the KO probability.

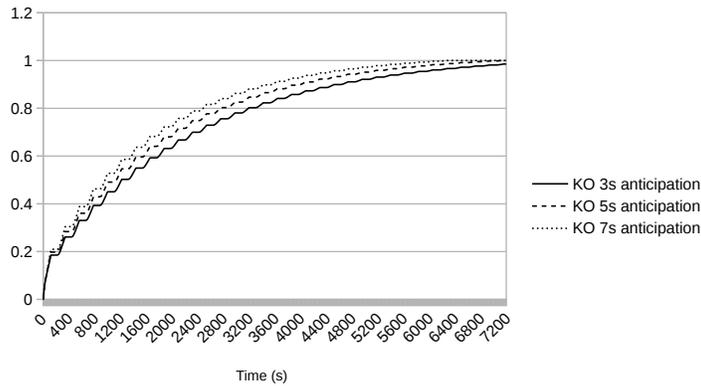


Fig. 5: Transient probability of the KO state for different values of the crosslight anticipation parameter.

Finally, Figure 6 shows the KO probability (obtained computing the instantaneous reward “ $KO > 0$ ”) for different values of the private traffic density. The probability of reaching the KO state increases when the traffic density is higher and reaches 0.7 in less than half an hour with extremely congested private traffic (i.e., 1500 cars per hour), while the same value is reached in more than an hour with moderately congested private traffic (i.e., 500 cars per hour).

We also argue that, for the planning of both tram timetables and traffic light timings, it is important to consider the correlation between the time of red signal, the time of green signal, and the tram headway, pointing out the need of

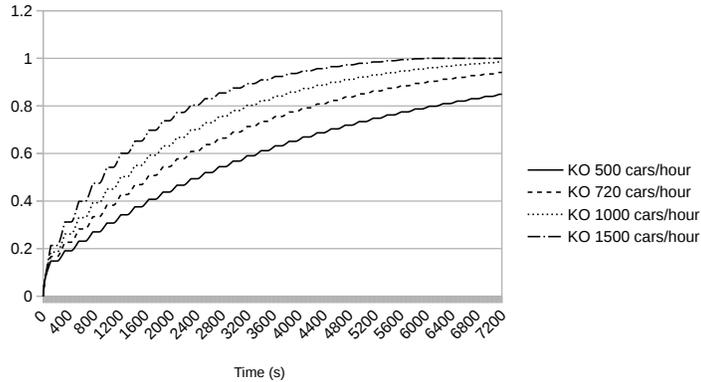


Fig. 6: Transient probability of the KO state for different values of traffic density.

an integrated management of the different transport systems in order to have a more robust and higher quality service. Furthermore, a more detailed analysis is needed to accurately model the behavior of private traffic during the day.

## 6 Conclusion

Modeling and analysis of complex intersections for the integration of private and public transport supports the evaluation of the perceived availability of public transport and the identification of robust traffic light plans and tram timetables. In this work, we presented an analytical approach to model and evaluate a critical intersection for the Florence tramway. Specifically, we used the ORIS tool to evaluate the probability of a traffic block, leveraging regenerative transient analysis based on the method of stochastic state classes to analyze a model of the intersection specified through Stochastic Time Petri Nets (STPNs). The analysis results showed a correlation between the frequency of tram rides, the traffic light plan, and the status of the queue of private vehicles, pointing out that the frequency of tram rides impacts on the road congestion. Therefore, compensating measures should be considered, such as synchronizing the passage of trams in opposite directions on the road crossing.

Within the context of modeling techniques to optimize the integration of public and private traffic, our work will go towards the following directions:

- analyze other road/tramway intersections, also considering the new tramway lines that will be opened in Florence, so as to compare differences and similarities and generalize the modeling methodology;
- improve the scalability of the approach by combining numerical solution of the tramway submodel through the ORIS tool with analytical evaluation of the traffic congestion level, which could permit to model private traffic more accurately (e.g., considering a larger number of congestion levels) without incurring in the state space explosion problem;

- evaluate to which extent the behavior of passengers and pedestrians as well as the weather conditions perturb the tramway performance, including them in the model of the road/tramway intersection [23].

## References

1. ACEA: The 2030 Urban Mobility Challenge. Tech. rep., European Automobile Manufacturers Association (May 2016)
2. Agarwal, A., Lämmel, G.: Modeling seepage behavior of smaller vehicles in mixed traffic conditions using an agent based simulation. *Transportation in Developing Economies* 2(2), 12 (2016)
3. Bernardi, S., Campos, J., Merseguer, J.: Timing-failure risk assessment of UML design using Time Petri Net bound techniques. *IEEE Transactions on Industrial Informatics* 7(1), 90–104 (2011)
4. Biagi, M., Carnevali, L., Paolieri, M., Vicario, E.: Performability evaluation of the ERTMS/ETCS - Level 3. *Transportation Research Part C: Emerging Technologies* 82, 314–336 (2017)
5. Bucci, G., Carnevali, L., Ridi, L., Vicario, E.: Oris: a tool for modeling, verification and evaluation of real-time systems. *International Journal for Software Tools for Technological Transfer* 12(5), 391–403 (September 2010)
6. Carnevali, L., Grassi, L., Vicario, E.: State-Density Functions over DBM Domains in the Analysis of Non-Markovian Models. *IEEE Transactions on Software Engineering* 35(2), 178–194 (2009)
7. Carnevali, L., Flammini, F., Paolieri, M., Vicario, E.: Non-Markovian Performability Evaluation of ERTMS/ETCS Level 3. In: *Lecture Notes in Computer Science* 9272. pp. 47–62. EPEW 2015, Springer (2015)
8. Charypar, D., Axhausen, K., Nagel, K.: Event-driven queue-based traffic flow microsimulation. *Transportation Research Record: Journal of the Transportation Research Board* pp. 35–40 (2007)
9. Choi, H., Kulkarni, V.G., Trivedi, K.S.: Markov regenerative stochastic petri nets. *Perform. Eval.* 20(1-3), 333–357 (May 1994)
10. Ciuti, I.: Jean-Luc Laugaa: “Ingorgo-trappola alla stazione, un rischio anche per la linea 2”. *Repubblica.it* (2014), <http://goo.gl/QxrXR4>.
11. Dill, D.L.: Timing assumptions and verification of finite-state concurrent systems. In: Sifakis, J. (ed.) *Automatic Verification Methods for Finite State Systems*. pp. 197–212. Springer Berlin Heidelberg, Berlin, Heidelberg (1990)
12. Dobler, C., Lämmel, G.: Integration of a multi-modal simulation module into a framework for large-scale transport systems simulation. In: *Pedestrian and Evacuation Dynamics 2012*. pp. 739–754. Springer International Publishing (2013)
13. ERTRAC: ERTRAC Road Transport Scenario 2030+ “Road to Implementation”. Tech. rep., European Road Transport Research Advisory Council (October 2009)
14. Fujii, H., Uchida, H., Yoshimura, S.: Agent-based simulation framework for mixed traffic of cars, pedestrians and trams. *Transportation Research Part C: Emerging Technologies* 85, 234–248 (2017)
15. Gawron, C.: An iterative algorithm to determine the dynamic user equilibrium in a traffic simulation model. *International Journal of Modern Physics C* 9(3), 393–407 (1998)
16. German, R.: *Performance Analysis of Communication Systems with Non-Markovian Stochastic Petri Nets*. John Wiley & Sons, Inc. (2000)

17. Higgins, A., Kozan, E., Ferreira, L.: Optimal scheduling of trains on a single line track. *Transportation Research Part B: Methodological* 30(2), 147 – 161 (1996)
18. Horváth, A., Paolieri, M., Ridi, L., Vicario, E.: Transient analysis of non-markovian models using stochastic state classes. *Perform. Eval.* 69(7-8), 315–335 (2012)
19. Kerner, B.S., Klenov, S.L., Wolf, D.E.: Cellular automata approach to three-phase traffic theory. *Journal of Physics A: Mathematical and General* 35(47), 9971–10013 (2002)
20. Krajzewicz, D., Erdmann, J., Härrri, J., Spyropoulos, T.: Including pedestrian and bicycle traffic into the traffic simulation SUMO. In: *ITS 2014, 10th ITS European Congress, 16-19 June 2014, Helsinki, Finland* (2014)
21. Krajzewicz, D., Hertkorn, G., Rössel, C., Wagner, P.: SUMO (Simulation of Urban MObility) - an open-source traffic simulation. In: *4th Middle East Symposium on Simulation and Modelling*. pp. 183–187 (2002)
22. Martina, S., Paolieri, M., Papini, T., Vicario, E.: Performance evaluation of fischer’s protocol through steady-state analysis of markov regenerative processes. In: *2016 IEEE 24th International Symposium on MASCOTS*. pp. 355–360 (2016)
23. Nagy, E., Csiszár, C.: Analysis of Delay Causes in Railway Passenger Transportation. *Periodica Polytechnica Transportation Engineering* 43(2), 73–80 (2015)
24. Ondráček, J., Schwarz, J., Ždímal, V., Andělová, L., Vodička, P., Bízek, V., Tsai, C.J., Chen, S.C., Smolík, J.: Contribution of the road traffic to air pollution in the Prague city (busy speedway and suburban crossroads). *Atmospheric Environment* 45(29), 5090–5100 (2011)
25. Paolieri, M., Horváth, A., Vicario, E.: Probabilistic model checking of regenerative concurrent systems. *IEEE Transactions on Software Engineering* 42(2), 153–169 (2016)
26. Peng, G., Cai, X., Liu, C., Cao, B., Tuo, M.: Optimal velocity difference model for a car-following theory. *Physics Letters A* 375(45), 3973 – 3977 (2011)
27. Shi, J., Sun, Y., Schonfeld, P., Qi, J.: Joint optimization of tram timetables and signal timing adjustments at intersections. *Transportation Research Part C: Emerging Technologies* 83, 104–119 (2017)
28. Tang, T., Wang, Y., Yang, X., Wu, Y.: A new car-following model accounting for varying road condition. *Nonlinear Dynamics* 70(2), 1397–1405 (2012)
29. Tonguz, O.K., Viriyasitavat, W., Bai, F.: Modeling urban traffic: A cellular automata approach. *IEEE Communications Magazine* 47(5), 142–150 (2009)
30. Vicario, E.: Static Analysis and Dynamic steering of time dependent systems using Time Petri Nets. *IEEE Transactions on Software Engineering* 27(1), 728–748 (2001)
31. Vicario, E., Sassoli, L., Carnevali, L.: Using stochastic state classes in quantitative evaluation of dense-time reactive systems. In: *IEEE Trans Software Engineering*. vol. 35, pp. 703–719 (2009)
32. Yang, J., Deng, W., Wang, J., Li, Q., Wang, Z.: Modeling pedestrians’ road crossing behavior in traffic system micro-simulation in china. *Transportation Research Part A: Policy and Practice* 40(3), 280–290 (2006)
33. Yoshimura, S.: MATES : Multi-agent based traffic and environmental simulator-theory, implementation and practical application. *Computer Modeling in Engineering and Sciences* 11(1), 17–25 (2006)
34. Zeng, W., Chen, P., Nakamura, H., Iryo-Asano, M.: Application of social force model to pedestrian behavior analysis at signalized crosswalk. *Transportation Research Part C: Emerging Technologies* 40, 143–159 (2014)
35. Zheng, L.J., Tian, C., Sun, D.H., Liu, W.N.: A new car-following model with consideration of anticipation driving behavior. *Nonlinear Dynamics* 70(2), 1205–1211 (2012)