Border crossing delay prediction using transient multi-server queueing models

Lei Lin, Qian Wang, Adel W. Sadek

Department of Civil, Structural and Environmental Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14260, USA

A R T I C L E   I N F O

Article history:
Received 30 November 2012
Received in revised form 3 September 2013
Accepted 24 March 2014
Available online 18 April 2014

Keywords:
Border-crossing delay prediction
Multi-server queueing models
Batch markovian arrival process
Phase type
Transient solution
Traffic simulation

A B S T R A C T

As a result of the continued increase in travel demand coupled with the need for tighter security and inspection procedures after September 11, border crossing delay has recently become a critical issue with tremendous economic and social costs. The current paper develops multi-server queueing models to estimate border crossing delay in support of a predictive traveler information system for the crossings. Two classes of multi-server models are considered: (1) models with exponential inter-arrival times and Erlang service times; and (2) a more generic model with a Batch Markovian Arrival Process (BMAP) and phase types (PH) services. As a case study, the models are developed based on real-time traffic volume and inspection time data collected at one of the major US–Canada border crossings, the Peace Bridge, and their transient solution is obtained using heuristic methods. For validation, the queueing models’ estimates are compared to the results from a detailed microscopic traffic simulation model of the Peace Bridge border crossing. The comparison shows that the transient queueing model, along its heuristic solution algorithm, is capable of predicting border crossing delay. Finally, a set of sensitivity analysis tests are conducted, and the developed models are incorporated within an optimization framework to help inform border crossing management strategies.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

In recent years, as a result of the continued increase in travel demand across the border coupled with the need for tighter security and inspection procedures after September 11, border crossing delay has become a critical problem with tremendous economic and social costs. A study conducted for the US Department of Transportation back in 2003 indicated that border crossing delays cost the US and Canada more than $13.2 billion every year (Taylor et al., 2003), and a report by the Ontario Chamber of Commerce (OCC) warned that if not adequately addressed, delay at the Detroit-Windsor border crossings may result in over 70,000 jobs lost by 2030 (OCC, 2004). More recently, in a press release in 2008, US former Transportation Secretary, Mary E. Peters, highlighted the border crossing delay problem by stating that such delays had cost businesses on the Canadian and the US sides as many as 14 billion dollars in 2007 (USDOT, Office of Public Affairs, 2008). Furthermore, a study by the Departments of Economics at the University of Waterloo and Wilfrid Laurier University in Canada concluded that the figures estimated by previous studies for the cost of the US–Canadian border crossing problem, while staggering, may have actually underestimated the true cost (Nguyen and Wigle, 2011).

* Corresponding author. Tel.: +1 716 645 4367; fax: +1 716 645 3733.
E-mail address: asadek@buffalo.edu (A.W. Sadek).

http://dx.doi.org/10.1016/j.tra.2014.03.013
0965-8564/© 2014 Elsevier Ltd. All rights reserved.
To alleviate the problem to some extent, transport authorities in the US and Canada presently provide current or instantaneous border-crossing delay information to the public, based on real-time traffic information collected at the border. The current border crossing times are communicated to the public via several information dissemination avenues, such as websites, on-road dynamic message signs and a traveler information phone system. While current border-crossing delay information is quite valuable in itself, it may be quite different from the travel times that the drivers would actually experience by the time they arrive at the border (i.e. experienced travel times). This is especially the case if there is a significant lag between the time when a traveler receives (or needs to act upon) the information and the time when she/he arrives at the border. This difference decreases the value of delay information in guiding traffic and mitigating congestion.

To address this, the current paper develops predictive models of border delay which can be used to provide drivers with estimates of the delay they are likely to experience when they arrive at the border. We view the problem of providing predictive information about border crossing delay as consisting of two sub-problems that need to be solved sequentially. The first is the short-term traffic volume forecasting problem, which is concerned with providing short-term predictions of the likely traffic volume at the border crossing for the next time period ranging from 30 min to a few hours. With this information, the second problem involves formulating and solving the appropriate queueing model in order to estimate the expected delay as a function of the predicted arrival volume from the solution of the first problem, the service rate and the number of customs inspection booths open.

Regarding the short-term traffic volume forecasting problem, extensive studies have been done for both freeway facilities (e.g. Smith and Demetsky, 1997; Park et al., 1998; Williams and Hoel, 2003; Smith et al., 2002; Huang and Sadek, 2009) and arterial streets (e.g. Lin et al., 2004; Liu et al., 2006). Moreover, for the special case of short-term predicting of border crossing traffic, the authors in a previous paper have proposed a multi-model combined forecast method which was shown to yield exceptionally good results (Lin et al., 2012).

Given this, the focus of this paper is on the second problem involving the formulation and solution of the appropriate queueing model for estimating border delay. Specifically, in this study, we formulate two classes of multi-server queueing models. The first looks at the special case of Erlang service times and exponential inter-arrival times (i.e. an \( M/\text{Ek}/n \) queueing model), and the second looks at the more generic case of a Batch Markovian Arrival Process (BMAP) and phase type (PH) services (i.e., \( \text{BMAP}/\text{PH}/n \) queueing model). The models are developed based on data collected from the Peace Bridge, one of the four major border crossings connecting Western New York, US to Southern Ontario, Canada, and their transient solutions are derived using heuristic approaches. For validation, the transient queueing model solutions are compared to those estimated from a well-calibrated microscopic traffic simulation model built for the Peace Bridge border crossing using VISSIM (PTV, 2010). Finally, a set of sensitivity analysis tests are conducted and the developed models are incorporated within an optimization framework to help inform border crossing management strategies.

The paper is organized as following. Section 2 provides background information about queueing models and their use for online travel time prediction. Section 3 introduces the Peace Bridge border crossing case study and describes the data collected. Section 4 discusses the details of the methodology. Specifically, that Section 1 describes the \( M/\text{Ek}/n \) queueing model formulation and its transient solution algorithm, followed by the \( \text{BMAP}/\text{PH}/n \) formulation along with its solution algorithm. It also describes the validation procedure which involved comparing the models’ results to the VISSIM simulation model’s results. Section 5 presents the validation and sensitivity analysis results. That section also includes an example on how the queueing models may be applied to develop “optimal” border management strategies. Finally, the main conclusions are summarized.

2. Literature review

Queueing models have been used extensively to solve problems related to manufacturing processes, transportation systems, product distribution systems, call centers, among other applications (Gontijo et al., 2011). Queueing models can be categorized in a number of different ways. One categorization divides them into stationary queueing models and transient queueing models as explained below.

Let \( N(t) \) denote the number of customers (i.e. vehicles) in the queueing system at time \( t \) measured from a fixed initial time moment \( t = 0 \), and let \( p_n(t) \) denote the probability that \( N(t) = n \) at time \( t \). Because it is usually difficult to find the time-dependent solution \( p_n(t) \) analytically, many applications in practice resort to consider only the steady state behavior of the queueing system after being in operation for a sufficiently long time. In that case, one is interested in the limiting behavior of \( p_n = \lim_{t \to \infty} p_n(t), n = 0, 1, 2, \ldots \) (Medhi, 2003). These queueing models that study the limiting probability of \( p_n(t) \) are called stationary queueing models.

Stationary queueing models are usually used to derive some useful performance measures such as the average waiting time, which in turn is often adopted as an objective function within an optimization framework to improve the efficiency of a system. For example, the study by Kim (2009) built a non-linear integer programming model to study the toll plaza optimization problem. In his model, the cost of the waiting time of the vehicles, as determined from the steady state solution of the queueing model, was minimized. Another example is the study by Ausin et al. (2007) which tried to minimize the steady state expected total waiting time cost by optimizing the number of servers based on real data from a bank. Moreover, Zhang et al. (2011) proposed a two-stage queueing model to balance security and customer service goals for a border crossing system. Besides waiting time, some researchers tried to estimate additional measures of queueing systems, such as traffic
intensity, based on stationary queueing models. Ke and Chu (2006), for example, proposed a consistent and asymptotically normal estimator of traffic intensity for a queueing system with distribution-free inter-arrival and service times. They also developed the confidence intervals for testing statistical hypothesis of intensity and derived the associated power function. Ke and Chu (2009) constructed and compared new confidence intervals of traffic intensity for a queueing system based on different bootstrap methods, and a new measure called relative coverage was introduced to assess the performances of the confidence intervals.

On the other hand, queueing models that consider time-dependent state probabilities are called transient queueing models. For systems that exhibit strong dynamic conditions, the transient solution of the queueing model is more meaningful. This is definitely the case for the border crossing problem considered in this study, where the arrival and service rates exhibit strong dynamic conditions. Hence, the focus of this paper is on the transient solution of the queueing models. An example of studies that considered transient queueing models is the study by Gupta (2011) which estimated air traffic delays using a transient $D(t)/M(t)/1$ queueing model. Moreover, the study by Escobar et al. (2002) provided some preliminary results regarding the approximate transient solution for multi-server queueing systems with Erlangian service times, based on the equally likely combination (ELC) heuristic. Because our solution approach to the queueing models with Erlang service times in this study is largely based on Escobar et al. (2002), their work is described in details in the Methodology section of this paper (i.e., Section 4).

Besides the study by Escobar et al. (2002), Ausín et al. (2008) used Bayesian inference to derive the transient behavior and the duration of the busy period for $GI/G/1$ queueing models, while Czachórski et al. (2009) studied the transient behavior of multi-servers with general service time and inter-arrival time distributions within the context of a call center. In the aforementioned studies, approximate solutions were proposed since exact solutions for transient queueing models with general distributions for arrival and service processes are notoriously hard to derive. Similar to the stationary models, transient queueing models can also be used within an optimization framework to identify optimal system operating policies. For example, Parlar and Sharafali (2008) derived the time-dependent operating characteristics of the queueing process that represents the operation of check-in counters at an airport. They then formulated a stochastic dynamic programming model to determine the optimal numbers of the check-in counters.

3. The Peace Bridge case study

As a case study for the development and solution of the border crossing delay prediction queueing models, this paper considers the Peace Bridge, one of the four main border crossings connecting Southern Ontario, Canada and Western New York, US. The Peace Bridge carries an estimated 4.76 million cars on the annual basis, and is thus one of the busiest border crossings between the US and Canada. When traveling through the bridge, traveling vehicles need to wait in line and go through security inspection before they can head to their destinations. The whole process can thus be considered as a queueing process with vehicle arrivals as the input flows, the inspection checkpoints as the service stations, and the queue length and waiting times as the performance indicators.

3.1. Estimation of arrival and service process distributions – Model 1

Information was collected regarding vehicle arrival patterns and security inspection processes in Peace Bridge in order to determine the appropriate distributions and the correct queueing model to use. Specifically, 700 observations of the vehicle inter-arrival times and 571 observations of the service times (i.e., inspection time) were collected from December 19, 2011 to January 10, 2012. The data were then used to define the appropriate probability distributions that best describe the arrival and service processes. As shown in Fig. 1, the distribution of headways (i.e., the inter-arrival times) was matched best by the exponential distribution $f(x) = \frac{\exp(-x/9.63)}{9.63}, x \geq 0$ with a mean value of 9.63 s. (for the first model, batch arrivals were

Fig. 1. Inter-arrival time distribution.
The R-square for fitting that curve to the collected inter-arrival time data was 0.8721, and the Root Mean Square Error (RMSE) was 0.0064.

For the service times, the best fitting distribution was found to be the Erlang distribution $f(x) = x \cdot \exp(-x/22.29)/22.29^2$ with order 2 and mean of 44.58 s, as shown in Fig. 2. Fitting that curve to the collected service time data points resulted in an R-square value of 0.8903, and RMSE of 0.009906.

3.2. Estimation of arrival and service process distributions – Model 2

In reality, the arrival process for transportation systems may not always be captured by an exponential distribution. In many cases, the arrival process may be the result of combining multiple streams with different exponential distributions. Moreover, it is quite common in the real-world for several vehicles to arrive simultaneously at the queueing system. Moreover, the service process could be more complicated than a simple Erlang distribution. To represent these complex arrival and service patterns, a more general modeling framework is to represent the system by a Batch Markov Arrival Process (BMAP) and a Phase Type (PH) distributed service process, as described below.

3.2.1. Batch Markov arrival process

Batch Markov Arrival Processes (BMAP) were introduced by Neuts (1979) in order to extend the standard Poisson process to account for more complex customer arrival processes in queueing systems. Let $J$ be an irreducible, continuous-time Markov chain with finite state space $E = \{1, 2, \ldots, e\}$, where $e$ is a finite, positive integer. Suppose $J$ has just entered state $i$, $1 \leq i \leq e$, the process spends an exponential distributed amount of time in state $i$ with mean $1/\lambda_i$ (Cordeiro and Kharoufeh, 2010). Let $\pi = \{\pi_1, \ldots, \pi_e\}$ denote the probability distribution of entering state $i$ for the process $J$, $p_{ij}$ be the probability for the system to switch from state $i$ to state $j$, $1 \leq j \leq m$ ($j$ may be equal to $i$), and $e_{b, i, j}$ represent the probability for $b$ vehicles to arrive in batch during the system’s transition from state $i$ to state $j$ (Daikoku et al., 2007). Obviously, the following conditions should exist:

$$\sum_{1 \leq i \leq e} \pi_i = 1, \quad \sum_{1 \leq i < e} p_{ij} = 1, \quad \sum_{b=0}^B e_{b, i, j} = 1, \text{ where } B \text{ is the maximum batch size}. \quad (1)$$

Without loss of generality, we assume $e_{0, i, i} = 0$, $1 \leq i \leq e$. Let $C$ and $D_b (1 \leq b \leq B)$ be $e \times e$ matrices. $C$ contains the transition rates of $J$ for which no arrivals occur, and $D_b (1 \leq b \leq B)$ contains the transition rates for which a batch size $b$ occurs (Cordeiro and Kharoufeh, 2010). The $(i, j)$th elements $C_{i, j}$ and $D_{b, i, j}$ in $C$ and $D_b$ are given as below:

$$C_{ij} = \begin{cases} -\lambda_i, & \text{if } i = j \\ \lambda_i p_{ij} e_{0, i, j}, & \text{otherwise} \end{cases} \quad (4)$$

$$D_{b, i, j} = \lambda_i p_{ij} e_{b, i, j}. \quad (5)$$

So at last, the BMAP can be characterized by a set of $e \times e$ matrices $(C, D_1, D_2, \ldots, D_B)$. And, $D = \sum_{1 \leq b \leq B} D_b \neq 0$, which means there must be some vehicles arriving.

As mentioned before, the Peace Bridge data included a total 700 observations of inter-arrival times. Out of those, 48 vehicles were observed to arrive in batch; unfortunately the study team did not record the exact size of batch for those
48 vehicles and the frequency at which this batch process occurred. For demonstrating our procedure therefore, we assume there are twelve batches with three vehicles and six batches with two vehicles and assign a random interval to each batch. In order to estimate BMAP (i.e. estimate parameters such as $\pi$, $C$ and $D_b$ ($1 \leq b \leq B$)), an expectation and maximization (EM) algorithm proposed by Lothar Breuer (2002) is used. The algorithm converged after four iterations, yielding the following estimates for BMAP’s parameters:

$$\pi = [0.7741, 0.2259],$$

$$C = \begin{bmatrix} -0.1063 & 0.0776 \\ 0.2694 & -0.3558 \end{bmatrix},$$

$$D_1 = \begin{bmatrix} 0.0046 & 0.0222 \\ 0.0727 & 0.0078 \end{bmatrix},$$

$$D_2 = \begin{bmatrix} 0.0001 & 0.0005 \\ 0.0016 & 0.0001 \end{bmatrix},$$

$$D_3 = \begin{bmatrix} 0.0002 & 0.0011 \\ 0.0037 & 0.0005 \end{bmatrix}.$$

Based on this, $p_{i,j}$ and $e_{b,i,j}$, $1 \leq i, j \leq e$ can be calculated and used to determine the arrival intervals in our approximation/simulation approach for deriving the transient solution of the queueing described later. Specifically, we can first determine the initial state $i$ by sampling from the inverse cumulative function of $\pi$. The next state of BMAP $j$ and the batch size of arrival $b$ in the state transition can also be sampled in the same way based on $p_{i,j}$ and $e_{b,i,j}$. Moreover, the inter-arrival time interval in this transition can be determined by sampling from the exponential distribution with rate $\lambda_i$ or using its mean value. When the system reaches state $j$, we continue to sample the next state, the arrival size and time interval. This procedure is repeated until the end of the prediction time horizon is reached.

### 3.2.2. Phase type distribution for service process

Consider a Markov process $J$ on a finite state space $(0, 1, \ldots, p)$ where 0 is absorbing and the other $p$ states are transient (Asmussen et al., 1996), a phase type (PH) distribution with parameter $(\pi, A)$ is the distribution of the time until absorption into state 0 in this Markov process. $\pi$ is the initial probability distribution of state, and it can be defined as $[\pi_0, \pi_p]$, where $\pi_0$ is the probability of starting the process at absorbing state 0, and $\pi_p$ is a $1 \times p$ vector containing the probabilities of starting at the other $p$ states. Obviously, $\pi_0 = 1 - \pi_p 1$, where 1 is a $1 \times 1$ vector with all elements as 1. The $p \times p$ dimensional matrix $A$ is called the phase-type generator (Asmussen et al., 1996). The $(i, j)$th elements $A_{i,j}$ are given as:

$$A_{i,j} = \begin{cases} - \sum_{k=0,k \neq i}^{p} \lambda_{ik}, & \text{if } i = j \\ \lambda_{ij}, & \text{otherwise} \end{cases},$$

where $\lambda_{ij}$ is the rate parameter of the exponential distribution, capturing the time that the Markov process spends at state $i$ before it goes to state $j$.

With this, the infinitesimal generator of this process can be written as:

$$Q = \begin{bmatrix} 0 & 0 \\ a & A \end{bmatrix},$$

where $a = -A 1$. Here, 1 is a $p \times 1$ vector with all elements as 1.

In this study, the KPC-Toolbox introduced by Casale et al. (2008), which uses the method of moment matching (Bobbio et al., 2005), was used to estimate the PH distribution of the observed service time for the Peace Bridge case study. The estimation results of $\pi$, $A$ and $Q$ are shown below:

$$\pi = [0.0163, 0.9837, 0],$$

$$A = \begin{bmatrix} -0.0126 & 0.0126 & 0 \\ 0 & -0.0488 & 0.0488 \\ 0 & 0 & -0.0488 \end{bmatrix},$$

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & -0.0126 & 0.0126 & 0 \\ 0 & 0 & -0.0488 & 0.0488 \\ 0.0488 & 0 & 0 & -0.0488 \end{bmatrix}.$$
4. Methodology

The methodology followed in this study can be viewed as consisting of two major steps: (1) queueing model development and solution; and (2) model validation. As previously mentioned, two groups of models were considered; first, the special case of an $M/E_\infty/n$ queueing model and then the more generic case of a $BMAP/PH/n$ model. The transient solution for such models was derived using an approximation/heuristic approach. Specifically for the $M/E_\infty/n$ model, a slightly modified version of the Equal Likely Combination (ELC) heuristic of Escobar et al. (2002), which reduces the computational burden, was utilized, whereas for the $BMAP/PH/n$, this study introduced a new heuristic (or assumption) which we call the Equally Likely Vehicle (ELV). In order to validate the model results empirically, the queue length and delay estimates derived from the queueing model solution were compared to those estimated from multiple runs of a detailed microscopic traffic simulation model. The details are shared below.

4.1. $M/E_\infty/n$ queueing models

As pointed out by Escobar et al. (2002), the exact solution of multi-server queueing models with exponential inter-arrival times and Erlangian service times ($M/E_\infty/n$) is quite challenging due to: (1) the very large number of possible system states, especially for systems with a large number of servers and/or Erlang distribution with high orders (i.e. high values of $k$); and (2) the complexity of the state transitions. To address these challenges, a handful of previous studies have proposed approximate solution methods to solve $M/E_\infty/n$ queueing model, utilizing ideas to simplify the size of the problem’s state space. Among those approximation method is the ELC heuristic (Escobar et al., 2002) that is applied to this study with some modifications to better suit the problem at hand. A brief description of how the solution proceeds is given below. While the details of the formulation and solution procedure can be found in Escobar et al. (2002), enough details are included herein to allow the reader to follow the presentation.

4.1.1. System state description

The key step in deriving either the steady state or the transient solution of an $M/E_\infty/n$ queueing model is to transform the complex Erlang distributed service process to a simpler version. Since an Erlang distribution with an integer order $k$ is equivalent to a sum of $k$ independent exponential distributions, a service station with $k$th-order Erlang distributed service times can be replaced by a chain of $k$ service stations with exponentially distributed service times. With this, the process for a vehicle to go through an Erlang service station becomes equivalent to the process of passing through a sequence of $k$ Exponential service stations.

Escobar et al. (2002) proposed a compact way to represent the system state. According to them, the system state can be represented by a three element descriptor $(l, m, r)$, where $l$ refers to the remaining number of exponential service stations that need to be completed for the vehicles currently in the system (this will be referred as stages thereafter), $m$ is the number of vehicles in the system, and $r$ is what the researchers called the “pattern identifier”. The pattern identifier is needed because there could be different instances or combinations where one would have $m$ vehicles in the system, and $l$ remaining stages or service stations. For example, in a queueing system with three Erlang service stations with order 3, the state $(6, 3)$ can represent either of the two patterns shown in Fig. 4. As can be seen, both patterns have three vehicles in the system. In pattern 1, one vehicle has just arrived at station 1, and the second vehicle has completed one stage of service in station 2, while the third vehicle has completed two stages in station 3. On the other hand, all three vehicles have completed one stage of service in Pattern 2. Finding the number of patterns associated with a given $(l, m)$ combination amounts to solving the problem of finding $m$ integers whose sum is $l$. This can be solved by writing a simple computer code or by enumeration.
4.1.2. State transition probabilities

Following the representation of the system states of a queueing model, the next step is to derive the state transition
probabilities (herein we use $P_{l,m}$ to denote the probability of a given state $(l,m)$ and we drop the pattern iden-
tifier index from the notation for simplicity). As previously, we calculate the state transition probabilities in this study on the basis of the equally likely combinations (ELC) heuristic method proposed by Escobar et al. (2002), which has been shown to be capable of simplifying the state transition calculation process, while maintaining the precision of the queueing model solution. While our solution approach is largely based on the framework proposed by Escobar et al. (2002), we introduce a slight modification to the solution algorithm which makes it more efficient. Specifically, the modification introduced involves updating the number of vehicles in the queueing system, when considering the vehicles’ arrival process, only when a new vehicle joins the queue, and because we are considering the transient solution, we determine the time when a new vehicle arrives at the system by randomly sampling from the inter-arrival time distribution curve (or by simply using the mean value of the inter-arrival distribution curve). This is slightly different from the original ELC heuristic where the arrival process is considered at every time step. The modification introduced helps reduce the state space that needs to be considered, and thus increases the computational efficiency of the algorithm. Table 1 defines the variables and parameters that are used in the following sections.

Now suppose that we are currently at time point, $t$, and that there are $m$ vehicles and $l$ unfinished stages in the system at that time step. For the next time step ($t + 1$), there may exist three possible types of state transition scenarios, as described below:

4.1.2.1. Scenario 1: one vehicle joins the queue. Let us assume first that the inter-arrival time period for vehicle $(N + 1)$ has been determined, either by sampling from the inverse cumulative function of the inter-arrival exponential distribution or by just using the mean value for that distribution as previously mentioned, and it is denoted by $t_N$ (for the sampled value) or by $t_{av}$ for the average value. Also let $t_N$ denote the time at which the $N$th vehicle had joined the queue. Now if the next time step, $t + 1$, which is equal to $t_N + t_e$ (or $t_N + t_{av}$ if we are using the average value), the state of the queueing system would transition from state $(l, m)$ to state $(l + k, m + 1)$ since the new arrival adds $k$ unfinished stages and one more vehicle to the system. Here, $k$ is the order of the Erlang distribution or the number of stages that need be finished for one vehicle to be fully served. In this case, the transition probability $P_{(l, m)\rightarrow(l+k, m+1)}$ should be 1. For other state transition scenarios mentioned later, $P_{(l, m)\rightarrow(l+k, m+1)}$ should be equal to 0, revealing that, without a new arrival, the transition from state $(l, m)$ to state $(l + k, m + 1)$ is impossible.

4.1.2.2. Scenario 2: one vehicle finishes its last stage of service and leaves the queue. Intuitively, the transition probability of this case can be represented as $P_{(l, m)\rightarrow(l-1, m-1)}$, and is calculated as follows. As was previously discussed, a given state may concern multiple patterns, $r$, and each pattern $i$ may involve multiple combinations of servers’ stages. For example, to produce pattern 1 shown on Fig. 4, we could have one server with three stages left, a second with two stages left, and a third with one stage left. Moreover, the specific station server with one, two or three stages left may vary (e.g. server number one may have three stages left, or two or just one). the number of different combinations for resulting in a given pattern can be calculated as:
\[ C_i = p_1!/(p_1!p_2! \ldots p_x!), \]  

\[ \gamma(t) = \text{The average arrival rate of the Exponential distribution} \]

\[ \mu = \text{The average service time of the Erlang distribution} \]

\[ \gamma = \text{The average service rate of the Erlang distribution} \]

where \( p = \min(m, n) \) is the number of active or busy service stations;

\( x \) is the largest number of unfinished stages for a server;

\( p_j \) is the number of servers with equal number of unfinished stages \( j, j = 1, 2, \ldots, x \).

Meanwhile, when relating patterns to system states, the total number of combinations producing a same state \((l, m)\) can be calculated as:

\[ C_{\text{total}(l,m)} = \sum_{i=D(l,m)} c_i, \]  

where \( D(l, m) \) is the set of the patterns that are associated with state \((l, m)\).

To illustrate, consider once again the example shown in Fig. 4. For pattern 1, there are three different numbers of unfinished stages for a server (i.e. 1, 2 and 3 stages). Among them, the largest number of unfinished stages is 3, which means that \( x = 3 \). Since each number of unfinished stages corresponds to one server, \( p_1 = 1, p_2 = 1, \) and \( p_3 = 1 \). Based on this, by using Eq. (8), the number of combinations for pattern 1, \( c_1 \), is equal to

\[ c_1 = 3!/(1!1!1!) = 6. \]

Similarly, for pattern 2, \( x = 2 \) and \( p_2 = 3 \) while \( p_1 = 0 \) since all the servers have two unfinished stages. Hence, the number of combinations for pattern 1 is equal to

\[ c_2 = 3!/0!3! = 1. \]

Since there are only two patterns associated with state \((6,3)\), the total number of combinations for the state is

\[ C_{\text{total}(6,3)} = c_1 + c_2 = 7. \]

The basic assumption of the ELC method is that all the possible combinations in \( C_{\text{total}(l,m)} \) are equally likely. Given this, the probability that one more stage of service is conducted and one vehicle leaves from the system can be calculated by:

\[ \alpha_{lm} = \sum_{i=D(l,m)} P_{i,l} = \sum_{i=D} (C_i/C_{\text{total}(l,m)}) \times (S_{1,i}/p) = \sum_{i=D} S_{1,i}C_i/pC_{\text{total}(l,m)}, \]  

where \( P_{i,l} = C_i/C_{\text{total}(l,m)} \) is the probability of pattern \( i \);

\( P_{1,l} = S_{1,l}/p \) is the probability of having servers with only one stage unfinished in pattern \( i \), and \( S_{1,l} \) is the number of servers with only one stage unfinished in pattern \( i \);

\( p = \min(m, n) \) is the number of active service stations;

\( D(l, m) \) is the set of the patterns that satisfy state \((l, m)\).

Finally, suppose \( \gamma(t) \) is the service rate that is sampled for each time step, \( t \), according to an Erlang distribution. Because the Erlang distribution with order \( k \) can be considered as a series of \( k \) consecutive, exponentially distributed tasks, the service rate for each stage is \( k\gamma(t) \), with a corresponding service time of \( 1/k\gamma(t) \). Given the distribution, for each time step (i.e. from the current time step \( t \) to the next time step \( t + 1 \)), the transition probability from state \((l, m)\) to state \((l - 1, m - 1)\) can be calculated as follows:
\[ P_{(1m)-(l-1,m)} = \alpha_{lm} * k_{l}(t) * p, \]

where \( p = \min(m, n) \) is the number of active service stations.

### 4.1.2.3. Scenario 3: one vehicle finishes one stage of service but still needs to stay in the queue for the other service stages.

The probability of this can be represented as \( P_{(l,m)\rightarrow(l-1,m)} \). Given the value of \( \alpha_{lm} \) (the probability that one more stage of service is conducted and one vehicle leaves from the system) was calculated as in Eq. (10) above, the probability that one stage is finished while no vehicles leave the queue, \( \beta_{l,m} \), can be calculated simply as:

\[ \beta_{l,m} = 1 - \alpha_{lm}. \]

Similarly, the transition probability from \((l, m)\) to state \((l - 1, m)\) can be calculated as follows:

\[ P_{(l,m)-(l-1,m)} = \beta_{l,m} * k_{l}(t) * p, \]

where \( p = \min (m, n) \) is the number of active service stations.

With these three scenarios discussed, the state-to-state transition diagram for the \( M|E_{k=2}/3 \) queueing model for example, can be depicted as shown in Fig. 5.

On Fig. 5, the number in the circle denotes the number of uncompleted stages in the queueing system (i.e. \( l \)), whereas the number on the top of each column represents the number of vehicles in the system (i.e. \( m \)). In other words, each circle represents a given state \((l, m)\). The different types of arrows show the different state to state transitions described in the left corner of Fig. 5. The solid arrow going toward the right represents the first transition case discussed in Section 4.1.2.1 (a new vehicle arrives). The dotted diagonal arrow represents the transition case described in Section 4.1.2.2 where one stage is completed and one vehicle leaves the system. Finally, the dotted vertical arrow moving downward represents the case described in Section 4.1.2.3 (one stage is completed but no vehicle departs).

### 4.1.3. State-to-state transitions and transient solution calculations

#### 4.1.3.1. Calculation approach overview

In this section, we briefly describe how the state-to-state transitions are calculated. Our discussion is once again largely based on the work of Escobar et al. (2002). However, we separate the description of the state transitions associated with serving vehicles from those associated with vehicle arrivals. In other words, in-between vehicle arrivals, we proceed in a time-step fashion to calculate the state transitions based on either state transition scenario 2 or 3 described above. On the other hand, when a new vehicle arrives, we calculate the state transitions based on scenario 1 above.

Because we are interested in the transient solution, our approach can be regarded as a hybrid between a simulation or numerical based approach on one hand, and an analytical approach on the other. The inter-arrival times are determined by sampling from the inter-arrival distribution curve, which determines when the next vehicle will arrive. In between the inter-arrival period, we sample the service rate from the service-time distribution at each time step which is one second in this study. The sampling mechanism was designed to facilitate the comparison to the VISSIM microscopic simulation results for validation as will be explained later in Section 4.2. Based on the sampled service rate, we calculate the state probabilities, as will be described in more detail in Section 4.1.3.2. Note that we also calculate the modeling results using the mean values of the inter-arrival and service times, and compare the results from both approaches (i.e. the randomly sampling based and the mean value based) to the VISSIM model’s results in the validation section.

#### 4.1.3.2. State probabilities calculations

Let \( P_{l,m}(t) \) represent the probability of state \((l, m)\) at the current time step \( t \). Naturally, at the initial state when no vehicle is in the system at time 0, \( P_{0,0}(0) = 1 \). Now consider the time interval in between when the \( N \)th vehicle arrives and when the \((N + 1)\)th vehicle arrives (as mentioned above, the inter-arrival time period between vehicle

---

**Fig. 5.** State to state transition process for the \( M|E_{k=2}/3 \) queueing model.
and $N + 1$ can be either sampled from the cumulative arrival distribution curve ($t_v$) or assumed as the mean value of the inter-arrival time period, $t_{av}$. Right before the $(N + 1)$th vehicle arrives, the possible values for the number of the vehicles $m$ in the system could range between 0 and $N$ (depending upon how many vehicles have already been served and have left the queue). In other words, all the states $(l, m)$, where $m \leq N$ may exist.

Now, at a given time step, the change in the probability for a state $(l, m)$ can be calculated according to the transition probabilities from and to that state, as indicated by the arrows going out of and toward the circles shown in Fig. 5 above. The following equations will describe exactly how the calculations proceed to update the state probabilities for each time step $t_i$, within the time period which starts at the time when the $N$th vehicle arrives and ends one time-step before the time when the $(N + 1)$th vehicle arrives (i.e. $t_i = t, \ldots, (t + t_{av} - 1)$ or $(t + t_{av} - 1)$). In these equations, $P_{lm}(t_i)$ represents the change in the value for $P_{lm}(t)$ at each time step, and therefore $P_{lm}(t_i + 1) = P_{lm}(t_i) + P_{lm}(t_i)$. This process is then repeated until the end of the analysis period of interest.

When calculating the change in the probabilities, one of the following two cases may occur, depending upon whether the number of vehicles in the system $(m)$ is less than the number of servers in the system (i.e. no queue exists) or not. For each case, the calculations are as follows:

Case (1): The number of vehicles in the system $(m)$ is less than the number of servers in the system $(n)$, $m < n$.

In this case, we have the following three possibilities:

(1a) For the special case of state $(0,0)$, the change in the state probability (see Fig. 5 for possible state transitions), can be calculated by Eq. (14) as shown below:

$$
\dot{P}_{0,0}(t_i) = k_1'(t_i)P_{1,1}(t_i).
$$

(1b) For the states where $l = mk$ (see Fig. 5 for possible state transitions for states $(mk, m)$ with $m = 1, 2, \ldots, n - 1$), we have:

$$
\dot{P}_{mk,m}(t_i) = -mk_1'(t_i)P_{mk,m}(t_i) + z_{mk-1,m+1}(m + 1)k_1'(t_i)P_{mk-1,m+1}(t_i).
$$

(1c) For the states where $l = mk - x$ with $m = 1, 2, \ldots, n - 1$, and $x = 1, 2, \ldots, m(k - 1)$, we have:

$$
\dot{P}_{mk-x,m}(t_i) = -mk_1'(t_i)P_{mk-x,m}(t_i) + \beta_{mk-x+1,m}mk_1'(t_i)P_{mk-x+1,m}(t_i) + z_{mk-x+1,m+1}(m + 1)k_1'(t_i)P_{mk-x+1,m+1}(t_i).
$$

Case (2): The number of vehicles in the system $(m)$ is equal to or greater than the number of servers $(n)$, $m \geq n$.

The difference between case (2) and case (1) considered above is that for case (2) arriving vehicles may start forming a queue behind the servers. In this case, we would have the following three possibilities:

(2a) For the states where $l = mk$ (i.e. states $(mk, m)$ with $m = n, (n + 1), (n + 2), \ldots, N$),

$$
\dot{P}_{mk,m}(t_i) = -mk_1'(t_i)P_{mk,m}(t_i) + z_{(n-1)k+1,n}mk_1'(t_i)P_{mk-1,m+1}(t_i).
$$

(2b) For the states where $l = mk - x$ with $m = n, (n + 1), (n + 2), \ldots, N$ and $x = 1, 2, \ldots, (n - 1)(k - 1)$,

$$
\dot{P}_{mk-x,m}(t_i) = -mk_1'(t_i)P_{mk-x,m}(t_i) + \beta_{mk-x+1,n}mk_1'(t_i)P_{mk-x+1,m}(t_i) + z_{(n-1)k-1,n}mk_1'(t_i)P_{mk-x+1,m+1}(t_i).
$$

(2c) For the states where $l = mk - y$ with $m = n, (n + 1), (n + 2), \ldots, N$ and $y = (n - 1)(k - 1) + 1, (n - 1)(k - 1) + 2, \ldots, n(k - 1)$,

$$
\dot{P}_{mk-y,m}(t_i) = -mk_1'(t_i)P_{mk-y,m}(t_i) + \beta_{mk-y+1,n}mk_1'(t_i)P_{mk-y+1,m}(t_i).
$$

After the $(N + 1)$th vehicle finally joins the queue at time point $t_i = (t + t_{av})$ or $t_i = (t + t_{av})$, all the possible states in the queueing system will be updated with probability 1, and it would look as if the probability values of the possible states moved one step to the right in Fig. 5 (i.e. the probability of a given state is now equal to the probability of the state to its left). The probabilities of the different states when a new vehicle arrives can thus be calculated as shown in Eq. (20) below:

$$
P_{mk-x,m}(t_i) = \begin{cases} P_{mk-x-m-1}(t_i), & \text{state } (mk - x - k, m - 1) \text{ exists} \\ 0, & \text{state } (mk - x - k, m - 1) \text{ not exist} \end{cases}
$$

for $m = (N + 1), N, \ldots, 1, 0$ and $x = \begin{cases} 0, \ldots, (n - 1)(k - 1), \ldots, n(k - 1), & \text{if } m > n \\ 0, \ldots, (m - 1)(k - 1), \ldots, m(k - 1), & \text{if } m \leq n \end{cases}$

An example of how the state probability calculations proceed and how the above equations are used is given in Appendix A for readers interested in the details.

4.1.4. Performance measurement calculations

With the probabilities calculated above, a number of useful performance measures can be calculated. In doing this, a unique advantage of our proposed hybrid numerical/analytical approach is that, as opposed to a purely simulation approach, the running time needed to derive these performance measures represents a fraction of the time needed to run a detailed microscopic traffic simulation model multiple times and to gather the required statistics. Specifically, after the state
transition probabilities for all states \((l, m)\) are calculated, the probability for \(m\) vehicles in the queueing system can be derived as follows:

\[
P_m(t) = \begin{cases} 
\sum_{l=m}^{km} P_{lm}(t), & 0 \leq m \leq n, \\
\sum_{l=n+(m-n)k}^{lm} P_{lm}(t), & m > n.
\end{cases}
\]  

With \(P_m(t)\) known, the average or most likely queue length at time \(t\), \(Q_m(t)\), can be calculated as:

\[
Q_m(t) = L_{veh} \cdot \sum_{m} P_m(t) \cdot m, 0 \leq m \leq N,
\]

where \(L_{veh}\) is the average length of the vehicle.

At the same time, the average delay for a vehicle that arrives at the border at time \(t\), can be calculated by estimating the time it would take for the queue in front of the vehicles to be served as follows:

\[
D_m(t) = \sum_{m} P_m(t) \cdot m \cdot \mu,
\]

where \(\mu\) is the average service time of the Erlang distribution.

Besides the mean values, the variance or standard deviation of the expected queue length and/or delay can be calculated. First, when the sampling is deployed (i.e. the inter-arrival and service times are determined by sampling from their probability distributions), the variance of the delay or queue length can be calculated by running the model multiple times. Alternatively, when using the mean values of the inter-arrival and service times, the variance of the delay, for example, may be calculated from Eq. (24), as follows.

\[
V_m(t) = \sum_{m} P_m(t) \cdot (m - \bar{m})^2 \cdot \mu,
\]

where \(\bar{m}\) is the average value of the number of vehicles.

4.2. BMAP/PH/n queueing model

Compared with \(M/E_k/n\) queueing model, the \(BMAP/PH/n\) queueing model has more complicated system states and state transition scenarios. At the onset, it is important to note that "system state" is used herein to describe the state of the queueing system itself, and that this is different from the "state space" mentioned in relation to the BMAP or PH distributions. To calculate the transient measures of the queueing system, this paper introduced a novel approach to describing the system states, along with a new assumption, which we call the Equally Likely Vehicles (ELV), to calculate the probabilities of the system states.

4.2.1. System state description

In order to make the description more understandable, we assume that "service type" \(i\) is equivalent to one of the service type distributions involved within that PH distribution (e.g., in our PH distribution, there are two service types: Exponential distribution and Erlang-2 distribution). Besides that, as was the case with the \(M/E_k/n\) we use "stage" \(l_i\) to represent the remaining number of exponential service stations that need to be completed for the vehicles \(m_i\) in service type \(i\).

Now suppose there are \(s\) service types in the PH distribution, a natural way is to use \((l_i, m_i)_{1 \leq i \leq s}\) to record the unfinished service stages \(l_i\) and number of vehicles \(m_i\) in service type \(i\). Here \(m_i \leq l_i \leq k_i \cdot m_i, m_i \geq 0, \) and \(k_i\) denotes the number of the total service stages of service type \(i\), like the order of the Erlang distribution. However, if there are \(N\) vehicles in the queueing system, the number of possible states will be \(s^N\). This can be a very huge number. In order to save some space, we only calculate \((l_i, m_i)_{1 \leq i \leq s}\) for the vehicles in the service stations, and use one more digit \(n_{ij}\) to represent the number of vehicles waiting in the queue and not being served. The complete state representation is now \((l_1, m_1, l_2, m_2, \ldots, l_s, m_s)n_{ij}\) in the BMAP/PH/n model. Using this representation, the total number of possible states will be \(s^N \cdot (N - n + 1)\) if \(N\) is greater than \(n\) and \(s^N\) if \(N\) is less than or equal to \(n\).

4.2.2. State transition probabilities

Suppose that at time point \(t\), \(N\) vehicles have joined the queue, and the queue is in state \((l_1, m_1, \ldots, l_i, m_i, \ldots, l_s, m_s)n_{ij}\), \(1 \leq i, j \leq s, i \neq j\). For the next time step \((t+1)\), there could be four possible state transition scenarios:

1. **Scenario 1: one or more vehicles join the queue**

   Different from the \(M/E_k/n\) queueing model, \(b\) vehicles (and not just one) could arrive in the queue at \(t_{N+b}\), and the initial service types when the \(b\) vehicles start being served can also be different. Now if the next time step, \(t+1\), is equal to \(t_{N+b}\), there are three situations:
(i) if there are empty servers, $\sum_{i=1}^{s} m_i < n$, and the number of empty servers is greater than or equal to $b$, all of the $b$ vehicles will instantly be served, and the queueing system state will change from $(l_1, m_1, \ldots, l_s, m_s)_{n_q}$ to $(l_1 + k_1 + m_{1bh}, m_1 + m_{1br}, \ldots, l_s + k_s + m_{sbh}, m_s + m_{sbr})_{n_q}$ with probability 1. Here, $m_{ibh}$ is the number of vehicles in $b$ with initial service type $i$;
(ii) if there are empty servers, $\sum_{i=1}^{s} m_i < n$, but the number of empty servers is less than $b$, $(e = n - \sum_{i=1}^{s} m_i)$ of the $b$ vehicles will start their service process, and the rest will wait for their service in the queue, the state $(l_1, m_1, \ldots, l_s, m_s)_{n_q}$ will become $(l_1 + k_1 + m_{1e}, m_1 + m_{1re}, \ldots, l_s + k_s + m_{se}, m_s + m_{se})_{n_q} + b - e$ with probability 1. Here, $m_{ie}$ is the number of vehicles in $e$ with initial service type $i$;
(iii) if there are no empty servers, $\sum_{i=1}^{s} m_i = n$, all the $b$ vehicles will wait in the queue without being served. The state will transfer from $(l_1, m_1, \ldots, l_s, m_s)_{n_q}$ to $(l_1, m_1, \ldots, l_s, m_s)_{n_q} + b$ with probability 1.

(2) Scenario 2: one vehicle finishes its last stage of service and leaves the queue
Suppose after finishing the last stage of service type $i$, the vehicle leaves the queue (corresponds to the absorption state 0 in PH distribution), the queueing system would thus transfer from state $(l_1, m_1, \ldots, l_s, m_s)_{n_q}$ to state $(l_1, m_1, \ldots, l_i - 1, m_i - 1, \ldots, l_s, m_s)_{n_q}, 1 \leq i \leq s$. If $n_q > 0$, which means there are still vehicles waiting in the queue to be served, state $(l_1, m_1, \ldots, l_i - 1, m_i - 1, \ldots, l_s, m_s)_{n_q}$ must be transferred to state $(l_1, m_1, \ldots, l_i - 1 + k_i, m_i + m_{ie}, \ldots, l_s, m_s)_{n_q} - 1$ or state $(l_1, m_1, \ldots, l_i - 1, m_i - 1, \ldots, l_j + k_j, m_j + 1, \ldots, l_s, m_s)_{n_q} - 1$ according to the initial service type of the $n_q^{th}$ vehicle in the reverse order from $N$.
Suppose $\mathcal{X}(l_1, m_1, \ldots, l_i, m_i)_n$ is the probability that one more stage of service type $i$ is finished and one out of $m_i$ vehicles leaves the queue. To calculate this, we still assume the ELC heuristic. Therefore,

$$\mathcal{X}(l_1, m_1, \ldots, l_i, m_i)_n = \mathcal{X}_i m_i,$$  
(25)

if the state transition is reasonable based on the Markov chain in PH distribution.
Otherwise if the state transition is impossible based on the Markov chain in PH distribution,

$$\mathcal{X}(l_1, m_1, \ldots, l_i, m_i)_n = 0.$$  
(26)

where $\mathcal{X}_i m_i$ can be calculated using the same way in $M/E_{\lambda}/n$ queueing model.

However, in order to calculate the queueing state transition probability that one stage of service type $i$ is finished and one vehicle leaves the system in one time step (i.e. from the current time step $t$ to the next time step $t+1$), one more assumption, which we call the Equally Likely Vehicles (ELV) heuristic, is needed. This heuristic assumes all vehicles have an equal probability to be served no matter what service type it is in, and therefore the probability that the vehicle to be served in the current time step is from service type $i$ is given by:

$$v_i = \frac{m_i}{\sum_{i=1}^{s} m_i}.$$  
(27)

Finally, the transition probability in this time step can be calculated as follows:

$$P(l_1, m_1, \ldots, l_i, m_i)_n St = v_i \cdot \mathcal{X}(l_1, m_1, \ldots, l_i, m_i)_n + k_i \gamma_i(t) \cdot m_i,$$  
(28)

where $St$ is the corresponding queueing state depending on the initial service type of the $n_q^{th}$ vehicle in the reverse order from $N$;

$$\gamma(t)$$ is the service rate at time step $t$ for service type $i$.

For example, let us consider how $\mathcal{X}_{(1,1,3,2)}$ is calculated. For the first pair $(1, 1, 3) = (1,1)$, which refers to the number of stages and the vehicles having the exponentially distributed service times, according to Eq. (26), $\mathcal{X}_{(1,1,3,2)} = 0$. This is because in this study we know that after the vehicle finishes the Exponential-2 service distribution, it will continue the process with the Erlang-2 service distribution. In other words, $(1,1,3,2) \rightarrow (0,0,3,2)$ is impossible. For the second pair $(1,2) = (3,2)$, which records the number of stages and the vehicles having service type of Erlang-2 distribution, according to Eq. (25), $\mathcal{X}_{(1,1,3,2)} = \mathcal{X}_{2,2} = \mathcal{X}_{3,2}$, and we know that $\mathcal{X}_{3,2} = 1$ from the previous section. Given this, we can finally estimate, $P(l_1, 1, 2, 0 \rightarrow (1,1,1,2),(0,0,1,2) = \mathcal{X}_2 + \mathcal{X}_{1,1,1,2}\times k_2 + \gamma_2(t) \cdot m_2 = \frac{m_i}{s m_i} + \mathcal{X}_{1,1,1,2} \times k_2 + \gamma_2(t) \cdot m_2$.

(3) Scenario 3: one vehicle finishes its last stage of service type $i$ and starts the first stage of another service type $j$ This means the transition from state $(l_1, m_1, \ldots, l_i, m_i, \ldots, l_s, m_s)_{n_q}$ to state $(l_1, m_1, \ldots, l_i - 1, m_i - 1, \ldots, l_j + k_j, m_j + 1, \ldots, l_s, m_s)_{n_q}, 1 \leq i, j \leq s, i \neq j$, from the current time step $t$ to the next time step $t+1$ (corresponds to the transition from state 1 to state 2 in Fig. 3).

Similarly, the probability can be calculated as:

$$P(l_1, m_1, \ldots, l_i, m_i, \ldots, l_s, m_s)_{n_q} \rightarrow (l_1, m_1, \ldots, l_i - 1, m_i - 1, \ldots, l_j + k_j, m_j + 1, \ldots, l_s, m_s)_{n_q} = v_i \cdot \mathcal{X}(l_1, m_1, \ldots, l_i, m_i)_n + k_i \gamma_i(t) \cdot m_i,$$  
(29)

where $v_i = \frac{m_i}{\sum_{i=1}^{s} m_i},$  
(30)

$$\mathcal{X}(l_1, m_1, \ldots, l_i, m_i)_n = \mathcal{X}_i m_i.$$  
(31)
if the state transition is reasonable based on the Markov chain in PH distribution. Otherwise,
\[ \tau_i(m_1, \ldots, l_i, m_n | n_q) = 0. \]

(32)

(4) Scenario 4: one vehicle finishes one stage of service type \( i \) but still needs to finish other stages of service type \( i \)

Now state \((l_i, m_1, \ldots, l_i, m_n)\) will become state \((l_i, m_1, \ldots, l_i - 1, m_n, \ldots, l_i, m_n)\) \( i \leq s \), from the current time step \( t \) to the next time step \( t + 1 \). Obviously, this probability is
\[ P_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n} = \nu_l \beta_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n} \cdot k_{l_i}^{(t)} \cdot m_{i}. \]

where
\[ \nu_l = \frac{m_{i}}{\sum_{i=1}^{s} m_i}, \]
\[ \beta_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n} = 1 - \alpha_{l_i, m_n}, \]
\[ \alpha_{l_i, m_n} = \frac{m_{i+1} + 1}{\sum_{i=1}^{s} m_{i+1} + 1}, \]
\[ m_{i+1} = m_{i} + 1, \]
\[ P_{s} = P_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n}. \]

4.2.3. State probabilities and transient solution calculations

In a simulation/approximation approach to determining the transient solution, the inter-arrival time and the arrival size can be sampled once the estimation of BMAP is realized. We can also estimate the initial service types of the newly arrived vehicles by sampling according to the PH distribution. Moreover, for each time step in the arrival intervals, the service rate can be determined either by sampling or using the mean value of the corresponding service time distribution, as done before. With these, the state probability can be calculated as follows:

For a given state \((l_i, m_1, \ldots, l_i, m_n)\) \( i \leq s \), \( i \neq j \), at time step \( t \), assuming that \( N \) vehicles have joined the queue, for each service type pair \((l_i, m_i)\), the probability \( P_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n} \) is calculated as:

When \( \sum_{i=1}^{s} m_i = 0 \) (which refers to the case when no vehicle is in the queue),
\[ \nu_{l_i} = k_{l_i}^{(t)}, \]
\[ \nu_{s} = \frac{m_{i} + 1}{\sum_{i=1}^{s} m_{i} + 1}, \]
\[ \nu_{l_i} = \frac{m_{i} + 1}{\sum_{i=1}^{s} m_{i} + 1}, \]
\[ \alpha_{l_i, m_n} = \frac{m_{i+1} + 1}{\sum_{i=1}^{s} m_{i+1} + 1}, \]
\[ m_{i+1} = m_{i} + 1, \]
\[ P_{s} = P_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n}. \]

When \( \sum_{i=1}^{s} m_i > 0 \) (some vehicles are in the queue), the transition probability can be generally expressed as,
\[ P_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n} = -m_i \cdot s_{l_i} \cdot P_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n} + \nu_{l_i} \cdot m_i \cdot s_{l_i} \cdot P_{s} + \nu_{s} \cdot m_i \cdot s_{l_i} \cdot P_{s} + \nu_{l_i} \cdot m_i \cdot s_{l_i} \cdot P_{s}, \]
\[ \text{where} \]
\[ s_{l_i} = k_{l_i}^{(t)}. \]

Now, if \( \sum_{i=1}^{s} m_{i+1} \leq n \),
\[ \nu_{l_i} = \frac{m_{i} + 1}{\sum_{i=1}^{s} m_{i} + 1}, \]
\[ \alpha_{l_i, m_n} = \frac{m_{i+1} + 1}{\sum_{i=1}^{s} m_{i+1} + 1}, \]
\[ m_{i+1} = m_{i} + 1, \]
\[ P_{s} = P_{l_i, m_1, \ldots, l_i, m_n, \ldots, l_i, m_n}. \]

Or if \( \sum_{i=1}^{s} m_{i+1} > n \),
\[ \nu_{l_i} = \frac{m_{i} + 1}{\sum_{i=1}^{s} m_{i} + 1}, \]
$$x = x_{(i, m_1, ..., l, m_l, ..., l, m_l)|n_q + 1},$$  \hspace{1cm} (50)$$

if the initial service type of the \((n_q + 1)\)th vehicle in the reverse order from \(N\) is \(i\); or

$$x = x_{(i, m_1, ..., l, m_l, ..., l, m_l)|n_q + 1},$$

if the initial service type of the \((n_q + 1)\)th vehicle in the reverse order from \(N\) is \(j, j \neq i\)

$$m_x = m_i,$$  \hspace{1cm} (51)

$$P_x = P_{(l, m_1, ..., l, m_l, ..., l, m_l)|n_q + 1},$$

if the initial service type of the \((n_q + 1)\)th vehicle in the reverse order from \(N\) is \(i\); or

$$P_x = P_{(l, m_1, ..., l, m_l, ..., l, m_l)|n_q + 1},$$

if the initial service type of the \((n_q + 1)\)th vehicle in the reverse order from \(N\) is \(j, j \neq i\)

$$v_x = \begin{cases} m_i + 1, & \text{if } (m_i + 1) \leq \sum_{l=1}^{s} m_l \\ 0, & \text{if } (m_i + 1) > \sum_{l=1}^{s} m_l \end{cases},$$  \hspace{1cm} (53)$$

$$\tau = \tau_{(l, m_1, ..., l, m_l, ..., l, m_l)|n_q},$$  \hspace{1cm} (54)

$$m_x = m_i + 1,$$  \hspace{1cm} (55)

$$P_x = P_{(l, m_1, ..., l, m_l, ..., l, m_l)|n_q},$$  \hspace{1cm} (56)

$$\nu_x = \frac{m_i}{\sum_{l=1}^{s} m_l},$$  \hspace{1cm} (57)

$$\beta = \beta_{(l, m_1, ..., l, m_l, ... \ldots)}$$

$$m_\beta = m_i,$$  \hspace{1cm} (58)

$$P_\beta = P_{(l, m_1, ..., l, m_l, ... \ldots)},$$  \hspace{1cm} (59)

At last, after all the service type pair \((l, m_i)\) is checked, \(P_{(l, m_1, ..., l, m_l, ... \ldots)}|n_q(t + 1)\) is updated as:

$$P_{(l, m_1, ..., l, m_l, ... \ldots)}|n_q(t + 1) = P_{(l, m_1, ..., l, m_l, ... \ldots)}|n_q(t) + \sum_{i=1}^{S} P_{(l, m_1, ..., l, m_l, ... \ldots)}|n_q(t).$$  \hspace{1cm} (61)$$

If at time step \(t + 1\), \(b\) vehicles will join the queue, for state \((l_1, m_{i_1}, ..., l_n, m_{i_n})|n_q\) according to the analysis of scenario 1 in Section 4.2.2, we can get the new state \((L_1, M_1, ..., L_n, M_n)|N_q\) and the probability of the new state should be:

$$P_{(l_1, m_{i_1}, ..., l_n, m_{i_n})|n_q(t + 1)} = P_{(l_1, m_{i_1}, ..., l_n, m_{i_n})|n_q(t + 1)},$$  \hspace{1cm} (62)

$$P_{(l_1, m_{i_1}, ..., l_n, m_{i_n})|n_q(t + 1)} = 0.$$  \hspace{1cm} (63)$$

This calculation should be conducted according to the possible number of vehicles in the queue in a decreasing order. \(\sum_{l=1}^{s} m_l + n_q = N, N - 1, ..., 1, 0\).

With all the state probabilities at any time step \(t\) known, the probability that there are \(v\) vehicles in the system can be finally calculated:

$$P_v(t) = \sum_{l=1}^{s} P_{(l_1, m_{i_1}, ..., l_n, m_{i_n})|n_q(t)} \text{ if } \sum_{l=1}^{s} m_l + n_q = v, \text{ for } v = 0, 1, ..., N.$$  \hspace{1cm} (64)$$

Similarly, this probability can be used to compute performance measures such as queue length and delay.

An example of how the state probabilities calculations proceed and how the above equations are to be applied is given in Appendix B for readers interested in the details.

4.3. The Baseline micro-simulation VISSIM model

To validate the proposed numerical/analytical approach, the queueing model's results were compared to the results derived from a detailed microscopic traffic simulation model of the border crossing area developed in VISSIM (PTV, 2010). The VISSIM model is used as a "baseline" model to validate the performance of the proposed queueing models and their approximate transient solution procedure, due to the current unavailability of detailed field observations regarding actually experienced delays and the corresponding number of inspection stations that were open at the time. As mentioned in the Conclusions section, we are planning to validate the models against field data in our future research. Fig. 6 shows a screen shot of the VISSIM animation of traffic at the Peace Bridge. The orange part in the figure is the US toll plaza for the private cars entering the US from Canada, which is the focus of this case study. As can be seen, the number of the lanes for private...
cars changes from 1 to 10 as one gets close to the border (i.e., the maximum number of inspection stations or servers for our queueing model is thus 10). In the example shown in Fig. 6, only 5 service stations are open.

In VISSIM, the dwell time distribution at the stop signs can be precisely controlled to follow a given probability distribution. Fig. 7 shows an example of how the dwell time distribution at a stop sign may be adjusted to follow any desired probability distribution. Given this, a stop-bar was placed upstream the inspection plaza in order to control the release of vehicles and to make sure that vehicles’ arrival at the toll plaza follow the desired probability distributions (i.e. the inter-arrival exponential distribution shown in Fig. 1). Similarly, a second stop bar was placed, where the inspection stations are, to simulate the service time Erlang distribution shown in Fig. 2. In other words, the VISSIM model was developed to mimic the operation of $M/E_k/n$ queueing system (i.e., the first multi-server queueing model considered in this study).

With this, the traffic demand was defined in a “.fma” file, and VISSIM’s optimal dynamic assignment module was used to simulate the drivers’ choices of lanes or inspection stations so as to ensure that the number of vehicles waiting in line for each service station is almost the same. Other settings were realized through the COM interface, which can be used to customize the VISSIM model (PTV, 2010), using the C# programming language. For example, the COM interface was used

Fig. 6. Queueing model in VISSIM.

Fig. 7. Setting of dwell time distribution in stop sign to simulate service time.
to control the number of lanes/inspection stations that are open at a given time by dynamically controlling the “LANE-CLOSED” parameter in VISSIM. The interface was also used to facilitate running the simulation model for multiple runs, using a different random seed number each time, and averaging the results to calculate the performance measures of interest (i.e., queue length and wait time) from the multiple runs.

5. Results

5.1. Validation results

As mentioned above, due to the unavailability of accurate field data on border crossing delay at the time this study was conducted, validating the queueing model results involved comparing them to those derived from running a VISSIM model, keeping in mind. For these comparisons, a 20 min prediction horizon was adopted (i.e., the models were used to estimate the queue length that a vehicle joining the queue 20 min later will encounter). The arrival volume was assumed to be equal to 400 vehicles per hour (vph), and an Erlang distribution with order 2 and mean 44.58 s was utilized to represent the service process for M/E_k/n queueing model, a mixture of an Exponential distribution with the mean of 79.36 s and an Erlang distribution with order 2 and the mean of 40.98 s was utilized to represent the service process for BMAP/PH/n queueing model. The number of service stations was varied from 3 to 6 stations.

As previously mentioned before, two slightly different methods were utilized to derive the transient solution of the queueing models, sampling the arrival and service times from the corresponding distributions and using the mean values of the corresponding distributions. For the M/E_k/n queueing model, when sampling, the heuristic queueing model solution procedure was repeated 100 times so that the mean and standard deviation of the queue length were calculated. For the BMAP/PH/n queueing model, because we always need to sample the arrival size and service types, the procedure based on sampling the arrival and service times method and the procedure based on using the mean values method were both repeated 100 times. The results from both methods were compared to the ones from the VISSIM model. For the VISSIM model, due to the fact that the model requires significantly more runtime compared to the runtime of the heuristic model solution, the model was run for only 10 times with different seed numbers. Finally for each model, the mean and the standard deviation of vehicles number in the queue can be seen in Table 2.

As can be seen, firstly, for the M/E_k/n queueing models, the results from the heuristic solution method of the queueing model appear to be quite close to the VISSIM model results. The advantage of the analytical approach is naturally the very high computational efficiency compared to the simulation based approach, and the ability to incorporate the models within an optimization framework as will be described later. Moreover, it can be seen that the results of the M/E_k/n queueing models from the random sampling method are also generally close to the results from the mean value based method. Meanwhile, using the mean values proves to be even more computationally efficient in comparison to the random sampling method because it does not require multiple runs. Based on this small-scale validation study, it can be concluded that the formulated queueing model M/E_k/n, and the ELC heuristic solution procedure, offers a more efficient approach to estimate likely queue lengths and border crossing delays than microscopic traffic simulation.

Secondly, we can see that the results of the BMAP/PH/n queueing models are quite close to those of the M/E_k/n queueing model and to the VISSIM simulation, except perhaps for when the number of the service stations was relatively small (i.e. 3) where the queueing lengths derived from the BMAP/PH/n queueing models are slightly longer. These results appear to make perfect sense, since from the analysis of PH service process, we know that the probability of a vehicle going through the Exponential service process followed by the Erlang service process is relatively small (i.e., 1.63%). For the majority of cases, the service process follows an Erlang distribution similar to the first model. Even though the probability is low, the differences are expected to be more obvious effect when the number of service stations is small. The transient results of the BMAP/PH/n queueing models thus appear to be realistic.

<table>
<thead>
<tr>
<th>Predicted traffic volume (vph)</th>
<th>Number of service stations</th>
<th>Number of vehicles in the queue (vehicles)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Random sampling method (M/E_k/n)</td>
<td>Mean</td>
</tr>
<tr>
<td></td>
<td>Mean</td>
<td>Standard variance</td>
</tr>
<tr>
<td>400</td>
<td>3</td>
<td>43.38</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>22.14</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>12.7</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>7</td>
</tr>
</tbody>
</table>
5.2. Sensitivity analysis

This section presents the results of a few sensitivity analysis tests conducted on the models. The purpose of this analysis is twofold, to demonstrate that the model results are reasonable and agree with intuition, and to provide some insight into how to effectively manage the border crossing in order to keep the delay within acceptable limits. Owing to the fact that there were minor differences between the $M/E_k/n$ and the $BMAP/PH/n$, the sensitivity analysis and the subsequent optimization are based on the $M/E_k/n$ for simplicity.

5.2.1. Impact of an increasing travel demand, $\lambda$

Starting with a base mean travel demand level or volume of $\lambda = 500$ veh/h, the demand level was increased in increments of 50 veh/h (up to a demand level of 1000 veh/h), and the expected queue length and the delay a vehicle joining the queue 20 min later would encounter, was calculated. In this test, the number of service stations was assumed to be equal to 3 stations, and the mean service time $\mu$ of the Erlang distribution with order 2 was set as 30 s. The results are shown in Table 3, where it can be seen that there is a significant increase in the delay with increasing volumes. For example, doubling the traffic volume from 500 to 1000 veh/h would result in an almost fourfold increase in average vehicle delay (from 26.2 min/vehicle to 109.4 min/vehicle).

Fig. 8 shows the evolution of the magnitude of the likely delay a vehicle would encounter as the prediction time period changes from 1 to 20 min (i.e., the figure shows how the delay a vehicle encounter would change if it arrives 1 min later versus 20 min later). The figure assumes a demand level of 500 veh/h, an average service time of 30 s, and 3 service stations open. As expected, the delay increases for vehicles joining the queue at later time periods. If an agency has a set policy to keep the delay under a certain threshold, for example, Fig. 8 can be used to determine when additional service stations would need to be opened. For example, if the border crossing authority desires to keep delay below say 10 min, and assuming the case shown in Fig. 8, the agency may need to open an additional inspection station around the 7th minute, when the delay is estimated to reach an average of 10 min/vehicle (see Fig. 8).

5.2.2. Impact of opening additional service stations, $n$

Another sensitivity analysis test performed involves varying the number of inspection stations from 3 to 10 for the base scenario considered in Section 5.2.1 (i.e., arrival rate of 500 veh/h and an average service time of 30 s). The results are shown in Fig. 8 which plots the delay for the vehicle joining the queue at the end of the 20-min period. As can be seen, there is a dramatic reduction in the value of the delay as the number of open service stations increase from 3 to 4, and also from 4 to 5 stations. Having more than 5 lanes open, however, does not appear to be quite beneficial from a delay saving standpoint, since the drop in the delay beyond that point is somewhat marginal. Plots such as Fig. 9 can also be used to determine the number of stations needed to keep delay below a certain threshold. In Fig. 9 for example, if the agency would like to keep delay around 10 min/vehicles, 4 service stations would be needed.

Table 3
Impact of increasing the traffic demand level.

<table>
<thead>
<tr>
<th>$\lambda$ (veh/h)</th>
<th>500</th>
<th>550</th>
<th>600</th>
<th>650</th>
<th>700</th>
<th>750</th>
<th>800</th>
<th>850</th>
<th>900</th>
<th>950</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queue length (no. of vehicles)</td>
<td>52.4</td>
<td>70.5</td>
<td>84.1</td>
<td>102.7</td>
<td>119.9</td>
<td>135.4</td>
<td>152.2</td>
<td>168.8</td>
<td>186.2</td>
<td>201.9</td>
<td>218.9</td>
</tr>
<tr>
<td>Delay (min)</td>
<td>26.2</td>
<td>35.2</td>
<td>42.0</td>
<td>51.3</td>
<td>59.9</td>
<td>67.7</td>
<td>76.1</td>
<td>84.4</td>
<td>93.1</td>
<td>100.9</td>
<td>109.4</td>
</tr>
</tbody>
</table>

Fig. 8. Delay curve of 20 min for $\lambda = 500$ veh/h and $\mu = 30$ s and $n = 3$. 
5.2.3. Impact of changes in the mean service time $\mu$

The last test performed involves varying the value of the mean service time $\mu$ of the Erlang Distribution (in increments of 15 s), and determining the number of inspection stations needed to keep the delay at the end of the 20th minute below 10 min/vehicle. The results are shown in Fig. 10, where it can be seen that if the mean service time is around 15 s, 3 service stations would be adequate. However, if the service time were to increase to 30 s, 4 stations would be needed, and so on.

5.3. Optimal operating policies

The sensitivity analysis described above pointed out the feasibility of using the queueing models to derive “optimal” operating policies for a customs and immigration border control agency. In this section, we include a simple example that demonstrates how the optimization problem may be formulated. We also provide a brief discussion of the results obtained and the insight they provide into the operation of the border crossing system.

5.3.1. Optimization problem formulation

The goal of the optimization problem formulated herein is to minimize the total cost of the queueing system for a given time period of analysis, $T$, including the cost for both the travelers as well as the operating agency. While doing that, the problem strives to keep the expected delay below a certain threshold. Specifically, we view the total cost as consisting of the following three elements. The first element is the operating cost of opening the inspection stations, calculated by multiplying the assumed hourly cost of operating one booth by the number of booths or inspection stations open by the length of assumed analysis period, $T$ (assumed to be 20 min in our study). The second element is the cost of the wait time travelers spent waiting in the queue at the border, calculated by multiplying the assumed monetary value for one hour of waiting time by the average number of vehicles in the queue during that time period by analysis period, $T$. The third element is a penalty term designed to capture the cost of switching between an open and a closed inspection lane (or vice versa). Two constraints are included: the first constraint is added to keep the average delay per vehicle below a certain threshold while a second

![Fig. 9. Delay at the end of 20 minutes for different service station number $n$ with $\lambda = 500$ veh/h and $\mu = 30$ s.](image)

![Fig. 10. Service station number for different average service time of Erlang distribution.](image)
constraint is included to make sure the number of inspection lanes open does not exceed the physical number of lanes available at the border crossing. The problem can thus be mathematically expressed as follows:

\[
\begin{align*}
\min & \quad C_t = (C_{\text{ope}} \cdot B_t + C_{\text{w}} \cdot V_t) \cdot T + C_{\text{pun}}, \\
\text{s.t.} & \quad \frac{V_t + \mu}{B_t} \leq T_{\text{w}}, \\
& \quad B_{\min} \leq B_t \leq B_{\max}.
\end{align*}
\]

where,

\begin{itemize}
  \item $C_t$ is the total cost of the queueing system during time period $T$;
  \item $C_{\text{ope}}$ is the cost per hour to operate one booth;
  \item $C_{\text{w}}$ is the hourly cost of waiting time per vehicle;
  \item $B_t$ is the number of open booths at time period $t$;
  \item $V_t$ is the average number of waiting vehicles at time period $T$;
  \item $\mu$ is the average service time (seconds);
  \item $T$ is the length of the analysis time period;
  \item $C_{\text{pun}}$ is the penalty cost for changing the number of open booths from one analysis time period to the next calculated as follows $C_{\text{pun}} = c \cdot |B_t - B_{t-1}|$, where $c$ is the penalty for switching for one booth.
  \item $\frac{V_t}{B_t} \leq T_{\text{w}}$ is the constraint that ensures that the average waiting time is less than a threshold value, $T_{\text{w}}$;
  \item $B_{\min} \leq B_t \leq B_{\max}$ is the constraint for the number of available booths.
\end{itemize}

5.3.2. Optimization problem results

To illustrate the application of the model for determining optimal operating policies for the border crossing, we considered the field-measured values for the hourly volumes at the Peace Bridge border crossing for a given day as shown in Fig. 11, and applied the optimization model formulated above successively for all the successive 20-min analysis periods of that day (i.e., we would calculate the predicted queue length and the average vehicle delay in 20-min increments throughout the day and calculate the total cost of the queueing system for that time period). The values assumed for the other model’s parameters were as follows. For the hourly operating cost of one booth ($C_{\text{ope}}$), four levels were assumed, a value of $50 per hour, $100 per hour, $150 per hour and a value of $200 per hour. For the monetary value of one hour of wait time ($C_{\text{w}}$), knowing the per capita annual income in New York in 2011 is $31,796 (United States Census Bureau, 2012), and assuming one person works 250 days per year, 6 h per day and there are 1.2 persons in one vehicle, the monetary value of one hour of wait time ($C_{\text{w}}$) was estimated to be around $25. The penalty for switching one booth ($c$) from closed to open (or vice versa) was set as $20. Given that the maximum number of inspection stations that can be opened at the Peace Bridge is 10, which meant that $B_{\min} = 1$ and $B_{\max} = 10$. Two levels of the accepted delay threshold ($T_{\text{w}}$) were considered, 10 min and 30 min.

Fig. 12 plots the values for the total cost obtained after solving the optimization model formulated above from the four assumed levels of hourly cost of opening a service booth (i.e., $50, $100, $150, and $200) and from two types of waiting time threshold (i.e., 10 min and 30 min) and from two types of waiting time threshold (i.e., 10 min and 30 min). The model was simply solved by enumeration since the solution space of the problem is quite limited (i.e., it ranged from 1 to 10 for each time period considered), and since the main focus herein is to illustrate the possible applications of the queueing models formulated. Fig. 13 shows the corresponding number of inspection lanes that are open for the four scenarios.

As can be seen and as to be expected, the total cost increases with the increase in the value assumed for the hourly operation cost (i.e., from $50 up to $200), and at the same time, the maximum number of open booths decreases. Specifically when the hourly operation cost ($C_{\text{ope}}$) is equal to $50 per hour, the maximum number of booths are opened (i.e. 10) and that number remains open for the rest of the day. This is because of the penalty for switching one booth every 20 min is set as $20, which is higher than the operation cost in 20 min ($50/3 or $16.67). Besides that, when $C_{\text{ope}} = 50$ or $C_{\text{ope}} = 100$, there were no differences in the total cost or in number of open booths when the waiting time threshold ($T_{\text{w}}$) was set to 10 min versus when it was set to 30 min. This is because in these cases the cost of opening new booths is relatively lower than the

![Fig. 11. Traffic volume of every 20 min interval for a whole day.](image-url)
delay cost for the travelers. As a result, the optimal solution was achieved at an average delay value less than the lower delay threshold (i.e. less than 10 min). In other words, the constraint \( \frac{V_t}{C_3} \geq T_{th} \) was not binding in this case. However, when \( C_{ope} \) is increased, for example, \( C_{ope} = \$150 \) or \( C_{ope} = \$200 \), minor differences in terms of the total cost between the two cases were observed (Fig. 12), along with discernible differences between the numbers of inspection booths open (Fig. 13). For the number of booths open, as to be expected, when a lower delay threshold is assumed (i.e. 10 min), more inspection stations will be needed to mitigate congestion.

Fig. 12. Total cost of optimizing the queueing system for a whole day.

Fig. 13. Number of open booth of optimizing the queueing system for a whole day.
Another interesting observation is that when \( C_{	ext{op}} = 150 \) or \( C_{	ext{op}} = 200 \), at some time intervals, the total cost for a 30-min average delay threshold \( (T_{th}) \) was slightly higher than that for the 10-min average delay threshold. This is because the increase in the waiting cost of the vehicles, when a higher delay threshold was allowed, slightly outweighed the savings in the operating cost resulting from opening fewer lanes.

6. Conclusions and future research

This study has formulated two groups of multi-server queueing models to predict border crossing delay, namely \( M/E_k/n \) and \( 
\text{BMAP}/\text{PH}/n \) queueing models. The models were formulated based on real-world data collected from the Peace Bridge, and their transient solution was numerically derived based on heuristic approaches. The solution derived was then validated by comparing the results to those estimated from a well-calibrated microscopic traffic simulation model. Sensitivity analyses were performed to check the validity of the model's predictions and to gain insight into how to effectively manage the border crossing. To further demonstrate the potential applications of the models, they were incorporated within an optimization framework and used to derive optimal management strategies for a customs and immigration border control agency. Among the main conclusions of the study are:

1. The transient solution of the \( M/E_k/n \) queueing models derived using the ELC heuristic appears to agree quite well with the values determined using the microscopic simulation models. The real advantage of the queueing modeling approach, however, is that its runtime is a fraction of the time needed to run the microscopic simulation model multiple times and gathering the required statistics.
2. For the case study considered herein, the results of \( 
\text{BMAP}/\text{PH}/n \) queueing models appear to be quite similar to \( M/E_k/n \) queueing models except for the case when the number of server stations was small.
3. The sensitivity analysis tests clearly demonstrate the reasonableness of the queueing model solution. Moreover, it shows that the models can be used to gain insight into how to best manage the border crossing; and
4. The solution of the border management optimization problem described in Section 5 shows that when the hourly cost of opening and operating a new inspection station is low, it becomes advantageous to open more lanes so as to keep the delay, and the associated wait time cost, on the low side. The case study considered also demonstrates that the solution of the optimal border crossing management strategy problem is quite sensitive to the assumptions regarding the cost of operating the inspection stations and the monetary value of waiting time.

The work described herein raises some interesting research questions which the authors hope to address in their future work. Some of these future research directions are listed below.

1. The accuracy of the border crossing delay, predicted by the queueing models formulated herein, is naturally dependent upon the ability to predict: (1) the future traffic volume; and (2) the number of inspection stations open. It would be quite interesting to study how robust the overall prediction system is to faulty assumptions regarding those two variables (i.e. predicted traffic volume and the number of inspection stations). Moreover, for the use of the models for online prediction (as a part of a real-time traveler information systems), future research should consider how to update the models’ predictions in real-time based upon real-world observations (i.e. measurements) of delay via technologies such as blue-tooth readers at border crossings.
2. As previously mentioned, the authors have previously developed models for predicting border crossing hourly traffic volumes at the Peace Bridge (Lin et al., 2012). The authors thus intend to integrate those traffic volume prediction models with the queueing models developed in this paper to construct a comprehensive tool for predicting the border crossing delay at the Peace Bridge. The overall predictive accuracy of the integrated system will then be assessed by comparing the integrated system’s results to field measurements of border crossing delay.
3. Finally, the authors plan to undertake a systematic study that looks at the three main Niagara Frontier’s border crossings (i.e., the Peace Bridge, the Rainbow Bridge, and the Lewiston-Queenston bridge), and the feasibility of developing a decision support system that balances the traffic load on the three bridges, through controlling the percentage of traffic recommended to use each border crossing.

Appendix A. An example of state probabilities calculations of \( M/E_k/n \) queueing model

In order to better understand the state probabilities calculation process discussed in session 4.1.3.2, this appendix illustrates how the equations for the \( M/E_{\infty}/3 \) queue may be applied. The appendix goes through the calculation procedure until all the equations shown in Section 4.1.3.2 have been applied at least once.

We assume initially that the queueing system is empty, and therefore \( P_{0,0}(0) = 1 \); (1) Suppose the first vehicle will arrive at time step \( t + t_1 \) or \( t + t_{ap} \) (\( t = 0 \) now). With this, and for the time steps, \( t_1 = t, \ldots, t + t_c - 1 \) or \( t + t_{ap} - 1 \), the only possible state is \((0,0)\), and therefore the number of vehicles, \( m = 0 \) and is less than the number of servers in the system, \( 0 < 3 \), and we are under Case (1) as discussed in Section 4.1.3.2. Therefore, using the
Eq. (14),
\[ P_{0,0}(t) = k^2 \gamma(t) + P_{1,1}(t) = 0, \]
where \( P_{1,1}(t) = 0; \) and hence, \( P_{0,0}(t + 1) = P_{0,0}(t) + P_{0,0}(t) = 1; \)

When the first vehicle joins the queue at the time step \( t = t \), or \( t = t + t_{av} \), the possible state becomes \((2,1)\). This can be determined according to Eq. (20) as follows. Because state \((0,0)\) exists, so
\[ P_{2,1}(t) = P_{0,0}(t) = 1; \]
but state \((-2, -1)\) does not exist, so
\[ P_{0,0}(t) = 0; \]

(2) Suppose the second vehicle will arrive at time step \( t + t \) or \( t + t_{av} \) is now the current time, which is equal to the time when the first vehicle joined the queue. With this, for the time steps, \( t = t, \ldots, t + t - 1 \) or \( t + t_{av} - 1 \), the possible states \((l, m)\) are \((0,0), (1,1), \) and \((2,1)\). Still \( m \) is equal to 0 or 1, and is thus less than the number of servers in the system, 3, and we are under Case (1) still.

Therefore, using the Eq. (14),
\[ \hat{P}_{0,0}(t) = k^2 \gamma(t) P_{1,1}(t); \]
Using Eq. (15),
\[ \hat{P}_{2,1}(t) = - (1 \times 2 \times \gamma(t)) P_{2,1}(t) + 2 \times 2 \times \gamma(t) P_{3,2}(t) = - (2 \times \gamma(t)) P_{2,1}(t); \]
Using Eq. (16),
\[ \hat{P}_{1,1}(t) = - 1 \times 2 \times \gamma(t) P_{1,1}(t) + 1 \times 2 \times \gamma(t) P_{2,1}(t) + 2 \times 2 \times \gamma(t) P_{2,2}(t) = - 2 \times \gamma(t) P_{2,1}(t) + 2 \times \beta_{2,1} \gamma(t) P_{2,1}(t); \]
The value of \( \beta_{2,1} \) is calculated by first determining the value of \( \alpha_{2,1} \) using Eq. (10). \( \alpha_{2,1} \) is equal to 0, because we have two stages left, not one, and therefore \( \beta_{2,1} = 1. \) As before, for each time step, \( P_{l,m}(t + 1) = P_{l,m}(t) + P_{l,m}(t) \) is used to update the state probabilities.

When the second vehicle joins the queue at the time step \( t = t + t \) or \( t + t_{av} \), the possible states are \((2,1), (3,2), \) and \((4,2)\), and therefore, according to Eq. (20), the state probabilities can be updated as follows:
\[ P_{2,1}(t) = P_{2,1}(t); \]
\[ P_{2,1}(t) = P_{0,0}(t); \]
\[ P_{2,2}(t) = P_{1,1}(t); \]
\[ P_{0,0}(t) = 0; \]
\[ P_{1,1}(t) = 0; \]

(3) Suppose the third vehicle will arrive at time step \( t + t \) or \( t + t_{av} \) (\( t \) is the current time, which is the time when the second vehicle joined the queue). With this, and for the time steps, \( t = t, \ldots, t + t - 1 \) or \( t + t_{av} - 1 \), the possible states \((l, m)\) are \((0,0), (1,1), (2,1), (2,2), (3,2), \) and \((4,2)\). Once again, \( m = 0, 1, \) or 2 and is less thus than the number of servers in the system, 3, therefore, we are still under Case (1).

Using the Eq. (14),
\[ \hat{P}_{0,0}(t) = k^2 \gamma(t) P_{1,1}(t); \]
Using Eq. (15),
\[ \hat{P}_{2,1}(t) = - (1 \times 2 \times \gamma(t)) P_{2,1}(t) + 2 \times 2 \times \gamma(t) P_{3,2}(t) = - (2 \times \gamma(t)) P_{2,1}(t); \]
Using Eq. (16),
\[ \hat{P}_{3,2}(t) = - (2 \times 2 \times \gamma(t)) P_{3,2}(t) + 2 \times 2 \times \gamma(t) P_{4,2}(t) + 3 \times 2 \times \gamma(t) P_{5,3}(t) = - (2 \times 2 \times \gamma(t)) P_{4,2}(t); \]
Here, we calculate, \( \alpha_{3,2} = 1/2, \) \( \beta_{4,2} = 1, \) \( \beta_{2,1} = 1, \) \( \beta_{2,2} = 1, \) \( \beta_{1,2} = 1/2. \) As before, for each time step, \( P_{l,m}(t + 1) = P_{l,m}(t) + P_{l,m}(t) \) is used to update the state probabilities.

When the third vehicle joins the queue at the time step \( t = t + t \) or \( t + t_{av} \), the possible states are \((2,1), (3,2), (4,2), \) \((4,3)\), \((5,3)\) and \((6,3)\).

According to Eq. (20),
\( P_{6,3}(t) = P_{4,2}(t); \)
\( P_{5,3}(t) = P_{3,2}(t); \)
\( P_{4,3}(t) = P_{2,2}(t); \)
\( P_{4,2}(t) = P_{2,1}(t); \)
\( P_{3,2}(t) = P_{1,1}(t); \)
\( P_{2,1}(t) = P_{0,0}(t); \)
\( P_{0,0}(t) = 0; \)
\( P_{2,2}(t) = 0; \)
\( P_{1,1}(t) = 0; \)

(4) Suppose the fourth vehicle will arrive at time step \( t + t_s \) or \( t + t_{av} \) (\( t \) is the current time, which is now equal to the time when the third vehicle joined the queue). With this, and for the time steps, \( t = t, \ldots, t + t_s - 1 \) or \( t + t_{av} - 1 \), the possible states are \((0,0), (1,1), (2,1), (2,2), (3,2), (4,2), (3,3), (4,3), (5,3) \) and \((6,3)\). Now when \( m = 3 \), the number of vehicles is equal to the number of servers in the system, 3, and we would be under Case (2).

Using Eq. (17),
\[ \dot{P}_{6,3}(t) = -3 * 2 + \gamma(t) P_{6,3}(t); \]

Using Eq. (18),
\[ \dot{P}_{5,3}(t) = -3 * 2 + \gamma(t) P_{5,3}(t) + 3 * 2 + \gamma(t) P_{5,4}(t) + 3 * 2 = 2 + \gamma(t) P_{5,3}(t) + 3 * 2 = \beta_{6,3} \gamma(t) P_{6,3}(t); \]

Using Eq. (19),
\[ \dot{P}_{4,3}(t) = -3 * 2 + \gamma(t) P_{4,3}(t) + 3 * 2 + \gamma(t) P_{4,4}(t) + 3 * 2 = 2 + \gamma(t) P_{4,3}(t) + 3 * 2 = \beta_{5,3} \gamma(t) P_{5,3}(t); \]

When \( m = 0, 1, 2 \) is less than the number of servers in the system 3, we go back to Case (1).

Therefore, using the Eq. (14),
\[ P_{0,0}(t) = k \gamma(t) P_{1,1}(t); \]

Using Eq. (15),
\[ \dot{P}_{2,1}(t) = -(1 * 2 + \gamma(t)) P_{2,1}(t) + 2 * 2 + \gamma(t) P_{3,2}(t); \]
\[ \dot{P}_{4,2}(t) = -(2 * 2 + \gamma(t)) P_{4,2}(t) + 3 * 2 + \gamma(t) P_{5,3}(t); \]

Using Eq. (16),
\[ \dot{P}_{3,2}(t) = -(2 * 2 + \gamma(t)) P_{3,2}(t) + 2 * 2 + \gamma(t) P_{4,3}(t) + 3 * 2 + \gamma(t) P_{4,4}(t); \]
\[ \dot{P}_{1,1}(t) = -1 * 2 + \gamma(t) P_{1,1}(t) + 1 * 2 + \gamma(t) P_{2,1}(t) + 2 * 2 + \gamma(t) P_{2,2}(t); \]
\[ \dot{P}_{2,2}(t) = -2 * 2 + \gamma(t) P_{2,2}(t) + 2 * 2 + \gamma(t) P_{3,2}(t) + 3 * 2 + \gamma(t) P_{3,3}(t); \]

Here, we calculate \( \beta_{6,3} = 1, \beta_{5,3} = 2/3, \beta_{4,3} = 1/3, \beta_{3,2} = 1/2, \beta_{4,2} = 1, \beta_{2,1} = 1, \beta_{2,2} = 1/2. \) As before, for each time step, \( P_{1,1}(t) = P_{1,1}(t + 1) + P_{1,1}(t) + P_{1,1}(t) \) is used to update the state probabilities.

When the fourth vehicle joins the queue, the possible states are \((2,1), (3,2), (4,2), (4,3), (5,3), (6,3), (5,4), (6,4), (7,4) \) and \((8,4)\), at the time step \( t = t + t_s \) or \( t = t + t_{av} \) according to Eq. (20),
\[ P_{5,4}(t) = P_{6,3}(t); \]
\[ P_{7,4}(t) = P_{5,3}(t); \]
\[ P_{6,4}(t) = P_{4,3}(t); \]
\[ P_{5,4}(t) = P_{3,3}(t); \]
\[ P_{6,3}(t) = P_{4,2}(t); \]
\[ P_{5,3}(t) = P_{3,2}(t); \]
\[ P_{4,3}(t) = P_{2,2}(t); \]
\[ P_{5,2}(t) = P_{2,1}(t); \]
\[ P_{4,2}(t) = P_{1,1}(t); \]
\[ P_{2,1}(t) = P_{0,0}(t); \]

For the other states, the probabilities should be 0.

So far, all the equations listed under the state probabilities calculation section have been applied at least once. The reader
may want to combine this appendix with Section 4.1.3.2 and Fig. 5 to fully understand the state probabilities calculation process.

**Appendix B. An example of state probabilities calculations of BMAP/PH/n queueing model**

To better understand BMAP/PH/n model, here we use an example to illustrate how to calculate the state probabilities as discussed in Section 4.2. The arrival process and service process have been discussed in Sections 4.2.1 and 4.2.2. Simply analyzing the first situation in scenario 1 and Eq. (62), $P(1,1,0,0)$ is as follows:

$$P(1,1,0,0) = \gamma_1(t_i).$$

According to Eq. (56), $P(0,0,0,0) = 0$, the first number captures the service type of this vehicle was determined to be an Exponential distribution. According to analysis of the first situation in scenario 1 and Eq. (62), $P(1,1,0,0) = 0$, because $P(1,1,0,0)$ is impossible, $\alpha_{(1,1,0,0)} = 0$.

(1) Suppose by sampling from $\pi_1$ and $\pi_2$, the initial state of the batch markov arrival process is 1, and that sampling from $p_1,1$ and $p_1,2$ indicates that it will transition to state 2. By sampling from $\sum_{i=0}^{b_2} e_{i} b_2 = 1$, assume only one vehicle will join the queue. Then by sampling or using the average value from the Exponential distribution with rate $\lambda_1$, we know that the first vehicle will arrive at time step $t + t_i$. Then we need check to the service type pair $(l_2, m_2) = (0, 0)$.

According to Eq. (43), $\gamma_1(t_i)$, the state transition from (0,0,0,0) to (1,1,0,0) is impossible, $\alpha_{(1,1,0,0)} = 0$. Let us assume that it will transition from (2,2,0,0) to (0,0,0,0).

Therefore, we need check to the service type pair $(l_2, m_2) = (0, 0)$, and a similar process can be performed,

$$P(0,0,0,0) = 0, \ P(0,0,0,0) = 0, \ P(0,0,0,0) = 0, \ P(0,0,0,0) = 0.$$

According to Eq. (42), $P_2 = P(2,2,0,0) = 0$, $P_3 = P(0,0,0,0) = 0$, $P_4 = P(1,1,0,0) = 0$.

According to analysis of the first situation in scenario 1 and Eq. (62), $P(1,1,0,0) = 0$, because $\sum_{i=0}^{b_2} e_{i} b_2 = 1$, assume only one vehicle will join the queue. Then by sampling or using the average value from the Exponential distribution with rate $\lambda_2$, we know that the 3 vehicles will arrive at time step $t + t_i$ and $t + t_i$. For time steps, $t_i = t, \ldots, t + t_i - 1$ or $t + t_i - 1$, the possible states are $(1,1,0,0)$, $(0,0,2,1)$, $(0,0,0,1)$, and $(0,0,0,0)$.

Then we check the service type pair $(l_2, m_2) = (0, 0)$, and a similar process can be performed,

$$P(0,0,0,0) = 0, \ P(0,0,0,0) = 0, \ P(0,0,0,0) = 0, \ P(0,0,0,0) = 0.$$

According to Eq. (46), $\alpha_{(2,2,0,0)} = 0$, according to Eq. (26), the state transition from (2,2,0,0) to (1,1,0,0) is impossible, $\alpha_{(2,2,0,0)} = 0$.

According to Eq. (47), $m_2 = 2$; According to Eq. (48), $P_2 = P(2,2,0,0) = 0$; According to Eq. (53), $\gamma_1 = \sum_{i=0}^{m_2} e_{i} m_2 = 0$; According to Eq. (54), $m_2 = \gamma_1 (2,2,0,0) = 0$, because $m_2 = i < 2$ should be met; According to Eq. (55), $m_2 = 2$; According to Eq. (56), $P_3 = P(2,2,0,0) = 0$; According to Eq. (57), $P_3 = \sum_{i=0}^{m_2} e_{i} m_2 = 0$.
According to Eq. (58), $\beta = \beta(2,1,0,0) = 0$, because $m_i \leq k_i \leq k_i * m_i * m_i \geq 0, 1 \leq i \leq 2$ should be met;
According to Eq. (59), $m_\beta = 1$;
According to Eq. (60), $P_\beta = P(2,1,0,0) = 0$;
So $P(1,1,0,0)(t_i) = -1 * \gamma_1(t_i) + P(1,1,0,0)(t_i) + \nu_2 * \alpha * m_* + \nu_2 * \tau * m_* + \nu_2 * \beta * m_* + \nu_2 = -1 * \gamma_1(t_i) + P(1,1,0,0)(t_i)$.
According to Eq. (61),
$$P(1,1,0,0)(t_i + 1) = P(1,1,0,0)(t_i) + P(1,1,0,0)(t_i).$$

Then check the service type pair $(l_2,m_2) = (0,0)$, and a similar process can be performed,
$$P(1,1,0,0)(t_i) = 0 * 2 \gamma_2(t_i) + P(1,1,0,0)(t_i) + \gamma_2 * m_* + \gamma_2 * \tau * m_* + \gamma_2 * \beta * m_* + \gamma_2 = 0,$n
$$P(1,1,0,0)(t_i + 1) = P(1,1,0,0)(t_i) + P(1,1,0,0)(t_i).$$

At time step $t_i = t + t_s$, or $t_i = t + t_{sv}$, 3 vehicles will join the queue, through sampling from the initial state distribution $\alpha$, assume the service types of the 3 vehicles are Exponential distribution, Erlang distribution and Exponential distribution, for states $(1,1,0,0), (0,0,2,1), (0,0,1,1), 0$, According to the analysis of the second situation in scenario 1 and Eqs. (62) and (63), with two available servers, they should become $(2,2,2,1), (1,1,4,2), 1$, and $(1,1,3,2), 1$,
$$P(2,2,2,1)(t_i) = P(1,1,0,0)(t_i), P(1,1,0,0)(t_i) = 0,$n$$P(1,1,4,2)(t_i) = P(0,0,2,1)(t_i), P(0,0,2,1)(t_i) = 0,$n$$P(1,1,3,2)(t_i) = P(0,0,1,1)(t_i), P(0,0,1,1)(t_i) = 0$$
for state $(0,0,0,0), 0$, According to the analysis of the first situation in scenario 1, it will become $(2,2,2,1), 0$, according to Eqs. (62) and (63),
$$P(2,2,2,1)(t_i) = P(0,0,0,0)(t_i), P(0,0,0,0)(t_i) = 0$$

(3) Now the BMAP is at state 1 again, and by continuing to sample from $p_{1,1}$ and $p_{1,2}$, let us assume that it will stay at state 1. By sampling from $\sum_{t=0}^{1} \alpha(t_i) = 1$, assume one vehicle will join the queue. Finally, by sampling or using the average value from the exponential distribution with rate $\lambda_1$, we now know the vehicle will arrive at time step $t + t_s$, and $t + t_{sv}$. For time steps, $t_i = t, t + t_s - 1$ or $t + t_{sv} - 1$, the possible states are $(0,0,6,3), (0,0,5,3), (0,0,4,3), (0,0,3,3), (1,1,4,2), 1$, $(1,1,3,2), 1$, $(1,1,2,2), 1$, $(2,2,1,1), 1$, $(3,3,0,0), 1$, $(0,0,6,3), 0$, $(0,0,5,3), 0$, $(0,0,4,3), 0$, $(0,0,3,3), 0$, $(1,1,4,2), 0$, $(1,1,3,2), 0$, $(1,1,2,2), 0$, $(2,2,1,1), 0$, $(3,3,0,0), 0$, $(0,0,4,2), 0$, $(0,0,3,2), 0$, $(0,0,2,2), 0$, $(1,1,2,1), 0$, $(1,1,1,1), 0$, $(2,2,0,0), 0$, $(1,1,0,0), 0$, $(0,0,2,1), 0$, $(0,0,1,1), 0$ and $(0,0,0,0), 0$: Take the probability calculation of $(3,3,0,0), 0$

as an example:
Because $\sum_{t=1}^{m} m_4 = 3$, Eq. (43) should be used to calculate $P(3,3,0,0)(t_i)$.
Firstly, we check the service type pair $(l_1, m_1) = (3,3)$,
According to Eq. (44), $\gamma_1(t_i)$, Because $\sum_{t=1}^{m} m_1 + 1 = 4 > n$, According to Eq. (49), $\nu_2 = \frac{m}{\sum_{t=1}^{m} m_1} = 1$.
According to Eq. (50), $\alpha = \alpha(3,3,0,0), 1$, and then according to Eq. (26), the state transition from $(3,3,0,0), 1$ to $(3,3,0,0), 0$ is impossible, $\alpha(3,3,0,0), 1 = 0$.
According to Eq. (51), $m_2 = 3$;
According to Eq. (52), $P_2 = P(3,3,0,0), 1$;
According to Eq. (53), $\nu_2 = 0$.
According to Eq. (54), $\tau = \frac{\tau_4}{t_4 - 0} = 0$, because $m_i \leq k_i \leq k_i * m_i, m_i \geq 0, 1 \leq i \leq 2$ should be met;
According to Eq. (55), $m_4 = 4$;
According to Eq. (56), $P_4 = P(4,4, - 1, 0) = 0$;
According to Eq. (57), $\nu_4 = \frac{m}{\sum_{t=1}^{m} m_4} = 1$.
According to Eq. (58), $\beta = \beta(4,3,0,0), 0 = 0$, because $m_i \leq k_i \leq k_i * m_i, m_i \geq 0, 1 \leq i \leq 2$ should be met;
According to Eq. (59), $m_5 = 3$;
According to Eq. (60),
$$P_5 = P(4,3,0,0), 0 = 0$$

So $P(3,3,0,0)(t_i) = -3 * \gamma_1(t_i) + P(3,3,0,0)(t_i) + \nu_2 * \alpha * m_* + \nu_2 * \tau * m_* + \nu_2 * \beta * m_* + \nu_2 = -3 * \gamma_1(t_i) * P(3,3,0,0)(t_i)$.
According to Eq. (61),
$$P(3,3,0,0)(t_i + 1) = P(3,3,0,0)(t_i) + P(3,3,0,0)(t_i).$$
Then we check the service type pair \((l_2, m_2) = (0,0)\), According to Eq. (44), \(s_{r_t} = \gamma_2(t_i)\), Because \(\sum_{i=1}^{2} m_i + 1 = 4 > n\), According to Eq. (49), \(v_a = \frac{m}{\sum_{i=1}^{2} m_i} = 0\), According to Eq. (50), \(a = \frac{1}{(2,2,1)} = 1\), because the initial service type of the 1st vehicle in the reverse order from \(N = 4\) is 1; According to Eq. (51), \(m_a = 0\), According to Eq. (52), \(P_s = P_{(2,2,1)} = 0\), According to Eq. (53), \(v_s = \frac{m}{\sum_{i=1}^{2} m_i} = \frac{1}{4}\), According to Eq. (54), \(\tau = \tau_{(2,2,1)}\), according to Eq. (26), the transition from state \((2,2,1)\) to state \((3,3,0)\) is impossible. According to Eq. (55), \(m_s = 1\); According to Eq. (56), \(P_s = P_{(2,2,1)} = 0\), According to Eq. (57), \(v_s = \frac{m}{\sum_{i=1}^{2} m_i} = 0\), According to Eq. (58), \(b = \frac{1}{(3,3,1,0)}\), because \(m_i \leq l_i < k_i \cdot m_i, m_i \geq 1, 1 < i \leq 2\) should be met; According to Eq. (59), \(m_b = 0\), According to Eq. (60), \(P_s = P_{(3,3,1,0)} = 0\), So \(P_{(3,3,0,0)}(t_i) = -0 \cdot \gamma_2(t_i) + P_{(3,3,0,0)}(t_i) + \sum_{i=1}^{2} m_i \cdot s_{r_i} P_s + \sum_{i=1}^{2} m_i \cdot s_{r_i} P_s + v_s \cdot \gamma_2(t_i) + v_s \cdot \gamma_2(t_i) = 0\), According to Eq. (61), \(P_{(3,3,0,0)}(t_i + 1) = P_{(3,3,0,0)}(t_i) + P_{(3,3,0,0)}(t_i)\).

At time step \(t + t_s\) and \(t + t_{av}\), the vehicle will join the queue, for states \((0,0,6,3)\), \((0,0,5,3)\), \((0,0,4,3)\), \((0,0,3,3)\), \((1,1,4,2)\), \((1,1,3,2)\), \((1,1,2,2)\), \((2,2,1,1)\), \((2,2,1,1)\), \((3,3,0,0)\), \((1,1,4,2)\), \((1,1,3,2)\), \((1,1,2,2)\), \((2,2,1,1)\), \((2,2,1,1)\), \((3,3,0,0)\), they will become \((0,0,6,3)\), \((0,0,5,3)\), \((0,0,4,3)\), \((0,0,3,3)\), \((1,1,4,2)\), \((1,1,3,2)\), \((1,1,2,2)\), \((2,2,1,1)\), \((2,2,1,1)\), \((3,3,0,0)\), according the analysis of situation 3 in scenario 1. For the other states, the probability transitions will happen as before. This process continues until all probabilities are calculated.

References
