Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

DESIGNING HIGHWAY ACCESS CONTROL SYSTEM USING MULTI-CLASS M/G/C/C STATE DEPENDENT QUEUEING MODEL AND CROSS-ENTROPY METHOD

Yifan Wang
Daniel Kim
Seong-Hee Kim

H. Milton Stewart School of
Industrial and Systems Engineering
Georgia Institute of Technology
765 Ferst Dr., NW
Atlanta, GA 30332-0205, USA

Haengju Lee

Department of Information
and Communication Engineering
Daegu Gyeongbuk Institute of
Science and Technology
333 Techno Jungang-daero
Daegu, 42988, REPUBLIC OF KOREA

ABSTRACT

In this paper, we consider a futuristic scenario where there exists a special lane in a segment of a highway; vehicles which wish to use the lane must send access requests ahead of time; and only the vehicles whose requests are accepted can use the lane. While some existing express lanes try to control traffic by dynamic pricing on the usage of the lanes based on consumer demand, we control traffic by accepting or rejecting access requests to the lane. Vehicle classes are defined by vehicle sizes and carry a different number of passengers. Our goal is to find an optimal allocation of the lane capacity among different vehicle classes which maximizes the long-run average passenger throughput. We use a multi-class M/G/C/C state dependent model to calculate the long-run average passenger throughput for a given allocation and use a cross-entropy method to find an optimal allocation among a large number of possible allocations. We briefly discuss how this problem formulation is applicable to general resource sharing problems and then discuss how to dynamically control acceptance/rejection of access requests to further enhance the real-time efficiency of our system.

1. INTRODUCTION

The highway is an essential part of the transportation system. It not only serves as a faster option for long distance traveling but also affects the whole traffic condition for metropolitan areas. Especially for those cities with major interstate highways going through the metro area, like Chicago and Atlanta, highway traffic is particularly critical. In these cities, congestion occurs regularly during rush hours as the demand increases, and it significantly undermines the efficiency of the highway and thus that of the whole road system in the city. The conventional way to solve the higher traffic demand problem is to expand the road system. Besides its expensive cost, this idea is often not even implementable due to geological reasons. A more innovative way is to regulate the traffic demand to ensure the traffic performance.

Ravi et al. (2007) present the idea of using dynamic pricing as a control mechanism for the high-priority lane. Theoretically, by varying the price of using the priority lane, the demand could be controlled to avoid congestion. For example, Peach Pass on I-85 in Georgia varies its rates between \$0.01 per mile to a flat \$13.95 for any length trip, and the price at a specific location and point in time is determined by consumer demand. In this paper, we consider a different approach, namely a highway access control system: there exists a special lane in a segment of a highway; vehicles which wish to use the lane must send access requests ahead of time; and only the vehicles which receive approvals can use the lane. This is a futuristic scenario, but with the proposal of self-driving vehicles and the wide use of wireless networks,

traffic surveillance and control could be done in real time and this idea could be implementable and effective in enhancing highway traffic.

The idea of a traffic access control system is not new and has been proposed in the literature. It is first introduced from the airway transportation system by Akahane and Kuwahara (1996) for the purpose of managing a travel demand by means of reservation. Later Wong (1997) proposes a booking procedure to control an individual vehicle's departure time in advance as a method of governing road traffic. The idea of traffic reservation or access control may take some time to implement, but the results tend to be promising according to Wong (1997). Feijter, Evers, and Lodewijks (2004) also advocate the advantages of the reservation system by explaining that a reservation system improves the reliability and the efficiency of the system. Zhao et al. (2010) provide another pioneering example using cordon-based control for congestion mitigation in a downtown area. A real-time module is used to make real-time operational decisions whether a vehicle should be permitted or declined to enter the control area. By stimulating users to reserve in advance, the system would make sure that the total number of vehicles always stays under the determined capacity, which would improve the efficiency and the traffic conditions in the controlled area. Liu et al. (2013) extend the reservation model by Zhao et al. (2010) to a highway system. They propose a token-based reservation system and a real-time scheduling algorithm to monitor the traffic condition and maintain the efficiency of the high priority lane. The token-based reservation is shown to increase the control accuracy. However, it requires a significant effort to monitor a large number of tokens, which makes it difficult to implement in practice.

For a highway system, Jain and Smith (1997) present an M/G/C/C state dependent queueing models for modeling the traffic flow where M/G/C/C represents a queueing model with (i) a Poisson arrival process, (ii) service times whose distribution follows any general distribution. (iii) C number of parallel servers and (iv) C number of total system capacity. So a new customer is rejected if the queueing system has C existing customers in the system at the time of its arrival. Jain and Smith (1997) consider a linearly or exponentially decreasing traffic flow speed model as the number of vehicles increases in a segment of highway and compare the analytical and simulation results with the actual traffic data. Furthermore, they provide an analytical model to calculate the rejection probability of a new arriving customer. This model can be used to calculate the performance of a road traffic system such as a rejection probability and long-run average throughput for a given lane capacity where a long-run average is defined as an average over an infinite time horizon.

If all vehicles are equal in size, the maximum capacity of a certain segment of a highway is usually set to the number of vehicles at which the traffic flow speed becomes zero. As the speed tends to decrease as the number of vehicles increases, using up the maximum capacity is not necessarily better especially if the goal is to maximize the long-run average throughput. Instead, the overall system performance might be better when the number of vehicles is forced to stay lower than the maximum capacity. Indeed, Greenshields et al. (1935) show that if a traffic flow speed model is linear and the incoming vehicle rate is infinity, then the maximum throughput is achieved when the maximum number of vehicles allowed to be entered in highway is limited to half of the maximum capacity. In this paper, we consider one lane in a segment of highway and design the access control system for the lane to maximize the long-run average passenger throughput. There are multiple classes of vehicles based on their sizes. We extend an M/G/C/C state dependent model from Jain and Smith (1997) to a multi-class M/G/C/C state dependent model, which enables us to calculate the long-run average throughput for a given space allocation among classes. When the service rates or times of a class only depend on the state of the class but independent of other classes' states, then a closed-form solution for the long-run average throughput can be derived as in Example 2 of Zachary (2007). However, when service rates depend on the state of the entire system including other classes' states as in our problem, a closed-form solution does not exist but a set of balance equations need to be solved for each possible allocation. The total number of possible allocations is usually large and balance equations are a large set of linear equations. Therefore it takes time to set up balance equations, calculate the throughput for each possible allocation, and find the best optimal allocation by the exhaustive approach. Instead, we

employ the cross-entropy method which has been shown to be useful for optimization when objective function values can be obtained only through a black-box simulation or calculation. Rubinstein (1997) introduces the cross-entropy method for estimating a rare event probability and Rubinstein (1999) shows that the cross-entropy can be used for (approximately) solving complicated optimization problems. The cross-entropy method is successfully applied to many deterministic and stochastic optimization problems including Alon et al. (2005) which apply the cross-entropy method to a buffer allocation problem in a production line. Note that the value of the objective function in our problem is deterministic in a sense that its exact value is calculated numerically while Alon et al. (2005) estimate their objective function value (i.e., throughput of a production line with a limited buffer space) by stochastic simulation. There are other types of methods which can be used for black-box optimization. However, the goal of this paper is to design a highway access control system rather than compare the performances of different optimization techniques. Therefore comparing different types of optimization techniques is out of scope in this paper.

Our problem can be viewed as a general case of resource sharing in a sense that limited resources are shared by customers from various classes and customers are rejected when there are no available resources. Each customer requires a number of resources simultaneously; the number of resources needed for each class is different; processing times are determined by the total current workload; and the utility or value of a completed job from each class is different to the decision maker. That is, one class is more valuable than the other class to the decision maker such as first-class passengers and economy-class passengers in the airline industry. This problem occurs in many other fields such as internet-based gaming service and online music streaming service, to which our method can be applicable.

The paper is organized as follows. Section 2 defines our problem and introduces notations. Section 3 presents how to find an optimal allocation. Section 4 proposes a dynamic acceptance/rejection policy to enhance the real-time traffic performance. Experimental results are presented in Section 5, followed by conclusions in Section 6.

2. PROBLEM AND NOTATION

This section formulates our problem and defines notations.

Consider one lane in a segment of a freeway of length L in which a vehicle is allowed to enter only when an access request to use the lane is accepted. Let R represent the number of vehicle classes. A vehicle in class r for $r = 1,\ldots,R$ has m_r as the expected number of passengers and s_r as the relative vehicle size to the vehicle size of class 1. We assume that class 1 has the smallest vehicle size and set its size to one.

Let A_r represent the maximum number of vehicles from class r that the access control system can have simultaneously on the lane at any point in time. $(s_1A_1,\ldots,s_RA_R)$ represent the assigned spaces to each class. The access request arrival rate from class r (during a certain time interval such as rush hours or non rush hours) is denoted by λ_r and it is assumed to follow a Poisson arrival process. The long-run rejection probability for class r is denoted by q_r. The long-run average vehicle throughput of class r is $(1-q_r)\lambda_r$ and the long-run average passenger throughput of class r is $m_r(1-q_r)\lambda_r$. Then the overall long-run average passenger throughput is $\sum_{r=1}^{R} m_r \cdot (1-q_r)\lambda_r$. Further, let $N_r(t)$ represent the number of vehicles of class r at time t. Then the total spaces occupied at time t is $N_T(t) = \sum_{r=1}^{R} s_r N_r(t)$. It is assumed that the traffic flow speed at time t is a non-increasing function of $N_T(t)$ and is denoted by $V(N_T(t))$ to represent its relationship to the total spaces taken at time t. We use $C_{\max}$ to denote the number of vehicles of class 1 at which the traffic flow speed becomes zero. Table 1 provides a summary of our notations.

Our problem is formulated as follows:

$$\begin{array}{ll} \max_{A_1,\ldots,A_R} & \sum_{r=1}^{R} m_r(1-q_r)\lambda_r \\ \text{subject to} & N_r(t) \le A_r, \quad r = 1,\ldots,R; \\ & \sum_{r=1}^{R} s_r A_r < C_{\max}; \\ & A_r \in \{0,\ldots,C_{\max}-1\}, \quad r = 1,\ldots,R. \end{array} \tag{1}$$

Table 1: Notation.

Notation	Definition
R	the number of vehicle classes (class 1 has the smallest vehicle size)
r	class index
L	length of a segment
$C_{\max}$	maximum capacity of the access control lane in the units of the number of vehicles of class 1 (thus an integer)
A_r	maximum number of vehicles from class r that the access control system can have at any point in time ($s_r A_r$ spaces are exclusively assigned to class r)
C_T	maximum number of total spaces the access control system can occupy at any point in time (all classes share C_T spaces together)
m_r	the expected number of passengers in a vehicle of class r
s_r	the relative vehicle size in class r compared to the size of vehicles in class 1 ($s_1 = 1$)
$s_r A_r$	spaces allocated to class r (not necessarily an integer)
λ_r	the arrival rate of access requests from class r
$N_r(t)$	number of vehicles in class r in the lane at time t
$N_T(t)$	total spaces occupied at time t, i.e., $\sum_{r=1}^{R} s_r N_r(t)$
$V(N_T(t))$	traffic flow speed at time t
q_r	long-run rejection probability of class r

The above formulation assumes that class r has space $s_r A_r$ exclusively dedicated to itself and a vehicle from other classes is not allowed to take the space. One may want to control access requests simply by one pooled space $C_T < C_{\max}$ and consider the following problem formulation:

$$\begin{array}{ll} \max_{C_T} & \sum_{r=1}^{R} m_r(1-q_r)\lambda_r \\ \text{subject to} & \sum_{r=1}^{R} s_r N_r(t) \le C_T; \\ & C_T \in \{0,\ldots,C_{\max}-1\}. \end{array} \tag{2}$$

In a usual optimization of a limited resource allocation, an optimal solution is often to use up all available resource. However, in a highway traffic system, the optimal solution never uses up $C_{\max}$ but keeps total spaces used by all classes below $C_{\max}$ because $V(N_T(t))$ is non-increasing function of $N_T(t)$ and $V(C_{\max}) = 0$. In this paper, we interpret m_r as the expected number of passengers in a vehicle of class r but depending on a problem, it could be interpreted as something else such as prices earned when a request from class r is accepted. For example, in an internet-based game site with a limited capacity, m_r may represent access fees paid by user class r. Then an optimal allocation of the limited capacity among the classes could be found by solving (1) or (2) to maximize the long-run average total collected fees.

Problem (1) finds a static allocation $(s_1 A_1, \ldots, s_r A_R)$ where $s_r A_r$ is the spaces assigned exclusively for class r. If class r uses up all spaces $s_r A_r$ but there are no vehicles of other classes on the lane, then a new request from class r is rejected although there are spaces available in the lane. In contrast, Problem (2) finds a pooled space C_T shared by all vehicle classes. In this case, a request from class r can be accepted as long as there are spaces available in the lane. We find optimal solutions of both problem formulations and compare them. For Problem (1), when a class has open spaces, it may be desirable to allow some spaces to be used for other classes. In Section 4 we implement this static allocation with a dynamic control policy which accepts a request from class r even when $N_r(t) = A_r$ if allowing the vehicle to use s_r spaces from $s_{r'} A_{r'}$ is unlikely to interrupt future acceptance/rejection decisions for class r'.

Throughout the paper, we assume that there are only two classes and thus $R = 2$ for simplicity.

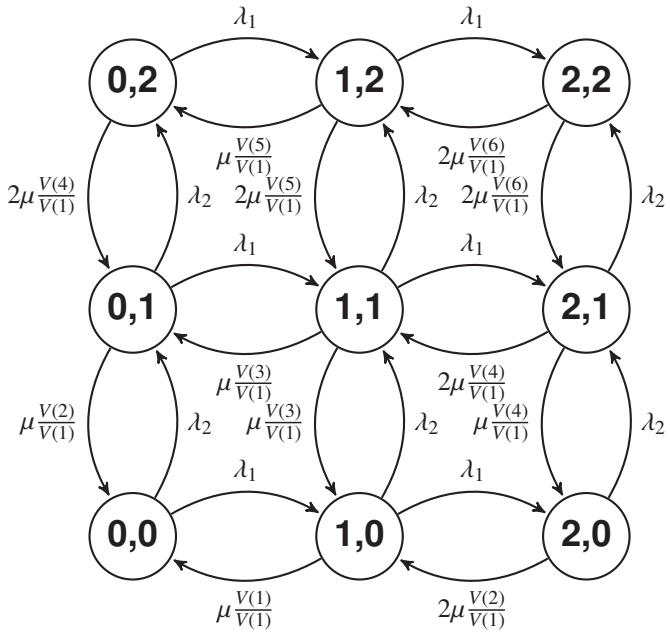

Figure 1: Rate diagram when $(s_1, s_2) = (1,2)$ and $(A_1, A_2) = (2,2)$ where each state (numbers in a circle) represents the number of vehicles in each class at time t, $(N_1(t), N_2(t))$.

3. STATIC ALLOCATION

In this section, we present a multi-class M/G/C/C state dependent queueing model to calculate the long-run average passenger throughput for a given allocation and then a cross-entropy method for optimization search over a large number of possible allocations.

3.1 Multi-Class M/G/C/C State Dependent Queueing Model

We describe a multi-class M/G/C/C state dependent queueing model for a specific allocation (A_1, A_2).

The mean travel time for the access control lane when there is only one car of class 1 is denoted by $\mathcal{L}$ and is set to $\mathcal{L} = L/V(1)$. Thus the mean travel time in state (i, j) is $L/V(s_1 i + s_2 j) = \mathcal{L}\frac{V(1)}{V(s_1 i + s_2 j)}$. In terms of service rates, the service rate for state (i, j) is $\mu\frac{V(s_1 i + s_2 j)}{V(1)}$ where μ is set to $1/\mathcal{L} = V(1)/L$. Then for a given allocation (A_1, A_2) from Problem (1), transition rates from state (i, j) to (i', j') are as follows:

$$\text{transition rate from } (i,j) \text{ to } (i',j') = \begin{cases} \lambda_1, & \text{if } (i',j') = (i+1,j) \text{ for } i = 0,1,\ldots,A_1-1; \\ \lambda_2, & \text{if } (i',j') = (i,j+1) \text{ for } j = 0,1,\ldots,A_2-1; \\ i \cdot \mu\frac{V(s_1 i + s_2 j)}{V(1)}, & \text{if } (i',j') = (i-1,j) \text{ for } i = 1,\ldots,A_1; \\ j \cdot \mu\frac{V(s_1 i + s_2 j)}{V(1)}, & \text{if } (i',j') = (i,j-1) \text{ for } j = 1,\ldots,A_2; \\ 0, & \text{otherwise.} \end{cases}$$

For allocation C_T from Problem (2), transition rates are the same as above except for the values of lower and upper bounds of i and j. Figures 1 and 2 show rate diagrams for $(A_1, A_2) = (2,2)$ and $C_T = 6$, respectively, when $(s_1, s_2) = (1,2)$ and $C_{\max}$ is a number greater than or equal to 6.

Kobayashi and Mark (2008) call multi-class M/G/C/C state dependent queues as a generalized Erlang loss queue and captures the multi-class nature of the generalized Erlang loss queue. They show that multi-class M/G/C/C state dependent queues are insensitive to distribution types if the classes of vehicles all arrive as independent Poisson arrival rates and there are no priorities. Their model is similar to our problem

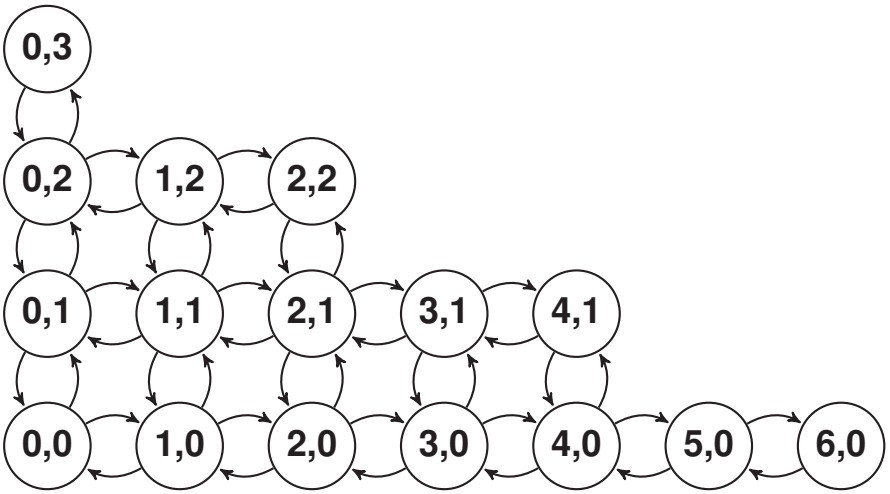

Figure 2: Rate diagram when $(s_1, s_2) = (1, 2)$ and $C_T = 6$ where each state (numbers in a circle) represents the number of vehicles in each class at time t, $(N_1(t), N_2(t))$.

formulation (2) except that service rates of class r are determined by $s_r N_r(t)$ rather than total workload $\sum_{r=1}^{R} s_r N_r(t)$. Zachary (2007) considers a more generalized case where service rates are determined by the total workload and shows that insensitivity in stochastic networks to service distributions still holds. Therefore, we calculate stationary distribution $\boldsymbol{\pi} = [\pi_{i,j}]$ for Problems (1) and (2) by setting up corresponding multi-class M/M/C/C state dependent queues and solving balance equations. Then the long-run rejection probabilities are

$$q_1 = \sum_{j=0}^{A_2} \pi_{A_1,j} \quad \text{and} \quad q_2 = \sum_{i=0}^{A_1} \pi_{i,A_2}$$

for Problem (1) and

$$q_1 = \sum_{(i,j):s_1 i + s_2 j = C_T} \pi_{i,j} \quad \text{and} \quad q_2 = \sum_{(i,j):0 \le C_T - (s_1 i + s_2 j) < s_2} \pi_{i,j}$$

for Problem (2). Then we can calculate the long-run passenger throughput for a given allocation.

3.2 Cross Entropy Method

The number of possible allocations for Problem (2) is $C_{\max}$. Thus the number of possible allocations linearly increases as $C_{\max}$ increases. This means that as long as $C_{\max}$ is not very large, the optimal allocation can be searched by the exhaustive method where $\boldsymbol{\pi}$ is calculated for every possible allocation.

On the other hand, in Problem (1), any (A_1, A_2) such that $s_1 A_1 + s_2 A_2 < C_{\max}$ can be a potential allocation and the number of pairs can be large. This problem is more pronouncing when $C_{\max}$ or the number of classes increases. One may find the best optimal allocation by setting up balance equations and calculating $\boldsymbol{\pi}$ for each $\boldsymbol{A} = (A_1, A_2)$ for every possible allocation, but this exhaustive method is not desirable due to computation time. Instead, we use a cross-entropy method known to be efficient for quickly searching for the best solution when the search space is large and evaluation of the objective function takes time. The main idea of the cross-entropy method is to convert a deterministic optimization problem to a rare-event probability estimation problem. Initially, a 'flat' sampling distribution is used for generating the n number of possible solutions. Then each solution's performance is calculated and the sampling distribution is updated by giving higher probabilities to solutions with good performances and lower probabilities to solutions with bad performances. These steps are repeated until the sampling distribution converges and a solution

with the largest probability is returned as the best solution. A full description of our cross-entropy method is described next.

Initialization: Pick a number of sampled allocations n, an integer for a stopping condition c, and a weight for updating a sampling distribution $0 < \alpha < 1$. Set a value ρ, a quantile probability. Set iteration counter $k = 0$ and the maximum possible values for each class M_1 and M_2 are $M_1 = C_{\max} - 1$ and $M_2 = \left\lfloor \frac{C_{\max}-1}{s_2} \right\rfloor$ where $\lfloor x \rfloor$ returns the greatest integer less than or equal to x.
The initial sampling distribution $\mathbb{P}^{(0)}$ is a 2 by (M_1+1) matrix of $[P_{rj}^{(0)}]$ such that

$$P_{rj}^{(0)} = \begin{cases} \frac{1}{M_1+1}, & r = 0, j = 0, \ldots, M_1; \\ \frac{1}{M_2+1}, & r = 1, j = 0, \ldots, M_2; \\ 0, & \text{otherwise}, \end{cases}$$

Sample Allocations: Sample n number of possible allocations $\boldsymbol{A}^{(i)} = (A_1^{(i)}, A_2^{(i)})$ for $i = 1, \ldots, n$ using $\mathbb{P} = \mathbb{P}^k$ as follows:

1. Generate a random permutation (w_1, w_2) of $\{1, 2\}$.
2. Sample $A_{w_1}^{(i)} \in \{0, 1, \ldots, M_{w_1}\}$ with pmf $(P_{w_1,0}, \ldots, P_{w_1,M_{w_1}})$.
3. Set $U = \left\lfloor \frac{C_{\max} - s_{w_1} A_{w_1}^{(i)}}{s_{w_2}} \right\rfloor$.
4. Sample $A_{w_2}^{(i)} \in \{0, \ldots, U\}$ with pmf $\left(\frac{P_{w_2,0}}{\sum_{i=0}^{U} P_{w_2,i}}, \ldots, \frac{P_{w_2,U}}{\sum_{i=0}^{U} P_{w_2,i}}, 0, \ldots, 0 \right)$.

Performance Calculation: Calculate $\boldsymbol{\pi}$ and the long-run average passenger throughput $\mathrm{TH}(\boldsymbol{A}^{(i)})$ for $i = 1, \ldots, n$ by solving balance equations and order these from largest to smallest, $\mathrm{TH}_1 \geq \cdots \geq \mathrm{TH}_n$. Define $\gamma = \mathrm{TH}_{\lfloor \rho n \rfloor}$.

Update: Using the same sample, update $\mathbb{P}^{(k+1)} = [P_{r,j}^{(k+1)}]$ as

$$P_{r,j}^{(k+1)} = \alpha \frac{\sum_{i=1}^{n} \mathbb{I}\left\{\mathrm{TH}(\boldsymbol{A}^{(i)}) \geq \gamma\right\} \cdot \mathbb{I}\left\{A_r^{(i)} = j\right\}}{\sum_{i=1}^{n} \mathbb{I}\left\{\mathrm{TH}(\boldsymbol{A}^{(i)}) \geq \gamma\right\}} + (1-\alpha) P_{r,j}^{(k)},$$

where $\mathbb{I}(A)$ is an indicator function of event A.
Set $k \leftarrow k+1$ and find ξ_r the index of the maximal element of the rth row of $\mathbb{P}^{(k)}$ for $r = 0, 1$. Return (ξ_1, ξ_2) as the current best allocation.

Stopping Rule: Stop if (ξ_1, ξ_2) has not changed for c consecutive times.

Note that **Sample Allocations** and **Update** steps are from Alon et al. (2005). Also, Alon et al. (2005) recommend that $0.7 \leq \alpha \leq 0.9$, $0.05 \leq \rho \leq 0.2$ and $c = 5$. We use $\alpha = 0.8$, $\rho = 0.2$ and $c = 5$ in this paper.

4. DYNAMIC CONTROL POLICY

Solutions to Problem (1) provide dedicated allocations in a sense that, for example, no access request from class 1 would be accepted when $(N_1(t), N_2(t)) = (A_1, 0)$ although there are $s_2 A_2$ spaces available. To further enhance space utilization, we dynamically control acceptance of requests depending on the current state $(N_1(t), N_2(t))$. The idea is simple. When there is a new request from class i whose spaces are full, we check the availability of spaces from class $j \neq i$. If spaces are available, then the probability that the acceptance of this new request of class i interferes future request acceptance of class j is computed. If the probability is low, then the new request is accepted to spaces assigned for class j. Otherwise, it is rejected.

Suppose that the current state is $(N_1(t), N_2(t))$ and there is a request from class i. Currently there is no space and thus $N_i(t) = A_i$. Category $j \neq i$ has $N_j(t)$ vehicles, occupying $s_j N_j(t)$ spaces. If there is no space to accept the new request (i.e., $s_j N_j(t) + s_i > s_j A_j$), then the new request is rejected. Otherwise, calculate the probability that the new request finishes its trip in the access control segment before it causes a rejection of a new request from class j. The trip finish time of the new request, denoted by end_i, is $end_i = L/V(N_T(t) + s_i)$, which is a constant. Category j currently has spaces for additional $(A_j - N_j(t))$ vehicles from class j. As the request arrivals are assumed to follow a Poisson process, the inter-arrival times are exponentially distributed with rate λ_j. The elapsed time of the $(A_j - N_j(t))$th new request from class j from the current clock t, denoted by S_j, is Erlang distributed with parameters $A_j - N_j(t)$ and λ_j. So if the $(A_j - N_j(t))$th new request from class j arrives before $L/V(N_T(t) + s_i)$, which corresponds to the event $S_j < \text{end}_i$, then the acceptance of the new request of class i interferes future acceptance and rejection decision for class j. We accept the new request and let it use spaces from class j if the probability that $S_j < \text{end}_i$ is smaller than some threshold $0 < \epsilon < 1$.

A formal description of the dynamic acceptance policy is given as follows:

Given: The current state is $(N_1(t), N_2(t))$. A request from class i arrives and currently the capacity is full as $N_i(t) = A_i$. Set a small constant $0 < \epsilon < 1$.

Initial Check: For $j \neq i$, if $s_j N_j(t) + s_i \leq s_j A_j$, calculate $p_j^a = \Pr(S_j < \text{end}_i)$ where S_j is an Erlang random variable with parameters $A_j - N_j(t)$ and λ_j and end_i is $L/V(N_T(t) + s_i)$; else, set $p_j^a = 1$.

Decision: Reject the new request if $p_j^a > \epsilon$ for all j. Otherwise, accept the new request and assign spaces from class j with smallest p_j^a. If the request from class i is accepted, set $N_j(t) \leftarrow N_j(t) + \frac{s_i}{s_j}$.

This dynamic decision is expected to increase space utilization for an allocation in Problem (1).

5. NUMERICAL RESULTS

In this section, we present analytical and simulation results using some traffic data.

5.1 Configurations

We consider two classes of vehicles. The overall arrival rate is denoted by λ with the proportion of traffic coming from each class (p_1, p_2). Then arrival rate of class r, λ_r, is set to λp_r. The overall arrival rate λ is set to $\lambda = 3960$ vehicles per hour for most settings but varies from 3960 to 7200 in Section 5.2. Two sets of (p_1, p_2) are tested: $(0.8, 0.2)$ and $(0.5, 0.5)$. The size of vehicles in each class s_r is assumed to be $(1, 2)$ and the expected number of passengers carried in a single vehicle from class r is assumed to be $(1, 1.5)$.

For traffic flow speed models, we consider linear and exponential models modified from Jain and Smith (1997). Suppose that there is only one class with size 1. A linear model assumes that $V(N_T(t))$ is calculated as $V(N_T(t)) = V(1) - \frac{V(1)}{C_{\max}} N_T(t)$, where $V(1)$ is referred to the free flow speed for a lone occupant and $C_{\max}$ is the jam density.

An exponential model represents traffic flow speed as $V(N_T(t)) = V(1) \exp\left[-\left(\frac{N_T(t)-1}{\beta}\right)^{-\phi}\right]$, where

$$\phi = \ln \frac{\ln(V(N_a)/V(1))}{\ln(V(N_b)/V(1))} \Big/ \ln \frac{(N_a - 1)}{(N_b - 1)}, \qquad \beta = \frac{N_a - 1}{[\ln(V(1)/V(N_a))]^{1/\phi}} = \frac{N_b - 1}{[\ln(V(1)/V(N_b))]^{1/\phi}},$$

and $(N_a, V(N_a))$ and $(N_b, V(N_b))$ are two points of a number of vehicles and its corresponding traffic flow speed which are selected from the empirical data to define traffic flow model.

5.2 Optimal Allocation with Various Arrival Rates for Single Class

In this section, we consider a single class with size 1 (i.e., $R = 1$ and $s_1 = 1$) and show how the optimal space allocation changes as λ changes when a linear traffic flow model with $V(1) = 75$ and $C_{\max} = 220$ is

considered. Arrival rate λ varies from 3960 vehicles per hour to 7000 vehicles per hour. Figure 3 shows rejection probabilities of vehicles at allocation $A_1 = 0, 1, \ldots, C_{\max} - 1$ for various λ. For each λ, an allocation with the smallest rejection probability produces the largest long-run average throughput and thus should be an optimal allocation. When $\lambda = 3960$, the optimal allocation is 192. When λ increases to 7000, the optimal allocation changes to 130. This demonstrates that using all possible spaces $C_{\max}$ is not necessarily good when service gets slower as there are more customers in the system. Also, note that as λ increases, the optimal allocation converges to a half of the maximum capacity $C_{\max}$ for the linear traffic flow model, which matches the results of Greenshields et al. (1935).

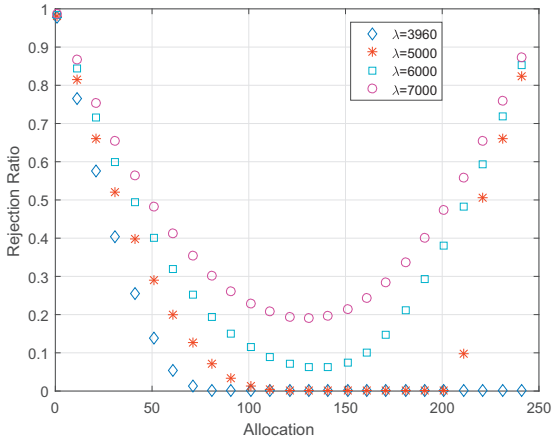

Figure 3: Rejection probability for allocation A_1 (or C_T) for various arrival rates λ.

5.3 Static Optimal Allocations for Two Classes

Now we consider two classes. Remind that $(s_1, s_2) = (1, 2)$ and $(m_1, m_2) = (1, 1.5)$. The overall arrival rate is $\lambda = 3960$ vehicles per hour with $(p_1, p_2) = (0.8, 0.2)$ or $(p_1, p_2) = (0.5, 0.5)$. A linear traffic flow model is used with $V(1) = 75$ miles per hour.

To demonstrate the efficiency of the cross-entropy (CE) method, we solve Problem (1) using both the exhaustive method and the CE method for $C_{\max} = 110, 160$ and 220 with $(p_1, p_2) = (0.5, 0.5)$. The CE method uses parameters $\alpha = 0.8, \rho = 0.2$ and $n = 400$. In all cases we tested, the CE method stops after 13 iterations. Both methods return the same allocation as the best and the CE method evaluates fewer allocations than the exhaustive method until it stops as shown in Table 2.

Table 2: Optimal solutions returned by the exhaustive method and the cross-entropy method.

$C_{\max}$	Optimal allocation (A_1, A_2)	Number of solutions searched by the exhaustive method	Number of solutions evaluated by the CE method
110	(66, 1)	3136	689
160	(76, 13)	6561	817
220	(83, 29)	12321	928

Table 3 compares optimal solutions of Problem (1) by the CE method and Problem (2) by the exhaustive method when $C_{\max} = 220$ and (p_1, p_2) is either (0.8,0.2) or (0.5,0.5). For both cases of (p_1, p_2), dedicated space allocations result in higher passenger throughput. However, we found that there are cases that the pooled allocation performs better than the dedicated allocation. For example, when (m_1, m_2) is $(1, 2)$ instead of $(1, 1.5)$ with $V(1) = 75$ and $(p_1, p_2) = (0.8, 0.2)$, the optimal pooled allocation is $C_T^* = 118$ with TH =

4139 and $(q_1, q_2) = (9.92, 18.83)$ while the optimal dedicated allocation is $(A_1^*, A_2^*) = (82, 21)$ with TH = 4132 and $(q_1, q_2) = (11.11, 16.93)$.

Table 4 reports simulation results for the optimal dedicated allocations with the dynamic control when ϵ varies from 0 to 1. When $\epsilon = 0$, estimated rejection probabilities and throughput from simulation are very close to analytical results from Table 3. This is expected because the dynamic control with $\epsilon = 0$ is same as the pure dedicated space allocation policy. The dynamic control slightly increases throughput as ϵ increases, but throughput drops significantly when $\epsilon = 1$. One would think that the case of $\epsilon = 1$ should be same as the pooled allocation with $C_T = A_1^* + s_2 A_2^*$ but, in fact, they are different. For example, when there is only one space left for each class, the dynamic control with $\epsilon = 1$ would reject a request from class 2 as size s_2 is 2 whereas the pooled policy would accept the request because two spaces are available in total. Indeed, the analytical results of a pooled allocation with $C_T = 83 + 2(29) = 141$ and $(p_1, p_2) = (0.5, 0.5)$ show $(q_1, q_2) = (22.69, 39.95)$ and TH = 3314 while the dynamic control with $\epsilon = 1$ has $(q_1, q_2) = (19.86, 41.13)$ and TH = 3331 (i.e., lower rejection for class 1 but higher rejection for class 2).

The big difference between $\epsilon = 0.99$ and $\epsilon = 1$ occurs because p_j^a — the probability that the acceptance of a new request of class i to class j's spaces interferes future request acceptance of class j — becomes 1 during most of simulation time. When $(p_1, p_2) = (0.8, 0.2)$ and $\lambda = 3960$ vehicles per hour, the highway tends to be full with a few spaces left if there are any. When only $2 \sim 3$ spaces are left, the mean of the Erlang random variable S_j is $2.14 \sim 3.42$ seconds for class 1 and $9.1 \sim 13.65$ seconds for class 2, respectively. On the other hand, the trip finish-time of the new request, end_i, is around $100 \sim 120$ seconds. Hence, end_i is much larger than S_j, which implies that accepting a request of class i will definitely interrupt the future service of class j or $p_j^a = 1$.

Table 3: Optimal solutions (A_1^*, A_2^*) for Problem (1) and C_T^* for Problem (2) when $C_{\max} = 220$. Rejection probabilities are in % and THs are in number of passengers per hour.

(p_1, p_2)	Optimal Allocation	(q_1, q_2)	TH
$(0.8, 0.2)$	$(A_1^*, A_2^*) = (108, 13)$	$(0.99, 40.96)$	3838
	$C_T^* = 118$	$(9.92, 18.83)$	3818
$(0.5, 0.5)$	$(A_1^*, A_2^*) = (83, 29)$	$(0.13, 45.89)$	3585
	$C_T^* = 114$	$(19.59, 35.27)$	3515

Table 4: Simulation results for optimal solutions of Problem (1) with the dynamic control and 24 hours run-length. (Rejection probabilities are in % and THs are in number of passengers per hour.)

ϵ	$(p_1, p_2) = (0.8, 0.2), (A_1, A_2) = (108, 13)$		$(p_1, p_2) = (0.5, 0.5), (A_1, A_2) = (83, 29)$	
	(q_1, q_2)	TH	(q_1, q_2)	TH
0	(0.90, 41.01)	3838	(0.10, 45.85)	3587
0.5	(0.90, 40.98)	3839	(0.10, 45.77)	3589
0.9	(0.90, 40.86)	3840	(0.11, 45.64)	3593
0.99	(0.93, 40.53)	3843	(0.15, 45.56)	3594
1	(10.95, 22.06)	3746	(19.86, 41.13)	3331

5.4 Various Traffic Flow Models for Two Classes

We find the optimal allocation under various traffic flow speed models for Problem (1): one linear model and three different exponential models with $V(1) = 75$, $\lambda = 3960$ and $(p_1, p_2) = (0.5, 0.5)$. These four traffic flow speed models are shown in Figure 4. Table 5 gives parameters for the four traffic flow speed models and returns optimal solutions along with their rejection probabilities and long-run average passenger throughput.

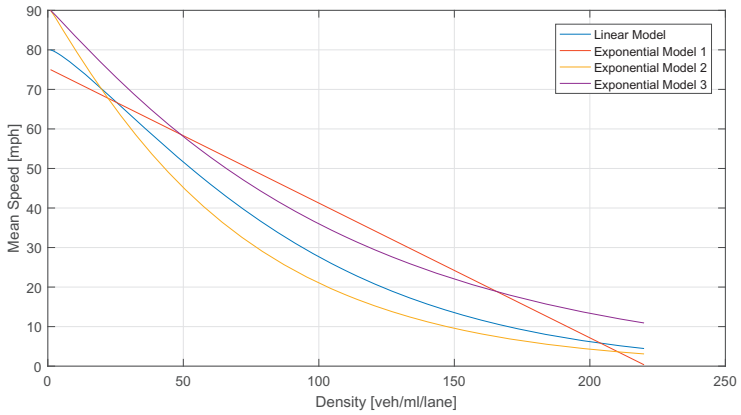

Figure 4: Different traffic flow models

Table 5: Optimal allocation (A_1^*, A_2^*) under various traffic flow models. Rejection probabilities are in % and THs are in number of passengers per hour.

Model	Parameters	(A_1^*, A_2^*)	(q_1, q_2)	TH
Linear	$V(1) = 75$, $C_{max} = 220$	(83, 29)	(0.13, 45.89)	3585
Exponential 1	$V(1) = 80$, $\phi = 1.26$, $\beta = 94.4$	(80, 12)	(0.79, 78.02)	2616
Exponential 2	$V(1) = 90$, $\phi = 1.06$, $\beta = 69.6$	(77, 5)	(1.97, 90.97)	2209
Exponential 3	$V(1) = 90$, $\phi = 1.05$, $\beta = 107.6$	(76, 35)	(13.02, 57.43)	2986

Space allocations are sensitive to speed models and more spaces tend to be assigned to class 1 because it takes fewer spaces than class 2 and thus contributes less to speed reduction compared to class 2. Also, we notice that the linear model produces higher throughput than exponential models because traffic flow speed decreases much slowly in the linear model than in exponential models.

6. CONCLUSIONS

In this paper, we present a freeway access control system which maximizes the long-run average passenger throughput. The work presented in this paper considers a simple case where there are only two classes and one lane in a single segment. Research is ongoing on extending it to more than 2 classes and a lane in multi-segments where part of vehicles from previous segments continue traveling to next segments. In addition, this paper assumes real-time requests and real-time control. It would be interesting to consider both real-time access requests and reservation requests ahead of time such as sending a request at 2 pm for traveling at 6 pm.

ACKNOWLEDGMENTS

We would like to thank J. MacGregor Smith and David Goldberg for their helpful comments. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2017R1D1A1B03030126) and Global Research Laboratory Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2013K1A1A2A02078326).

REFERENCES

Akahane, H., and M. Kuwahara. 1996. "A basic study on trip reservatipn systems for recreational trips on motorways". In *Proceedings of the Third World Congress on Intelligent Transportation Systems*.

Alon, G., D. Kroese, T. Raviv, and R. Rubinstein. 2005. "Application of the cross-entropy method to the buffer allocation problem in a simulation-based environment". *Annals of Operations Research* 134:137–151.

Feijter, R., J. Evers, and G. Lodewijks. 2004. "Improving travel time reliability by the use of trip booking". *IEEE Transactions on Intelligent Transportation Systems* 5:288–292.

Greenshields, B., J. Bibbins, W. Channing, and H. Miller. 1935. "A study of traffic capacity". In *Highway Research Board Proceedings*.

Jain, R., and J. Smith. 1997. "Modeling vehicular traffic flow using M/G/C/C state dependent queuing models". *Transportation Science* 31:324–336.

Kobayashi, H., and B. Mark. 2008. *System Modeling and Analysis: Foundations of System Performance Evaluation*. UK: Pearson.

Liu, K., E. Chan, V. Lee, K. Kapitanova, and S. Son. 2013. "Design and evaluation of token-based reservation for a road system". *Transportation Research Part C* 26:184–202.

Ravi, N., S. Smaldone, and M. Gerla. 2007. "Lane reservation for highways". In *Proceeding of the IEEE Intelligent Transportation System Conference*, 795–800. IEEE, Inc.

Rubinstein, R. 1997. "Optimization of computer simulation models with rare events". *European Journal of Operational Research* 99:89–112.

Rubinstein, R. 1999. "The cross-entropy method for combinatorial and continuous optimization". *Methodology and Computing in Applied Probability* 2:127–190.

Wong, J. 1997. "Basic concepts for a system for advance booking for highway use". *Transportation Policy* 4:109–114.

Zachary, S. 2007. "A note on insensitivity in stochastic networks". *Journal of Applied Probability* 44:238–248.

Zhao, Y., K. Triantis, D. Teodorovic, and P. Edara. 2010. "A travel demand management strategy: The downtown space reservation system". *European Journal of Operational Research* 205:584–594.

AUTHOR BIOGRAPHIES

YIFAN WANG is a first-year PhD student in H. Milton Stewart School of Industrial and Systems Engineering at Georgia Institute of Technology. He received his B.S. in Aerospace Engineering from Georgia Institute of Technology in 2016. His e-mail address is ywang832@gatech.edu.

DANIEL KIM is an undergraduate student in H. Milton Stewart School of Industrial and Systems Engineering at Georgia Institute of Technology. His email address is dkim608@gatech.edu.

SEONG-HEE KIM is a Coca-Cola Professor in H. Milton Stewart School of Industrial and Systems Engineering at Georgia Institute of Technology. She received her Ph.D. in Industrial Engineering and Management Sciences from Northwestern University in 2001. Her email address is skim@isye.gatech.edu.

HAENGJU LEE is a Research Professor in Department of Information and Communication Engineering at Daegu Gyeongbuk Institute of Science and Technology, South Korea. She received her Ph.D. in Industrial Engineering and Operations Research from Columbia University in 2005. Her email address is haengjulee@dgist.ac.kr.