8-2013

"Network-Theoretic" Queuing Delay Estimation in Theme Park Attractions

Ajay ARAVAMUDHAN

Archan MISRA
Singapore Management University, archanm@smu.edu.sg

Hoong Chuin LAU
Singapore Management University, hclau@smu.edu.sg

Citation

ARAVAMUDHAN, Ajay; MISRA, Archan; and LAU, Hoong Chuin. "Network-Theoretic" Queuing Delay Estimation in Theme Park Attractions. (2013). *IEEE Conference on Automation Science and Engineering (CASE), 17-20 August 2013*. 776-782.
Available at: https://ink.library.smu.edu.sg/sis_research/1924

“Network-Theoretic” Queuing Delay Estimation in Theme Park Attractions

Ajay Aravamudhan, Archan Misra, Hoong Chuin Lau

School of Information Systems, Singapore Management University
E-mail: {ajaysa,archam,hclau}@smu.edu.sg

***Abstract*— Queuing is a common phenomenon in theme parks which negatively affects visitor experience and revenue yields. There is thus a need for park operators to infer the real queuing delays without expensive investment in human effort or complex tracking infrastructure. In this paper, we depart from the classical queuing theory approach and provide a *data-driven* and online approach for estimating the time-varying queuing delays experienced at different attractions in a theme park. This work is novel in that it relies purely on empirical observations of the entry time of individual visitors at different attractions, and also accommodates the reality that visitors often perform other unobserved activities between moving from one attraction to the next. We solve the resulting inverse estimation problem via a modified Expectation Maximization (EM) algorithm. Experiments on data obtained from, and modeled after, a real theme park setting show that our approach converges to a fixed-point solution quite rapidly, and is fairly accurate in identifying the per-attraction mean queuing delay, with estimation errors of 7-8% for congested attractions.**

I. Introduction

Correctly and robustly estimating the queuing (or ’waiting’) time at different attractions of a theme park is a critical operational imperative, as it directly affects both visitor satisfaction and revenue yields. Knowledge of such queuing delays can be used, for example, to offer dynamic itinerary recommendations or to provide specially-timed passes, as a means of alleviating peak loads at popular attractions. For the most part, estimates of such likely delays have been computed *off-line* (e..g, see [1]), as part of capacity planning and policy formulation, by applying models of visitor movement that are typically computed based on aggregated surveys (e..g, [6]) or, more recently, based on empirical data collected via mobile devices (e.g., [3]).

In this paper, we instead propose and investigate a more *data-driven* and *online* estimation approach, where we try to infer the dynamically-varying queuing delays *actually experienced* at different attractions, based purely on empirical observations of the *entry time* of individual visitors at different attractions. This approach posits that the entry times serve as an *indirect indicator* of the queuing delays experienced, as the time between two successive entries will obviously be longer if the visitor spends a longer time waiting in an intervening queue. More specifically, this problem is based on the operational realities of a major theme park partner located in Singapore: the theme park provides each visitor, or some fraction of the overall set of visitors, with an electronic tag/card that he or she “taps” only when entering an attraction, but not when exiting the attraction. As a result of this “tapping” behavior, our observable dataset consists of a set of $Visit$ tuples of the form:

$$\langle Timestamp, CardID, AttrID \rangle$$

where $AttrID$ refers to the ID of a specific attraction, $CardID$ is the identity associated with the electronic tag (unique for each visitor) and $Timestamp$ denotes the time at which the visitor ‘taps’ the card (at some stage of “entering” the attraction). Table I shows a small sample of taps recorded for a set of three users.

Given such a corpus of $Visit$ tuples (from different visitors across all attractions), the problem of estimating the queuing delay at individual attractions is an ‘inverse-estimation’ problem. Fundamentally, the interval between successive ‘tap’ timestamps by an individual provides the total *transit time* between the corresponding attraction pair, and the inverse estimation problem then becomes one of identifying the fraction of this transit time that is spent waiting in one or more queues.

Such problems have been addressed previously in Internet tomography literature (e.g., [4], [8]). However, our problem is unique due to the fact, that human beings (unlike objects such as Internet packets or goods in a supply chain) do not directly proceed from one attraction to the next, but exercise “free will” to potentially engage in other unobserved activities, of widely varying duration, between attraction

Timestamp	CardID	AttrID
2011-10-21 19:29:42	304101984	1
2011-10-21 19:31:44	303930755	6
2011-10-21 19:34:51	304193107	6
2011-10-21 20:10:06	303930755	1
2011-10-21 20:14:15	304193107	1
2011-10-21 20:19:43	304101984	6
2011-10-21 21:01:47	304193107	5
2011-10-21 21:05:19	304101984	5
2011-10-21 21:27:32	304101984	3
2011-10-21 21:09:02	303930755	5
2011-10-21 22:05:39	303930755	2

TABLE I: Snapshot of visitor activity history

This research project is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

visits. In other words, the time between consecutive 'taps' consists of not only one or more usually-deterministic and well-known *attraction delays* (i.e., the time taken to complete a ride) and one or more unknown *queuing delays*, but also an unobserved *'sojourn delay'*. For example, one visitor might take a short restroom break between two attractions, while several others may collectively take a longer lunch break.

In this paper, we shall show that this inverse estimation problem can be modeled by a linear set of 'network of queues' equations, driven by unobservables that include the queuing delay and the 'random' *sojourn delay* between attractions. Moreover, the sojourn delay itself can be modeled as a *mixture* of distributions, each of which intuitively corresponds to a different type of activity that a visitor may undertake between attractions. We shall then derive and evaluate a modified Expectation Maximization (EM) algorithm to solve for the various unknowns of this 'network of queues' model. The EM approach is necessary, as opposed to more conventional MLE techniques, due to the underlying mixture model, whereby it is not observable if the visitor's inter-attraction time is a result of just the queuing delay or delays due to other intermediate activities.

Research Questions: As part of our investigation, we shall address the following research questions:

- How do we statistically model the 'random' sojourn delay (which differs for every individual and for every attraction pair) in our inverse estimation framework?
- Given the pairwise inter-attraction 'tap' times for different individuals, how do model the queuing delay estimation problem, while incorporating the unobservable 'sojourn delays'?
- How do we then statistically solve the resulting inverse estimation problem?

Key Contributions: We believe that this paper makes the following key contributions:

1) **'Network of Queues' Model:** Based on the real-life characteristics of theme park attractions, we develop a 'network of queues' model, where each attraction is modeled as a node that is preceded by a variable-delay queue, with the visitor 'tapping' action performed either before or after enduring the queuing delay.
2) **Bimodal mixture for Sojourn Delay:** We leverage on the empirically-observed *bimodal distribution* (see Figure 2) of the sojourn delay, aggregated across visitors. This reflects the intutition that most visitors transit between Attraction i and Attraction j either directly or after a reasonably long gap, and allows us to model the sojourn delay as a *mixture* of two Gamma-distributed random variables.
3) **Modified EM Algorithm:** To solve the resulting inverse estimation problem, we shall propose a modified Expectation Maximization (EM) algorithm. More specifically, unlike the basic EM algorithm, our modified 'M' step consists of an additional regression process to determine the best values (at present) of both the per-attraction queueing delays and the two Gamma-distributed sojourn variables. We shall show that this approach provides reasonably good estimates of the queuing delays for our experimental theme park layout.

II. Related Work

Most of the prior work related to queuing delays in theme parks focuses on the *planning phase*, and aims to understand the likelihood and impact of the delays that may result from different visitor movement and usage patterns. In the pioneering work in [1], Ahmadi developed a series of statistical movement models and evaluation criteria for ride management, short-term capacity planning and visitor flow management in a theme park environment, with a goal of understanding how changes in movement behavior affect the customer experience at different attractions. Motivated by a study at Universal Studios Hollywood (USH), [9] developed a flow management model that illustrates the impact of visitor flows on the retail store sales and suggests tuning the capacities and schedules of the major attractions so as to increase visitor flows to high-profit retail areas. More recently, the ability to capture individual-level preferences of visitors via their personal mobile devices has enabled the creation of more detailed person-specific movement models. For exmple, [3] recently proposed a data-driven approach, where the aggregate visit patterns to different attractions are based on underlying agent models, which have been built from individual-level movement data collected via mobile devices.

All of these prior works do not attempt to *dynamically* estimate the actual queuing delays, experienced during the daily, "real-time operation" of the park. One example of operational management of attraction delays is the FASTPASS system (whose distribution mechanism has been studied in [7]) used at Disneyland, which offers visitors special wait-free entry to specific attractions at a later stipulated time, under certain conditions. There appears to be little prior work in *online* estimation of queuing delays, based on observed timestamps of attraction visits.

As mentioned before, such online estimation of queuing delays has, however, been addressed in the Internet "network tomography" community, where end-to-end measurements of Internet traffic properties are used to infer "hidden" bottlenecks in the Internet core. More specifically, Coates et. al [4] survey the use of signal processing and statistical estimation techniques to uncover the available bandwidth and queuing delays of intermediate network links, based on end-to-end measurement of Internet traffic. Similarly, the problem of estimating the delays experienced by multicast packets and the traffic loads between origin-destination pairs was addressed in [8], which developed a maximum pseudo-likelihood estimation technique (a modified version of classical MLE). In general, the tomography approaches model the problem via a linear network of edges and buffers; this can be applied to our theme park-oriented formulation as well. However, as explained earlier, our problem's uniqueness lies in the need to accommodate an additional, unobserved amount of visitor-specific "sojourn delay" between attractions: we shall see that solving the resulting *mixture model* is not possible using the conventional MLE approaches, but requires a modified version of the Expectation Maximization (EM) [5] algorithm.

III. Empirical Dataset and Observations

Our techniques are developed and based on a real-world dataset (similar to the representation in Table I) obtained from our theme park partner. The data corresponds to a specially themed 5-day event organized during Halloween 2011, and contains a total of 28,878 individual 'tapping' actions, corresponding to a total set of 9,324 visitors. For this special event, the attractions were located fairly close to one another; hence, visitors transited between the attractions only on foot and along a clearly delineated path (transportation-related delays, due to bus or monorail schedules, are thus not relevant to this setting). Moreover, the transit time between any two attractions is itself relatively small (less than 5 minutes).

Figure 1 shows the schematic layout of the 6 theme park attractions associated with this special event. Each attraction is associated with a buffer, representing the physical queue that visitors experience before entering the event. Note that the event (and our model) must distinguish between two types of attractions:

- *pre-queuing*, where the visitor experiences the queuing delay *prior* to 'tapping' the card, and
- *post-queuing*, where the visitor 'taps' first and then experiences a possible queuing-related wait.

For our event, attractions 1-5 were of the pre-queuing type, while attraction 6 was the only one of the post-queuing variety. Figure 1 also shows a sample route (in dotted lines) possibly taken by an individual visitor–in the example, the visitor generates the 'tap' histories in the sequence $\{4,6,1,3\}$.

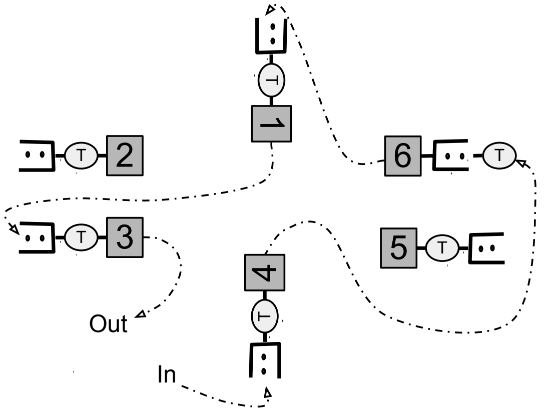

Fig. 1: Six attraction park with heterogenous queues

A. Bimodal sojourn times

From the observed $\langle Timestamp, CardID, AttrID\rangle$ tuples, we can construct a total inter-attraction delay table as shown in Table II. In general, for any two attractions i and j, we can find a set of inter-attraction delays, denoted by t_{ij}, of all people who tapped at i and tapped next at j.

	Delay (mins)	CardID
t_{15}	59, 47	303930755, 304193107
t_{16}	50	304101984
t_{52}	57	303930755
t_{53}	22	304101984
t_{61}	38, 39	303930755, 304193107
t_{65}	45	304101984

TABLE II: Inter-attraction time interval of visitors (corresponding to Table I)

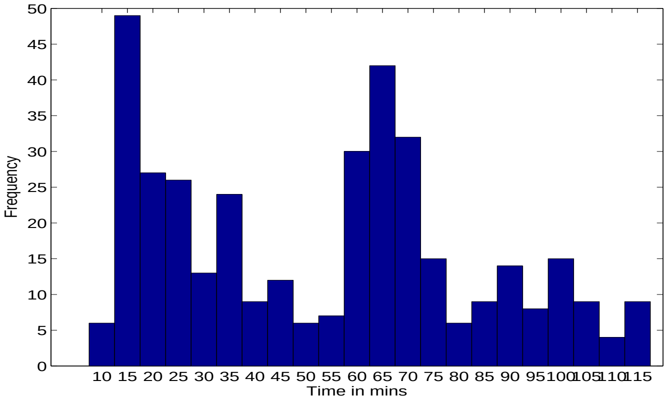

Fig. 2: Bimodal distribution of inter-attraction times.

One of our key hypotheses is that the inter-attraction delay is not just due to the queuing delays experienced prior to entering each individual attraction, but also due to the fact that visitors will possibly engage in additional unobserved activities *between* different attractions. To validate this hypothesis, Figure 2 plots the empirically observed histogram of the inter-attraction delays across all attraction pairs (and all days). Given that the actual travel time between attractions is less than 5 minutes, the delays suggest that almost all visitors experienced some degree of queuing delay during the event. There is also clearly a significantly *bimodal* nature to this delay. We believe that this can be directly attributed to the choices visitors make while moving between attractions: some visitors walk directly to the next attraction, while others may choose to take photographs or have a meal, for instance. This is reinforced by anecdotal observations that "typical" queuing (or wait) times for visitors was often around 40-60 minutes, suggesting that the large spread of the inter-attraction times was driven by significant other 'activity' undertaken by the visitors between attractions. Accordingly, in Section IV, we shall model the sojourn delay component, denoted by d, as a mixture of two random variables.

IV. Problem Description

We first describe our mathematical model of queuing and sojourn-induced delays and then explain how the observations in a multi-attraction theme park setting can be modeled as a network graph and represented by an adjacency matrix representation.

A. Mathematical Notation and Model

Table III shows the notations used in this paper. Visitors to the theme park with N attractions move from attraction to attraction until they leave the park. Each attraction i has a queue associated with it, and we denote the queuing delay experienced[1] in the queue at i by q_i. The *sojourn delays* are denoted by d, and since we assume a bimodal distribution, we denote the component distributions as d_1 and d_2 (with d_1 having the lower mean than d_2, and implicitly representing the case where the visitor moves directly from one attraction to the next). We assume that the sojourn delay d is independent of the attraction pair that the visitor is traveling between (the model can be generalized to consider pair-specific sojourn delays).

t_{ij} represents the set of observations of inter-attraction duration between i and j, each element of which is denoted by $t_{ij}^k, k \in \{1, \ldots, |t_{ij}|\}$, where $|t_{ij}|$ represents the number of observations of people tapping at attraction i followed by attraction j. $P_{ij}^k(l)$ represents the probability (not observable, but something that we must estimate) that the inter-attraction duration t_{ij}^k was caused by an individual while being subject to a sojourn delay caused by $d^l, l \in \{1, 2\}$. Intuitively, $P_{ij}^k(1)$ represents the probability that t_{ij}^k was generated by a visitor moving directly from attraction i to j, whereas $P_{ij}^k(2)$ is the corresponding probability that the visitor performed some other activities during the intervening period.

We follow earlier research on the stochastic Orienteering Problem which models the the inter-attraction duration via a Gamma-distributed random variable [2]. As an extension, the queue variables q_i and the sojourn delay variables d_l are also Gamma distributed, with k being the *shape* and θ the *scale* parameter. The mean of a Gamma distribution is defined as $k\theta$. For our formulation, we do not vary the scale parameter θ. This is both mathematically expedient (as it allows us to use the additive property of Gamma distributions) and also intuitively appealing (as the queuing and sojourn delays are typically roughly of the same magnitude–e.g., tens of minutes). We denote the mean of the queue delay variable q_i as α_i, and the mean of the sojourn delay variable d_i as δ_i. We assume that the shape parameter for *all* random variables is 1; therefore we can use the mean of the distribution and the shape variable interchangeably.[2]

B. Network-Graph Representation

Symbol	Description
N	Number of attractions in the park
q_i	Random variable representing the distribution of the queuing time at attraction $i, i \in \{1, \ldots, N\}$
$\tilde{d}$	Random variable representing human sojourn delay, mixture of component distributions d_l
d_l	Random variable representing the $j^{th}, j = \{1, 2\}$ component distribution of human sojourn delay
α_i	The estimated mean of the RV $q_i, i \in \{1, \ldots, N\}$
δ_l	The estimated mean of the RV $d_l, l = \{1, 2\}$
$\tilde{t_{ij}}$	Random variable representing guest movement from attraction i to j, $i, j \in \{1, \ldots, N\}$
t_{ij}	The *set* of observations of guest movement from attraction i to j, $i, j \in \{1, \ldots, N\}$
t_{ij}^k	The k^{th} observation of guest movement from attraction i to j, $i, j \in \{1, \ldots, N\}$
$\|t_{ij}\|$	The number of observations of guest movement from attraction i to j, $i, j \in \{1, \ldots, N\}$
$P_{ij}^k(l)$	The probability mass of the k^{th} observation of $\langle i, j \rangle$ movement being generated by the d_l (the human sojourn delay RV).
$\hat{P}_{ij}^k(l)$	The normalized probability that the k^{th} observation of $\langle i, j \rangle$ movement was caused by the use of the human delay RV d_l.

TABLE III: Symbols used

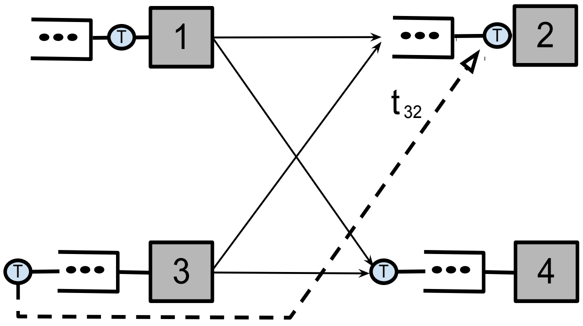

Fig. 3: A four attraction theme park with heterogeneous queues

We now represent the set of observations t_{ij} by a set of linear equations. To illustrate the generic process simply, we consider (see Figure 3) a small four-attraction theme park. In this example, attractions 1 & 2 are of the pre-queuing type (visitors experience queuing delays before they tap their pass), whereas 3 & 4 are post-queuing attractions. The figure shows a path from attraction 3, where visitors tap before queueing, to attraction 2, where visitors queue before tapping. The random variable, $\tilde{t}_{32}$, denoting the time taken between taps at attraction 3 and attraction 2 is then governed by the relationship:

$$\tilde{t}_{32} = q_3 + q_2 + \tilde{d} \tag{1}$$

where, as noted in Table III, q_i is the queue delay random variable and $\tilde{d}$ is the human sojourn delay random variable. In other words, each of the individual observations in the set t_{32} is a realization of the sum of the random variables q_3, q_2 and $\tilde{d}$.

Similarly, we can write the equations for the other r.v.s $\tilde{t}_{12}$, $\tilde{t}_{14}$, and $\tilde{t}_{34}$ in figure 3 as follows:

$$\tilde{t}_{12} = q_2 + \tilde{d} \tag{2}$$

$$\tilde{t}_{14} = \tilde{d} \tag{3}$$

$$\tilde{t}_{34} = q_3 + \tilde{d} \tag{4}$$

Proceeding in this fashion, we can then represent the relationships between the *random variables* in Figure 3 via

[1] In the rest of this paper, we assume that each attraction's queuing delay is a time-invariant a constant. In subsequent work, which we do not cover here due to space limitations, we have extended this model to capture the fact that the queuing delay $q_i(t)$ does, in fact, vary with time.

[2] Our modeling approach, in fact, works well for any distribution that shows additive properties, such as the Normal distribution. For the Normal, which we have also successfuly modeled in our work, we would need to estimate both the variance as well as the mean.

the following matrix representation:

$$\begin{pmatrix}0 & 1 & 0 & 0 & 1\\ 0 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0 & 1\\ 1 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0 & 1\\ 1 & 0 & 1 & 0 & 1\\ 0 & 1 & 1 & 0 & 1\\ 0 & 0 & 1 & 0 & 1\\ 1 & 0 & 0 & 1 & 1\\ 0 & 1 & 0 & 1 & 1\\ 0 & 0 & 0 & 1 & 1\end{pmatrix} \begin{pmatrix}q_1\\ q_2\\ q_3\\ q_4\\ \tilde{d}\end{pmatrix} = \begin{pmatrix}\tilde{t}_{12}\\ \tilde{t}_{13}\\ \tilde{t}_{14}\\ \tilde{t}_{21}\\ \tilde{t}_{23}\\ \tilde{t}_{24}\\ \tilde{t}_{31}\\ \tilde{t}_{32}\\ \tilde{t}_{34}\\ \tilde{t}_{41}\\ \tilde{t}_{42}\\ \tilde{t}_{43}\end{pmatrix} \tag{5}$$

Note that this matrix captures the relations between the different *random variables* – our estimation problem is one of estimating various parameters (e.g., the mean) of the different r.v.s, given the set of empirical observations.

In general, this relationship can be expressed as:

$$A * Q = T, \tag{6}$$

where A is the 'routing adjacency matrix', Q is the vector of random variables whose parameters we have to determine and T is the vector of random variables representing the inter-attraction tap times.

The above equation models the relationship between the distributions of the queue times, delays and movements. Given the set of actual movement observations T, we shall now address the question: "how do we compute the values of α_i and δ_i (i.e., the means of these random variables)?" In other words, taking expectations on both sides of Equation 6, we get the following relationship:

$$A * E(Q) = E(T). \tag{7}$$

More specifically, looking at the eighth row of the above equation, we have:

$$\alpha_2 + \alpha_3 + E[\tilde{d}] = E(t_{32}), \tag{8}$$

where α_2 is the mean of q_2, α_3 is the mean of q_3 and $E(t_{32})$ is the sample mean of the observation set t_{32}.

V. Expectation Maximization Algorithm

We now present an EM-based method to solve Equation 7. The EM algorithm ([5], [10]) is an iterative method for finding maximum likelihood estimates of parameters in statistical models with unobserved latent variables. It alternates between performing an expectation (E) step, which computes the likelihood of the observation being generated by a specific latent variable, and a maximization (M) step, which computes the parameters of the latent variable that maximize the expected likelihood found on the E step.

A. Description of EM for theme park queue estimation

We first outline the basic steps of the EM algorithm, using the theme park of Figure 3 as reference. We start with randomly chosen values for α_i (the mean of the queuing delays), and δ_l, the 'human sojourn time' means. Algorithm 1 outlines the pseudocode for EM algorithm, which we describe in detail next.

Algorithm 1: Expectation Maximization Algorithm

```
Data: $A$: routing matrix, $T$: vector of sets of sojourn
    delays, $t_{ij} \forall i, j$
Result: Estimates of $\alpha_i, \forall i \in \{1, \ldots, N\}, \delta_l, \forall l \in \{1, 2\}$
Initialization: Set $\alpha_i$, $\delta_l$ to 0 $\forall i, l$;
$i \longleftarrow 0$, $MaxIter \longleftarrow 50$;
while $i \leq MaxIter$ do
    E-Step:;
        $m^1 = A * [\alpha_1 \, \alpha_2 \, \ldots \, \alpha_N \, \delta_1]'$ ;
        $m^2 = A * [\alpha_1 \, \alpha_2 \, \ldots \, \alpha_N \, \delta_2]'$ ;
        for each $t^k_{ij} \in t_{ij}$ in $T$ do
            $t_1 = \Gamma(t^k_{ij}, m^1_{ij}, 1)$;
            $t_2 = \Gamma(t^k_{ij}, m^2_{ij}, 1)$;
            $\hat{P_{ij}}^k(1) = t_1/(t_1 + t_2)$;
            $\hat{P_{ij}}^k(2) = t_2/(t_1 + t_2)$;
    M-Step:;
        for each $t_{ij} \in T$ do
            $w^1_{ij} = t_{ij} * P_{ij}(1)$;
            $w^2_{ij} = t_{ij} * P_{ij}(2)$;
            $W = [w^1_{ij} w^2_{ij}]'$;
            $Q = NNLSE(A', W)$;
    $i \leftarrow i + 1$;
```

1) The "E" step: For observations between attractions 3 and 2, we can write:

$$t^k_{32} = q_3 + q_2 + \tilde{d}; \forall k = 1, \ldots.., |t_{32}| \tag{9}$$

We first estimate the relative probabilities for *each* of the $|t_{32}|$ observations that it came from the Gamma-distributed random variable d_1, or from the RV d_2. As the sum of two independently distributed Gamma random variables with the same scale parameter is also Gamma, we will get:

$$P^k_{32}(1) = \Gamma(t^k_{32}, \alpha_3 + \alpha_2 + \delta_1, 1) \tag{10}$$

$$P^k_{32}(2) = \Gamma(t^k_{32}, \alpha_3 + \alpha_2 + \delta_2, 1) \tag{11}$$

We can then compute the normalized probability for the observation t^k_{32} coming from either of these two distributions as:

$$\hat{P}^k_{32}(1) = \frac{P^k_{32}(1)}{P^k_{32}(1) + P^k_{32}(2)} \; \hat{P}^k_{32}(2) = \frac{P^k_{32}(2)}{P^k_{32}(1) + P^k_{32}(2)} \tag{12}$$

In this way, we can compute the relative normalized probabilities $\hat{P}^k_{ij}(m), m = \{1, 2\}; \forall k \in |t_{ij}|$ for all pair-wise observations, thereby completing the E step.

2) The "M" step: In the M-step, for each pairwise attraction $\langle i, j \rangle$, we *weigh* each observation t^k_{ij} by the relative probability of it coming from the appropriate 'mixture' distribution. To illustrate the process, we continue with the observations of the transitions between attractions 3 and 2.

We now create *separate* weighted equations for each of the two 'human sojourn' RVs, i.e., $\forall k \in 1 \ldots |t_{32}|$:

$$t^k_{32} * \hat{P}^k_{32}(1) = q_3 + q_2 + d_1 \tag{13}$$

$$t^k_{32} * \hat{P}^k_{32}(2) = q_3 + q_2 + d_2 \tag{14}$$

By the sufficient statistic property of all the readings we will get the following relationships:

$$\alpha_3 + \alpha_2 + \delta_1 = \frac{\sum_{k=1}^{|t_{32}|} t_{32}^k * \hat{P}_{32}^k(1)}{\sum_{k=1}^{|t_{32}|} \hat{P}_{32}^k(1)} \tag{15}$$

$$\alpha_3 + \alpha_2 + \delta_2 = \frac{\sum_{k=1}^{|t_{32}|} t_{32}^k * \hat{P}_{32}^k(2)}{\sum_{k=1}^{|t_{32}|} \hat{P}_{32}^k(2)} \tag{16}$$

Repeating this for all pairs of observations, we will get two values for each t_{ij}, one drawn from a combination of queuing delays and sojourn delay variable d_1, and the other drawn from a combination of queuing delays with sojourn delay variable d_2. If we denote the sample mean of the observed sojourn delays between attractions i and j as w_{ij}^1 and w_{ij}^2, where w_{ij}^1 & w_{ij}^2 are drawn from d_1 and d_2 respectively, we obtain the equations as shown in Equation 17.

3) Least Square Error Regression: From Equation 17, we obtain a set of simultaneous equations, where the number of equations is greater than the number of unknowns. We then use the Non-Negative Least Square Error (NNLSE) regression to find the 'best fit' estimate that most closely satisfies the set of equations. This is done at the end of the M-Step, and the estimates of the NNLSE regression are used as input to the subsequent E-Step, until the solution converges.

$$\begin{pmatrix} 0&1&0&0&1&0\\ 0&0&0&0&1&0\\ 0&0&0&0&1&0\\ 1&0&0&0&1&0\\ 0&0&0&0&1&0\\ 0&0&0&0&1&0\\ 1&0&1&0&1&0\\ 0&1&1&0&1&0\\ 0&0&1&0&1&0\\ 1&0&0&1&1&0\\ 0&1&0&1&1&0\\ 0&0&0&1&1&0\\ 0&1&0&0&0&1\\ 0&0&0&0&0&1\\ 0&0&0&0&0&1\\ 1&0&0&0&0&1\\ 0&0&0&0&0&1\\ 0&0&0&0&0&1\\ 1&0&1&0&0&1\\ 0&1&1&0&0&1\\ 0&0&1&0&0&1\\ 1&0&0&1&0&1\\ 0&1&0&1&0&1\\ 0&0&0&1&0&1 \end{pmatrix} \begin{pmatrix} \alpha_1\\ \alpha_2\\ \alpha_3\\ \alpha_4\\ \delta_1\\ \delta_2 \end{pmatrix} = \begin{pmatrix} w_{12}^1\\ w_{13}^1\\ w_{14}^1\\ w_{21}^1\\ w_{23}^1\\ w_{24}^1\\ w_{31}^1\\ w_{32}^1\\ w_{34}^1\\ w_{41}^1\\ w_{42}^1\\ w_{43}^1\\ w_{12}^2\\ w_{13}^2\\ w_{14}^2\\ w_{21}^2\\ w_{23}^2\\ w_{24}^2\\ w_{31}^2\\ w_{32}^2\\ w_{34}^2\\ w_{41}^2\\ w_{42}^2\\ w_{43}^2 \end{pmatrix} \tag{17}$$

VI. Experimental Results

We now study the estimation accuracy and other performance characteristics of the proposed EM algorithm. As the theme park dataset does not contain the "ground truth" for queuing delays, we resort to the use of a synthetic dataset. In this approach, the trace of visitor movements and wait times are generated using a synthetic movement model, whose parameters mimic the real-world dataset (we use the same transition probabilities, and number of visitors observed, for the 6-node theme park layout presented in Figure 1), but with artificially-generated queue and human sojourn delays. Table IV presents the specific parameters used for this data generation. We focus on two key metrics: *a)* the *estimation accuracy* and *b)* the *speed of EM convergence*.

Parameter	Value
Number of attractions (N)	6
Number of observations ($\|t_{ij}\|$)	[100, 500]
Proportion of sojourn delay	15(70%) 35(30%)
Maximum number of iterations	50

TABLE IV: Parameters for numerical evaluation

A. Estimation Accuracy

Variable	Mean Delay	Estimated	Error
q_1	4.00	4.90	+22.5%
q_2	8.00	7.86	-1.75%
q_3	12.00	12.95	+7.92%
q_4	16.00	16.55	+3.44%
q_5	20.00	21.35	+6.75%
q_6	24.00	25.03	+4.29%
d_1	15.00	16.12	+7.47%
d_2	35.00	35.38	+1.08%

TABLE V: Estimation Accuracy Results

Table V shows the converged estimates (for both the queuing and sojourn delays) after the algorithm finishes running. The algorithm performs well as the queue delay increases; for smaller delays the error is as high as 22%, whereas for higher values, the error is under 5%. (While different runs of the EM algorithm can converge to slightly different values depending on the starting point, the trend of greater error for smaller queue delays holds.) This property works to our advantage, as the park operator is principally interested only in those attractions that are 'congested'-i.e., are experiencing fairly significant wait times.

B. Varying the relative proportion of sojourn delay variables

We observe the performance of the algorithm when the relative proportions of the component distributions are changed. Specifically, we change the ratios of d_1 and d_2 variables, and observe the changes in the estimated values of α_i (the means of the q_i variables). Variations in sojourn delay mixture proportions occur because visitors may decide on a particular behavior more than another.

We tabulate the results in table VI. We observe that as the percentage of people who travel faster between attractions (i.e., with sojourn times generated according to d_1) increases, the estimated queue delays also increases. Conversely, as more visitors exhibit higher sojourn delay, the queuing delays are under-estimated.

C. Convergence Speed

Figure 4 shows rate of convergence for q_3, for different proportions of sojourn delay variables d_1 and d_2. We note that q_3 converges fairly fast (around the 20^{th} iteration)

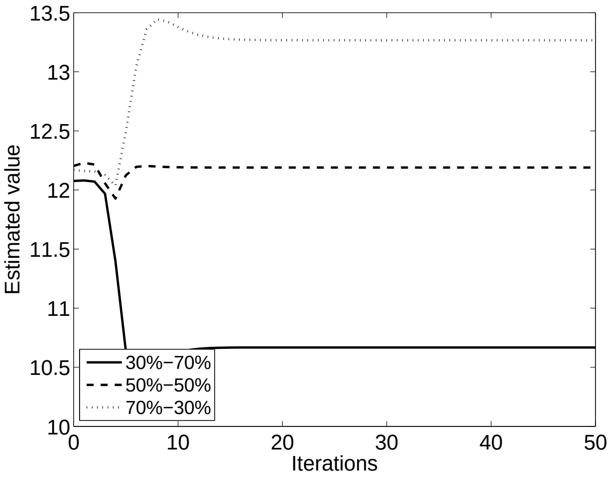

Fig. 4: Convergence of q_3 over 50 iterations with different d_i proportions

and that this rate of convergence seems to be essentially unaffected by different mixture proportions.

VII. Conclusion

In this paper, we have addressed the problem of dynamically estimating the queuing-related wait times at different theme park attractions, based on electronic records of visitor entry times, instead of the traditional methods of using pre-computed models of movement behavior. This estimation problem is challenging as the inter-attraction tap time interval arises not just from queuing delays, but from additional unpredictable and unobserved *sojourn delays* due to individualized visitor behavior. We characterize such a delay as a mixture model of two Gamma-distributed random variables, and then apply a modified EM algorithm (with the M-step consisting of an additional regression operation) to jointly identify the unknown parameters of both the queuing and sojourn delay distributions. Synthetic behavorial traces, based on a real-world theme park setting and empirically gathered visitor tap time records, show that our statistical delay estimator converges to a fixed-point solution quite rapidly and is fairly accurate in identifying the mean queuing delay, with estimation errors of $\leq 7-8\%$ for congested attractions. Note that this technique can be applied for dynamic delay estimation in other 'networked' queuing systems–e.g., for estimating congestion in road networks based on the times at which specific vehicles pass different toll booths.

Variable	Mean Delay	30%-70%	50%-50%	70%-30%
q_1	4.00	3.67	3.88	4.24
q_2	8.00	6.56	8.15	9.14
q_3	12.00	10.67	12.19	13.27
q_4	16.00	14.56	15.92	17.69
q_5	20.00	17.61	20.07	22.02
q_6	24.00	21.79	24.07	26.16
d_1	15.00	15.59	14.86	16.01
d_2	35.00	34.04	34.96	34.64

TABLE VI: Effect of change in mixture composition (ratio of $d_1 : d2$ observations) on queuing delay estimates

In ongoing work, we have been addressing two important open questions. First, the solution presented here assumes that the queuing delay distribution is uniform throughout the entire observation period. In reality, as user arrivals are not time-homogeneous, the distribution of the queuing delays will vary with time. More recently, we have extended this approach to perform continually-updated delay estimation in an *online* fashion, where the tap data is viewed as a continuously arriving data stream. In this approach, successive time intervals are assumed to have different distributions for the queuing random variables, with some correlation among successive intervals (as delays usually do not change abruptly, but build up or recede gradually). Second, to generalize this work to other layouts, we need to study the sensitivity of the results to the mix of pre-queue and post-queue attractions (as their relative order changes the number of unobserved variables that contribute to a single inter-tap duration).

References

[1] R. H. Ahmadi. Managing capacity and flow at theme parks. *Operations Research*, 45(1):1–13, 1997.

[2] A. Campbell, M. Gendreau, and B. Thomas. The orienteering problem with stochastic travel and service times. *Annals of Operations Research*, 186(1):61–81, 2011.

[3] S.-F. Cheng, L. Lin, J. Du, H. C. Lau, and P. Varakantham. An agent-based approach to dynamic experience management in theme parks. In *MABS 2013: Proceedings of 14th International Workshop on Multi-Agent-Based Simulation, in conjunction with AAMAS 2013 (Twelfth International Conference on Autonomous Agents and Multiagent Systems)*, 2013.

[4] M. Coates, A. Hero, R. Nowak, and B. Yu. Internet tomography. In *IEEE Signal Processing Magazine*, 2002.

[5] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[6] A. Kemperman, A. Borgers, H. Oppewal, and H. Timmermans. Predicting the duration of theme park visitors' activities: An ordered logit model using conjoint choice data. *Journal of Travel Research*, 41(4):375–384, 2003.

[7] V. Kostami and A. R. Ward. Managing service systems with an offline waiting option and customer abandonment. *Manufacturing & Service Operations Management*, 11(4):644–656, 2009.

[8] G. Liang and B. Yu. Maximum pseudo likelihood estimation in network tomography. In *IEEE Trans. on Signal Processing*, 2003.

[9] K. Rajaram and R. H. Ahmadi. Flow management to optimize retail profits at theme parks. *Operations Research*, 51(2):175–184, 2003.

[10] C. Wu. On the convergence properties of the em algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.