Moment-based availability prediction for bike-sharing systems

Cheng Feng [a],[*], Jane Hillston [b], Daniël Reijsbergen [b]

[a] *Institute for Security Science and Technology, Imperial College London, London, UK*
[b] *LFCS, School of Informatics, University of Edinburgh, Informatics Forum, Edinburgh, UK*

ARTICLE INFO

ABSTRACT

We study the problem of predicting the future availability of bikes in a bike station through the moment analysis of a PCTMC model with time-dependent rates. Given a target station for prediction, the moments of the number of available bikes in the station at a future time can be derived by a set of moment equations with an initial set-up given by the snapshot of the current state of all stations in the system. A directed contribution graph is constructed, and a contribution propagation method is proposed to prune the PCTMC so that it only contains stations which have significant contribution to the journey flows to the target station. Once the moments have been derived, the underlying probability distribution of the available number of bikes is reconstructed through the maximum entropy approach. We illustrate our approach on Santander Cycles, the bike-sharing system in London. The model is parameterized using historical data from Santander Cycles. Experimental results show that our model outperforms a time-inhomogeneous Markov queueing model with respect to several performance metrics for bike availability prediction.

1. Introduction

In recent years, we have seen significant growth of bike-sharing programs all over the world [1]. Public bike-sharing systems have been launched in many major cities such as London, Paris, and Vienna. Indeed, they have become an important part of urban transportation which provides improved connectivity to other modes of public transit. The concept of bike-sharing systems is rather simple: the system consists of a number of bike stations distributed over a geographic area (city). Each station is equipped with a limited number of bike slots in which public bikes can be parked. When users arrive at a station, they pick up a bike, use it for a while, and then return it to another station of their choice.

With the increasing popularity of the smart transport theme, there has been great interest from the research community in the intelligent management of bike-sharing systems. Topics include, but are not limited to, policy design [2,3], intelligent bike redistribution [4–6], and user journey planning [7,8]. The focus of this paper is on the probabilistic prediction of the number of available bikes in stations. Having a predictive model is of vital interest to both the user and the system administrator. The user can use it to identify likely origin/destination stations between which a trip can be successfully made. System administrators can use the model to undertake service level agreement checking, and plan bike redistribution for stations which are likely to break the service level requirement.

In this paper we present a novel moment-based prediction model that can provide probabilistic forecasts for the number of available bikes in a bike station. By representing the bike-sharing system as a Population Continuous Time Markov

---

* Corresponding author.
 E-mail addresses: c.feng@imperial.ac.uk (C. Feng), jane.hillston@ed.ac.uk (J. Hillston), dreijsbe@inf.ed.ac.uk (D. Reijsbergen).

Chain (PCTMC) with time-dependent rates, our model is explanatory as the dynamics of the system is explicitly given. Gast et al. [8] show the benefits of predicting (*forecasting*) the entire probability distributions of possible bike availabilities in a station, compared with previous models that were only able to produce point estimates, often using time-series-based techniques [7,9,10]. However, unlike [8], in which all the considered forecasting methods worked on the level of isolated stations, our model also captures the journey dynamics between stations.

Guenther and Bradley [11] also provide a PCTMC model with time-dependent rates for bike availability prediction, however there are several key differences between their model and ours. Firstly, our model provides the full probability distribution of the number of available bikes in a station whereas their model only provides a point estimate. Secondly, we use a model reduction method to prune our PCTMC such that the significant journey dynamics with respect to the target station are guaranteed to be preserved. However, their model aggregates stations which are spatially close, assuming that they have similar journey durations to the target station, which causes the information about the emptiness and fullness of stations to be lost.

Contribution. We summarize the contribution of our paper as follows. Firstly, a novel PCTMC model with time-dependent rates is presented to successfully capture the journey dynamics between bike stations. Secondly, we propose a novel model reduction technique to prune the PCTMC model based on the directed contribution graph with a contribution propagation method for a given target station for bike availability prediction as well as a correlation heuristic to reduce the size of ODEs for joint moments in order to achieve a fast moment analysis. Finally, we reconstruct the underlying probability distribution of the number of available bikes in the target station using the maximum entropy principle based on a few moments generated from fluid approximation of the PCTMC, and show that the model has a better performance on a set of metrics for bike availability prediction compared with the Markov single-station queueing model.

This paper is a substantial revision of the paper that appeared in QEST 2016 [12]: we give more detailed explanations of the technical aspects of the work, particularly the moment analysis techniques used, how parameters are estimated from data that is commonly available for bike sharing systems. We also present the detail of a novel correlation heuristic to accelerating moment analysis of the reduced PCTMC by a further reduction of joint moment ODEs. More experiments have been undertaken and we have done further work on evaluating the reliability of the predictions, including journeys involving multiple people.

The rest of this paper is structured as follows. Section 2 presents the background material related to our work: we briefly introduce the concepts of PCTMC with time-dependent rates and the Markov queueing model for bike availability prediction which we use as a baseline comparison. We also present our assumptions about the availability of data and the broad framework for fitting a model from the data. In Section 3, we present our PCTMC model for the bike-sharing scenario in detail, and the model reduction techniques that we use to make fast evaluation feasible. In the next section, we show how to reconstruct the probability distribution of the number of available bikes using the maximum entropy approach. Section 5 presents the experimental results of our model on the London bike-sharing system compared with the Markov queueing model. Finally, Section 6 discusses possible extensions of our model and draws final conclusions.

2. Background

In this section we present the modelling framework that we use for developing our basic model, PCTMC with time-dependent rates, and discuss the idea of the basic Markov queueing model used in [8], which we use as a baseline model for comparison. We also discuss the data typically available for bike-sharing systems and how this can be used to parameterize the models.
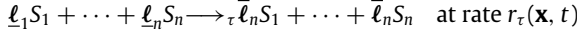
2.1. PCTMC with time-dependent rates

A PCTMC is a stochastic process which consists of a number of distinct agent populations and a set of transition classes. The state of a PCTMC is captured by an integer vector counting the number of each agent type. The model evolves with the firing of transitions. When a transition fires, one or more agent populations are updated. Each transition is associated with a rate function, which assigns a rate governed by an exponential distribution to the transition based on the current state of the PCTMC.

In this paper, we specifically consider *time-inhomogeneous* PCTMCs, in which transition rates are time-dependent. Such models are suitable for systems which experience periods of different dynamics at different times of the day, such as the bike-sharing systems in which the direction of travel often reflects the daily commute. Specifically, a PCTMC with time-dependent rates can be expressed as a tuple $\mathcal{P} = (\mathbf{x}, \mathcal{T}, \mathbf{x}_0)$:

- $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_{\geq 0}^n$ is an integer vector with the ith ($1 \leq i \leq n$) component representing the current number of an agent type S_i. Each x_i takes values in a finite domain $\mathcal{D}_i \subseteq \mathbb{Z}_{\geq 0}$. Hence, $\mathcal{D} = \prod_{i=1}^n \mathcal{D}_i$ is the state space of the model.
- $\mathcal{T} = \{\tau_1, \ldots, \tau_m\}$ is the set of transition classes, of the form $\tau = (r_\tau(\mathbf{x}, t), \mathbf{d}_\tau)$, where:
  1. $r_\tau(\mathbf{x}, t) \in \mathbb{R} \geq 0$ is a time-dependent rate function, associating with each transition the rate of an exponential distribution, depending on the state of the PCTMC $\mathbf{x}$ as well as the current time t.
  2. $\mathbf{d}_\tau \in \mathbb{Z}^n$ is the update vector which gives the net change for each element of $\mathbf{x}$ caused by transition τ.
- $\mathbf{x}_0 \in \mathbb{Z}_{\geq 0}^n$ is the initial state of the model.

We will assume that there is a set of deterministic time points $t_1, t_2, \ldots, t_k$ which mark the times at which rates within the system change.

For readability, transitions in PCTMCs can be expressed in the chemical reaction style, as

$$\underline{\ell}_1 S_1 + \cdots + \underline{\ell}_n S_n \longrightarrow_\tau \overline{\ell}_n S_1 + \cdots + \overline{\ell}_n S_n \quad \text{at rate } r_\tau(\mathbf{x}, t)$$

where the net change of agents of type S_i due to transition τ is given by $d_\tau^i = \overline{\ell}_i - \underline{\ell}_i \, (1 \leq i \leq n)$, and the transition rate is

$$r_\tau(\mathbf{x}, t) = \begin{cases} r_\tau(\mathbf{x}, t_1) & \text{if } x_i \geq \underline{\ell}_i \, \forall i = 1, 2, \ldots, n \text{ and } t < t_1 \\ r_\tau(\mathbf{x}, t_2) & \text{if } x_i \geq \underline{\ell}_i \, \forall i = 1, 2, \ldots, n \text{ and } t_1 \leq t < t_2 \\ \ldots \\ r_\tau(\mathbf{x}, t_k) & \text{if } x_i \geq \underline{\ell}_i \, \forall i = 1, 2, \ldots, n \text{ and } t_{k-1} \leq t < t_k \\ 0 & \text{otherwise.} \end{cases}$$

As the state space of PCTMC models is often very large or even infinite, numerical techniques traditionally used for performance analysis, based on a Markovian approach, are entirely infeasible. Stochastic simulation is feasible, but deriving useful metrics such as mean, variance, and probability distribution of populations often requires an excessively large number of simulation runs, thus making this approach extremely costly in terms of computational resources, particularly when estimating full probability distributions over large state spaces.

In this paper, we will adopt a much more computationally efficient approach to analyse the PCTMC for the bike-sharing model. Specifically, let $M(\mathbf{x}) : \mathbb{R}_{\geq 0}^n \rightarrow \mathbb{R}$ be a moment function, then the moment described by M evolves according to the following set of ODEs [13]:

$$\frac{\mathrm{d}}{\mathrm{d}t} \mathbb{E}[M(\mathbf{x}(t))] = \sum_{\tau \in \mathcal{T}} \mathbb{E}[(M(\mathbf{x}(t) + \mathbf{d}_\tau) - M(\mathbf{x}(t))) r_\tau(\mathbf{x}, t)] \tag{1}$$

with $E[M(\mathbf{x}(0))] = M(\mathbf{x}_0)$. For instance, if we set $M(\mathbf{x}) = x_i$, $M(\mathbf{x}) = x_i^2$, $M(\mathbf{x}) = x_i x_j$, we can get the following set of ODEs to describe the first moment, second moment and second-order joint moment respectively, of population variables in an arbitrary PCTMC model:

$$\frac{\mathrm{d}}{\mathrm{d}t} \mathbb{E}[x_i] = \sum_{\tau \in \mathcal{T}} \mathbb{E}[(x_i + d_\tau^i - x_i) r_\tau] = \sum_{\tau \in \mathcal{T}} d_\tau^i \mathbb{E}[r_\tau] \tag{2}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} \mathbb{E}[x_i^2] = \sum_{\tau \in \mathcal{T}} \mathbb{E}[((x_i + d_\tau^i)^2 - x_i^2) r_\tau]$$

$$= 2 \sum_{\tau \in \mathcal{T}} d_\tau^i \mathbb{E}[x_i \times r_\tau] + \sum_{\tau \in \mathcal{T}} {d_\tau^i}^2 \mathbb{E}[r_\tau] \tag{3}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} \mathbb{E}[x_i x_j] = \sum_{\tau \in \mathcal{T}} \mathbb{E}[((x_i + d_\tau^i)(x_j + d_\tau^j) - x_i x_j) r_\tau]$$

$$= \sum_{\tau \in \mathcal{T}} d_\tau^i \mathbb{E}[x_j \times r_\tau] + \sum_{\tau \in \mathcal{T}} d_\tau^j \mathbb{E}[x_i \times r_\tau] + \sum_{\tau \in \mathcal{T}} d_\tau^i \times d_\tau^j \mathbb{E}[r_\tau] \tag{4}$$

where we use x_i as short for $x_i(t)$, and r_τ as short for $r_\tau(\mathbf{x}, t)$ for convenience. The above system of ODEs is not closed if there exist transition rates in the PCTMC which are nonlinear functions of the population variables, as then the equation for one moment can only be expressed in terms of higher moments. As a consequence of this, moment-closure methods [14–16] must be applied to close the system of ODEs before it can be numerically solved. However, in this work, all the transition rates in the PCTMC for the bike-sharing model are either constants or linear functions of population variables, thus the derived system of moment ODEs can be directly solved by standard numerical simulation methods. With time-dependent rates, the system becomes hybrid rather than continuous, with discrete jumps of rates at some specific points of the numerical simulation.

2.2. Markov queueing model

It is intuitive to consider each bike station as a simple Markov queueing model [8], and we will use such a model as our comparator, i.e. we will compare the predictions based on our time inhomogeneous PCTMC model with predictions derived from a time inhomogeneous queueing model.

Concretely, the most straightforward way to evaluate the behaviour of a station is to analyse it in isolation. In this case, a station can be modelled as a time inhomogeneous Markov $M/M/1/c_i$ queue, as illustrated in Fig. 1.

Specifically, c_i denotes the capacity of a station i, $\lambda_i(t)$ and $\mu_i(t)$ are the time-dependent bike arrival and pickup rates of station i at time t of a day. Usually, the time of a day is split into k even slots, $[t_0, t_1), [t_1, t_2), \ldots, [t_{k-1}, t_k)$. Then, both $\lambda_i(t)$ and $\mu_i(t)$ can be estimated based on $|D|$ days of observation (as discussed below, days in D should be either all weekdays or
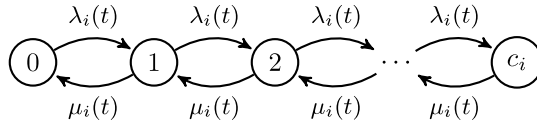
Fig. 1. The time-inhomogeneous Markov queue for station i.

all weekends), for $t_{j-1} < t < t_j$:

$$\lambda_i(t) = \frac{\sum_{d \in D} \text{No. of bike arrivals at station } i \text{ in } (t_{j-1}, t_j) \text{ on day } d}{\sum_{d \in D} \text{time length in } (t_{j-1}, t_j) \text{ on day } d \text{ during which station } i \text{ is not full}}$$

$$\mu_i(t) = \frac{\sum_{d \in D} \text{No. of bike pickups at station } i \text{ in } (t_{j-1}, t_j) \text{ on day } d}{\sum_{d \in D} \text{time length in } (t_{j-1}, t_j) \text{ on day } d \text{ during which station } i \text{ is not empty}}$$

Once the parameters for the model are known we can construct the transition rate matrix for station i: $Q^i(t)$, as

$$Q^i(t) = \begin{pmatrix} -\mu_i(t) & \mu_i(t) & & & & \\ \lambda_i(t) & -(\mu_i(t) + \lambda_i(t)) & \mu_i(t) & & & \\ & \ddots & \ddots & \ddots & & \\ & & \lambda_i(t) & -(\mu_i(t) + \lambda_i(t)) & \mu_i(t) \\ & & & \lambda_i(t) & -\lambda_i(t) \end{pmatrix},$$

and use it to predict the probability that there are y bikes in station i at time $t + h$ given the station has x bikes at time t, by the following equation:

$$\Pr(y \mid x, t, h) = \exp\left(\int_0^h Q^i(t + s) \mathrm{d}s \right)_{x,y}$$

where $\exp(M)_{x,y}$ is the element at row x and column y of the matrix exponential of M. Such a model has been used to make bike availability or station inventory level predictions in several papers in the literature (e.g. [6,8,17,18]).

Two assumptions are made in this model. First, the bike arrivals and pickups at stations form Poisson processes. Second, the state of a particular station does not depend on the state of the others. The first assumption has been successfully validated for busy stations in [8], using historical data from the Vélib' bike-sharing system in Paris. The second assumption is justified by mean field conditions in the case of populations that tend to infinity. However, we conjecture that the second assumption is generally not true in practice. For example, when a station is empty, no bikes can depart from it, therefore the arrival rate at other stations should be reduced. Hence, we seek a more realistic model, which captures the journey dynamics between stations.

2.3. Data on bike sharing systems

In order to parameterize the model of this paper, we need to be able to estimate the arrival and pick-up rates $\lambda_i(t)$ and $\mu_i(t)$ at each station i throughout the day. This can be done in two main ways. The first is to use the station occupancy data that is typically published online by operators in order to facilitate the planning of trips by users. This data can be logged manually,[1] resulting in a dataset of station occupancy changes at the time points at which the server was queried. This can be sufficient for estimating $\lambda_i(t)$ and $\mu_i(t)$, although both may be underestimated since a bike removal and arrival between two queries will cancel each other out. A second approach is possible when (in particular for Vélib' in Paris and Santander Cycles in London) operators are willing to share a more refined dataset of all the individual journeys, which allows for far more accurate estimation of the model parameters. We use the second approach for the London dataset which contains journey data between 751 stations in Section 5.

Some caution in choosing what data to include for parameterization is necessary, since in reality the arrival and departure rates of bikes are dependent on several other factors than just the time of day. These include the day-to-day weather, seasonal effects, and whether a day is a working day or not. To illustrate this, we have displayed the difference between weekday and weekend usage patterns in Fig. 2. It is clearly visible that the weekdays have pronounced rush hour peaks, whereas during the weekends the bikes are used most intensively during lunch time. To limit these external factors, we have only used data from the winter and early spring for the experiments, and excluded weekends and holidays.

---

[1] For an example of a service that aggregates this type of data for different bike-sharing systems, see https://citybik.es/.
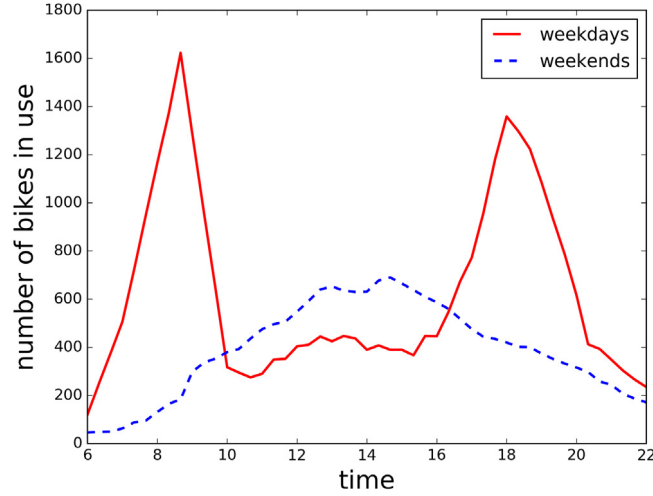
Fig. 2. The number of bikes in use in 20 min slots from 06:00 to 22:00 in Santander Cycles, London during weekdays and weekends.

3. PCTMC of bike-sharing model

In this section we present our time-inhomogeneous PCTMC model and explain how we use it as a basis from which to derive a computationally efficient reduced model of the moment equations suitable for deriving the probability distribution of the number of available bikes within a future time point. Since each journey directly involves just two stations, the departure station and the arrival station, we focus particularly on these. Moreover, from a prediction point of view we can consider two separate problems: given the current time t, the prediction of the number of bikes at the departure station at time $t + h_1$, where $t + h_1$ is the desired start time; and the prediction of the number of slots at the arrival station at time $t + h_2$ where $h_2 - h_1$ is the expected journey duration. In the following presentation we discuss the prediction of the number of available bikes at time $t + h$, but it is clear that the prediction of the number of available slots at the journey's end can be treated analogously.

3.1. A naive PCTMC model

In the PCTMC model we can identify three kinds of agents that determine the state of the system: bikes in stations, slots in stations and bikes which are en route between stations. These are represented as agent types $Bike_i$, $Slot_i$ and $Journey_j^i@P_l$. Journey durations are generally not exponentially distributed, so we fit the journey duration to a phase type distribution, where for the journey from station i to station j we have P_j^i identical phases each with rate P_j^i/d_j^i, where d_j^i is the mean journey duration. To faithfully represent the journey dynamics between bike stations in a bike-sharing system with N stations, we first propose a naive PCTMC model which contains the following transitions:

$$
\begin{aligned}
&Bike_i \longrightarrow Slot_i + Journey_j^i@P_1 && \text{at } \mu_i(t)p_j^i(t) && \forall i,j \in (1,N) \\
&Journey_j^i@P_l \longrightarrow Journey_j^i@P_{l+1} && \text{at } (P_j^i/d_j^i)\#(Journey_j^i@P_l) && 1 \leq l < P_j^i, \\
&&&&& \forall i,j \in (1,N) \\
&Journey_j^i@P_{P_j^i} + Slot_j \longrightarrow Bike_j && \text{at } (P_j^i/d_j^i)\#(Journey_j^i@P_{P_j^i}) && \forall i,j \in (1,N)
\end{aligned}
$$

where $Bike_i$, $Slot_i$ represent a bike and a slot agent in station i respectively; $Journey_j^i@P_l$ represents a bike agent which is currently on a journey from station i to station j at phase l, $\mu_i(t)$ is the fitted bike pickup rate governed by an exponential distribution in station i at time t, and p_j^i is the probability that a journey will end at station j given that it started from station i at time t. $\#(S)$ denotes the population of an agent type S.

Obviously, the above model is not scalable, even for solution by simulation. Since the total number of bike stations N is usually very large (for example there are around 750 bike stations in London), it is computationally infeasible to analyse a model which captures the full set of bike stations. Fortunately, since we are only interested in the prediction of bike availability at a single target station at a time, we only need to model stations which have a significant contribution to the journey flows to the target station (knowing the state of a station which has a very small contribution to the journey flows to the target station will have negligible impact on the accuracy of bike availability prediction for the target station). Thus, a directed contribution graph together with a contribution propagation method is proposed to automatically identify the set of stations which need to be modelled with respect to a given target station for bike availability prediction.
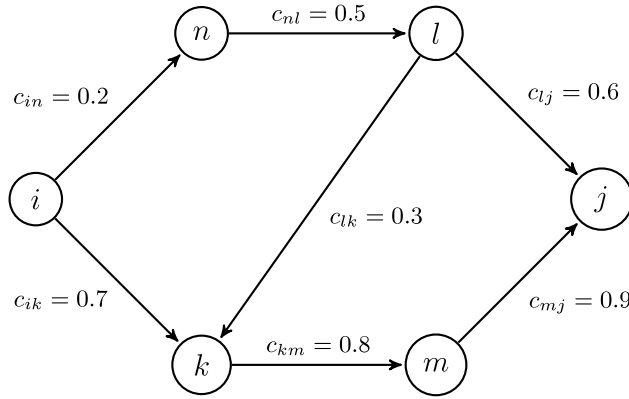
Fig. 3. An example directed contribution graph with six stations.

3.2. Directed contribution graph with contribution propagation

Here we show how to derive a set of bike stations $\Theta(v)$ in which all stations have a significant contribution to the journey flows to a given target station $v \in (1, 2, \ldots, N)$ for bike availability prediction. Concretely, we first need a way to quantify the contribution of one station to the journey flows to another station. Specifically, we let C_{ij} denote the *contribution coefficient* of station j to station i which quantifies the contribution of station j to the journey flows to station i.

One station can contribute to the journey flows to another station both directly and indirectly. The definition of a *direct contribution coefficient* at time t is given by the following simple formula:

$$c_{ij}(t) = \lambda_i^j(t)/\lambda_i(t)$$

in which $\lambda_i^j(t)$ represents the bike arrival rate from station j to station i at time t and $\lambda_i(t) = \sum_j \lambda_i^j(t)$. Then, it is clear that $c_{ij}(t) \in [0, 1], 0 \le \sum_{j \ne i} c_{ij}(t) \le 1$.

With the definition of the directed contribution coefficient, we can construct a directed contribution graph for the bike-sharing system at each time slot of a day. The definition of the directed contribution graph is given as follows:

Definition 1. At an arbitrary time t, the directed contribution graph for a bike-sharing system at this moment is a graph in which nodes represent the stations in the system, and there is a weighted directed edge from node i to node j if $c_{ij}(t) > 0$, and in this case the weight of the edge is $c_{ij}(t)$. In particular, note that the direction of edges is the inverse of contribution flows.

Fig. 3 shows a sample directed contribution graph which consists of six bike stations only for illustration purposes (in a real case, the graph will be more connected).

For those stations which are not directly connected in the directed contribution graph, by using a contribution propagation method, we can evaluate the *indirect contribution coefficient* of one station on the journey flows to another station. Specifically, the indirect contribution coefficient is quantified by a path dependent coefficient $c_{ij,\gamma}(t)$, which is the product of the direct contribution coefficients along an acyclic path γ from node i to node j:

$$c_{ij,\gamma}(t) = \prod_{kl \in \gamma} c_{kl}(t)$$

Intuitively, $c_{ij,\gamma}(t)$ measures the estimated contribution of station j to the journey flows to station i propagated through the path γ. More specifically, taking the graph in Fig. 3 as an example, suppose station i is the target station, then the contribution of station j to the journey flows to station i has to propagate through its direct contribution to station m, station m's direct contribution to station k, and finally reach station i through station k's direct contribution. Doing so, this indirect contribution is estimated by the product of the direct contribution coefficients along the path $\gamma = \{ik, km, mj\}$ to capture this diminishing propagation effect.

Furthermore, for the purpose of model reduction, we assume any contribution (either direct or indirect) to the journey flows to the target station which is less than a specific threshold is insignificant. Then, we can characterize the *contribution coefficient of station j to station i* by the maximum of the path dependent coefficients:

$$C_{ij}(t) = \begin{cases} \max_{\text{all paths } \gamma} c_{ij,\gamma}(t) & \text{if there exists a path from node } i \text{ to node } j \\ 0, & \text{otherwise} \end{cases}$$
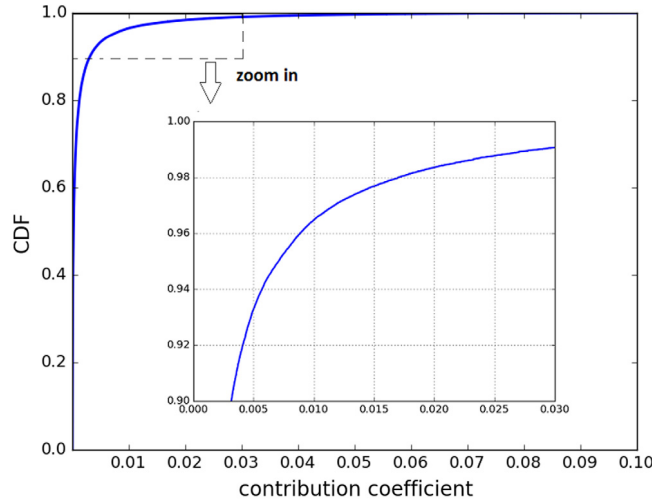
Fig. 4. CDF of contribution coefficients between all the bike stations in Santander Cycles during all time slots.

In the following we will abbreviate $c_{ij}(t)$, $c_{ij,\gamma}(t)$ and $C_{ij}(t)$ as c_{ij}, $c_{ij,\gamma}$ and C_{ij} when the value of t is unimportant. For example, according to Fig. 3, the contribution coefficient of station j to station i is $C_{ij} = c_{ik} \times c_{km} \times c_{mj} = 0.504$, since $c_{ik} \times c_{km} \times c_{mj} > c_{in} \times c_{nl} \times c_{lj} > c_{in} \times c_{nl} \times c_{lk} \times c_{km} \times c_{mj}$.

Lastly, with $\Theta(v)$ being the set of bike stations with a significant contribution to the journey flows into station v. Given a target station v, and its contribution coefficient for station $i \in (1, 2, \ldots, N)$, we can infer:

$$i \in \Theta(v) \quad \text{if } C_{vi} > \theta$$
$$i \notin \Theta(v) \quad \text{if } C_{vi} \leq \theta$$

where $\theta \in (0, 1)$ is the threshold value which can be used to control the extent of model reduction.

Remark. Capturing indirect contributions to the journey flows to the target station is important because indirect journey flows with significant coefficients can largely decide the emptiness of the stations which have a significant direct journey flow to the target station, thus can cause considerable impact on the overall journey flow to the target station. Furthermore, we choose to characterize contribution coefficients by the maximum instead of the sum of path dependent coefficients because we only want to model stations which have at least a significant (direct or indirect) journey flow to the target station. To model stations which have many insignificant journey flows to the target station is costly but can have limited impact on the predictions. Moreover, the maximum of path dependent coefficients has the nice property that if if $i \in \Theta(v)$ and $C_{vi} = c_{vi,\gamma}$, then for a station j which is on the path γ, it is certain that $C_{vj} > \theta$, thus $j \in \Theta(v)$. As a result, for all stations which have a significant journey flow to the target station, that journey flow will certainly be captured in the resulting reduced PCTMC. In contrast, this property would not be preserved if we use the sum of path dependent coefficients. For example in Fig. 3, if we set $\theta = 0.55$ and use the sum instead of the maximal of path dependent coefficients to characterize contribution coefficients, we get $C_{ij} = \sum_{\gamma} c_{ij,\gamma} > 0.55$, thus station j is included in the reduced PCTMC. However, since $\sum_{\gamma} c_{il,\gamma} < 0.55$, station l will not be included in the reduced PCTMC. As a result, $\sum_{\gamma} c_{ij,\gamma} > 0.55$ will not actually be satisfied in the reduced PCTMC after station l is excluded.

As an illustration of the extent of model reduction, Fig. 4 shows the empirical cumulative distribution function (CDF) of contribution coefficients during all time slots between any two bike stations in Santander Cycles (which is computed by historical journey data with 20 min slot duration). It can be seen that more than 96% of the computed contribution coefficients are smaller than 0.01. This means that on average more than 96% stations can be excluded even if θ is set to a small value 0.01 for the PCTMC of a random station in Santander Cycles.

3.3. The reduced PCTMC model

Given a target station v and current time t, suppose we are interested in the number of bikes at the station at time $t + h$. Letting $s = (s_1, s_2, \ldots, s_n)$ be the minimal set of time slots which cover $[t, t + h]$, we obtain $\Theta(v) = \Theta(v, s_1) \cup \Theta(v, s_2) \cup \cdots \cup \Theta(v, s_n) \cup v$, where $\Theta(v, s_i)$ is the set of bike stations which have significant contribution to the journey flows to the target station within time slot s_i. In other words, we take the union of all the stations that make a significant contribution to the journey flows across all the relevant time slots.

The PCTMC for the prediction of bike availability at station v at time $t + h$ is presented in Table 1

Table 1

The PCTMC model for the prediction of bike availability at station v at time $t + h$; $\mathbf{1}(Slot_j(t) = 0)$ is an indicator function which returns 1 when the number of empty slots at station j at time t is zero, otherwise returns 0.

Transitions	Rate	Scope	Explanation
$Bike_i \longrightarrow Slot_i$	$\mu_i(t)\left(1 - \sum_{j \notin \Theta(v) \vee c_{ji} \leq \theta} p_j^i(t)\right)$	$\forall i \in \Theta(v)$	A bike in station i is picked for a journey to a station outside $\Theta(v)$ or a station to which the journey flow is negligible.
$Slot_i \longrightarrow Bike_i$	$\sum_{j \notin \Theta(v) \vee c_{ij} \leq \theta} \lambda_i^j(t)$	$\forall i \in \Theta(v)$	A bike is returned to station i from a station outside $\Theta(v)$ or a station from which the journey flow is negligible.
$Bike_i \longrightarrow Slot_i + Journey_j^i@P_1$	$\mu_i(t)p_j^i(t)$	$\forall i, j \in \Theta(v) \wedge c_{ji} > \theta$	A bike in station i is picked for a journey to a station j inside $\Theta(v)$ and the journey flow is significant.
$Journey_j^i@P_l \longrightarrow Journey_j^i@P_{l+1}$	$(P_j^i/d_j^i)\ \#(Journey_j^i@P_l) \quad 1 \leq l < P_j^i$	$\forall i, j \in \Theta(v) \wedge c_{ji} > \theta$	Progress of a journey to the next stage.
$Slot_j + Journey_j^i@P_{P_j^i} \longrightarrow Bike_j$	$(P_j^i/d_j^i)\ \#(Journey_j^i@P_{P_j^i})$	$\forall i, j \in \Theta(v) \wedge c_{ji} > \theta$	Completion of a journey.
$Journey_j^i@P_{P_j^i} \longrightarrow \varnothing$	$\mathbf{1}(Slot_j(t) = 0)(P_j^i/d_j^i)\ \#(Journey_j^i@P_{P_j^i})$	$\forall i, j \in \Theta(v) \wedge c_{ji} > \theta$	A bike in transit from station i to station j is returned to another station outside $\Theta(v)$ because there is no empty slot in station j.

3.3.1. Approximating the indicator function

We are going to analyse the PCTMC using moment ODEs derived by Eq. (1). For clarity of notation we let u_i^m denote $\mathbb{E}[(Slot_i(t))^m]$, where m is the order of the moment. We can only access the moments of the number of empty slots at a station i at time t, u_i^m, during numerical simulation, whereas the number of empty slots at station i at time t is an unknown variable. Thus, we propose a method to approximate the indicator function $\mathbf{1}(Slot_i(t) = 0)$ by a function of the moments of the number of empty slots and the capacity of the station: $\mathbf{1}(Slot_i(t) = 0) \sim f(u_i^1, u_i^2, \ldots, u_i^m, k_i)$.

Concretely, given the first m moments of the random variable $Slot_i(t)$, and the value domain $Slot_i(t) \in [0, 1, \ldots, k_i]$, we can approximate the probability distribution of $Slot_i(t)$ by a discrete distribution with finite support k_i. For example, if we only know the first moment of $Slot_i(t)$ (which is u_i^1), we can fit a binomial distribution $Slot_i(t) \sim Binomial(k_i, u_i^1/k_i)$ to the probability distribution of $Slot_i(t)$. In this case, we get $P(Slot_i(t) = 0) = (1 - u_i^1/k_i)^{k_i}$. Furthermore, if we know the first two moments (u_i^1, u_i^2), then we can fit a beta-binomial distribution $Slot_i(t) \sim BetaBinomial(k_i, \alpha, \beta)$, where

$$\alpha = \frac{u_i^1 u_i^2 - k_i(u_i^1)^2}{k_i(u_i^1)^2 + k_i u_i^1 - k_i u_i^2 - (u_i^1)^2} \quad \beta = \frac{(k_i - u_i^1)(k_i u_i^1 - u_i^2)}{k_i(u_i^1)^2 + k_i u_i^1 - k_i u_i^2 - (u_i^1)^2}$$

Thus, we get

$$P(Slot_i(t) = 0) = \frac{B(\alpha, k_i + \beta)}{B(\alpha, \beta)}$$

where $B(a, b)$ is a beta function. Theoretically, with knowledge of more moments of $Slot_i(t)$, the estimation of $P(Slot_i(t) = 0)$ will be more accurate. Finally, we let

$$\mathbf{1}(Slot_i(t) = 0) = \begin{cases} 1 & \text{if } P(Slot_i(t) = 0) > p \\ 0 & \text{if } P(Slot_i(t) = 0) \leq p \end{cases}$$

where $P(Slot_i(t) = 0) = f(u_i^1, u_i^2, \ldots, u_i^m, k_i)$, p is a threshold value above which we believe the number of available slots in station i is zero. In general, p should be set to a value close to 1 in order to make sure the station is only treated as full when there is no empty slot with a high confidence. In our later experiments, we explicitly set $p = 0.9$.

3.3.2. Specifying the initial state

When we wish to use the model to make a prediction, say at time $t + h$, we must first initialize the model state for time t. A snapshot of the bike-sharing system a time t will contain the following information[2]:

$$Bike_i(t), \ldots, Slot_i(t), \ldots, Journey^i(t, \Delta t), \ldots$$

where $Bike_i(t)$ and $Slot_i(t)$ are the current number of available bikes and empty slots at a station i; $Journey^i(t, \Delta t)$ represents that there is a bike currently en route from station i, and the journey started at time $t - \Delta t$. Then, for each $Journey^i(t, \Delta t)$, we use a random number to determine the destination of the journey, and the time Δt to determine the appropriate phase of the journey time. Specifically, let $P_i^{t-\Delta t}(Des = k)$, $P_i(Dur > \Delta t \mid Des = k)$ $\forall k$ be the probability that the journey will end at station k given that the journey started from station i at time $t - \Delta t$, and the probability that the duration of a journey from station i to station k lasts more than Δt, respectively. Using Bayes' theorem, we can infer the probability that the destination of the journey will end up at a station k given its duration lasts more than Δt as follows:

$$P_i^{t-\Delta t}(Des = k \mid Dur > \Delta t) = \frac{P_i^{t-\Delta t}(Des = k) \times P_i(Dur > \Delta t \mid Des = k)}{\sum_{j=1}^{N} P_i^{t-\Delta t}(Des = j) \times P_i(Dur > \Delta t \mid Des = j)}$$

Then, we can generate a random number α uniformly distributed in $(0, 1)$, and decide $Journey^i(t, \Delta t) = Journey_j^i(t, \Delta t)$ if:

$$\sum_{k=0}^{j-1} P_i^{t-\Delta t}(Des = k \mid Dur > \Delta t) \leq \alpha < \sum_{k=0}^{j} P_i^{t-\Delta t}(Des = k \mid Dur > \Delta t)$$

Furthermore, we let

$$Journey_j^i(t, \Delta t) = Journey_j^i@P_l \quad \text{if } (l - 1)d_j^i/P_j^i \leq \Delta t < l \times d_j^i/P_j^i,$$

where $l \leq P_j^i$. Otherwise, if $l > P_j^i$, we let $Journey_j^i(t, \Delta t) = Journey_j^i@P_{P_j^i}$.

3.4. Further reduction of moment ODEs for the reduced PCTMC

We derive the moment ODEs following Eq. (1) for the reduced PCTMC for the first m order of moments. The reduced PCTMC has significantly fewer population variables than the naive PCTMC, however, we may still face a large number of moment ODEs which are slow to analyse especially for real-time bike availability predictions when m is large. Thus, in order

---

[2] This information is actually recorded for the London bike-sharing system.
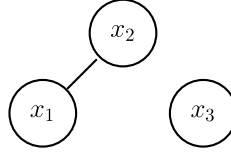
Fig. 5. The correlation graph of a moment variable $\mathbb{E}[x_1x_2x_3]$.

to accelerate moment analysis of the reduced PCTMC, we propose a correlation heuristic to achieve a further reduction on the size of the moment ODEs for the reduced PCTMC, utilizing the neighbourhood relation between population variables which is firstly introduced in [16].

Concretely, we observe that most of the derived ODEs are used to describe the evolution of joint moments of population variables. For instance, suppose $m = 2$, then there will be $2 \times n$ ODEs to describe the evolution of $\mathbb{E}[x_i]$ and $\mathbb{E}[x_i^2]$, $(n^2 - n)/2$ ODEs to describe joint moments $\mathbb{E}[x_ix_j]$ for all $i, j \in (1, 2, \ldots, n) \wedge i \neq j$, where n is the number of population variables in the reduced PCTMC. Clearly, the number of ODEs for joint moments grows exponentially as m increases. Therefore, our strategy is to reduce the number of ODEs for joint moments to accelerate moment analysis.

Specifically, for each distinct moment variable $\mathbb{E}[\mathbf{x^m}] = \mathbb{E}[x_1^{m_1} \cdots x_k^{m_k}]$ that appears in the left hand size of the derived moment ODEs for the reduced PCTMC, we construct a *correlation graph* for it. The definition of a correlation graph is given below:

Definition 2. The correlation graph G of a moment variable $\mathbb{E}[\mathbf{x^m}]$ is a graph, in which there is a node for each population variable x_i that appears in the expression of the moment variable. Moreover, there is an edge $Edge(x_i, x_j)$ between two nodes if and only if

$$\exists \, \tau \in \mathcal{T} \text{ such that } (d_\tau^i \neq 0 \wedge \delta_\tau^j = 1) \vee (d_\tau^j \neq 0 \wedge \delta_\tau^i = 1),$$

where δ_τ^j is an indicator equal to 1 if and only if x_j is updated after transition τ ($d_\tau^j \neq 0$) or x_j appears in the rate function ($r_\tau(\mathbf{x}, t)$) of τ.

Intuitively, this means that two population variables (x_i, x_j) are connected in the correlation graph if there exists a transition in the reduced PCTMC, in which one of the two population variables is updated, and the other is also involved. More specifically, a connected pair of population variables in the correlation graph are referred to one-hop neighbours in [16]. Empirically, it has been shown that only capturing the correlation between one-hop neighbours can achieve similar accuracy with moment analysis which captures the correlation between all population variables in a PCTMC. We refer to [16] for more details.

By definition, the correlation graph of a moment variable can consist of one or more *correlation islands*. The correlation islands are defined as follows:

Definition 3. A correlation island $\mathcal{I}$ is a subgraph of a correlation graph G such that:

$$\forall \, x_i, x_j \in \mathcal{I} \longrightarrow x_i \text{ and } x_j \text{ are connected}$$
$$\forall \, x_i \in \mathcal{I}, x_j \notin \mathcal{I} \longrightarrow x_i \text{ and } x_j \text{ are not connected}$$

Fig. 5 illustrates the correlation graph of a moment variable $\mathbb{E}[x_1x_2x_3]$ which consists of two correlation islands. Each correlation island in a correlation graph represents a decoupled moment variable with a lower order than the moment variable represented by the correlation graph. Thus, with the identification of correlation islands in a correlation graph, we can decouple the moment variable for joint moment ODE reduction. Specifically, let $\mathbb{E}[\mathbf{x^m}] = \mathbb{E}[x_1^{m_1} \cdots x_k^{m_k}]$ be an arbitrary moment variable that appears on the left hand side of a moment ODE, G be the corresponding correlation graph, and $\mathcal{I}$ be a correlation island in G. We can approximate $\mathbb{E}[\mathbf{x^m}]$ by the following formula:

$$\mathbb{E}[\mathbf{x^m}] \approx \prod_{\mathcal{I} \in G} \mathbb{E}[\prod_{x_i \in \mathcal{I}} x_i^{m_i}]$$

According to the above formula, for a moment variable $\mathbb{E}[\mathbf{x^m}]$ which appears on the left hand side of a moment ODE, if its correlation graph consists of more than one correlation island, then this moment ODE can be eliminated since it can be approximated by the product of moment variables with lower orders.

After the reduction of joint moment ODEs using the above method, we can achieve a much faster moment analysis of the reduced PCTMC for the bike-sharing model. The remaining set of moment ODEs is solved by numerical simulation using the Dormand–Prince method [19].

4. Probability distribution reconstruction

From the moment analysis of the PCTMC for the bike-sharing model, we obtain the first m moments of the number of available bikes in the target station v at the prediction time $t + h$, i.e.

$$\left(\mathbb{E}[(Bike_v(t+h))^1], \mathbb{E}[(Bike_v(t+h))^2], \ldots, \mathbb{E}[(Bike_v(t+h))^m] \right)$$

which we denote as $(u^1, u^2, \ldots, u^m)$ in the following. Our goal is to predict the probability that the station has a specific number of bikes at time $t + h$. This means the problem is to reveal $P\left(Bike_v(t+h) = i \mid u^1, u^2, \ldots, u^m, k_v\right)$, where $i \in (1, 2, \ldots, k_v)$. Therefore, we need to reconstruct the entire probability distribution of the random variable $Bike_v(t+h)$ based on its first m moments. The corresponding distribution is generally not uniquely determined. Hence, to select a particular distribution, we apply the maximum entropy principle to minimize the amount of bias in the reconstruction process. In this way, we assume the least amount of prior information about the true distribution. Note that the maximum entropy approach has been successfully applied to reconstruct distributions based on moments in many areas, e.g. physics [20], stochastic chemical kinetics [21], and performance analysis [22].

4.1. The maximum entropy approach

For convenience we will denote $Bike_v(t + h)$ by X_v and let $\mathcal{G}$ be the set of all possible probability distributions for X_v. Then, based on the maximum entropy principle, the goal is to select a distribution g to maximize the entropy $H(g)$ over all distributions in $\mathcal{G}$. The problem can be denoted as follows:

$$\arg \max_{g \in \mathcal{G}} H(g) = \arg \max_{g \in \mathcal{G}} \left(- \sum_{x=0}^{k_v} g(x) \ln g(x) \right)$$

Furthermore, given $(u^1, u^2, \ldots, u^m)$, we know the following constraints should be satisfied:

$$\sum_{x=0}^{k_v} x^n g(x) = u^n, \quad n = 0, 1, \ldots, m$$

where $u^0 = 1$ to ensure that g is a probability distribution. Now, the problem becomes a constrained optimization program. Thus to perform the constrained maximization of the entropy, we introduce one Lagrange multiplier λ_n per moment constraint. We thus seek extrema of the Lagrangian functional:

$$L(g, \lambda) = - \sum_{x=0}^{k_v} g(x) \ln g(x) - \sum_{n=0}^{m} \lambda_n \left(\sum_{x=0}^{k_v} x^n g(x) - u^n \right)$$

Functional variation with respect to the unknown probability mass function $g(x)$ yields:

$$\frac{\partial L}{\partial g(x)} = 0 \implies g(x) = \exp\left(-1 - \lambda_0 - \sum_{n=1}^{m} \lambda_n x^n \right)$$

Since $u^0 = 1$, we get

$$\sum_{x=0}^{k_v} \exp\left(-1 - \lambda_0 - \sum_{n=1}^{m} \lambda_n x^n \right) = 1.$$

Thus we can express λ_0 in terms of the remaining Lagrange multipliers

$$e^{1+\lambda_0} = \sum_{x=0}^{k_v} \exp\left(- \sum_{n=1}^{m} \lambda_n x^n \right) \equiv Z$$

Then, the general form of $g(x)$ can be given as follows:

$$g(x) = \frac{1}{Z} \exp\left(- \sum_{n=1}^{m} \lambda_n x^n \right)$$

Inserting the preceding equation into the Lagrangian, we can then transform the problem into an unconstrained minimization problem of a function Γ with respect to variables $\lambda_1, \lambda_2, \ldots, \lambda_m$:

$$\arg \min_{\lambda_1 \lambda_2, \ldots, \lambda_m} \Gamma(\lambda_1, \lambda_2, \ldots, \lambda_m)$$

where

$$\Gamma(\lambda_1, \lambda_2, \ldots, \lambda_m) = \ln Z + \sum_{n=1}^{m} \lambda_n u^n.$$

The convexity of the function Γ is proved in [20], which guarantees the existence of a unique solution of the minimization problem.

4.2. Solving the unconstrained minimization problem

Theoretically, the optimal $\boldsymbol{\lambda}^* = (\lambda_1^*, \lambda_2^*, \ldots, \lambda_m^*)$ which minimizes function Γ takes the solution of the following systems of equations:

$$\frac{\partial}{\partial \lambda_1} \Gamma(\lambda_1, \lambda_2, \ldots, \lambda_m) = 0$$
$$\frac{\partial}{\partial \lambda_2} \Gamma(\lambda_1, \lambda_2, \ldots, \lambda_m) = 0$$
$$\cdots$$
$$\frac{\partial}{\partial \lambda_n} \Gamma(\lambda_1, \lambda_2, \ldots, \lambda_m) = 0$$

However, in practice, it is impossible to solve the above equations analytically when $m \geq 2$. Thus, we apply the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [23–26] which is an iterative method commonly used for solving unconstrained nonlinear optimization problems to find a close approximation of the optimal $\boldsymbol{\lambda}^*$.

Specifically, with an initial guess $\boldsymbol{\lambda}^0$ and an initial Hessian matrix approximation $B_0 \in \mathbb{R}^{m \times m}$ which is positive definite (e.g., B_0 can be set to the identity matrix), the optimal $\boldsymbol{\lambda}^*$ can be obtained by the iteration (here $\boldsymbol{\lambda}^k$ is the value of $(\lambda_1, \lambda_2, \ldots, \lambda_m)$ after the kth iteration):

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \alpha_k * \mathbf{d}_k$$
$$B_{k+1} = B_k + \frac{\mathbf{y}_k \mathbf{y}_k^{\mathsf{T}}}{\mathbf{y}_k^{\mathsf{T}} \mathbf{s}_k} - \frac{B_k \mathbf{s}_k \mathbf{s}_k^{\mathsf{T}} B_k}{\mathbf{s}_k^{\mathsf{T}} B_k \mathbf{s}_k}$$

until the norm of the gradient satisfies $|\nabla \Gamma(\boldsymbol{\lambda}^k)| < \epsilon$ for a small threshold ϵ, where $\alpha_k > 0$ is an acceptable step size obtained by performing a line search along $\mathbf{d}_k$ [27], and

$$\mathbf{d}_k = -B_k^{-1} \nabla \Gamma(\boldsymbol{\lambda}^k)$$
$$\mathbf{y}_k = \nabla \Gamma(\boldsymbol{\lambda}^{k+1}) - \nabla \Gamma(\boldsymbol{\lambda}^k)$$
$$\mathbf{s}_k = \alpha_k \mathbf{d}_k.$$

After finding $\boldsymbol{\lambda}^*$ using the above algorithm, we can finally predict the probability that there are an arbitrary number of available bikes in the target station by:

$$P(X_v = x) = \frac{\exp\left(-\sum_{n=1}^{m} \lambda_n^* x^n\right)}{\sum_{i=0}^{k_v} \exp\left(-\sum_{n=1}^{m} \lambda_n^* i^n\right)}, \quad \forall x \in (1, 2, \ldots, k_v)$$

Clearly we can use an analogous distribution for also predicting the available number of slots at the end of a journey.

5. Experiments

In this section, we test the time cost and accuracy of our prediction model in different cases and compare the accuracy of our model with the baseline Markov queueing model. Specifically, we split the historic journey data and bike availability data from January 2015 to April 2015 from the London Santander Cycles Hire scheme to 10 parts, and a 10-fold cross validation is done where in each experiment 9/10 of data is used to fit the parameters of our PCTMC model as well as the Markov queueing model, the remaining 1/10 data is used to test their prediction accuracy. As in [11], we fit the number of journey phases between stations using the HyperStar tool [28] command line interface. Specifically, we set the maximum value of P_j^i to 20 to make our model compact and also to avoid overfitting. Moreover, for parameter estimation, we split a day into slots of 20 min duration. In our experiments, given the bike availability in a station at time t, we predict the probability distribution of the number of available bikes in that station at time $t + h$, where h is set to 10 min for short range prediction and 40 min for long range prediction.

The evaluation of our model is twofold. The first is accuracy, the second is efficiency. These two aspects are both influenced by the value of two important parameters, namely m, the highest order of moments being derived, and θ, the coefficient

Table 2
The mean and standard deviation of RMSEs on the prediction of the number of available bikes in the 10-fold cross validation.

	10 min	40 min	
ARIMA model	1.44 ± 0.12	2.94 ± 0.17	
Markov queueing model	1.45 ± 0.14	3.01 ± 0.18	
PCTMC with $\theta = 0.03$	1.44 ± 0.14	2.78 ± 0.17	$m = 1, 2, 3$
PCTMC with $\theta = 0.02$	1.44 ± 0.13	2.76 ± 0.17	$m = 1, 2, 3$
PCTMC with $\theta = 0.01$	1.42 ± 0.13	2.75 ± 0.17	$m = 1, 2, 3$

threshold for the identification of bike stations which have significant contribution to the journey flow to the target station. For higher values of m, the solution cost of our model becomes larger since more moment ODEs are derived, however the model should become more accurate due to more constraints in the probability distribution reconstruction based on the maximum entropy principle. For higher values of θ, more stations are excluded in the reduced PCTMC for a target station whereas the model accuracy can be potentially reduced. Thus, to observe the effects on these two parameters, we do experiments with values $m = 1, 2, 3$, and $\theta = 0.01, 0.02, 0.03$.

5.1. Root mean square error

For prediction accuracy, we first consider the classic criterion based on root mean square error (RMSE), a commonly used metric for evaluating point predictions (i.e., predictions that only state the expected number of bikes). Specifically, given a vector **x** of predictions and **y** of observations, with A the set of prediction/observation pairs, the RMSE is defined as:

$$\sqrt{\frac{1}{|A|} \sum_{i \in A} (x_i - y_i)^2}$$

Furthermore, in order to convincingly show the benefits of using our PCTMC model for bike availability prediction, we also compare our RMSE with an AutoRegressive Integrated Moving Average (ARIMA) model which is commonly used to provide point prediction for bike availability in stations [7,11]. Specifically, let $Y_i(t) = \{y_i(1), y_i(2), \ldots\}$ be a time series in which each element $y_i(t)$ represents the number of available bikes at station i at discrete time step t, we provide a spatial–temporal ARIMA(p,d,q) model as follows for bike availability prediction:

$$y_i^d(t) = \sum_{j=1}^{p} \alpha_j \, y_i^d(t - j) + \sum_{k \in ne(i)} \sum_{j=1}^{p} \gamma_j^k \, y_k^d(t - j) + \sum_{j=1}^{q} \beta_j \, \varepsilon_i(t - j) + \varepsilon_i(t)$$

where $y_i^d(t)$ denotes the dth differencing of $y_i(t)$ to impose stationarity, $ne(i)$ is the set of neighbour stations of station i for capturing spatial pattern correlation and the $\varepsilon_i(t)$ are error terms which are assumed to be independent, identically distributed variables sampled from a normal distribution with zero mean. The α_j are the parameters of the autoregressive part of the model, the γ_j^k are the parameters of the spatial pattern part, the β_j are the parameters of the moving average part. Here, we let the five nearest stations to denote the set of neighbour stations for any given station. The values of p, d, q for each station are tuned to achieve best prediction accuracy. All the parameters are fitted using the statsmodels Python library [29]. Moreover, like in [7,11], an observation frequency of 5 min is chosen to construct the time series. Thus, in order to predict the number of available bikes in a station at 10 min later, one needs to predict the number of available bikes at 5 minutes later first, and then use the predicted number to estimate the number at the target time. Clearly, we can also use this iterative approach to obtain the prediction of the number of available bikes at 40 min later.

Table 2 compares the RMSE of the prediction results of our PCTMC model with the Markov queueing model as well as the ARIMA model. As can be seen, although the improvement of accuracy in the short range is not significant, our PCTMC model clearly outperforms both the Markov queueing model and the ARIMA model in the long prediction range. This means capturing the journey dynamics provides more significant information for the long prediction range. For the PCTMC models, smaller values of θ only reduce the RMSE slightly. This means capturing less significant journey flows will have little impact on the prediction accuracy. Moreover, we find that the derived highest moments have almost no impact on the RMSE. This is obvious since the expected number of available bikes is only decided by the first moment.

5.2. Probability of making incorrect recommendations

Predicting the expected number of available bikes is important for system administrators when they want to decide how to redistribute bikes in the system. However, users are interested in whether there are at least N bikes in the target station when they want to pick up N bikes from there. We are specifically interested in the probabilities of making incorrect

recommendations for the queries "Will there be at least one bike?" and "Will there be at least two bikes?" in order to measure the accuracy of our model.[3]

Concretely, for the "Will there be at least N bikes?" query, we respond "Yes" if the predicted probability of that station having more than N bikes is greater than a threshold value $p*$, and respond "No" if the predicted probability of that station having more than N bikes is less than or equal to $p*$. Therefore, we define P(Incorrect, N) as the probability of making incorrect recommendations for the "Will there be at least N bikes?" query:

$$P(\text{Incorrect}, N) = P\Big(P(X_v \geq N) > p * \wedge x_v < N\Big) +$$
$$P\Big(P(X_v \geq N) \leq p * \wedge x_v \geq N\Big)$$

where $P\Big(P(X_v \geq N) > p * \wedge x_v < N\Big)$ is the probability of a "Yes" prediction when the actual number of available bikes is less than N, thus making an incorrect recommendation of "Yes"; whereas $P\Big(P(X_v \geq N) \leq p * \wedge x_v \geq N\Big)$ is the probability of making incorrect recommendations of "No". A larger value of the threshold $p*$ means a more conservative policy for recommendations of "Yes". For example, $p* = 0$ means the policy will (almost) always recommend "Yes" for the query, whereas $p* = 1$ means the policy will always recommend "No" for the query (in these two extreme cases, using different prediction models will make no difference).

In Fig. 6, we show the probabilities of making incorrect recommendations for the "Will there be at least one bike?" and "Will there be at least two bikes?" queries, using the Markov queueing model (MQM) and our PCTMC model with different parameters as a function of $p*$, where $p* \in \{0, 0.1, 0.2, \ldots, 0.9\}$ (letting $p* = 1$ makes no sense since it is useless to have an always "No" recommendation policy). *Note that the ARIMA model can only provide point prediction, thus is excluded here. The PCTMC model with $m = 1$ is also excluded since at least two moments are needed to make a meaningful reconstruction of the probability distribution.*

It can be seen that it is generally more likely to make incorrect recommendations for the "Will there be at least two bikes?" query than the "Will there be at least one bike?" query using both prediction models. Furthermore, we observe that for the short prediction range, our PCTMC model tends to have slightly lower probabilities of making incorrect recommendations for both queries compared with the Markov queueing model while increasing $p*$, which makes the policy become more conservative. For the long range prediction, the difference between our PCTMC model and the Markov queueing model becomes more clear. This means using our PCTMC model can effectively improve the accuracy of bike availability recommendations compared with the Markov queueing model. Moreover, for the long prediction range, we also find that increasing the value of m, which makes the reconstructed probability distribution closer to the true distribution, has a greater effect on improving recommendation accuracy than decreasing the value of θ so as to include more stations in the reduced PCTMC.

5.3. Scoring for recommendations

In the previous evaluation, we assume that making an incorrect recommendation of "Yes" is equally bad as making an incorrect recommendation of "No". However, in reality, users are more likely to be frustrated when they are recommended to go to a station and find that there are not enough bikes there than when they are recommended to not go to a station when there are actually enough bikes available in that station. Thus, the authors in [8] proposed a *proper scoring rule* in which correct recommendations will be awarded with a unique positive score and incorrect recommendations will be penalized by varying negative scores. A key feature of a proper scoring rule is that it ensures that the most accurate prediction model can always be expected to obtain the highest score. Specifically, it is proved in [8] that the following scoring rule is proper:

$$\text{Score} = \begin{cases} 1 & \text{if } P(X_v \geq N) > 0.8 \wedge x_v \geq N \\ -4 & \text{if } P(X_v \geq N) > 0.8 \wedge x_v < N \\ 1 & \text{if } P(X_v \geq N) < 0.8 \wedge x_v < N \\ -\dfrac{1}{4} & \text{if } P(X_v \geq N) < 0.8 \wedge x_v \geq N \end{cases}$$

in which we set the penalty scores for incorrect recommendations for "Yes" and "No" recommendations to -4 and $-\frac{1}{4}$, respectively; the threshold value $p*$ is set to 0.8 which is equal to:

$$p* = \frac{\text{Score(Incorrect, Yes)} - \text{Score(Correct, Yes)}}{\text{Score(Incorrect, Yes)} + \text{Score(Incorrect, No)} - \text{Score(Correct, Yes)} - \text{Score(Correct, No)}}.$$

The above scoring rule can be easily extended to the evaluation of the recommendations to the "Will there be at least one/two slots?" queries.

We show the average score with 95% confidence interval of our PCTMC model with different parameters as well as the Markov queueing model for the queries "Will there be at least one/two bikes?" and "Will there be at least one/two slots?"

---

[3] We assume it is relatively uncommon for more than two people to plan bike journeys together.
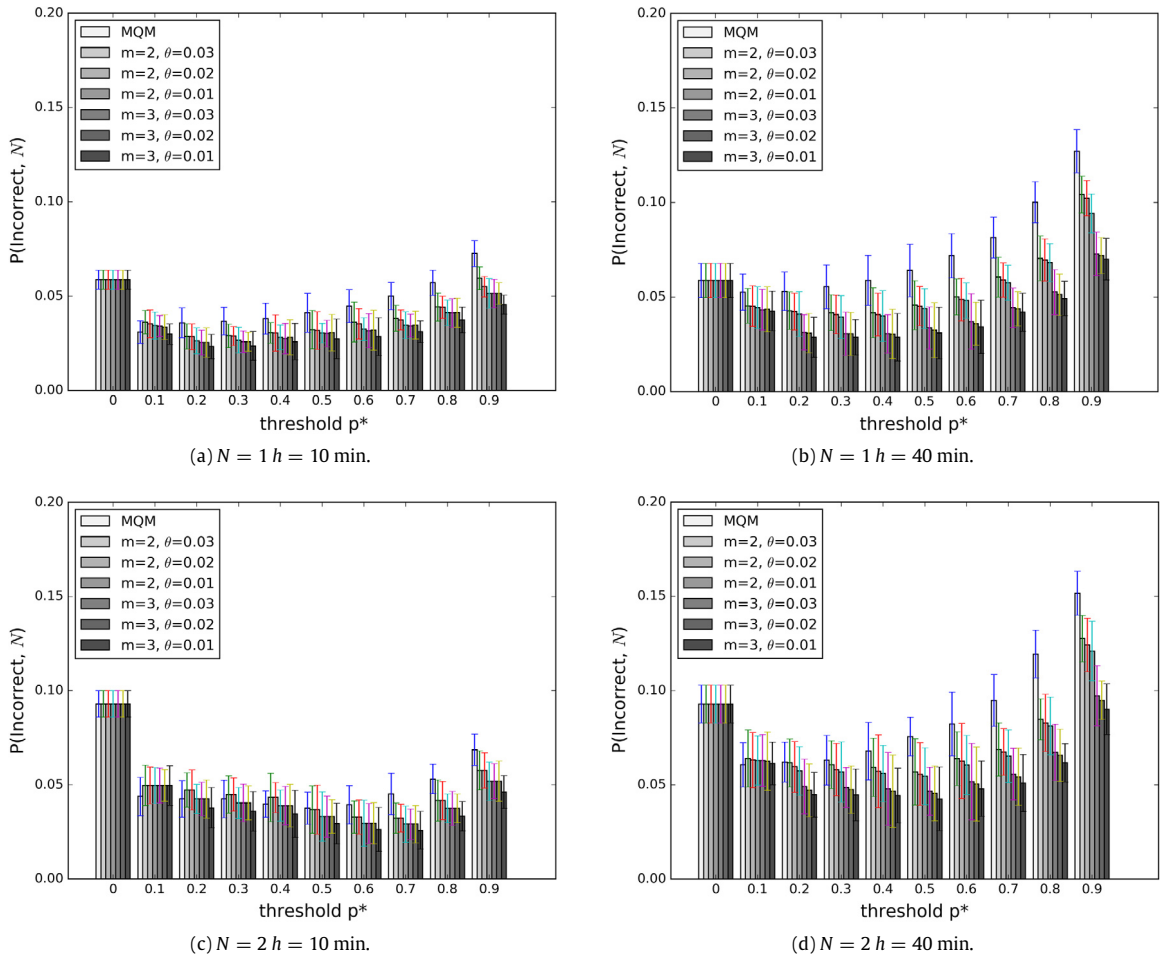
Fig. 6. The mean and standard deviation of probabilities of making incorrect recommendations for "Will there be at least one and two bikes?" queries in the 10-fold cross validation.

Table 3
The average score of making a recommendation to the "Will there be at least one bike?" and "Will there be at least two bikes?" queries with 95% confidence interval based on all the predictions made in the 10-fold cross validation.

	10 min		40 min		
	$N = 1$	$N = 2$	$N = 1$	$N = 2$	
Markov queueing model	0.91 ± 0.02	0.90 ± 0.02	0.88 ± 0.02	0.85 ± 0.03	
PCTMC with $\theta = 0.03$	0.92 ± 0.02	0.91 ± 0.03	0.91 ± 0.03	0.87 ± 0.03	$m = 2$
	0.93 ± 0.03	0.92 ± 0.03	0.92 ± 0.03	0.89 ± 0.03	$m = 3$
PCTMC with $\theta = 0.02$	0.92 ± 0.02	0.91 ± 0.03	0.91 ± 0.03	0.87 ± 0.02	$m = 2$
	0.93 ± 0.03	0.92 ± 0.03	0.92 ± 0.03	0.89 ± 0.03	$m = 3$
PCTMC with $\theta = 0.01$	0.93 ± 0.02	0.91 ± 0.03	0.91 ± 0.03	0.87 ± 0.02	$m = 2$
	0.93 ± 0.03	0.92 ± 0.03	0.92 ± 0.03	0.89 ± 0.03	$m = 3$

based on all the recommendations made in the 10-fold cross validation in Tables 3 and 4, respectively. As can be seen from the tables, the PCTMC model has a better average score for the recommendation for both queries. In the long prediction range, the improvement is more clear. In agreement with the results of Fig. 6, we also observe that with higher values of m, the average score increases. Moreover, decreasing the value of θ again has a rather limited effect on the scores which means that the PCTMC with $\theta = 0.03$ has already captured almost all the significant journey dynamics to the target station for an accurate availability prediction for enhancing user experience compared with the Markov queueing model. To summarize, the results in the two tables again demonstrate that using our PCTMC model for bike availability prediction can actually improve the user experience of the public bike-sharing system.

Table 4
The average score of making a recommendation to the "Will there be at least one slot?" and 'Will there be at least two slots?" queries with 95% confidence interval based on all the predictions made in the 10-fold cross validation.

	10 min		40 min		
	$N = 1$	$N = 2$	$N = 1$	$N = 2$	
Markov queueing model	0.92 ± 0.03	0.90 ± 0.03	0.89 ± 0.03	0.85 ± 0.03	
PCTMC with $\theta = 0.03$	0.93 ± 0.03	0.91 ± 0.03	0.92 ± 0.02	0.88 ± 0.02	$m = 2$
	0.93 ± 0.03	0.91 ± 0.03	0.92 ± 0.03	0.88 ± 0.03	$m = 3$
PCTMC with $\theta = 0.02$	0.93 ± 0.02	0.91 ± 0.02	0.92 ± 0.02	0.88 ± 0.03	$m = 2$
	0.93 ± 0.03	0.92 ± 0.03	0.92 ± 0.03	0.89 ± 0.02	$m = 3$
PCTMC with $\theta = 0.01$	0.93 ± 0.03	0.91 ± 0.02	0.92 ± 0.02	0.88 ± 0.02	$m = 2$
	0.93 ± 0.03	0.92 ± 0.03	0.93 ± 0.02	0.89 ± 0.02	$m = 3$

Table 5
Time cost to make a prediction with 95% confidence interval.

	10 min	40 min	
PCTMC with $\theta = 0.03$	1.76 ± 0.2 ms	6.98 ± 0.77 ms	$m = 1$
	103 ± 13.7 ms	328 ± 43 ms	$m = 2$
	2.2 ± 0.2s	8.9 ± 0.83s	$m = 3$
PCTMC with $\theta = 0.02$	4.25 ± 0.4 ms	15.72 ± 1.42 ms	$m = 1$
	251 ± 25.5 ms	1.1 ± 0.1s	$m = 2$
	8.9 ± 1.2s	37 ± 3.5s	$m = 3$
PCTMC with $\theta = 0.01$	13.5 ± 0.9 ms	49.1 ± 3.92 ms	$m = 1$
	8.8 ± 1.1s	30.1 ± 0.31s	$m = 2$
	33.9 ± 5.4s	157 ± 17.8s	$m = 3$

5.4. Time cost

The time cost of making a prediction is also important. Table 5 shows the time cost for making a prediction using our PCTMC model with different parameters (we do not show the time costs for the Markov queueing model since they are negligible due to its small state space because of the independence assumption). As can be seen from the table, although increasing the value of m has been shown to have a great effect on reducing the probability of making incorrect recommendations and improving the scores for recommendations, it also incurs a greater time cost for making a prediction due to the large number of ODEs for the derived higher moments. Decreasing the value of θ will also lead to larger time cost because more journey dynamics are explicitly captured in the reduced PCTMC. As a result, in general, given an acceptable time cost threshold, we want to have as small a value of θ as possible to have more accurate prediction for point estimates as illustrated in Section 5.1. For probability distribution predictions, one should prioritize the choice of m since it has a greater impact on the accuracy of predictions as demonstrated in Sections 5.2 and 5.3.

In this case, we assume that the time cost of making a prediction must be less than one second for real time application. Thus, for point predictions, we recommend to set $\theta = 0.01$, $m = 1$ for both prediction ranges. For probability distribution predictions, we recommend to set $\theta = 0.02$, $m = 2$ for short range prediction, $\theta = 0.03$, $m = 2$ for long range prediction. Note that we used an Intel CORE i7 laptop with 8GB RAM to run our experiments; the time cost could be considerably reduced if a more powerful machine, e.g. a server, were used.

6. Conclusions

We have presented a moment-based approach to make predictions of availability in bike-sharing systems. The moments of the number of available bikes are automatically derived via a PCTMC with time-inhomogeneous rates, fitted from historical data. The entire probability distribution is reconstructed using a maximum entropy approach. Our model is easy to understand since it explicitly captures the dynamics of the bike-sharing system. We have demonstrated that it outperforms a spatial–temporal ARIMA model for point estimates, and the Markov queueing model in several performance metrics for prediction accuracy. Moreover we have also shown that by using the directed contribution graph and the moment ODE reduction method, the model size can be significantly reduced to such an extent that it is suitable for real-time application.

In future work we plan to explore the impact of neighbouring stations, and extend our model to capture their effects. For example, if a station is empty, then the user is likely to pick up a bike from a neighbouring station, thus increasing the pick-up rate at the neighbouring station. Conversely, if a station is full, then the user is likely to return a bike to a neighbouring station, increasing the bike arrival rate there. We think another merit of our PCTMC model is that it can be easily extended to capture such impact by using the indicator function to check whether a neighbouring station is empty or full in order to alter the bike arrival and pick-up rate of a station.

Acknowledgement

References

[1] E. Fishman, Bikeshare: A review of recent literature, Transp. Rev. 36 (1) (2016) 92–113. http://dx.doi.org/10.1080/01441647.2015.1033036.

[2] J.-R. Lin, T.-H. Yang, Strategic design of public bicycle sharing systems with service level constraints, Transp. Res. E: Logist. Transp. Rev. 47 (2) (2011) 284–294. http://dx.doi.org/10.1016/j.tre.2010.09.004.

[3] J. Pfrommer, J. Warrington, G. Schildbach, M. Morari, Dynamic vehicle redistribution and online price incentives in shared mobility systems, IEEE Trans. Intell. Transp. Syst. 15 (4) (2014) 1567–1578. http://dx.doi.org/10.1109/TITS.2014.2303986.

[4] R. Nair, E. Miller-Hooks, Fleet management for vehicle sharing operations, Transp. Sci. 45 (4) (2011) 524–540. http://dx.doi.org/10.1287/trsc.1100.0347.

[5] C. Contardo, C. Morency, L.-M. Rousseau, Balancing A Dynamic Public Bike-Sharing System. Tech. Rep. CIRRELT-2012-09, 2012.

[6] J. Schuijbroek, R. Hampshire, W.-J. van Hoeve, Inventory Rebalancing and Vehicle Routing in Bike Sharing Systems. Tech. Rep. 2013-E1, Tepper School of Business, Carnegie Mellon University, 2013.

[7] J.W. Yoon, F. Pinelli, F. Calabrese, Cityride: a predictive bike sharing journey advisor, in: 13th IEEE International Conference on Mobile Data Management (MDM), IEEE, 2012, pp. 306–311. http://dx.doi.org/10.1109/MDM.2012.16.

[8] N. Gast, G. Massonnet, D. Reijsbergen, M. Tribastone, Probabilistic forecasts of bike-sharing systems for journey planning, in: The 24th ACM International Conference on Information and Knowledge Management (CIKM 2015), 2015. http://dx.doi.org/10.1145/2806416.2806569.

[9] J. Froehlich, J. Neumann, N. Oliver, Sensing and predicting the pulse of the city through shared bicycling, in: IJCAI, vol. 9, 2009, pp. 1420–1426.

[10] A. Kaltenbrunner, R. Meza, J. Grivolla, J. Codina, R. Banchs, Urban cycles and mobility patterns: exploring and predicting trends in a bicycle-based public transport system, Pervasive Mob. Comput. 6 (4) (2010) 455–466. http://dx.doi.org/10.1016/j.pmcj.2010.07.002.

[11] M.C. Guenther, J.T. Bradley, Journey data based arrival forecasting for bicycle hire schemes, in: Analytical and Stochastic Modeling Techniques and Applications, in: LNCS, vol. 7984, Springer, 2013, pp. 214–231. http://dx.doi.org/10.1007/978-3-642-39408-9_16.

[12] C. Feng, J. Hillston, D. Reijsbergen, Moment-based probabilistic prediction of bike availability for bike-sharing systems, in: G. Agha, B.V. Houdt (Eds.), 13th International Conference on Quantitative Evaluation of Systems, QEST 2016, in: LNCS, 9826, Springer, 2016, pp. 139–155. http://dx.doi.org/10.1007/978-3-319-43425-4_9.

[13] S. Engblom, Computing the moments of high dimensional solutions of the master equation, Appl. Math. Comput. 180 (2) (2006) 498–515. http://dx.doi.org/10.1016/j.amc.2005.12.032.

[14] A. Ale, P. Kirk, M.P. Stumpf, A general moment expansion method for stochastic kinetic models, J. Chem. Phys. 138 (17) (2013) 174101. http://dx.doi.org/10.1063/1.4802475.

[15] M.C. Guenther, A. Stefanek, J.T. Bradley, Moment closures for performance models with highly non-linear rates, in: Computer Performance Engineering, in: LNCS, vol. 7587, Springer, 2013, pp. 32–47. http://dx.doi.org/10.1007/978-3-642-36781-6_3.

[16] C. Feng, J. Hillston, V. Galpin, Automatic moment-closure approximation of spatially distributed collective adaptive systems, ACM Trans. Model. Comput. Simul. 26 (4) (2016) 26. http://dx.doi.org/10.1145/2883608.

[17] C. Fricker, N. Gast, H. Mohamed, Mean field analysis for inhomogeneous bike sharing systems, in: Aofa 2012, International Meeting on Probabilistic, Combinatorial and Asymptotic Methods for the Analysis of Algorithms, 2012.

[18] T. Raviv, O. Kolka, Optimal inventory management of a bike-sharing station, IIE Trans. 45 (10) (2013) 1077–1093. http://dx.doi.org/10.1080/0740817X.2013.770186.

[19] J.R. Dormand, P.J. Prince, A family of embedded Runge-Kutta formulae, J. Comput. Appl. Math. 6 (1) (1980) 19–26. http://dx.doi.org/10.1016/0771-050X(80)90013-3.

[20] L.R. Mead, N. Papanicolaou, Maximum entropy in the problem of moments, J. Math. Phys. 25 (8) (1984) 2404–2417. http://dx.doi.org/10.1063/1.526446.

[21] A. Andreychenko, L. Mikeev, V. Wolf, Model reconstruction for moment-based stochastic chemical kinetics, ACM Trans. Model. Comput. Simul. 25 (2) (2015) 12. http://dx.doi.org/10.1145/2699712.

[22] Á. Tari, M. Telek, P. Buchholz, A unified approach to the moments based distribution estimation–unbounded support, in: Formal Techniques for Computer Systems and Business Processes, Springer, 2005, pp. 79–93. http://dx.doi.org/10.1007/11549970_7.

[23] C.G. Broyden, The convergence of a class of double-rank minimization algorithms 2. the new algorithm, IMA J. Appl. Math. 6 (3) (1970) 222–231. http://dx.doi.org/10.1093/imamat/6.3.222.

[24] R. Fletcher, A new approach to variable metric algorithms, Comput. J. 13 (3) (1970) 317–322. http://dx.doi.org/10.1093/comjnl/13.3.317.

[25] D. Goldfarb, A family of variable-metric methods derived by variational means, Math. Comp. 24 (109) (1970) 23–26. http://dx.doi.org/10.2307/2004873.

[26] D.F. Shanno, Conditioning of quasi-newton methods for function minimization, Math. Comp. 24 (111) (1970) 647–656. http://dx.doi.org/10.2307/2004840.

[27] Y.-H. Dai, Convergence properties of the bfgs algoritm, SIAM J. Optim. 13 (3) (2002) 693–701. http://dx.doi.org/10.1137/S1052623401383455.

[28] P. Reinecke, T. Krauss, K. Wolter, Hyperstar: phase-type fitting made easy, in: 2012 Ninth International Conference on Quantitative Evaluation of Systems, IEEE, 2012, pp. 201–202. http://dx.doi.org/10.1109/QEST.2012.29.

[29] S. Seabold, J. Perktold, Statsmodels: econometric and statistical modeling with python, in: 9th Python in Science Conference, 2010.