Performance modelling and analysis for IoT services

Jiwei Huang*

State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications,
Beijing 100876, China
Fax: +86 10 6119 8034
Email: huangjw@bupt.edu.cn
*Corresponding author

Songyuan Li

School of Computer Science,
Beijing University of Posts and Telecommunications,
Beijing 100876, China
Email: sylee1416@gmail.com

Ying Chen

Department of Computer Science and Technology,
Tsinghua University,
Beijing 100084, China
Email: chenying12@mails.tsinghua.edu.cn

Junliang Chen

State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications,
Beijing 100876, China
Email: chjl@bupt.edu.cn

Abstract: With the growing popularity of Internet of Things (IoT) services on the internet, performance has become an important issue in the design and optimisation of IoT service systems. This paper proposes a theoretical approach of performance evaluation for IoT services, which is expected to provide a mathematical prediction on performance metrics at the design phase before system implementation. In specific, we first formulate an atomic service by a queueing system, and then an IoT system can be modelled by a queueing network. Detailed quantitative model analyses under different request arrival distributions are presented, and performance metrics are obtained. Furthermore, we briefly study two popular problems which are resource management and task scheduling in order to illuminate how the models and analytical results can be applied in the design and optimization of IoT systems. Finally, simulation experiments based on real-world data are conducted to validate the effectiveness of our approach.

Keywords: Internet of Things; IoT; IoT service; performance evaluation; quality of service; queueing model.

Reference to this paper should be made as follows: Huang, J., Li, S., Chen, Y. and Chen, J. (2018) 'Performance modelling and analysis for IoT services', *Int. J. Web and Grid Services*, Vol. 14, No. 2, pp.146–169.

Biographical notes: Jiwei Huang received both PhD and BE in Computer Science and Technology from Tsinghua University in 2014 and 2009, respectively. He is now an Assistant Professor in the State Key Laboratory of Networking and Switching Technology at Beijing University of Posts and Telecommunications. He was a Visiting Scholar at Georgia Institute of Technology. His research interests include services computing, Internet of Things and performance evaluation. He has published more than 20 papers in international journals and conference proceedings, e.g. IEEE Transactions on Services Computing, IEEE Transactions on Cloud Computing, ACM SIGMETRICS, IEEE ICWS, IEEE SCC, etc.

Currently, Songyuan Li is pursuing BE with the School of Computer Science at Beijing University of Posts and Telecommunications. His research interests include services computing, Internet of Things and edge computing.

Ying Chen received BE degree in the School of Computer Science from Beijing University of Posts and Telecommunications, Beijing, China, in 2012. Currently, she is pursuing PhD with the Department of Computer Science and Technology, Tsinghua University. Her research interests include modelling, performance evaluation and optimisation of web services and services computing.

Junliang Chen received the Doctor of Engineering degree from the Moscow Institute of Electrical Telecommunications, former Soviet Union, in 1961. He was a Visiting Scholar at the University of California, Berkeley and the University of California, Los Angeles from 1979 to 1981. Currently, he is a Professor and the Director of the Institute of Network Technology of Beijing University of Posts and Telecommunications. He is an academician of both the Chinese Academy of Sciences and the Chinese Academy of Engineering, and the fellow of China Computer Federation (CCF). His research interests include services computing, network intelligence and services, Internet of Things technology, switching technology and communication software.

1 Introduction

The Internet of Things (IoT) is an emerging technique providing advanced connectivity of devices, systems, services, and human beings that go beyond machine-to-machine communications (Höller et al., 2014). It can be regarded as the natural continuity of paradigms coming from several domains, such as cloud computing, ubiquitous computing, services computing and ambient intelligence (Al-Fuqaha et al., 2015). With the rapid development of the underlying techniques related to the IoT such as electronics, sensors, networking, embedded systems and upper-layer software, the IoT has become widely applied in many aspects of human life, such as industrial manufacturing, medical

and healthcare systems, environmental monitoring, transportation, building and home automation, etc.

From services computing viewpoint (Zhang et al., 2007), IoT environments can be seen as smart environments composed of pervasively distributed things (e.g. devices, sensors, actuators, smartphones and appliances) offering heterogeneous capabilities abstracted as services. With the rapid development of hardware and software technologies, besides sensing, each thing has been capable of handling basic computational tasks related to its sensing data. Such capability makes it possible for IoT devices to react immediately according to its surrounding environment. For complex tasks that involve large amount of computational procedures and database operations which are beyond the capabilities of the devices, they will be thrown to computing centres which are equipped with a number of powerful servers connected with high-speed networking. Such computing paradigm conforms to an emerging computing model, namely edge computing which pushes the frontier of computing applications, data and services away from centralised nodes to the logical extremes of a network, enabling analytics and knowledge generation to occur at the source of the data (Garcia Lopez et al., 2015). This novel paradigm introduces an edge layer to leveraging all the resources and makes it possible for users to retake control of their information for guaranteeing its privacy. By leveraging the well-developed technologies of cloud computing, the computing capabilities of mobile devices can be significantly enhanced in this paradigm (Abolfazli et al., 2014). Therefore, the edge computing has become quite popular for providing services in various applications, especially in mobile environments (Corcoran and Datta, 2016; Jararweh et al., 2016).

Since the IoT has been applied in a growing number of real-life applications, the performance of IoT services has become one of the most important requirements. First, the requirements of real-time adaptive sensing as well as data analytics arise significantly, especially in sensor-based IoT systems, and thus put forward strict demands on the performance of both devices at the edge layer and computers in the data centres. Second, due to the limited energy levels of IoT devices, all the resources have to be fully utilised to fulfil user requirements. Also, greening data centres at the centre level can bring significant profit to service providers, and thus all the computers as well as networking equipments should be operated with high efficiency. Third, the huge number of different links and interactions between edge nodes in IoT makes it a complex system, whose scalability on performance should be guaranteed in its whole life cycle.

In order to ensure the performance of IoT services, the very foundation is to evaluate their performance, which can be conducted from the following two aspects. The most intuitive way is to directly measure the performance after the services have been developed and deployed. The results obtained by such performance measurement are absolute metrics reflecting the characteristics of IoT services. However, the measurement is so expensive and even impractical, especially in large-scale IoT systems. Another alternative approach is to design a predictive mathematical model according to the characteristics of the services/systems at the design phase and conduct quantitative analyses of the performance metrics. Although sometimes assumptions as well as approximations have to be made in some scenarios, the model-based evaluation can provide guidance for the design and development of the systems before their establishments. Also, the quantitative results can provide theoretical support for the optimisation of system architectures and parameters.

Performance evaluation of IoT services is an emerging topic in this community. Most of the existing researches studied the performance issue from a measurement viewpoint, which analysed or optimised the performance of existing systems when they

have been deployed in reality. However, few of them designed a model-based approach of performance evaluation for well-organised IoT systems and deeply studied the model parameters. In this paper, we make an attempt at filling this gap. Theoretical models are proposed for describing dynamic characteristics of IoT services, and mathematical analyses are presented. Based on the quantitative results, some discussions on resource management and task scheduling are presented for the guidance on the design and optimisation of hierarchical IoT service systems. Finally, simulation experiments based on real-life data are conducted to validate the effectiveness of our approaches. This paper is expected to provide a theoretical predictive approach of performance evaluation for IoT services, which can be conducted with little cost before the implementation of systems at the design phase.

The remainder of this paper is organised as follows. In Section 2, we discuss related work most pertinent to this paper. In Section 3, we present fundamental models of both atomic IoT services and IoT service systems. In Section 4, we conduct mathematical analyses based on our models, and analytical solutions are presented for performance evaluation. In Section 5, we discuss two popular optimisation problems in IoT systems and show how to apply our models and analyses to system design and optimisation. In Section 6, we conduct real data-based experiments to validate our approach. Finally, we conclude the paper in Section 7.

2 Related work

Internet of Things is an emerging technique which is being widely applied in several aspects of our daily life. With the rapid development of IoT technology, researchers have begun to pay their attention on the performance issue. In this section, we briefly survey the existing work on this topic and describe prior approaches on performance evaluation, especially for IoT services.

2.1 Measurement-based approaches

Measurement is the most straightforward way for performance evaluation in IoT systems. Measurement-based performance evaluation approaches design and implement some hardware equipments or computer programs and deploy them in the real-life running systems or emulators to obtain performance metrics. All the performance data collected by measurement is mostly realistic, and measurement-based approaches are the most accurate and intuitive methods for performance evaluation.

There have been quite a few research works dedicating to performance measurement in IoT environments. Stusek et al. (2015) conducted a number of experiments for testing several OSGi-based IoT frameworks. Software programs as well as hardware testers were applied enabling comprehensive measurement within the systems. Chen and Kunz (2016) implemented a hardware-based test bed and applied network emulators to analyse the performance of several IoT protocols. Wang et al. (2015, 2016b) involved user feedbacks in performance evaluation of cloud and multimedia services and deeply studied the trust management issue of quality of service (QoS) measurement data (user feedback ratings), which could significantly improve the accuracy of performance evaluation by 50%.

2.2 Prediction-based approaches

With the rapidly growing population of services on the internet, researchers have found that real-world performance evaluation for geographically dispersed services is not an easy task. It appears to be extremely difficult and expensive, if not impossible, to evaluate all the services efficiently. Therefore, some of them have begun studying prediction-based approach for performance evaluation, which is a methodology that analyses historical performance data from previous users of invoked services and uses it to predict the performance that will be experienced by a current user on the particular service (Wang et al., 2010). Since the prediction-based approaches allow for data missing, they appear powerful strength in performance evaluation, especially in large-scale services computing systems.

Recently, there have been some seminal works that brought to the fore the prediction on performance or QoS in IoT environments. Luo et al. (2016) proposed a data-driven scheme of predicting the missing QoS values for large-scale web service-based IoT systems. A kernel machine learning algorithm was used to analyse the hidden relationships between all the known QoS data and corresponding QoS data with the highest similarities, based on which the unknown QoS values were predicted. Wang et al. (2016a) exploited the structural relationships among multidimensional QoS data and proposed a spatiotemporal model-based approach for QoS prediction, especially in mobile service environments. Moreover, the performance prediction for web services has been studied for years, which should also provide some insights. Zheng et al. (2013) applied collaborative filtering techniques and proposed a neighbourhood-integrated approach for personalised web service QoS value prediction. Furthermore, Tang et al. (2016) considered the influence of underlying network, and designed a network-aware web service QoS prediction approach by integrating matrix factorisation with the network map.

2.3 Model-based approaches

Although performance prediction has been successfully applied in several services computing systems, a part of the real-world data of existing services is mandatory. Such requirement makes it quite difficult to predict the performance of a services computing system before its design and implementation. Therefore, mathematical models built upon the structural and dynamic characteristics of the systems instead of the measurement data of existing ones are urgently required.

To attack this challenge, some researchers have proposed model-based approaches for performance evaluation. The basic idea is to build a mathematical model for formulating the dynamics of the system or services and conduct quantitative analysis of the performance according to the model parameters. These approaches make it possible to avoid from implementing and deploying any system or service in reality which is quite expensive and time-consuming. Matos et al. (2013) applied Markov chain model for the analysis of performance and reliability of web services, based on which an optimisation algorithm was designed for QoS-aware service composition. Xia et al. (2015) applied Markovian queueing models to describe cloud services and presented detailed analyses that captured various realistic features in Infrastructure-as-a-Service (IaaS) clouds. Li et al. (2014) used $M/M/k$ queueing model to anticipate the service time on IoT nodes and designed a three-layer QoS scheduling model for service-oriented IoT. Hsu et al. (2015) used system dynamics to build models of operational time, and proposed a quantitative simulation model for performance evaluation of IoT services. Zhang et al.

(2016) presented probabilistic models for analysing the performance for IoT in long-term evolution advanced heterogeneous networks with partial spectrum usage, based on which QoS provisioning is deeply studied.

However, few of the existing work studies the performance evaluation issue from a systematic viewpoint, and there lacks of a methodology that is able to comprehensively model and analyse an IoT system, especially when the system is well organised according to the edge computing paradigm. Also, the model parameters and task arrival distributions are largely unexplored. Our performance evaluation approach, to be described next, is designed to fill these gaps.

3 Basic models of IoT services

In this section, we present fundamental models of both IoT services and IoT systems. Dynamic behaviours of atomic IoT services are formulated by queueing models, while the systems consisting of a number of IoT services with complex interrelationships are described as queueing network models. Basic model parameters and analytical methodologies will be presented.

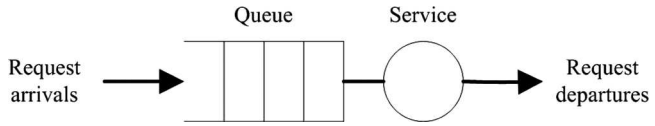
3.1 Queueing model of atomic service

An atomic service represents a type of relationship-based interactions or activities between the service provider and the service consumer to achieve a certain business goal or solution objective (Zhang et al., 2007). In an IoT system, there are a number of atomic services providing different functionalities. For example, several sensing services are deployed on the sensors at the edge levels, gathering data from the physical world and sending them to the nodes at upper layers for further operations. Another example is the data analytical services running on the data centre, which are often deployed on powerful servers and able to provide efficient analysis functionalities on massive data.

The dynamics of an atomic service include the following three fundamental parts. First, requests arrive at the service nodes for completing certain tasks according to their requirements. Such requests can be regular sensing tasks on the sensors, basic calculations of sensing data on the servers or complex data analysis in data centres. Second, since the resources at the service nodes are not unlimited, the requests sometimes have to wait in the queues until the service is available. Otherwise, arriving requests immediately proceed to the service without queueing. Third, the requests get served and finally depart from the system.

According to such dynamics, an atomic service can be formulated by a queueing model. Graphically, we represent a queueing model for IoT services as shown in Figure 1. The circle represents a service, and an open box represents a buffer (queue) preceding this service, where the slots in the queue are meant to indicate waiting requests. The requests are thought of as *arriving* at the queue and *departing* from the service, and it is assumed that the process of services normally takes a strictly positive amount of time.

Figure 1 Queueing model of atomic service



The queueing model can be precisely formulated by a discrete event system (DES), where its 'events' consist of a sequence of arrival events and departure ones. Specifically, we define the *state* of the model as the number of requests in the queue, and thus the state space forms a set of non-negative integers. In stochastic timed automata, we associate each state with an underlying clock sequence, denoted by $X(t) \in \{0, 1, 2, \ldots\}$. Moreover, we associate with arrival events a stochastic sequence $\{Y_1, Y_2, \ldots\}$ where Y_k $(k \geq 1)$ is a random variable defined by the time interval between the $(k-1)$-th and k-th arrivals and Y_1 is the time of the first arrival. Similarly, a stochastic sequence $\{Z_1, Z_2, \ldots\}$ is associated to departure events, where Z_k $(k \geq 1)$ is defined by the random variable indicating the service time for the k-th request.

Without loss of generality, we assume that the stochastic sequence $\{Y_k\}$ is independent and identically distributed, and thus the probability distribution defined by (1) is able to describe the interarrival time sequence, which is quite an important model parameter in the following discussions. Furthermore, it is customary to use the notation of (2) to define the average *arrival rate* by the inverse of $A(t)$'s mean value.

$$A(t) = \Pr(Y \leq t); \tag{1}$$

$$\lambda \equiv \frac{1}{\mathbf{E}[Y]}. \tag{2}$$

Similarly, we introduce a random variable $B(t)$ defined by (3), whose mean value indicates the average *service rate* denoted by μ shown as (4).

$$B(t) = \Pr(Z \leq t); \tag{3}$$

$$\mu \equiv \frac{1}{\mathbf{E}[Z]}. \tag{4}$$

Besides the distribution of arrivals and services, the queueing discipline which describes the order in which the server selects requests to be processed is another factor affecting the mathematical properties of the queueing system. In most of the existing service systems including IoT environments, the queueing discipline is commonly First-Come-First-Served (FCFS), and thus we adopt such discipline in the following discussions of this paper.

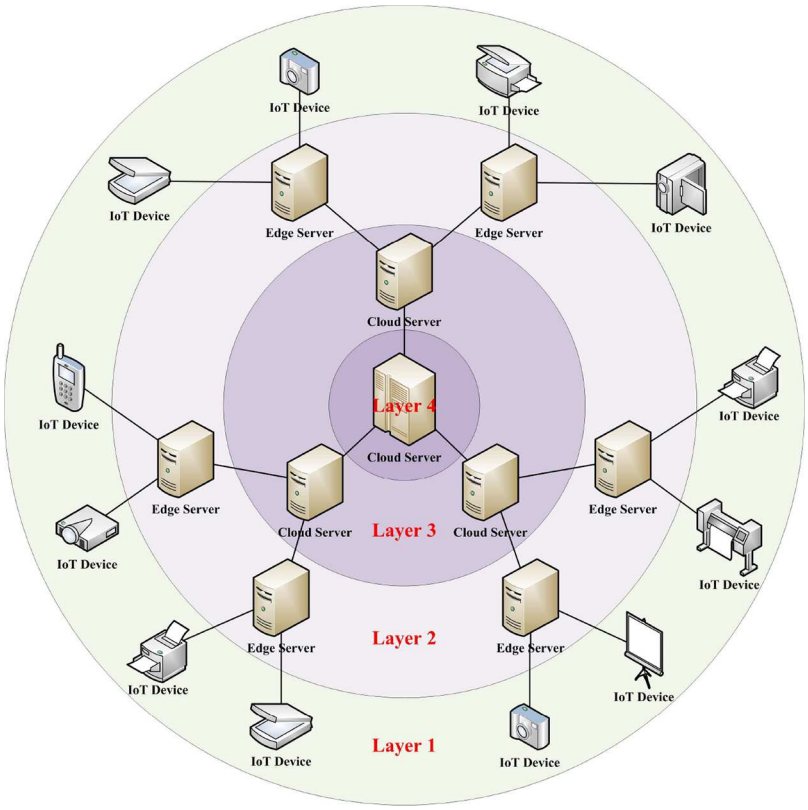
We define S_k as the response time (from arrival instant until departure) of the k-th request. With *Little's law* which provides an all-purpose steady-state performance analysis of queueing systems with any stochastic clock structure, we have (5) indicating that the average response time of a request is proportional to the mean queue length with $1/\lambda$ being the constant of proportionality.

$$\mathbf{E}[S] = \frac{1}{\lambda}\mathbf{E}[X]. \tag{5}$$

3.2 Queueing network model of IoT systems

In an IoT system which is commonly designed according to edge computing paradigm, new intermediate edge layers between IoT devices and cloud are introduced to process in part workload and services locally (Deng et al., 2016). Such layer is composed of geodistributed servers which are highly virtualised similar to lightweight cloud servers. Furthermore, since most of the cloud servers are organised in a well-defined hierarchy, the IoT system can be regarded as a hierarchically structural system, shown by Figure 2.

Figure 2 Hierarchical structure of IoT systems (see online version for colours)

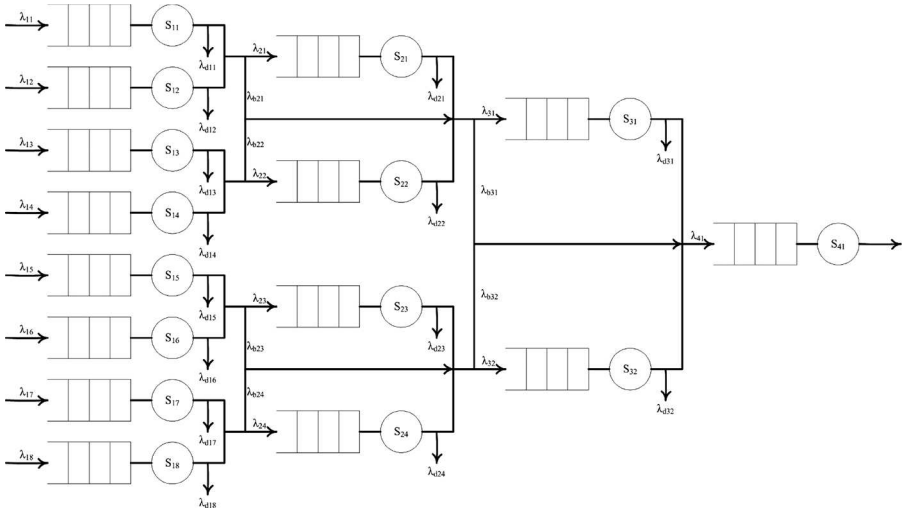


In the following discussions of this paper, we introduce some preliminary definitions on the hierarchical structure of an IoT system. According to the characteristics of edge computing paradigm, we assign each device/server to a layer, and number the layers from 1 to l. More specifically, the IoT devices gathering primitive data are classified into the first layer; the edge servers that are directly connected to the devices and processing their data are regarded as the second layer; cloud servers with more powerful configurations that handle submitted tasks from the edge servers are labelled with layer 3; and so forth. We should note that there is a possibility that some of the connections may skip one or more layers in the hierarchy. For instance, in some cloud-centric approaches, physical instruments may communicate directly with the centralised cloud servers (Lauro et al., 2012), and thus there might be a connection line from an IoT device to the cloud server at layer 4 in Figure 2. However, such situations will not affect the modelling and analysis of our performance evaluation, which will be shown in detail in the following discussions.

With the queueing model for describing atomic services presented in Section 3.1, we extend our study to an open queueing network for formulating interconnected services in a complex IoT system. According to the cooperative pattern of edge computing paradigm, requests are usually initiated by activities on IoT devices, which are handled by services on the devices. Some of the requests are quite simple that they can be completed locally on the devices, while the others need cooperations with physical servers equipped with more powerful computational resources upon their local completion. The movements from IoT

devices to upper-layer servers are formulated by directed connections from the departures of the local services to the arrivals of the next queues. For the cases that requests can be handled locally without emitting to the next layer, there exist departures from the whole system at the end of each service. At the layers where physical servers are located, it is possible for a server (except the one at the highest layer) to determine to whether process each request locally or submit it to its upper-layer service node. In order to formulate such characteristic, we assign some bypass connections before the intermediate arrivals within the queueing network model. A basic queueing network model of IoT systems is illustrated in Figure 3.

Figure 3 Queueing network model of IoT systems



Quantitatively, we conduct performance analysis of the queueing network model. First, we number each service by a two-dimensional matrix as S_{ij}, where the first indicator $i \in \{1, 2, \ldots, l\}$ illuminates the layer that the service is in and the second one $j \in \{1, 2, \ldots, n_i\}$ is the index of the service uniquely in its layer, supposing there are n_i services/devices in layer i. Second, we let λ_{ij} denote the arrival rate of service node S_{ij}, λ_{dij} be the request departure rate from the whole system at the end of S_{ij} and λ_{bij} indicate the bypass request rate from S_{ij} to its upper-layer node. Hence, the internal arrival rates can be calculated according to the connective relationships among the services. Taking the queueing network in Figure 3 for example, the arrival rate of service S_{21} can be obtained by $\lambda_{21} = (\lambda_{11} - \lambda_{d11}) + (\lambda_{21} - \lambda_{d21}) - \lambda_{d21}$. Similar to the previous subsection, we define the states of each service S_{ij} by the stochastic variable $X_{ij} \in \{0, 1, 2, \ldots\}$. Therefore, with *Little's law*, the average response time of the queueing network model can be obtained from the following expression:

$$\mathbf{E}[S] = \frac{1}{\lambda}\mathbf{E}[X] = \frac{1}{\sum_{j=1}^{n_1} \lambda_{1j}} \mathbf{E}\left[\sum_{i=1}^{l}\sum_{j=1}^{n_i} X_{ij}\right]. \tag{6}$$

With the models presented above, the performance of an IoT system can be evaluated from a modelling perspective. On the one hand, the evaluation can be conducted numerically.

With the models that describe the dynamic behaviour of the IoT system, discrete event simulation techniques can be applied, especially when we need to try out several different configurations and parameter settings in the design of systems. Besides designing and running DES simulations, some well-developed software tools for general queueing network analytics can also be very helpful in practice. Examples of the software tools include QNet Approximator (Veatch, 2005), Q-MAM (Van Velthoven et al., 2007), etc. On the other hand, analytical solutions can be obtained under some certain statistical distributions of arrivals and departures or assumptions on such distributions. Explicit mathematical expressions for quantities of interest are yielded, and the design and improvement of the IoT system can be formulated by mathematical optimisation problems. Although sometimes some assumptions have to be made in order to simplify a model, analytical solutions are able to provide valuable reference for the system behaviour to guide the system design and its optimisation with extremely small cost, making mathematical analysis a very attractive way for performance evaluation. In the next section, we will give several methodologies for obtaining analytical solutions of our models under different mathematical distributions.

4 Performance analysis of IoT services

In order to obtain analytical solutions of the queueing models, some assumptions on the distributions of arrivals or service times have to be made. Some of the assumptions are mostly in conformity with reality, while the others may give lower-bound estimations on the performance metrics. In this section, different assumptions are discussed, under which mathematical analyses are conducted. The analyses are expected to provide performance estimations of IoT systems in different scenarios and thus to guide the optimisation of system design.

4.1 Markovian queueing models

Chlebus and Brazier (2007) have shown that the task arrivals above the session level in distributed systems can be basically formulated by Poisson distribution. Such distribution is commonly an essential building block in the stochastic modelling and analysis of DES because of its exponentially distributed interevent times and the memoryless property. The properties allow us to formulate a Poisson process by Markov chains which are much easier to obtain analytical solutions.

The service times of requests are assumed to conform the exponential distribution (Guerra et al., 2008). This assumption is usually made in performance evaluation for lower-bound analysis, because of the following reason. In queueing theory, a notation c_B is introduced to indicate the relative deviation of the service times from their mean value, denoted by $c_B = \frac{\sigma_Z}{\mathbf{E}[Z]}$. For an exponential distribution, we always have $E[Z] = \sigma_Z = \frac{1}{\mu}$, and thus $c_B = 1$. This means that the standard deviation of the service time as a random variable equals to its average value, which is quite a large deviation in reality. It has been proved by basic queueing theory that the queueing system with the same arrival distribution and $c_B < 1$ has smaller average queue length and thus smaller response time (better performance). Therefore, we apply this assumption in the following part of this paper, in order to make our model robust even to some strict optimisation requirements.

The dynamics of a queueing system for an atomic service with Poisson arrivals and exponentially distributed service times can be formulated by a birth-death Markov chain

with birth rate λ and death rate μ. It is commonly assumed that $\lambda < \mu$ so the underlying Markov chain is stable. Therefore, by solving its Markov chain together with Little's theorem, the queueing system can be theoretically analysed as the following expressions, where ρ is defined as the utilisation of the service, π_i is the steady-state probability of state i, q is the average queue length and T is the mean response time in steady state.

$$\rho \equiv \frac{\lambda}{\mu}; \tag{7}$$

$$\pi_i \equiv \lim_{t\to\infty} \Pr(X(t) = i) = (1-\rho)\rho^i; \tag{8}$$

$$q \equiv \mathbf{E}[X] = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu-\lambda}; \tag{9}$$

$$T \equiv \mathbf{E}[S] = \frac{1}{\mu-\lambda}. \tag{10}$$

With these analytical solutions of a basic queueing system for an atomic service, we expand to the queueing network model. With *Burke's theorem* saying that a Poisson process supplying arrivals to a server with exponentially distributed service times results in a Poisson departure process with the exact same rate, and *Jackson's theorem* showing that a product from solution always exists in such system, it allows us to treat each service node independently in a Markovian queueing network model. Therefore, the average queue length of the whole network can be obtained by (11), and thus with Little's law, we have the analytical solution of the average response time shown by (12).

$$q \equiv \mathbf{E}[X] = \mathbf{E}\left[\sum_{i=1}^{l}\sum_{j=1}^{n_i} X_{ij}\right] = \sum_{i=1}^{l}\sum_{j=1}^{n_i}\mathbf{E}[X_{ij}] = \sum_{i=1}^{l}\sum_{j=1}^{n_i}\frac{\lambda_{ij}}{\mu_{ij}-\lambda_{ij}}; \tag{11}$$

$$T \equiv \mathbf{E}[S] = \frac{1}{\lambda}\mathbf{E}[X] = \frac{\sum_{i=1}^{l}\sum_{j=1}^{n_i}\frac{\lambda_{ij}}{\mu_{ij}-\lambda_{ij}}}{\sum_{j=1}^{n_1}\lambda_{1j}}. \tag{12}$$

4.2 Semi-Markovian queueing models

Although Poisson arrival assumption has been generally accepted at service layer, we now turn our attention to the situations in IoT systems with non-Markovian event processes. A common case of IoT services is that a sensor detects events and uploads its data at a deterministic rate. Another example is that several sensors may submit their data to the same sensor hub for integrated processing, resulting in a normal-like or exponential-like distributed arrival at the sensor hub. In the absence of memoryless (Markovian) property of these situations, we have to 'remember' the residual lifetimes of all the non-Markovian events. Therefore, the system state can no longer be specified as the queue length all by itself, leading to the difficulties of model analysis.

The service times are also assumed to be exponentially distributed for giving a lower-bound analysis. This assumption also helps us to find some special embedded Markovian time points to analytically solve the queueing model, and hence the model is called 'semi-Markovian'. In specific, these embedded time points are the times just before the request arrivals occur.

First, we analyse a queueing system of an atomic service with arbitrary distributed arrivals and exponential service times. We let L_k denote the number of requests in the system just before the k-th arriving request, and D_{k+1} be the number of requests served between the arrivals of the k-th and $(k+1)$-th requests. Thus, the following equation holds when we analyse their relationship.

$$L_{k+1} = L_k + 1 - D_{k+1}. \tag{13}$$

Since the service processes are assumed exponentially distributed, the sequence $\{L_1, L_2, \ldots\}$ naturally forms a Markov chain at the embedded time points. Its transition probabilities are defined as (14).

$$p_{ij} = \Pr\left(L_{k+1} = j | L_k = i\right), \quad i \geq 0, \;\; j \geq 0. \tag{14}$$

For all $j > i+1$, it is obvious that $p_{ij} = 0$. For $0 < j \leq i+1$, p_{ij} equals to the probability that exactly $i - j + 1$ requests are served and the service is fully utilised during the interarrival time. For the condition of $j = 0$, p_{ij} can be obtained by calculations from the previous expressions since the transition probabilities from a state should add up to one. Finally, p_{ij} can be expressed as follows, where $f_Y(\cdot)$ is defined as the probability density function (PDF) of the interarrival times.

$$p_{ij} = \begin{cases} 0; & j > i+1; \\ \int\limits_{t=0}^{\infty} \dfrac{(\mu t)^{i-j+1}}{(i-j+1)!} e^{-\mu t} f_Y(t) \mathrm{d}t; & 0 < j \leq i+1; \\ 1 - \sum\limits_{k=1}^{i+1} p_{ik}; & j = 0. \end{cases} \tag{15}$$

With the transition probabilities, one can obtain the following equations of the stationary distribution $\nu = [\nu_0, \nu_1, \ldots]$ of the embedded Markov chain.

$$\nu_0 = \sum_{i=0}^{\infty} \nu_i p_{i0}; \tag{16}$$

$$\nu_n = \sum_{i=0}^{\infty} \nu_{n+i-1} \int\limits_{t=0}^{\infty} \frac{(\mu t)^i}{i!} e^{-\mu t} f_Y(t) \mathrm{d}t, \quad n \geq 1; \tag{17}$$

$$\sum_{i=0}^{\infty} \nu_i = 1. \tag{18}$$

We solve the Eqs. (16)–(18) and yield the solutions, which have the following form:

$$\nu_n = (1-\sigma)\sigma^n, \tag{19}$$

where σ is the unique root of Eq. (20) in the range of $0 < \sigma < 1$. Here, $\tilde{Y}$ is the Laplace-Stieltjes transform of the interarrival time, i.e. $\tilde{Y}(s) = \int\limits_{t=0}^{\infty} e^{-st} f_Y(t) \mathrm{d}t$.

$$\sigma = \tilde{Y}(\mu - \mu\sigma). \tag{20}$$

Therefore, the steady-state probabilities of the semi-Markovian queueing system in each state can be obtained by (21).

$$\begin{aligned}\pi_i &= \Pr(X > i-1) - \Pr(X > i) \\ &= \rho\left[\Pr(L > i-2) - \Pr(L > i-1)\right] \\ &= \rho\nu_{i-1} \\ &= \rho(1-\sigma)\sigma^{i-1}. \end{aligned} \tag{21}$$

Finally, the average response time within an atomic service can be calculated using Little's law by (22).

$$T = \frac{1}{\lambda}\mathbf{E}[X] = \frac{1}{\lambda}\sum_{i=0}^{\infty} i \cdot \pi_i = \frac{1}{\lambda} \cdot \frac{\rho}{1-\sigma} = \frac{1}{\mu(1-\sigma)}. \tag{22}$$

With the queue lengths of all the atomic services, the overall performance metric can be easily obtained using Little's law on the queueing network model as shown in (6).

5 Discussion

In this section, we illuminate how the models and analytical results can be applied in the design and optimisation of hierarchical services computing systems that provide IoT services in edge computing paradigm. We discuss two problems, i.e. resource management and task scheduling, which are both hot topics in this community and have been studied for years. Basic problem formulations are presented and optimal solutions are provided based on analytical results introduced in the previous section.

5.1 Resource management of service nodes

Resource management, which is a fundamental research problem in computer science, is to properly allocate or fully utilise the computational and storage resources of one or a group of physical servers for meeting user requirements at certain QoS level. In services computing systems with virtualisation as its foundation underlying technique, many of the researchers in this community dedicate to study how the virtual resources should be assigned to different services in order to meet their service-level agreements (SLA) (Chen et al., 2012). By all means, resource management finally determines the performance of the services, and thus it can be basically transferred to a service rate optimisation problem in our queueing network model. The objective is to find optimal service rates in different hierarchies in order to obtain maximal profits for service providers.

In this paper, we present a primitive formulation of resource management for IoT services. Our objective is to illuminate how our modelling approach can provide theoretical reference for system design and optimisation, instead of deeply studying the resource management problem itself. Therefore, detailed schemes and techniques of resource allocation and dispatching are out of the scope of our discussion.

Basically, resource management from service provider viewpoint is to meet user SLA with (physical or virtual) resources as few as possible. We formulate resource management as an optimisation problem. The input arguments are service rates of the services above

the second layer, since the performances of the IoT devices (e.g. sensors and controllers) are commonly determined by the hardware and thus are out of the scope of resource management in services computing systems. The objective of the optimisation problem is minimising the cost brought by resource consumption expressed by $c(\mu_{ij})$, which can be directly defined by the money spend by service provides for renting virtual machines with certain predefined configurations. The constraints are defined by SLA, which can be expressed by the upper bound of the overall average response time of the system, i.e. T_{SLA}. Hence, the optimisation problem is mathematically described as follows.

$$\underset{\mu_{ij}, i>1}{\text{minimise}} \quad c(\mu_{ij}); \tag{23}$$

subject to

$$T \leq T_{\mathrm{SLA}}. \tag{24}$$

Different optimisation techniques may be applied to solve the problem. For the cases where analytical solutions can be obtained, traditional mathematical optimisation methods could help for finding the optimal configuration. On the other hand, when the performance metrics can only be evaluated by simulations, heuristic search or ordinal optimisation techniques can be applied to solve the problem with relatively high efficiency. Next, we present a case study showing how to combine our theoretical analyses and existing optimisation theories for solving the resource management problem in an IoT service system.

Example 1: We study an illustrative IoT system with Poisson arrivals and exponentially distributed service times. There are four layers and it is assumed that the services in each layer i are homogenous with the same arrival rate and service rate, denoted by λ_i and μ_i respectively. The rental costs of the services are assumed to be proportional to their service rates with the multiplicative factor c_0.

In order to solve the resource management problem, we first model the system as a Markovian queueing network. With Burke's theorem and Jackson's theorem, we are able to analyse each service node independently. The average queue length of the service buffer at layer i can be obtained by the following expression:

$$q_i = \frac{\lambda_i}{\mu_i - \lambda_i}. \tag{25}$$

With Little's law, the overall average response time of the system is expressed by:

$$T = \frac{q}{\lambda} = \frac{\sum_{i=1}^{4} q_i n_i}{\lambda_1 n_1} = \frac{\frac{\lambda_1 n_1}{\mu_1 - \lambda_1} + \frac{\lambda_2 n_2}{\mu_2 - \lambda_2} + \frac{\lambda_3 n_3}{\mu_3 - \lambda_3} + \frac{\lambda_4 n_4}{\mu_4 - \lambda_4}}{\lambda_1 n_1}. \tag{26}$$

Therefore, the formulation of optimisation problem can be specified as follows.

$$\underset{\mu_2, \mu_3, \mu_4}{\text{minimise}} \quad c(\mu_2, \mu_3, \mu_4) = c_0(n_2\mu_2 + n_3\mu_3 + n_4\mu_4); \tag{27}$$

subject to

$$\frac{\frac{\lambda_1 n_1}{\mu_1 - \lambda_1} + \frac{\lambda_2 n_2}{\mu_2 - \lambda_2} + \frac{\lambda_3 n_3}{\mu_3 - \lambda_3} + \frac{\lambda_4 n_4}{\mu_4 - \lambda_4}}{\lambda_1 n_1} \leq T_{\mathrm{SLA}}. \tag{28}$$

To solve the non-linear optimisation problem, we first transform (28) into the following expression:

$$g(\mu_2, \mu_3, \mu_4) = \lambda_1 n_1 T_{\text{SLA}} - \left(\frac{\lambda_1 n_1}{\mu_1 - \lambda_1} + \frac{\lambda_2 n_2}{\mu_2 - \lambda_2} + \frac{\lambda_3 n_3}{\mu_3 - \lambda_3} + \frac{\lambda_4 n_4}{\mu_4 - \lambda_4} \right) \geq 0. \quad (29)$$

Then we introduce a non-negative parameter θ. With *Karush-Kuhn-Tucker (KKT) conditions*, the optimum lies in the solution of the following equations:

$$\frac{\partial c(\mu_2, \mu_3, \mu_4)}{\partial \mu_2} - \theta \frac{\partial g(\mu_2, \mu_3, \mu_4)}{\partial \mu_2} = c_0 n_2 - \theta \frac{\lambda_2 n_2}{(\mu_2 - \lambda_2)^2} = 0; \quad (30)$$

$$\frac{\partial c(\mu_2, \mu_3, \mu_4)}{\partial \mu_3} - \theta \frac{\partial g(\mu_2, \mu_3, \mu_4)}{\partial \mu_3} = c_0 n_3 - \theta \frac{\lambda_3 n_3}{(\mu_3 - \lambda_3)^2} = 0; \quad (31)$$

$$\frac{\partial c(\mu_2, \mu_3, \mu_4)}{\partial \mu_4} - \theta \frac{\partial g(\mu_2, \mu_3, \mu_4)}{\partial \mu_4} = c_0 n_4 - \theta \frac{\lambda_4 n_4}{(\mu_4 - \lambda_4)^2} = 0; \quad (32)$$

$$\theta \cdot g(\mu_2, \mu_3, \mu_4) = 0; \quad (33)$$

$$\theta \geq 0. \quad (34)$$

Finally, we obtain the optimal solutions as follows.

$$\mu_2 = \frac{n_2 \lambda_2 + n_3 \sqrt{\lambda_2 \lambda_3} + n_4 \sqrt{\lambda_2 \lambda_4}}{\lambda_1 n_1 (T_{\text{SLA}} - \frac{1}{\mu_1 - \lambda_1})} + \lambda_2; \quad (35)$$

$$\mu_3 = \frac{n_2 \sqrt{\lambda_2 \lambda_3} + n_3 \lambda_3 + n_4 \sqrt{\lambda_3 \lambda_4}}{\lambda_1 n_1 (T_{\text{SLA}} - \frac{1}{\mu_1 - \lambda_1})} + \lambda_3; \quad (36)$$

$$\mu_4 = \frac{n_2 \sqrt{\lambda_2 \lambda_4} + n_3 \sqrt{\lambda_3 \lambda_4} + n_4 \lambda_4}{\lambda_1 n_1 (T_{\text{SLA}} - \frac{1}{\mu_1 - \lambda_1})} + \lambda_4. \quad (37)$$

5.2 Task scheduling within hierarchical structures

Task scheduling is another hot topic that has attracted the attention of the researchers and engineers from both academia and industry. It targets at designing proper schemes for dispatching requests to different services/servers upon their arrivals in order to obtain optimal profits while meeting user requirements. In our models, task scheduling can be regarded as determining the arrival rates of the services deployed at the physical or virtual servers.

Task scheduling in our model can be also formulated by a constrained non-linear optimisation problem. The objective is to minimise the cost brought by running services, while the input arguments are arrival rates of the services above the second layer. The constraints include two parts. First, the SLA has to be satisfied, which is expressed by an upper bound of the average response time. Second, it has to be guaranteed that all the requests are served, and thus all the arrivals from the outside to the system should be covered.

Specifically, the formulation of the optimisation problem for task scheduling is as follows.

$$\underset{\lambda_{ij}, i>1}{\text{minimise}} \quad c(\lambda_{ij}); \quad (38)$$

subject to

$$T \leq T_{\mathrm{SLA}}; \tag{39}$$

$$\sum_{i=2}^{l}\sum_{j=1}^{n_i}\lambda_{ij} \geq \sum_{j=1}^{n_1}\lambda_{1j}. \tag{40}$$

Similar to the previous part, we present an example to illuminate how to solve the problem with our models.

Example 2: We study an illustrative IoT system with Poisson arrivals and exponentially distributed service times. There are four layers and it is assumed that the services in each layer i are homogenous with the same arrival rate and service rate, denoted by λ_i and μ_i respectively. The rental costs of the services are assumed to be proportional to their arrival rates with the multiplicative factor c_i.

Parts of the quantitative analyses are similar to the ones in Example 1, and (25) and (26) also hold in this current case study. Therefore, the specified formulation of the optimisation problem is as follows.

$$\underset{\lambda_2,\lambda_3,\lambda_4}{\text{minimise}} \quad c(\lambda_2,\lambda_3,\lambda_4) = c_2n_2\lambda_2 + c_3n_3\lambda_3 + c_4n_4\lambda_4; \tag{41}$$

subject to

$$\frac{\frac{\lambda_1 n_1}{\mu_1-\lambda_1} + \frac{\lambda_2 n_2}{\mu_2-\lambda_2} + \frac{\lambda_3 n_3}{\mu_3-\lambda_3} + \frac{\lambda_4 n_4}{\mu_4-\lambda_4}}{\lambda_1 n_1} \leq T_{\mathrm{SLA}}; \tag{42}$$

$$n_2\lambda_2 + n_3\lambda_3 + n_4\lambda_4 \geq n_1\lambda_1. \tag{43}$$

The constraints are transformed into a standard formation. Equation (42) can be expressed by (44) similar to (29), while (43) is transformed to (45).

$$g(\lambda_2,\lambda_3,\lambda_4) = \lambda_1 n_1 T_{\mathrm{SLA}} - \left(\frac{\lambda_1 n_1}{\mu_1-\lambda_1} + \frac{\lambda_2 n_2}{\mu_2-\lambda_2} + \frac{\lambda_3 n_3}{\mu_3-\lambda_3} + \frac{\lambda_4 n_4}{\mu_4-\lambda_4}\right) \geq 0; \tag{44}$$

$$h(\lambda_2,\lambda_3,\lambda_4) = n_2\lambda_2 + n_3\lambda_3 + n_4\lambda_4 - n_1\lambda_1 \geq 0. \tag{45}$$

Therefore, the optimisation problem can be solved with KKT conditions, by solving the following equations.

$$\frac{\partial c(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_2} - \theta_1\frac{\partial g(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_2} - \theta_2\frac{\partial h(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_2} = 0; \tag{46}$$

$$\frac{\partial c(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_3} - \theta_1\frac{\partial g(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_3} - \theta_2\frac{\partial h(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_3} = 0; \tag{47}$$

$$\frac{\partial c(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_4} - \theta_1\frac{\partial g(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_4} - \theta_2\frac{\partial h(\lambda_2,\lambda_3,\lambda_4)}{\partial\lambda_4} = 0; \tag{48}$$

$$\theta_1 \cdot g(\lambda_2,\lambda_3,\lambda_4) = 0; \tag{49}$$

$$\theta_2 \cdot h(\lambda_2,\lambda_3,\lambda_4) = 0; \tag{50}$$

$$\theta_1 \geq 0, \theta_2 \geq 0. \tag{51}$$

Since the expressions of the analytical solutions to the optimisation problem are much more complex than the previous ones, we do not write them down here. Also, the optimisation problem can be solved in other ways. On the one hand, numerical approaches can be applied, which are able to find the optimal solution efficiently with the help of modern computers. Well-known examples of such approaches include Newton-Raphson method, Quasi-Newton method, conjugate gradient method, etc. On the other hand, some heuristic algorithms can also be helpful for finding the global optimal solutions. Approaches including genetic algorithm, simulated annealing and ant colony algorithm are proved to be effective ways for solving complex optimisation problems in reality.

6 Empirical results

In this section, we conduct experiments based on real-life data to validate our models and analyses. Discrete event simulations with real workload are designed and implemented, and their output data are finally used for assessing the validity of the theoretical analyses.

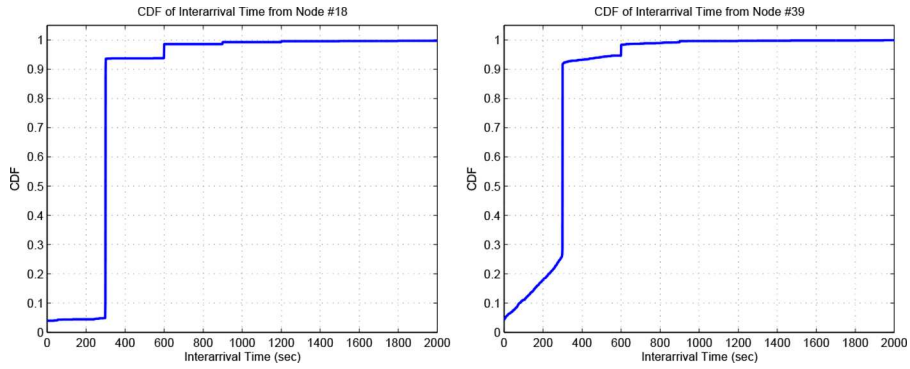
6.1 Data set

A data set named 'T-Drive' (Yuan et al., 2010, 2011) released by Microsoft Research is applied to validate the effectiveness of our model by simulation experiments. The data in this data set is collected by a number of GPS loggers and GPS phones equipped on several taxis. The data set contains the GPS trajectories of 10,357 taxis within the city of Beijing during a period of 1 week in 2008. Totally, there are nearly 15 million pieces of data, and the total distance of the trajectories reaches 9 million kilometers.

Besides detailed GPS information including the longitude and latitude, there is a timestamp in each piece of data recording the exact time of the data being submitted to the system. These timestamps provide detailed information of the task arrivals from different sensors, and the arrival distributions are key factors for our modelling and analysis. Since GPS services that gather a volume of GPS data from geographically dispersed GPS sensors and calculate the locations of the objects are typical IoT services which have been widely applied in reality, we use T-Drive data set to conduct experiments for performance evaluation.

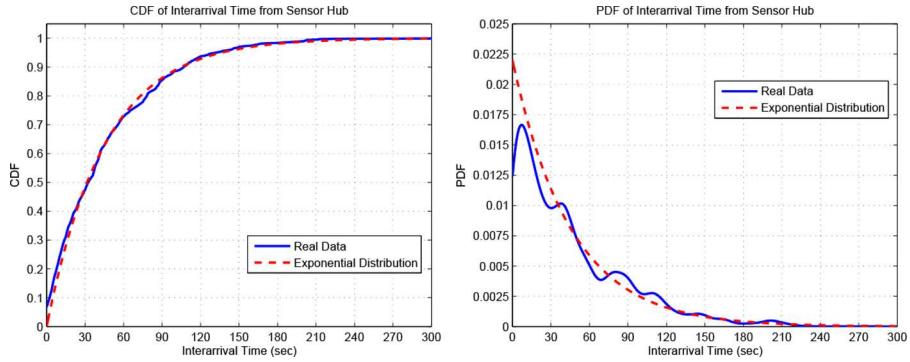
By deeply studying the arrival patterns from the data set, we obtain that the GPS sensors have a variety of sampling rates, and most of the trajectories are logged every 1–20 min. In specific, we study the task arrival distributions statistically. Figure 4 plots the distributions of time intervals between two consecutive points in two selected cars as examples. The first subfigure shows the cumulative distribution function (CDF) of the interarrival time generated from the taxi whose ID is $\#18$, which seems similar to a staircase function. One can clearly obtain from the statistic that nearly 90% of the time intervals are exactly 300 s, i.e. the GPS sensor deployed on that taxi commonly uploads its sensing data every 5 min. About 5% of the data is valued 0, while another 4% of the interarrival times is 600 s. These facts may be caused by some unusual duplicate transmissions and packet losses. From a steady-state point of view, the interarrivals can be approximately regarded as deterministic. After examining the data set, we find that the majority of the data is similar to this case. However, another example shows that the distributions of some other data are less staircase-like, shown by the second subfigure in Figure. 4. It can be concluded from the CDF that more than 95% of the data lies in the interval from 0 to 10 min (600 s), and there is a possibility of each piece of data to be any value, especially in the interval between 0 and 300 s.

Figure 4 Distributions of interarrival times on sensors (see online version for colours)



Furthermore, we aggregate all the arrivals from 10 random taxis and simulate a sensor hub gathering sensing data from multiple IoT devices. The PDF and CDF of the aggregated arrivals are shown in Figure 5. It is clear that the PDF is not a typical mathematical distribution, but we may approximately consider it as an exponential one. Therefore, all the distribution functions can be applied in our analyses presented in the previous sections, in order to make the assumptions and evaluations correspond to the reality.

Figure 5 Distribution of interarrival times for aggregated arrivals (see online version for colours)



6.2 Experimental results for atomic services

We conduct experiments simulating an atomic service handling requests generated from IoT devices. The request arrivals are directly obtained from the T-Drive data set, and the service times are randomly generated conforming to exponential distribution with the average of 150 s. Sample paths are generated according to the dynamics of the queueing system, and their output data is carefully analysed for estimating the performance of the system.

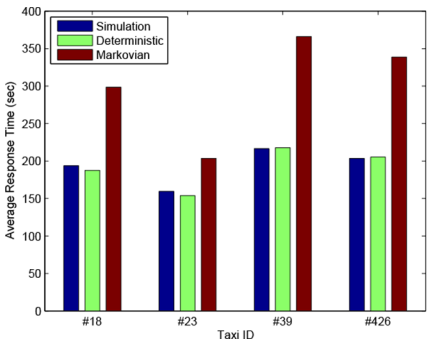
We should note that there are a few 'gaps' between some pieces of the data. For example, for taxi #18, the GPS sensor reports its location at 16:50:32, and then the next data arrive at the system more than 3 h later at 19:58:45. Although such huge gap does not affect the dynamics of the queueing system, it indeed has influence on the model analysis at certain level, because the gap will cause a large variance to the calculation of the average

arrival rate. Also, we obtain from Figure 4 that the population of the gaps is quite small, and thus they bring little effect on the dynamics of the system. Therefore, such gaps are omitted in our model parameter calculations.

We compare the experimental results with the theoretical solutions to validate the effectiveness of our approach. We conduct two groups of theoretical analysis, in which the request arrivals are assumed to conform to deterministic distribution and Poisson distribution, respectively. According to the statistical analysis in the previous subsection, it is logical that the deterministic arrival model should better correspond to the simulation results, while the analytical results obtained by the model with Poisson arrivals should provide us with a worst-case solution.

Figure 6 shows both the experimental results and the analytical solutions. On the one hand, it can be concluded from the comparisons that the relative error of our semi-Markovian model under deterministic arrival assumption is below 4%, which significantly proves the effectiveness of our approach for performance evaluation. Although some of the arrival distributions are not typically deterministic (e.g. taxis $\#39$ and $\#426$), the error can be bounded in a relatively small range. On the other hand, we obtain that the Poisson arrivals result in much higher variance to the queueing process, and hence the response time of the system with Poisson arrivals is more than 50% longer than the one with deterministic interarrival time. Therefore, the analytical results obtained by our Markovian models can be helpful for worst-case analysis, especially in the tough situations where interarrival times of the requests are of large variance.

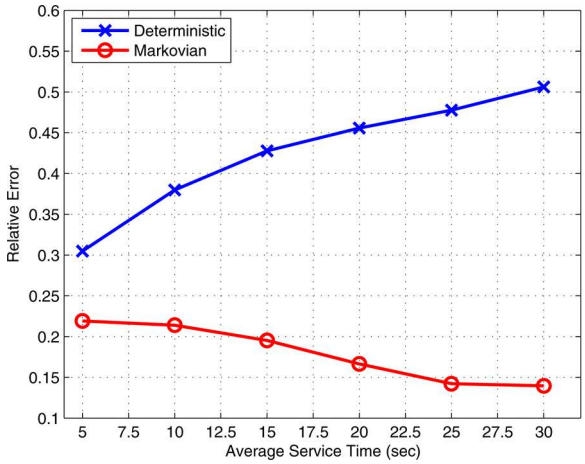
Figure 6 Empirical results of atomic services (see online version for colours)



Taxi ID	Experimental Result	Deterministic Model	Markovian Model	Relative Error
#18	193.7868	187.6446	298.5885	3.17%
#23	159.5771	153.7986	203.6725	3.62%
#39	216.5673	217.7914	365.9622	0.57%
#426	203.5256	205.3785	338.7867	0.91%

For a service node with aggregated arrivals, which simulates a sensor hub in reality, there is no precise mathematical distribution to formulate the request arrivals shown by Figure 5. Therefore, we have to try deterministic and Poisson distributions as the input of our model to predict the overall performance. Figure 7 shows the relative error rates of the prediction results obtained by our approach. First, it can be concluded that the Markovian model works much better than the deterministic one, since the distribution of arrivals is more similar to exponential other than deterministic. Second, the predictive result on performance obtained by the Markovian model is basically reasonable at certain extent, and the relative error is bounded by 25%. Third, the error rate goes down with an increase in average service time. Longer service time means higher possibility for the request to be congested in the queue, and thus the experimental results prove the suitability of the Markovian model for worst-case analysis.

Figure 7 Empirical results of sensor hubs (see online version for colours)

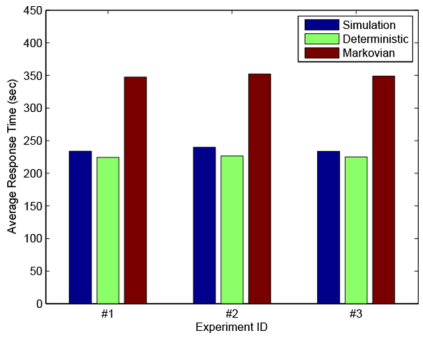


6.3 Experimental results for IoT systems

In order to validate our queueing network model, we conduct experiments simulating an IoT service system with multiple atomic services. It is assumed that there are four layers in the system, and the service nodes are well organised within the hierarchy as shown in Figure 3. Each of the GPS sensor submits its data to the respective edge server, and the edge server can possibly route the requests to its upper-layer nodes. Sensing data arrivals obtained from the T-Drive data set are used as the input of our model.

On the one hand, we validate the accuracy of our analysis by trying different groups of data that are randomly selected from the T-Drive data set. Figure 8 shows the comparisons between the experimental results and theoretical analyses. The results indicate that our queueing network model with the deterministic arrival assumption is able to precisely predict the performance of the system, and the predicting errors are mostly in the range of 3–6% with different groups of data. Also, the Markovian model is proved more suitable for other arrivals with larger deviation.

Figure 8 Empirical results of IoT systems (see online version for colours)

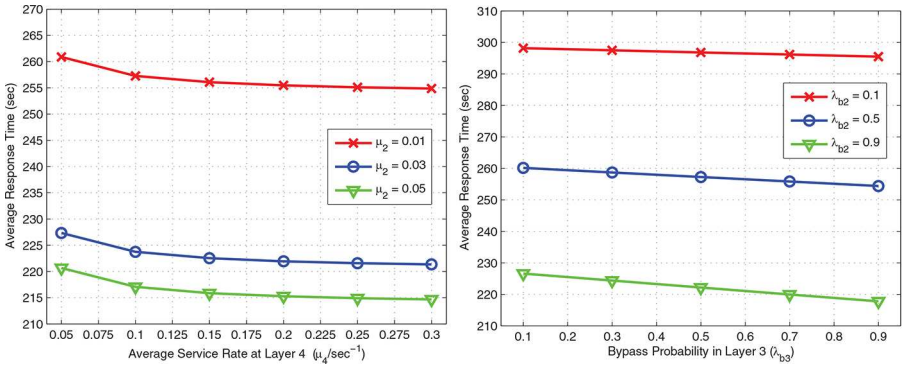


Experiment ID	Experimental Result	Deterministic Model	Markovian Model	Relative Error
#1	233.8625	224.4584	347.4407	4.02%
#2	239.8189	226.5554	352.1432	5.53%
#3	233.6540	225.1433	349.0132	3.64%

On the other hand, we tune the model parameters with the same task arrivals to see how they affect the overall performance of the system, and thus brief suggestions on system

optimisation on the parameters can be proposed. The results obtained by these experiments are shown in Figure 9. The first subfigure shows the experimental results when we tune the service rates within the edge layer (layer 2) and the central cloud layer (layer 4), respectively. It is shown that the increase of the service rate of the edge nodes (μ_2) is able to bring much more benefit on the average response time than increasing μ_4, and thus the services at the edge layer are the performance bottleneck in these parameter settings. Therefore, with limited resources, the edge services should be upgraded at the first place in this situation. The second subfigure demonstrates an example on task scheduling. We tune the bypass probabilities to assign different workload levels to different layers. Since the cloud services are set with higher service rate, it is logical for the average response time going down with an increase in bypass probabilities. Also, it can be obtained from the figure that transforming the requests from edge layers to cloud layers (i.e. increasing λ_{b2} in our queueing model) significantly benefits the performance, comparing with tuning the workload assignment within the multilayer cloud servers. Therefore, the resource management and task scheduling of the service nodes at the edge layer are quite important issues in IoT systems and should be well treated accordingly.

Figure 9 Empirical results with different parameters in IoT systems (see online version for colours)



7 Conclusion

Performance has always been an important issue for IoT services and systems. In this paper, we present a theoretical approach of performance modelling and analysis for IoT services. An atomic service is formulated by a queueing model, and detailed quantitative analysis is conducted for different task arrival distributions. Then a hierarchical services computing system that provides IoT services in edge computing paradigm can be modelled by a queueing network, and the methodology of its performance analysis is presented. Two popular research problems in this community which are resource management and task scheduling are discussed as an applicative case study, in order to illustrate the connection between model analysis and optimisation. Finally, simulation experiments based on real-life data sets are conducted, which validate the effectiveness of our models and analyses. This work is expected to provide system designers and managers with a predictive approach for performance evaluation without implementing the services and systems in

reality, which can be helpful for their design and optimisation with high efficiency and little cost.

There are several avenues for the future work. First, detailed distributions of task arrivals and service times can be further studied in different systems, and their corresponding queueing models and related mathematical analyses can be specified making our performance evaluation more precise. Second, the methodology of system modelling and performance analysis presented in this paper can be extended to other types of systems such as fog computing systems, hierarchical cloud systems, etc. The patterns of task arrivals and interconnection relationships among service nodes can be formulated by queueing network models, and their performance metrics can be calculated using the similar methodology. Third, optimisation problems can be further specified according to user requirements, and efficient algorithms for solving them need to be investigated accordingly. Finally, it will be an interesting and valuable work to apply the models and algorithms to real-world large-scale systems. The experimental results obtained from reality can provide us with better understanding of the model description, estimation of parameters, and effectiveness and overhead of the algorithms.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 61502043 and No. 61132001), Beijing Natural Science Foundation (No. 4162042), BeiJing Talents Fund (No. 2015000020124G082) and the Fundamental Research Funds for the Central Universities (No. 2015RC22).

References

Abolfazli, S., Sanaei, Z., Ahmed, E., Gani, A. and Buyya, R. (2014) 'Cloud-based augmentation for mobile devices: motivation, taxonomies, and open challenges', *IEEE Communications Surveys Tutorials*, Vol. 16, No. 1, pp.337–368.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. and Ayyash, M. (2015) 'Internet of Things: a survey on enabling technologies, protocols, and applications', *IEEE Communications Surveys Tutorials*, Vol. 17, No. 4, pp.2347–2376.

Chen, Y. and Kunz, T. (2016) 'Performance evaluation of IoT protocols under a constrained wireless access network', *Proceedings of the 2016 International Conference on Selected Topics in Mobile Wireless Networking (MoWNeT 2016)*, 11–13 April, 2016, Cairo, Egypt, pp.1–7.

Chen, W., Qiao, X., Wei, J. and Huang, T. (2012) 'A profit-aware virtual machine deployment optimization framework for cloud platform providers', *Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD 2012)*, 24–29 June, 2012, Honolulu, HI, US, pp.17–24.

Chlebus, E. and Brazier, J. (2007) 'Nonstationary poisson modeling of web browsing session arrivals', *Information Processing Letters*, Vol. 102, No. 5, pp.187–190.

Corcoran, P. and Datta, S.K. (2016) 'Mobile-edge computing and the Internet of Things for consumers: extending cloud computing and services to the edge of the network', *IEEE Consumer Electronics Magazine*, Vol. 5, No. 4, pp.73–74.

Deng, R., Lu, R., Lai, C., Luan, T.H. and Liang, H. (2016) 'Optimal workload allocation in fog-cloud computing towards balanced delay and power consumption', *IEEE Internet of Things Journal*, Vol. PP, No. 99, pp.1–1.

Garcia Lopez, P., Montresor, A., Epema, D., Datta, A., Higashino, T., Iamnitchi, A., Barcellos, M., Felber, P. and Riviere, E. (2015) 'Edge-centric computing: vision and challenges', *ACM SIGCOMM Computer Communication Review*, Vol. 45, No. 5, pp.37–42.

Guerra, R., Leite, J. and Fohler, G. (2008) 'Attaining soft real-time constraint and energy-efficiency in web servers', *Proceedings of the 2008 ACM Symposium on Applied Computing (SAC 2008)*, 16–20 March, 2008, Fortaleza, Ceara, Brazil, pp.2085–2089.

Höller, J., Tsiatsis, V., Mulligan, C., Karnouskos, S., Avesand, S. and Boyle, D. (2014) *From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*, Academic Press, Oxford.

Hsu, A.P.T., Lee, W.T., Trappey, A.J.C., Trappey, C.V. and Chang, A.C. (2015) 'Using system dynamics analysis for performance evaluation of IoT enabled one-stop logistic services', *Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2015)*, 9–12 October, 2015, Kowloon Tong, Hong kong, pp.1291–1296.

Jararweh, Y., Doulat, A., AlQudah, O., Ahmed, E., Al-Ayyoub, M. and Benkhelifa, E. (2016) 'The future of mobile cloud computing: integrating cloudlets and mobile edge computing', *Proceedings of the 23rd International Conference on Telecommunications (ICT 2016)*, 16–18, May, 2016, Thessaloniki, Greece, pp.1–5.

Lauro, R.D., Lucarelli, F. and Montella, R. (2012) 'SIaaS - sensing instrument as a service using cloud computing to turn physical instrument into ubiquitous service', *2012 IEEE 10th International Symposium on Parallel and Distributed Processing with Applications*, 10–13, July, 2012, Madrid, Spain, pp.861–862.

Li, L., Li, S. and Zhao, S. (2014) 'QoS-aware scheduling of services-oriented Internet of Things', *IEEE Transactions on Industrial Informatics*, Vol. 10, No. 2, pp.1497–1505.

Luo, X., Liu, J., Zhang, D. and Chang, X. (2016) 'A large-scale web QoS prediction scheme for the industrial Internet of Things based on a kernel machine learning algorithm', *Computer Networks*, Vol. 101, pp.81–89.

Matos, R.S., Maciel, P.R. and Silva, R.M. (2013) 'QoS-driven optimisation of composite web services: an approach based on GRASP and analytical models', *International Journal of Web and Grid Services*, Vol. 9, No. 3, pp.304–321.

Stusek, M., Hosek, J., Kovac, D., Masek, P., Cika, P., Masek, J. and Kröpfl, F. (2015) 'Performance analysis of the OSGi-based iot frameworks on restricted devices as enablers for connected-home', *Proceedings of the 7th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT 2015)*, 6–8 October, 2015, Brno, Czech republic, pp.178–183.

Tang, M., Zheng, Z., Kang, G., Liu, J., Yang, Y. and Zhang, T. (2016) 'Collaborative web service quality prediction via exploiting matrix factorization and network map', *IEEE Transactions on Network and Service Management*, Vol. 13, No. 1, pp.126–137.

Van Velthoven, J., Van Houdt, B. and Blondia, C. (2007) 'Simultaneous transient analysis of QBD Markov chains for all initial configurations using a level based recursion', *Proceedings of the 4th International Conference on the Quantitative Evaluation of Systems (QEST 2007)*, 17–19 September, 2007, Edinburgh, UK, pp.79–88.

Veatch, M.H. (2005) *Approximate Dynamic Programming for Networks: Fluid Models and Constraint Reduction*, Technical Report, Gordon College.

Wang, S., Ma, Y., Cheng, B., Yang, F. and Chang, R. (2016a) 'Multi-dimensional QoS prediction for service recommendations', *IEEE Transactions on Services Computing*, Vol. PP, No. 99, pp.1–12.

Wang, S., Sun, L., Sun, Q., Wei, J. and Yang, F. (2015) 'Reputation measurement of cloud services based on unstable feedback ratings', *International Journal of Web and Grid Services*, Vol. 11, No. 4, pp.362–376.

Wang, S., Sun, Q. and Yang, F. (2010) 'Towards web service selection based on QoS estimation', *International Journal of Web and Grid Services*, Vol. 6, No. 4, pp.424–443.

Wang, S., Zhou, A., Lei, W., Yu, Z., Hsu, C.-H. and Yang, F. (2016b) 'Enhanced user context-aware reputation measurement of multimedia service', *ACM Transactions on Multimedia Computing, Communications and Applications*, Vol. 12, No. 4s, pp.59:1–59:18.

Xia, Y., Zhou, M., Luo, X., Zhu, Q., Li, J. and Huang, Y. (2015) 'Stochastic modeling and quality evaluation of infrastructure-as-a-service clouds', *IEEE Transactions on Automation Science and Engineering*, Vol. 12, no. 1, pp.162–170.

Yuan, J., Zheng, Y., Xie, X. and Sun, G. (2011) 'Driving with knowledge from the physical world', *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2011)*, 21–24 August, 2011, San Diego, CA, US, pp.316–324.

Yuan, J., Zheng, Y., Zhang, C., Xie, W., Xie, X., Sun, G. and Huang, Y. (2010) 'T-drive: driving directions based on taxi trajectories', *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS 2010)*, 2–5 November, 2010, San Jose, CA, US, pp.99–108.

Zhang, R., Wang, M., Shen, X. and Xie, L.L. (2016) 'Probabilistic analysis on QoS provisioning for Internet of Things in LTE-A heterogeneous networks with partial spectrum usage', *IEEE Internet of Things Journal*, Vol. 3, No. 3, pp.354–365.

Zhang, L.-J., Zhang, J. and Cai, H. (2007) *Services Computing*, Springer, Berlin, Heidelberg.

Zheng, Z., Ma, H., Lyu, M.R. and King, I. (2013) 'Collaborative web service QoS prediction via neighborhood integrated matrix factorization', *IEEE Transactions on Services Computing*, Vol. 6, No. 3, pp.289–299.