Statistically managing cloud operations for latency-tail-tolerance in IoT-enabled smart cities

Daniel Sun a,b,d,*, Guoqiang Li b, Yuanyuan Zhang c, Liming Zhu a,d, Raj Gaire a

a Data61, CSIRO, Australia
b School of Software, Shanghai Jiao Tong University, China
c National Computational Infrastructure, Australia
d University of New South Wales, Australia

HIGHLIGHTS

- The impact on latency variability due to cloud operations is formulated for IoT-enabled Smart City.
- Our online management does not take any assumptions on predictability and periodicity.
- Our policy manipulates two parameters to control priorities and is more flexible than the others.
- A simulation is driven by the real data and demonstrates the advantage of our proposed policy.

ABSTRACT

Smart City is typically large scale and IoT-enabled online services for huge amount of streaming data generated by sensors, and the services are often deployed in clouds, where infrastructure and services need to be maintained and optimised to achieve the best quality of service by using various cloud operations that are running in the background of Smart City business. However, on one hand the background operations inevitably press a negative impact on the latency of normal requests to the services, while on the other hand to maintain the short latency of requests usually results in unreasonably long latency of background operations. In this paper, on the architecture of IoT-enabled Smart City services deployed in clouds, our motivation is to find the best management policy of operations for normal request traffic, which is stable and stationary and to which operations are inserted. We focus on the system environment in which administrators are dealing with in most time other than discuss the special cases, such as outages or rush hours, and take the assumptions e.g. predictability and periodicity. The suitable management techniques for us are a class of online management policies, which are simple and applicable in practice. These policies harmonise the requests and the operations by giving higher priority to the normal requests other than forcing the background operations to wait unreasonably. We also propose a statistical policy, which is asymptotically optimal in the class. In order to compute the numeric solution, we employ a queueing model to analyse our policy. The evaluation indicates that our technique can effectively maintain the short latency of the requests, and meanwhile the latency of operations is not unreasonably long in comparison to the competitors.

1. Introduction

Although no formal definition of “Smart City” has been widely accepted, informally Smart City refers to conceptually adopting ICT (Information and Communication Technology) solutions in the management of public affairs for many national governments [1].

In typical architecture of Smart City, sensors are networked into a cloud via Internet of Things (IoT), and streaming data are analysed in the cloud for decision making. As a new communication paradigm, IoT has become a major means linking data generating devices and data processing platforms for added-value services. The data instances generated by sensors and transmitted to the clouds via IoT are cleaned, transformed, analysed and stored by data processing workflows and services in clouds. The requests to each service component may be sent by sensors, users, or other components.
IoT-enabled Smart City is a set of large scale services, for which reducing latency is essential for responsive functionalities. Innovations and improvements on network connectivity, software design, system configuration, scalability, replication and redundancy and so on are all more or less helpful. However, just like unavoidable failures, in web services, latency variability cannot be eliminated thoroughly due to the scale and the complexity [2]. Many factors can influence latency variability such as resource sharing, maintenance, queuing, power, and memory management, background activities and operations. Redundant resource, careful real-time engineering of software, and improved reliability can narrow down the variability, but in this paper we study how background operations and queuing can influence latency variability, and focus on system level solutions for managing operations and harmonising operations and requests effectively. Our goal is the art of balance other than arbitrarily postponing the operations.

Technically, request traffic can be represented as time series data, and using time series analysis and prediction has been reported to optimally allocate idle time slots for various purposes [3–7]. In real-time systems, system workload may have the property of periodicity. In this case, bandwidth reservation can be employed to harmonise periodic tasks and sporadic tasks [8]. However, in the context of IoT-enabled Smart City, the assumptions on predictability and periodicity are not available. Although an individual sensor may generate data periodically, thousands of sensors only present random generation of data. The assumptions are off the practice completely. This is the reason why administrators and practitioners have to adopt naive methods to maintain short latency tail in practice. Usually, these methods do sacrifice background operations for normal requests, and this results in significant delay of operations, i.e. long latency tail of operations.

This paper aims at maintaining short latency for both requests and operations, given statistically stable systems and queues. Computer and network systems have been successfully modelled by queuing theory, in which the utilisation is a key indicator. Theoretically, the waiting time in a queue will not grow unlimitedly if the utilisation is less than 1 in many queuing systems. Thus, inserting operations into a request traffic properly can help reach our goal. Operation management has been studied and can schedule operations to the idle computing machines. However, without the strong assumptions on predictability and periodicity, few policies can work well to maintain short latency for both requests and operations. To solve this problem, we propose a statistical management policy, which makes use of the processing capacity that remains from the normal workload. In order to compute the optimal parameters, we resort to a priority queuing model from which the asymptotical optimality can be computed. We measure the key metrics and collect the data from a small and real IoT system with some sensors and a service deployed in a cloud. Then, we use the data for scaling-up in a simulation, in which we can evaluate arbitrary possibilities, scales, and workflows. The results show that our statistical management can reach close to the best reference or even to a system that assumes no operation at all. This implies that the remaining capacity of the system can be used efficiently. In summary, our contribution in this paper is the proposed management technique, which is simple, optimal, and applicable in practice.

The remainder of this paper is organised as follows. In Section 2, we review related work. In Section 3, we introduce and discuss the details about IoT enabled smart cities, long latency tail, cloud operations, our system model, and the goal of this paper. In Section 4, we introduce the general form of the statistical management policy, and then the modelling and the computation of the optimality are also presented. In Section 5, our experiments are described, and the results are presented. Section 6 concludes this paper and justifies our contributions.

2. Related work

Using IoT and cloud computing for Smart City has been well discussed in concept [1,9] and implemented in practice [10,11]. In the ecosystem of IoT, big data, clouds, and Smart City services, there are still many technical challenges that need to be dealt with [12,13]. One is that Smart City needs latency-tolerant services, while in clouds it is difficult to assure timely responses. One of the difficulties is the management of the background operations running aside the services in clouds. Cloud operations are inevitable for dependability and availability of cloud systems [14–18].

Latency tail has been noticed in interactive services, which often have large-scale parallel implementations, and in order to assure fast responses the median and the tail of latencies of a service’s components must be low [19]. In clouds, the variability of latency is caused by many factors, e.g. resource sharing, maintenance, queuing, power or energy management, garbage collection, and so on. Most of them are carried out as cloud operations. Differentiating service classes and higher-level queuing, reducing head-of-line blocking, and managing background activities are most effective means to reduce component-level variability [2]. In this paper, our focus is on queuing, service classes, and managing background activities, and we do not sacrifice background operations arbitrarily.

In dedicated computational systems, prediction grants system administrators effective privilege to optimise systems [3–6]. However, accurate prediction is only feasible if the detailed knowledge can be acquired and the predictable systems are not so complicated as Smart City. For example, if the workload is periodic, bandwidth reservation can be applied [20–22]. Rich information and knowledge can also help build queuing models for managing operations [23,24], and once a system can be represented by such a queuing model, the optimisation becomes much easier. Compared to many queuing models, our model deals with more than one type of clients. We need to consider both normal requests to Smart City services and background operations of clouds.

Serving two or more classes of clients in a queuing system has been studied in the literature. Usually the classes are granted different priorities, and the clients are served in terms of the class priorities. In [25,26], low priority clients are served only if less than a number of servers are busy. In [27], a probabilistic method is proved to be optimal in the class of policies that make decisions for low-priority clients on the system load. The essence of these techniques is that the idle time can be utilised up and meanwhile the system utilisation is controlled [7]. In this paper, we adopt this type of policies for normal requests to Smart City services and background operations in clouds. We develop our theory by extending and correcting [27] for IoT-enabled Smart City.

Most management techniques control either the threshold of system utilisation or the probability of running low priority tasks as shown above. Our proposed technique controls both of the two parameters in order to well exploit the system potential, and achieves not only the lowest latency of requests but also the possibly lowest latency of operations with guaranteeing the priority. The technical concern is that the probability of blocking requests and operations by previous ones is asymptotically minimised.

3. Preliminaries

In order to clearly introduce our technique, we need to clarify the concepts, the problem, and the scope of this paper in the first place. In this section, first, we discuss the problem of long latency, and show an example. Second, we enumerate the background operations in IoT-enabled smart city services deployed in clouds, and specify the operations suitable for our techniques. Third, we define the system model on which our technique is proposed.
دریافت فوری
متن کامل مقاله

امکان دانلود نسخه تمام متن مقالات انگلیسی
امکان دانلود نسخه ترجمه شده مقالات
پذیرش سفارش ترجمه تخصصی
امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
امکان دانلود رایگان ۲ صفحه اول هر مقاله
امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
دانلود فوری مقاله پس از پرداخت آنلاین
پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات