ARTICLE IN PRESS

Mechatronics xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

Cloud resource allocation for cloud-based automotive applications☆

Zhaojian Li[a], Tianshu Chu[b], Ilya V. Kolmanovsky[c], Xiang Yin[1,*,d], Xunyuan Yin[e]

[a] Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48824, USA
[b] Department of Civil and Environmental Engineering, Stanford University, CA 94305, USA
[c] Department of Aerospace Engineering, The University of Michigan, Ann Arbor, MI 48109, USA
[d] Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China
[e] Department of Chemical and Materials Engineering, University of Alberta, Edmonton, AB T6G 1H9, Canada

ARTICLE INFO

Keywords:
Resource allocation
Vehicle-to-Cloud
Chance constrained optimization
Communication delays
Deep deterministic policy gradient
Reinforcement learning

ABSTRACT

There is a rapidly growing interest in the use of cloud computing for automotive vehicles to facilitate computation and data intensive tasks. Efficient utilization of on-demand cloud resources holds a significant potential to improve future vehicle safety, comfort, and fuel economy. In the meanwhile, issues like cyber security and resource allocation pose great challenges. In this paper, we treat the resource allocation problem for cloud-based automotive systems. Both private and public cloud paradigms are considered where a private cloud provides an internal, company-owned internet service dedicated to its own vehicles while a public cloud serves all subscribed vehicles. This paper establishes comprehensive models of cloud resource provisioning for both private and public cloud-based automotive systems. Complications such as stochastic communication delays and task deadlines are explicitly considered. In particular, a centralized resource provisioning model is developed for private cloud and chance constrained optimization is exploited to utilize the cloud resources for best Quality of Services. On the other hand, a decentralized auction-based model is developed for public cloud and reinforcement learning is employed to obtain an optimal bidding policy for a "selfish" agent. Numerical examples are presented to illustrate the effectiveness of the developed techniques.

1. Introduction

There is growing interest in employing cloud computing in automotive applications [4,8,14,17,25,30–32]. Ready access to distributed information and computing resources can enable computation and data intensive vehicular applications for improved safety, drivability, fuel economy, and infotainment. Several cloud-based automotive applications have been identified. For instance, a cloud-based driving speed optimizer is studied in [22] to improve fuel economy for everyday driving. In [16], a cloud-aided comfort-based route planner is prototyped to improve driving comfort by considering both travel time and ride comfort in route planning. A cloud-based semi-active suspension control is studied in [15] to enhance suspension performance by utilizing road preview and powerful computation resources on the cloud.

As such, cloud computing has been both an immense opportunity and a crucial challenge for vehicular applications: opportunity because of the great potential to improve safety, comfort, and enjoyment; challenge because cyber-security and resource allocation are critical issues that need to be carefully considered. A cloud resource allocation scheme determines how a cloud server such as Amazon "EC2" or Google Cloud Platform distributes resources to its many clients (vehicles in our context) efficiently, effectively, and profitably. This allocation design becomes even more challenging when it comes to cloud-based automotive systems in which issues like communication delays and task deadlines arise. These complexities make a good resource allocation design a non-trivial, yet important task.

Not surprisingly, extensive studies have been dedicated to the development of efficient and profitable cloud resource allocation schemes. A dynamic bin packing method, MinTotal, is developed in [13] to minimize the total service cost. In [5], a distributed and hierarchical component placement algorithm is proposed for large-scale cloud systems. A series of game theoretical cloud resource allocation approaches have also been developed, see e.g., [2,3,11,19]. However, as far as the authors are aware, a resource allocation scheme for cloud-based automotive systems that accounts for communication delays and task deadlines is still lacking.

In this paper, we develop resource allocation schemes for cloud-based automotive systems that optimally tradeoff costs and Quality of Service

☆ This paper was recommended for publication by Associate Editor Junmin Wang.

* Corresponding author.

E-mail addresses: lizhaoj1@egr.msu.edu (Z. Li), cts1988@stanford.edu (T. Chu), ilya@umich.edu (I.V. Kolmanovsky), xiangyin@umich.edu (X. Yin), xunyuan@ualberta.ca (X. Yin).

[1] The material in this paper was not presented at any IFAC conference.

https://doi.org/10.1016/j.mechatronics.2017.10.010
Received 31 December 2016; Received in revised form 18 September 2017; Accepted 27 October 2017

(QoS) with the presence of stochastic communication delays and task deadlines. In particular, we consider allocation schemes under two cloud paradigms, private and public cloud. A private cloud is a company-owned resource center which provides computation, storage and network communication services and is only accessible by cars made by the car company. The private cloud therefore has a high level of security and information is easy and safe to share and manage. On the other hand, a public cloud relies on a third-party service provider (e.g., Amazon EC2) that provides services to all subscribed vehicles. A public cloud can eliminate the capital expenses for infrastructure acquisition and maintenance, and can provide the service on an as-needed basis.

The objectives of resource allocation are quite different between private and public cloud paradigms. Since the private cloud resources are pre-acquired, the company basically "use them or waste them". Therefore, the goal of private cloud resource allocation is to best utilize its resources to provide good QoS to its subscribed vehicles. Since the information exchange between vehicles and the server is more secure and convenient, the resource allocation can be achieved in a centralized manner. On the other hand, public cloud provides services to subscribed vehicles from a variety of makers, e.g., Ford, GM, Toyota, etc. Due to security and privacy issues, these vehicles typically will not share their information nor be interested in coordination; hence each vehicle becomes a "selfish" agent. The goal of each agent is to minimize its service cost while maintaining good QoS.

In this work, we develop mathematical models to formalize the resource allocation problems for both private and public cloud paradigms. Stochastic communication delays and onboard task deadlines are explicitly considered. A centralized resource-provisioning scheme is developed for private cloud and chance constrained optimization is employed to obtain an optimal allocation strategy. On the other hand, an auction-based bidding framework is developed for public cloud and reinforcement learning is exploited to train an optimal bidding policy to minimize the cost while maintaining good QoS. Numerical examples are presented to demonstrate the effectiveness of the proposed schemes.

The main contributions of this paper include the following. Firstly, compared to the previous literature on cloud resource allocation, issues important to automotive vehicles such as communication delays and onboard task deadlines are explicitly treated in this paper. Secondly, resource allocation within a private cloud paradigm is formalized as a centralized resource partitioning problem. Chance constrained optimization techniques are employed to obtain the optimal partitioning by solving a convex optimization problem. Thirdly, a decentralized, auction-based bidding framework is developed for public cloud-based resource allocation and the best response dynamics assuming a constant time delay and bidding is derived. Furthermore, a Deep Deterministic Policy Gradient (DDPG) algorithm is exploited to train the optimal bidding policy with stochastic time delay and unknown bidding from other vehicles. Sensitivity analysis is also performed to show how the bidding policy can change by varying task parameters such as workload and deadline.

The rest of our paper is organized as follows. Section 2 describes the model of cloud resource provisioning for private cloud-based automotive systems. The problem formulation and a chance constrained optimization approach are also presented. In Section 3, a numerical example is given to illustrate the allocation scheme for private cloud. The resource allocation problem with a public cloud is formalized in Section 4. The best response dynamics with constant time delay and bidding is also derived. A DDPG algorithm is exploited in Section 5 to train the optimal bidding policy with stochastic time delay and unknown bidding from other vehicles. A numerical case study is also presented with sensitivity analysis on task parameters. Finally, conclusions are drawn in Section 6.

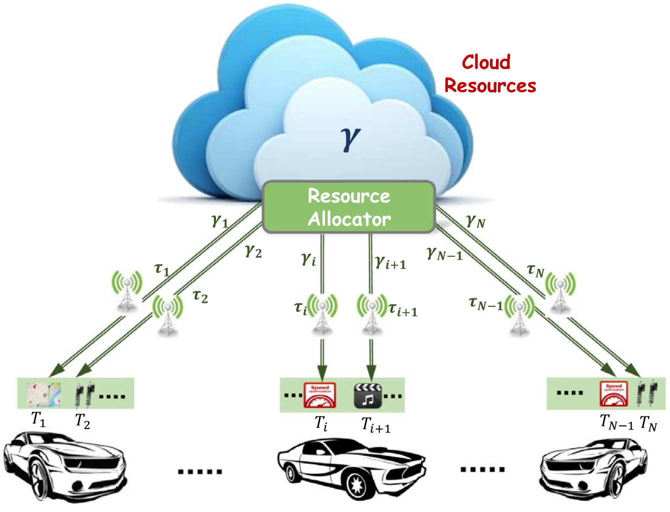

Fig. 1. Schematic diagram of private cloud-based resource allocation.

2. Centralized resource allocation with a private cloud

It is more secure and manageable for automotive manufacturers to acquire and maintain its own private cloud infrastructure which provides computation, data storage and network services only to vehicles made by the manufacturer. A schematic diagram of resource allocation for private cloud-based automotive systems is illustrated in Fig. 1. Suppose that a set of cloud-based vehicular applications are available (e.g., cloud-based route planning, cloud-based suspension control, etc.) and we consider a general case that each vehicle runs a subset of these applications. Let us consider a total number of N applications running on M vehicles as in Fig. 1. Each application i, $i = 1, 2, \ldots, N$, corresponds to a periodic task associated with a tuple, $\mathscr{T}_i = \{T_i, w_i, d_i, \tau_i\}$, where

- T_i is the period of task i in seconds;
- w_i is the workload of task i in million instructions;
- $d_i \leq T_i$ is the deadline of task i in seconds;
- τ_i is a random time delay of the communication channel associated with task i in seconds.

For each task i, the Quality of Service (QoS) is characterized by the following cost function adopted from [33]:

$$C_i(\gamma_i; \tau_i) = \begin{cases} B_i\left(\frac{w_i}{\gamma_i} + \tau_i\right), & \text{if } \frac{w_i}{\gamma_i} + \tau_i \leq d_i \\ M_i, & \text{Otherwise,} \end{cases} \tag{1}$$

where γ_i is the process rate that the cloud resource allocator assigns to task i and $\sum_{i=1}^{N} \gamma_i = \gamma$ with γ being the total resource available on the cloud; $B_i(\cdot)$: $\mathbb{R}^+ \rightarrow \mathbb{R}^+$ is a non-decreasing function reflecting the QoS of task i; $M_i \geq B_i(d_i)$ is a positive scalar representing the penalty for missing the deadline; the condition $\frac{w_i}{\gamma_i} + \tau_i > d_i$ indicates that the deadline has been missed. Note that task priorities are reflected in the deadline-missing penalty M_i. For safety-critical tasks (e.g., cloud-based functions involved in powertrain or vehicle control), a large penalty, M, should be given while a small M can be assigned to some non-critical tasks such as online video streaming.

Since a private cloud is a pre-acquired "use it or waste it" capability, the goal of resource allocation for private cloud-based automotive systems is to distribute the cloud resources to the N tasks such that the total expected QoS cost as in (1) is minimized. Basically, the cloud collects task information (i.e., workload, deadline, time delay statistics[2]) of the N tasks and determines how optimally to partition the total resources into N parts so that the expected QoS cost is minimized. The problem can be mathematically formalized as a constrained optimization problem

[2] Note that the task period T_i is not used here but we include it as one of the four task attributes for completeness. The task period will appear when it comes to the public cloud-based resource allocation in Section 4.