Vehicular Delay-Tolerant Networks for Smart Grid Data Management Using Mobile Edge Computing

Neeraj Kumar, Sherali Zeadally, Joel J. P. C. Rodrigues

The authors investigate the use of VDTNs as one of the solutions for data dissemination to various devices in the SG environment using mobile edge computing. In the proposed architecture, most of the computation for making decisions about charging and discharging is done by mobile devices such as vehicles located at the edge of the network (also called mobile edge computing).

ABSTRACT

With the widespread popularity and usage of ICT around the world, there is increasing interest in replacing the traditional electric grid by the smart grid in the near future. Many smart devices exist in the smart grid environment. These devices may share their data with one another using the ICT-based infrastructure. The analysis of the data generated from various smart devices in the smart grid environment is one of the most challenging tasks to be performed as it varies with respect to parameters such as size, volume, velocity, and variety. The output of the data analysis needs to be transferred to the end users using various networks and smart appliances. But sometimes networks may become overloaded during such data transmissions to various smart devices. Consequently, significant delays may be incurred, which affect the overall performance of any implemented solution in this environment. We investigate the use of VDTNs as one of the solutions for data dissemination to various devices in the smart grid environment using mobile edge computing. VDTNs use the store-and-carry forward mechanism for message dissemination to various smart devices so that delays can be reduced during overloading and congestion situations in the core networks. As vehicles have high mobility, we propose mobile edge network support assisted by the cloud environment to manage the handoff and the processing of large data sets generated by various smart devices in the smart grid environment. In the proposed architecture, most of the computation for making decisions about charging and discharging is done by mobile devices such as vehicles located at the edge of the network (also called mobile edge computing). The computing and communication aspects are explored to analyze the impact of mobile edge computing on performance metrics such as message transmission delay, response time, and throughput to the end users using vehicles as the mobile nodes. Our empirical results demonstrate an improved performance 10–15 percent increase in throughput, 20 percent decrease in response time, and 10 percent decrease in the delay incurred with our proposed solution compared to existing state-of-the-art solutions in the literature.

INTRODUCTION

Vehicular delay-tolerant networks (VDTNs) have gained lot of popularity over the last few years due to their usage in a large number of delay-sensitive applications such as video streaming, gaming, and safety. VDTNs use the mechanism of store-carry-and-forward in which one of the vehicles may act as the relay node for message transmission between the source and the destination. The relay vehicles keep the message with them until suitable vehicles within range are found. However, message transmission to the recipient vehicles is a challenging task due to high velocity and the constant topological changes of vehicles in this environment [1].

Electric vehicles perform energy trading with a centralized authority known as the smart grid (SG), which consists of a large number of smart devices which may be deployed in different geographical regions for data collection. For example, smart meters and sensors may be deployed across different areas to collect electricity consumption data from different homes as well as to monitor household activities. The data collected from different homes may be transferred to the nearest access points (APs) using short-, medium-, and long-range communications. The collected electricity consumption data from various smart devices in smart homes is used to estimate demand and supply so that a unified policy can be designed for smooth operation of power scheduling and redistribution in the SG environment [2, 3].

As the amount of data generated from various smart devices is huge and highly heterogeneous, we require an efficient computing infrastructure such as the cloud. In this case, smart devices offload the data to the cloud. Using a cloud computing infrastructure, data can be placed at a centralized/distributed repository, which can be accessed from anywhere, anytime. As the vehicles move around, mobility management and data acquisition by these vehicles need to be analyzed in the SG environment. As vehicles move, they

Neeraj Kumar is with Thapar University; Sherali Zeadally is with the University of Kentucky; Joel J. P. C. Rodrigues is with National Institute of Telecommunications (Inatel), Instituto de Telecomunicações, UBI, and ITMO University.

0163-6804/16/$25.00 © 2016 IEEE

may access data from the cloud through seamless handoffs provided by the APs and gateways when they cross boundaries and coverage areas of APs. On the other hand, the processing of data at the cloud servers needs to be performed efficiently so that a quick response can be sent to the vehicles in order that they can make timely and optimized decisions for route optimization and energy trading from the grid. Using the cloud computing infrastructure, vehicles can access resources from anywhere, anytime, which increases the efficiency of data transmission and communication in this environment [4, 5].

The Need for Mobile Edge Computing

With the popularity of smart city and big data, the use of unconventional sources of energy such as photovoltaic array (PV), fuel cells, hydro, and wind energy sources are expected to increase many-folds in the years to come. During this era, plug-in hybrid electric vehicles (PHEVs) may act as one of the distributed energy sources from which energy can be taken from and supplied back to the SG at any instant. PHEVs are vehicles that operate on both conventional and unconventional sources of energy. As decisions about energy transfers are made in real time, there is a requirement of computing in the proximity of users so that delay can be minimized; at the same time, users need access to computing and storage resources closer to them. To stabilize the grid with respect to power, a large number of PHEVs act as distributed energy resources. For example, a large number of PHEVs may participate in making decisions about charging and discharging from SG. However, as PHEVs move, long delays may be incurred if the data is processed by using the core-based infrastructure, which may be located far away from the actual deployment of the PHEVs. For example, Amazon EC2 and Windows Azure public core-based infrastructure may be deployed at distant locations. One solution to this problem is the usage of cloudlet-based mobile data offloading [4]. However, cloudlet-based solutions are based on the usage of a WiFi network, which may have some coverage problem, along with limited bandwidth and frequent disconnections. Hence, there is a need for computational processing at the edge of the network so that delays can be reduced. Although in the past some efforts have been made using mobile edge computing [4, 6, 7], these solutions have addressed mainly the energy consumption issue of the users' mobile devices. The devices consume more energy during the access of various network resources such as bandwidth and number of channels. In contrast to the traditional core-based infrastructure, our solution provides better computation offloading with reduced delay and response time, and an overall increase in throughput.

Research Contributions of this Work

Based on the above discussion, the main research contributions of this work can be summarized as follows:

- We propose an architecture for the integration of mobile edge computing and the SG environment. Using the proposed architecture, vehicles can make decisions about charging and discharging from various charging stations deployed across a geographical location.
- We propose a virtual machine migration approach that aims to minimize the energy consumption at the data center.
- We evaluate the performance of our proposed scheme and compare it with the existing core-based scheme with respect to various metrics.

Organization

The rest of the article is structured as follows. The next section discusses the proposed architecture used for SG data management using a mobile cloud computing environment. Then we present an analysis in terms of computing and communication operations in edge and core cloud computing. Following that, we describe the energy consumption and optimization at data centers using efficient VM utilization. Next, we evaluate the proposed scheme using various performance metrics. Finally, we conclude the article.

Architecture for Smart Grid Integration with Mobile Cloud

Figure 1 shows a generalized architecture for SG integration with the mobile cloud environment using edge computing. In the mobile cloud environment, most of the devices (e.g., vehicles, computers, laptops, PDAs) used for data dissemination and communication with the grid are mobile. The concept of mobile edge computing has emerged because of the need to provide fast access to various services to mobile clients [7, 8]. As the core cloud computing infrastructure may be overloaded due to the large number of requests, we may have performance degradation with respect to the execution and maintenance of various real-time applications that require a fast response time. Hence, some of the applications can be executed at the edge of the network for shorter response times. Consequently, there has been some interest in mobile edge computing recently [4, 6]. As shown in the figure, there is a four-layer architecture in the mobile cloud environment and the SG environment. These four layers consist of mobile devices at the lowest layer followed by utilities and the communication infrastructure backbone consisting of routers and gateways deployed in the backbone. The routers and gateways are used for routing and handoffs when the mobile devices that are connected to vehicles move from one network domain to another. PHEVs are the vehicles that operate on both conventional and unconventional sources of energy. The data from the backbone infrastructure is captured by data centers (layer 3), which are distributed at different geographical locations. The data centers store the data collected, process it at the edge of the network, and send the results to each PHEV on time. The data centers house different types of servers such as database servers (for storage and data repository), file servers (for maintaining different files since the virtualized environment may contain different operating systems), a certificate authority server (for certificate distribution and security maintenance) at various levels in this environ-

With the popularity of smart city and big data, the usage of unconventional sources of energy such as photovoltaic array, fuel cells, hydro, and wind energy sources is expected to increase many-folds in the years to come. During this era, plug-in hybrid electric vehicles may act as one of the distributed energy sources from which energy can be taken from and supplied back to the SG at any instant.

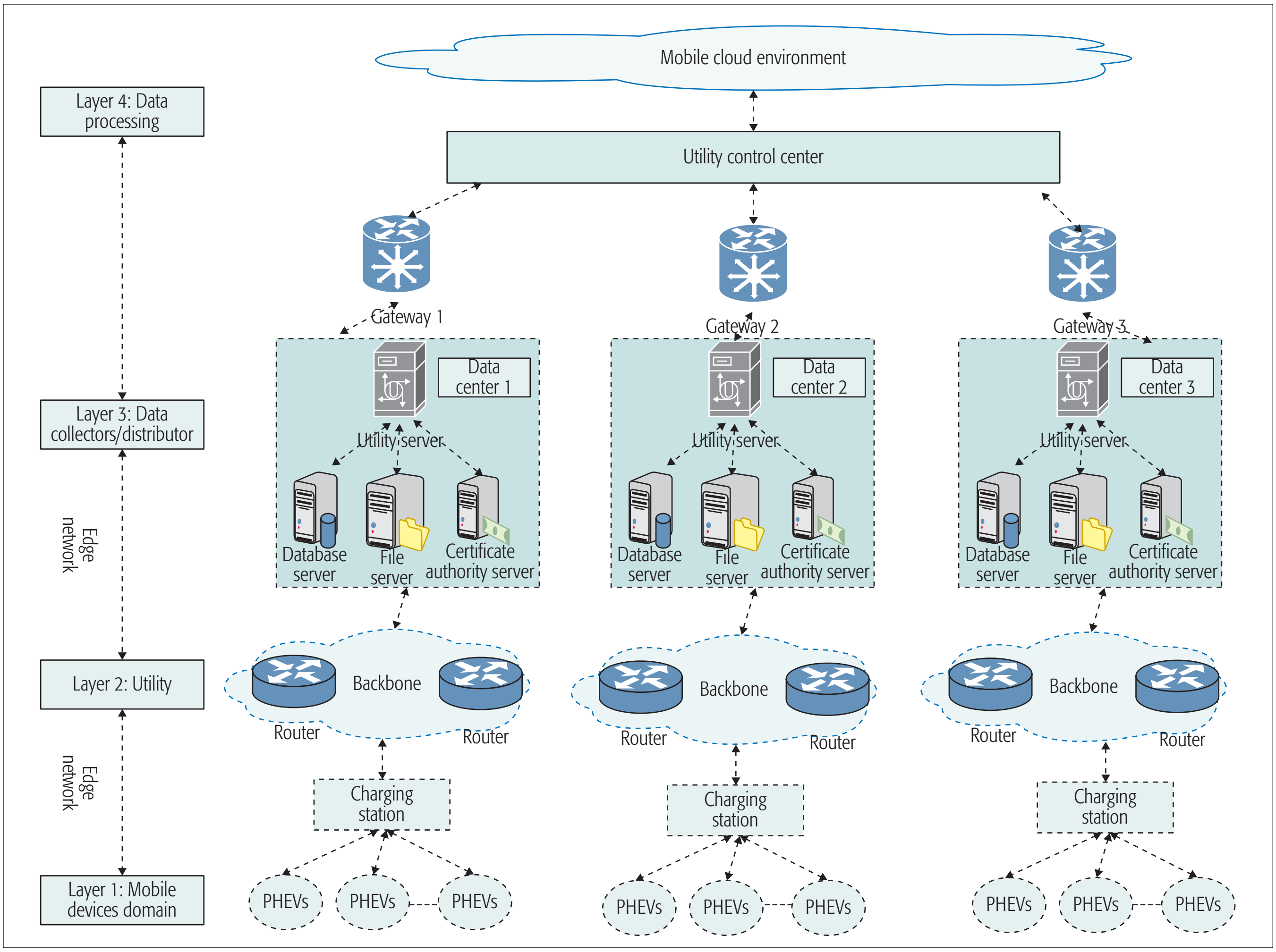

Figure 1. Architecture of the proposed system.

ment. Finally, at layer 4, a core distributed cloud environment is located where infrastructure as a service (IaaS) is assumed in our proposed architecture. These four layers coordinate with each other to execute various jobs in this environment [9–11].

Mobile Edge Computing

Computing and Communication at the Edge: An Analysis

As shown in Fig. 1, many mobile devices are connected to the mobile environment at the edge of the network, and are used to execute various programs related to charging and discharging activities of vehicles. For load shedding using an intelligent scheduler at the cloud, various jobs/applications are executed at either the nearest APs or the data centers. This reduces the burden of executing all the jobs at the centralized cloud server. This type of computing paradigm, where many jobs can be executed at the boundary of the cloud infrastructure, is called mobile edge computing [7, 8].

Figure 2a illustrates the delay incurred using mobile edge computing and the traditional core cloud-computing-based infrastructure for various real-time applications. We have collected real-time traces of the Amazon EC2 cloud data center and our own private cloud data center over a 24-hour period. As shown in the figure, with an increase in the number of interconnected devices, the response time for various real-time applications is reduced. This is mainly because mobile edge computing provides various resources closer to users so that they can execute their applications without performance degradation [8].

Efficient Virtual Machine Utilization and Load Balancing for Optimized Energy Consumption Using Mobile Edge Computing

As we stated earlier, a large amount of heterogeneous data is generated from various compute-intensive applications across the edge network. For example, PHEVs make decisions about charging and discharging with the SG, which causes a lot of information to be exchanged between the SG and the PHEVs. Hence, we need effective design strategies that can optimize energy consumption when various resources are used across the edge network. Mobile gateways or APs may become overloaded as they execute various instructions in the core network. Mobile devices have limited resources (computing power, storage, etc.). Parallel applications can be executed by efficient virtual machine (VM) utilization, which is one of the most challenging tasks to perform in a mobile edge computing network because the high mobility of the devices causes frequent

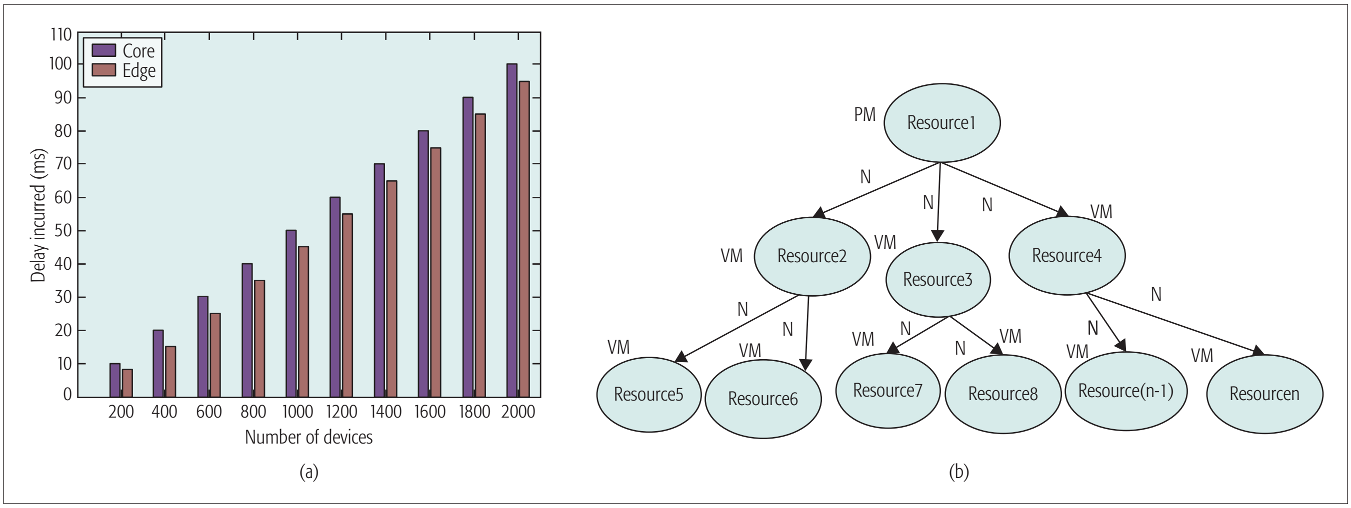

Figure 2. Results for a) impact on the delay incurred with an increase in the number of devices; b) decision tree representation.

handoff between the devices and network. The VM migration for load balancing and parallel execution of various applications have been studied in our earlier work [12].

Several tasks are performed for efficient VM migration and utilization in our proposed scheme. These are described as follows. First, all the requests from the edge nodes are passed to the nearest data collector, which collects and analyzes the data and communicates with the nearest APs (e.g., to the road side units [RSUs]in the case of vehicular ad hoc networks). These RSUs are connected to the backbone at layer 2 (as shown in Fig. 1). As depicted in the figure, PHEVs act as the mobile nodes in the proposed approach. The data generated by these PHEVs may be in the form of demand generated as a result of charging and discharging of PHEVs at the nearest charging stations (CSs). As the requests generated for charging and discharging are on a real-time basis, the proposed architecture of mobile edge computing is used. Moreover, the real-time pricing for PHEVs is also communicated by the utility from time to time so that these PHEVs can make decisions with respect to charging and discharging to reap the maximum benefits from the service provider, which, in the case of mobile edge computing, is the utility located at the edge of the network.

Data Center Configuration and Placement for Energy Optimization

Recent advances in cloud computing and information technology enable users to access hardware, software, or PaaS on the fly. However, energy optimization at data centers remains one of the most challenging tasks because it often requires specialized solutions. Since most of the applications will be executed from resource-constrained mobile devices to cloud servers, minimizing energy consumption through the effective utilization of various resources at the cloud data centers is an important goal. The development and usage of information and communications technologies (ICT) is one of the major factors causing an increase in energy consumption and global warming. This is due to the increase in application offloading from handheld devices to data center servers. As a result, the maintenance and operational costs of these data centers have increased exponentially in recent years, prompting the owners of the data centers to investigate viable solutions that can minimize energy consumption in these data centers.

Data centers may contain various clusters that reside at different geographical locations maintained by the different utilities in the SG environment. The vehicles may make decisions with respect to charging and discharging from any of the charging stations controlled by the utilities. These utilities have different pricing policies which are frequently advertised. A cluster is selected based on the pricing policy at different regions along with the load balancing strategy being used. This strategy helps in optimizing the energy consumption at the cloud data centers. Based on the available price at different data centers/clusters, each vehicle trades energy to reap the maximum benefit. During off peak hours, the vehicles consume energy from the grid and store additional energy for future usage. During peak hours they discharge their stored energy so that it can be utilized for load balancing purposes. Thus, vehicles/PHEVs act as distributed energy resources from which energy can be retrieved (vehicle to grid) as well as transferred to (grid to vehicle).

Energy Optimization with Smart Charging

This section describes how our proposed solution is effective in optimizing energy consumption at the data centers using smart charging with mobile edge computing support. At the data centers, multiple jobs are executed by creating a virtual environment using VMVare and XEN [8]. As discussed above, VM migration and utilization are used for smart energy management. Vehicles in the proposed solution act as distributed energy sources that participate in both charging and discharging decisions. In our approach, we have used a Bayesian cooperative coalition game approach in which each vehicle is assumed to be the player, and data centers located at the utilities are assumed to be mul-

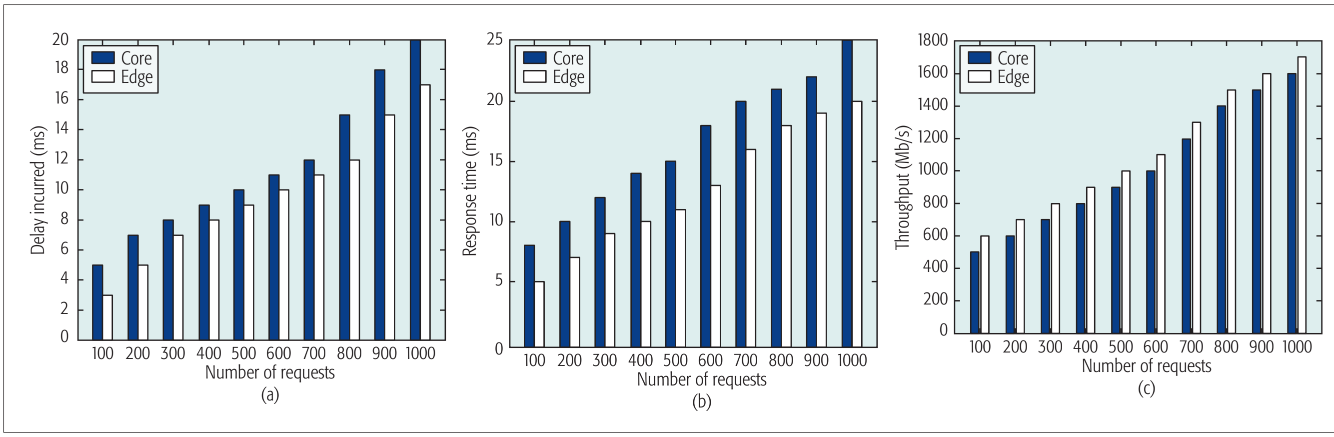

Figure 3. Results for: (a) variation of delay incurred with number of requests; b) variation of response time with the number of requests; c) variation of throughput (megabits per second) with the number of requests.

tiple service providers. Each vehicle is given a variable payoff function based on its action and is computed as follows:

$$Payoff = (operational^{cost} + maintenance^{cost}) - (p) \quad (1)$$

where, pricing p is computed as follows:

$$p = N \times t^{slot} \times dur \times freq \quad (2)$$

where N represents the number of vehicles, t^{slot} represents the time slot in which vehicles charge or discharge, dur represents the duration for which charging is done, and $freq$ represents the number of times vehicles charge or discharge from the CS. All these factors are used to compute the price offered to a moving vehicle so that it can decide on charging or discharging. For example, if a vehicle's frequency $freq$ is higher at a charging station, it may be offered charging at some discount. Similarly, if it is during peak hours, the vehicle may be offered a different charging rate compared to off-peak hours.

The charging rates at the charging stations can be varied with respect to the demand and supply at the charging stations. For instance, during peak hours, the charging rate may be high, while during off-peak hours the charging rate may be lower. Customers want to avail of the best deal with respect to charging and discharging as per the rates available at different locations. By reducing the load on the control center, the energy consumption with respect to the computation of multiple jobs is reduced because intelligent decisions are made based on the payoff of the players as defined in Eq. 2. Players and their payoff functions are represented as a directed acyclic graph in the proposed solution. In one unit interval of time, only those players whose payoff values are above a predefined threshold are allowed to perform an action such as charging and discharging. There are multiple thresholds for the players in the game. Initially, all the players are assumed to play the game with the same probability, but as the game proceeds, payoff values of the players get changed according to their actions.

Live Virtual Machine Migration Using Load Forecasting

In this section, we describe how our proposed scheme performs a live migration of VMs using adaptive load forecasting at the data center. Data centers use many physical machines (PMs) which may be located at different geographical locations, and these PMs consume a lot of energy [13]. Each PM may host many VMs, which may coexist on a single PM running many different applications. A VM needs to be allocated to a PM for the successful execution of various applications in the virtualized environment. Two aspects, load and global load, need to be considered for VM migration on PM for efficient resource utilization. Local load is computed on the PM with respect to the CPU, memory utilization, and I/O resources used, while global load is computed with respect to the current status of all the PMs. The communication load includes network bandwidth, number of channels, and interference from the neighboring channels. The following equations are used to allocate VMs to the PMs both at the local and global levels:

$$load^{local} = N \times n^{app}$$

$$load^{global} = B \times n^{ch} \times I$$

$$CPU^{util} = \sum_{1}^{n} service_i$$

where N represents the total number of VMs competing for PMs for application execution. For this purpose, they require CPUs, memory, and I/O resources. n^{app} is the number of applications running on the VMs. B is the bandwidth allocated, n^{ch} is the number of channels required, and I is the interface. CPU^{util} represents the CPU utilization with respect to a finite number of services, i, running on the CPUs.

In the proposed scheme, we have used a decision-tree-based structure for making the decision about the global scheduling for VM migration from one location to another. This scheduler is called the *decision tree-based VM migration manager* (DTVMM). The input parameter to DTVMM is $load^{local}$. Based on this input parameter, *DTVMM* creates a global schedule that it uses to decide where to shift the VMs so that effective resource utilization can be done without violating a service level agreement (SLA) with respect to

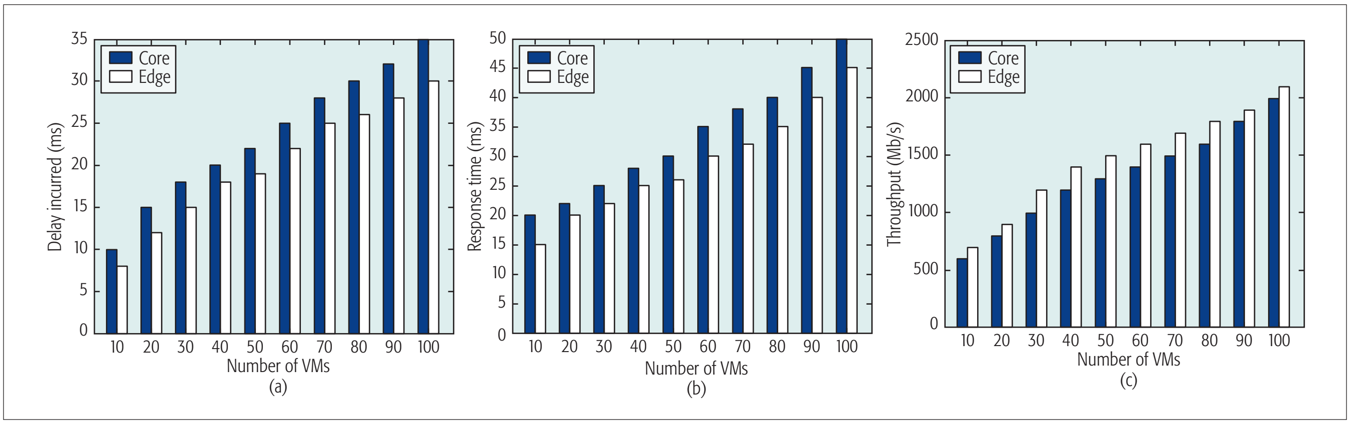

Figure 4. Results for: a) variation of delay incurred with the number of VMs; b) variation of response time with the number of VMs; c) variation of throughput (megabits per second) with the number of VMs.

various quality of service (QoS) parameters. As shown in Fig. 2b, various resources ($Resource_1$, $Resource_2$, …, $Resource_n$) are represented at different locations in the decision tree. The various resources considered in the proposal are CPU utilization, memory usage, and I/O requests. The size of the decision tree expands or shrinks depending on the allocation of VMs using Eq. 3. Two types of scheduling are considered in the proposed scheme: local and global. In the case of local scheduling, the load of one VM is transferred to another VM without SLA violations, while a global scheduler is used to determine where to shift the VMs for better resource utilization [14].

The major decisions about what to migrate, where to migrate, and when to migrate are made by the global scheduler in the proposed scheme. Various attributes such as network bandwidth available, current load on the source and destination, number of signal levels, distance between cloud data centers and other factors are also considered. In the preliminary stage, the hosts where VMs are to be migrated are identified. Based on the load on the hosts, a preliminary list of underloaded and overloaded machines is set up using the decision tree method. The global scheduler uses this list to decide where VMs are to be transferred for load balancing in the proposed solution.

Performance Evaluation

For all simulation tests conducted, the topology shown in Fig. 1 was used where a finite number of PHEVs submit their requests for charging from CSs. For the evaluation of the proposed scheme, 100 charging stations of capacities of 5000 W for homes and 50 charging stations of 20,000 W in neighborhood area networks are considered. 100 VMs with 2 GB capacity each are considered. We evaluate the proposed scheme by using the following performance metrics on ns-2 [12]:

- *Delay incurred:* This is the time it takes to carry the total number of data units from the source to the destination by the vehicles.
- *Response time:* This is the time it takes for the first response to come after processing of a dataset.
- *Throughput:* This is the total number of data units delivered per unit time interval.

Figure 3a shows the variation of the delay incurred with the number of requests generated from a finite number of PHEVs. As shown in the figure, with an increase in the number of requests generated for services from the service providers attached with the data centers, there is an increase in the delay incurred. But this delay incurred is lower in the proposed scheme as compared to when the core-based cloud computing infrastructure is used, demonstrating the effectiveness of the proposed scheme. This is because all computations are done at the edge of the network where datasets about charging and discharging from the grid are offloaded onto the edge nodes, resulting in faster computations for making the decisions about charging and discharging. We obtained similar results when the number of VMs is varied (as shown in Fig. 4a).

Figure 3b shows the response time of end users waiting for various services from the grid. When the number of services is increased, the proposed edge computing approach yields a lower response time compared to the approach making use of the traditional core cloud computing infrastructure. The lower response time is because when the core is used, services need to be accessed far away from remote sites that reduce the overload on the core. In the proposed scheme, the services are immediately available from the edge computing resources, thereby minimizing the response time. Hence, lower delays are incurred with the proposed scheme compared to the core infrastructure. Similar results are observed in Fig. 4b when the number of VMs is varied.

Figure 3c shows the variation of throughput using the proposed edge computing approach and the traditional core-based cloud computing infrastructure. As observed from the results obtained, the proposed edge computing scheme delivered higher throughput compared to the core-based cloud computing infrastructure. As before, the improvements are mainly because of the availability of services at the edge of the network. Applications running at the edge of the network take less time to offload the data about charging and discharging to the cloud environment, so the execution time of such applications is lower because of high resource availability for all such applications. As shown in Fig. 4c, similar results are obtained with the variation of the number of VMs. Tables 1 and 2 summarize the improvements obtained with our proposed scheme compared to the core approach for the various performance metrics.

Compared to the traditional core-based cloud computing infrastructure, mobile edge computing leads to lower delays and response times, and high throughput for many applications because most of the devices used in mobile edge computing need various services at the edge of the network.

Performance metrics	Percent improvement range with edge computing approach over core approach for different number of requests (100–1000)
Delay (ms)	15–30%
Response time (ms)	20–40%
Throughput (Mb/s)	10–30%

Table 1. Percent improvements of the proposed scheme with variation in number of requests.

Performance metrics	Percent improvement with edge computing approach over core approach for different number of VMs (10–100)
Delay (ms)	15–40%
Response time (ms)	20–50%
Throughput (Mb/s)	15–30%

Table 2. Percent improvements of the proposed scheme with variation in number of VMs.

Conclusion

Mobile edge computing has emerged as one of the most promising networking solutions. Compared to the traditional core-based cloud computing infrastructure, mobile edge computing leads to lower delays and response times, and high throughput for many applications because most of the devices used in mobile edge computing need various services at the edge of the network. In this work, we propose a vehicular delay-tolerant network-based smart grid data management scheme that leverages the mobile edge computing paradigm. In the smart grid environment, plug-in hybrid electric vehicles need to charge using the nearest charging stations. During this process, many requests may be generated for charging at the charging stations. To handle all these requests, we propose the use of vehicular delay-tolerant networks to deliver messages related to charging and discharging decisions along with the safety messages from source to destination so that real-time decisions can be made. Both computing and communications issues are explored in the proposed scheme. The performance evaluation results obtained clearly indicate that mobile edge computing is a more effective solution for mobile data offloading to the cloud compared to the traditional core-based cloud computing infrastructure.

In the future, we will explore game theoretical aspects of the proposed mobile edge computing for dynamic energy trading with the smart grid.

Acknowledgments

We thank the anonymous reviewers for their valuable comments which helped improve the quality and presentation of this article. This work has been partially supported by Instituto de Telecomunicacoes, Next Generation Networks and Applications Group (NetGNA), Covilha Delegation, the Government of Russian Federation, Grant 074-U01, and by National Funding from the FCT — Fundacao para a Ciencia e a Tecnologia through the UID/EEA/500008/2013 Project.

References

[1] S. Mehar *et al.*, "Sustainable Transportation Management System for a Fleet of Electric Vehicles," *IEEE Trans. Intelligent Transportation Systems*, vol. 16, no. 3, 2015, pp. 1401–14.
[2] N. Kumar, N. Chilamkurti, and S. C. Misra, "Bayesian Coalition Game for the Internet of Things: An Ambient Intelligence-Based Evaluation," *IEEE Commun. Mag.*, vol. 53, no. 1, Jan. 2015, pp. 48–55.
[3] Guangjie Han *et al*., "Two Novel DoA Estimation Approaches for Real Time Assistant Calibration System in Future Vehicle Industrial," *IEEE Sys. J*., 2015; DOI: 10.1109/JSYST.2015.2434822.
.[4] X. Chen *et al*., "Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing," *IEEE Trans. Net*., DOI:10.1109/TNET.2015.2487344.
[5] N. Kumar *et al*., "Playing the Smart Grid Game: Performance Analysis of Intelligent Energy Harvesting and Traffic Flow Forecasting for Plug-In Electric Vehicles," *IEEE Vehic. Tech. Mag*., vol. 10, no.4, 2015, pp. 81–92.
[6] D. Huang, P. Wang, and D. Niyato, "A Dynamic Offloading Algorithm for Mobile Computing," *IEEE Trans. Wireless Commun*., vol. 11, no. 6, June 2012, pp. 1991–95.
[7] J. N. G. Isento *et al*., "Vehicular Delay-Tolerant Networks – A Novel Solution for Vehicular Communications," *IEEE Intelligent Transportation Systems Mag*., vol. 5, no. 4, Winter 2013, pp. 10–19.
[8] A. Mondal and S. Misra, "Game-Theoretic Energy Trading Network Topology Control for Electric Vehicles in Mobile Smart Grid," *IET Netwroks*, vol. 4, no. 4, 2015, pp. 220–28.
[9] P. B. F. Duarte *et al*., "On the Partially Overlapped Channel Assignment on Wireless Mesh Network Backbone: A Game Theoretic Approach," *IEEE JSAC*, vol. 30, no. 1, 2012, pp. 119–27.
[10] A. V. Vasilakos and G. I. Papadimitriou, "A New Approach to the Design of Reinforcement Schemes for Learning Automata: Stochastic Estimator Learning Algorithm," *Neurocomputing*, vol. 7, no. 3, 1995, pp. 275–97.
[11] N. Kumar, J. H. Lee, and J. P. C. Rodrigues, "Intelligent Mobile Video Surveillance System as a Bayesian Coalition Game in Vehicular Sensor Networks: Learning Automata Approach," *IEEE Trans. Intelligent Transportation Systems*, vol. 16, no. 3, 2015, pp. 1149–62.
[12] N. Kumar *et al*., "Performance Analysis of Bayesian Coalition Game-Based Energy-Aware Virtual Machine Migration in Vehicular Mobile Cloud," *IEEE Network*, vol. 29, no. 2, Mar. 2015, pp. 62–69.
[13] S. Zeadally, S. Khan, and N. Chilamkurti, "Energy-Efficient Networking: Past, Present, and Future," *J. Supercomputing*, vol. 62, no. 3, 2012, pp. 1093–1118.
[14] A. Gelman *et al*., *Bayesian Data Analysis*, 2nd ed., CRC Press, 2003.

Biographies

Neeraj Kumar [M'16] is working as an associate professor in the Department of Computer Science and Engineering, Thapar University, Patiala (Pb.). He received his M.Tech. from Kurukshetra University, Kurukshetra (Haryana) followed by his Ph.D. from SMVD University, Katra (J&K) in CSE. He was a postdoctoral research fellow at Coventry University, United Kingdom. He has more than 100 research papers in leading journals and conferences of repute. His research is supported by UGC, DST, and TCS. He is an Associate Editor of *IJCS*, Wiley.

Sherali Zeadally received a Bachelor's degree from the University of Cambridge, United Kingdom, and his doctorate from the University of Buckingham, United Kingdom, both in computer science. He is an associate professor with the College of Communication and Information, University of Kentucky, Lexington. He is a Fellow of the British Computer Society and the Institution of Engineering Technology, United Kingdom.

Joel J. P. C. Rodrigues [S'01, M'06, SM'06] (joeljr@ieee.org) is a professor and senior researcher at the National Institute of Telecommunications (Inatel), Brazil, and a senior researcher at the Institute of Telecommunication, Portugal. He is the leader of the NetGNA Research Group (http://netgna.it.ubi.pt), Past Chair of the IEEE ComSoc Technical Committees on eHealth and Communications Software, and a Steering Committee member of the IEEE Life Sciences Technical Community. He is the Editor-in-Chief of three international journals, and a co-author of over 450 papers, three books, and three patents. He is the recipient of several Outstanding Leadership and Outstanding Service Awards from IEEE Communications Society and several best paper awards.