Mobility Prediction for Efficient Resources Management in Vehicular Cloud Computing

Ahmad M. Mustafa  
IoT Service Delivery, Readiness and Support  
Vodafone Group Services Ltd., UK  
Email: ahmad.mustafa1@vodafone.com

Omar M. Abubakr  
Information Technology Institute, Egypt  
Email: omar.abubakr.alexeng@gmail.com

Omar Ahmadien  
Department of Electronics and Computer Engineering  
Istanbul Şehr University, Turkey  
Email: omarahmadien@std.sehir.edu.tr

Ahmed Ahmedin  
University of California, Davis  
Email: ahmedin@ucdavis.edu

Bassem Mokhtar  
Department of Electrical Engineering  
Faculty of Engineering  
Alexandria University, Egypt  
Email: bmokhtar@alexu.edu.eg

Abstract—Vehicular Cloud Computing (VCC) has become a significant research area recently, due to its potential advantages and applications, especially in the field of Intelligent Transportation Systems (ITS). However, the high mobility of vehicular environment poses crucial challenges to resources allocation and management in VCC, which makes its implementation more complex than conventional clouds. Many works have been introduced to address various issues and aspects of VCC, including resources management and Virtual Machine Migration in vehicular clouds. However, using mobility prediction in VCC has not been studied previously. In this paper, we introduce a novel solution to reduce the effect of resources mobility on the performance of vehicular cloud, using an efficient resources management scheme based on vehicles mobility prediction. This approach enables the vehicular cloud to take pre-planned procedures, based on the output of an Artificial Neural Network (ANN) mobility prediction model. The aim is to reduce the negative impact of sudden changes in vehicles locations on vehicular cloud performance. A simulation scenario is introduced to compare between the performance of our resources management scheme and other resources management approaches introduced in the literature. The simulation environment is based on Nagel-Shreckenberg cellular automata (CA) discrete model for traffic simulation. Simulation results show that our proposed approach has leveraged the performance of vehicular cloud effectively without overusing available vehicular cloud resources.

Keywords— Vehicular Cloud Computing (VCC), Resources Management, Virtual Machine Migration, Mobility Prediction, Traffic Modeling and Simulation

I. INTRODUCTION

Advances in vehicular technology and Intelligent Transportation Systems (ITS) have provided new applications that demand complex computation and large storage. However, due to hardware and cost limitations, a single vehicle has limited computation and storage resources. Hence, it is difficult for an individual vehicle to efficiently support these applications. A possible solution is to share the computation and storage resources among physically nearby vehicles [1]. This led to the introduction of the cloud concept to Vehicular Ad-hoc Networks (VANETs) [2], which resulted in the evolution of VCC paradigm. VCC belongs to the wider title of Mobile Cloud Computing (MCC).

However, vehicular cloud implementation is more complex than clouds in traditional computer networks. Unlike conventional cloud systems, resources in VCC are moving vehicles which rapidly change their locations [3]. The high mobility in vehicular cloud environment is one of the crucial challenges in VCC.

Many works have been introduced to address various issues and aspects of VCC [4], [5]. Resources management and VM migration in vehicular clouds are among those aspects discussed in the literature recently.

For example, R. Yu et al. [6] proposed a game-theoretical approach for optimal vehicular cloud resources allocation. A resource reservation scheme is proposed as well to increase VM migration efficiency and reduce service dropping.

In their work, H. Arikan et al. [1] suggested clustering technique as a solution to overcome high mobility and frequent topology changes in vehicular cloud environment, where vehicles form dynamic clusters and the ones that are more suitable become cluster heads. Fuzzy-logic is used to determine the optimal cluster head in terms of stability. The cluster head optimizes resources management for its neighboring vehicles using reinforcement learning. However, the works mentioned previously do not deal directly with the problem of resources mobility using mobility prediction.

Moreover, P. Ghazizadeh et al. [7] proposed a fault-tolerant job assignment strategy, based on redundancy, which mitigates the effect of resource volatility of resource availability in vehicular clouds. They assumed that the vehicular cloud is implemented in a parking lot. However, parking is the least complex scenario to implement vehicular cloud due to zero or negligible mobility of vehicles [8]. This study does not suggest a solution for environments with higher mobility like highways and urban areas. In our study, we are suggesting a solution for implementing vehicular cloud in a harsh highway...
environment, where different vehicles collaborate for a very short span due to very high mobility.

T. Refaat et al. [3], [9] have introduced the interesting concept of mobility-aware VM migration in VCC. The mobility-aware VM migration scheme incorporates prediction of the vehicle’s future path in order to avoid unsuccessful migrations. However, this study points to the importance of mobility-awareness in VCC, without providing a methodology for mobility prediction in a real-life vehicular cloud environment. In our work, we extend this concept by suggesting a mobility prediction model to improve the performance of VCC in a real-life vehicular cloud environment.

In this paper, we introduce a novel solution to reduce the effect of VM migrations and job failures on the performance vehicular cloud, using an efficient resources management scheme based on vehicles mobility prediction. Our proposed solution aims to deal directly with the main challenge in VCC, which is high mobility. This approach enables the vehicular cloud to take pre-planned procedures based on the output of an ANN mobility prediction model, to reduce the negative impact of sudden changes in vehicles locations on vehicular cloud performance.

A simulation scenario is introduced to compare between the performance of our resources management scheme and other resources management approaches, and to show the effect of our solution on reducing the impacts of vehicles mobility to improve VCC performance. The simulation environment is based on Nagel-Shreckenberg CA vehicular traffic model. Our simulation results show that the proposed approach has leveraged the performance of vehicular cloud effectively without overusing available vehicular cloud resources.

The rest of this paper is organized as follows: Section 2 discusses our system model. The Virtual Machine Migration (VMM) schemes included in our study, including our proposed scheme, are discussed in Section 3. Section 4 introduces the CA vehicular traffic model used in our simulation environment. Section 5 introduces our ANN mobility prediction model and its performance. Our simulation setup and configuration is represented in Section 6. A performance evaluation based on our simulation results is mentioned in Section 7. Finally, Section 8 concludes the paper and highlights our future work.

II. SYSTEM MODEL

The cloud architecture adopted in this study is introduced in [6], [8], [10]. As shown in figure 1, the architecture consists of three layers: i. Vehicular Cloud: a local cloud established among a group of cooperative vehicles formed by vehicle-to-vehicle (V2V) communications. ii. Roadside Cloud: a local cloud established among a set of adjacent roadside units (RSUs), vehicles will access a roadside cloud by vehicle-to-infrastructure (V2I) communications. iii. Central Cloud: can be driven by either dedicated servers in vehicular networks data center or servers in the Internet. A central cloud is mainly used for complicated computation, massive data storage, and global decision.

In our vehicular cloud scenario, each group of vehicles located within the radio coverage of an RSU, create a cloudlet. Similar to conventional cloud system, virtualization enables cloud users sharing the same physical resources in an isolated manner. During service time, the Virtual Machines (VMs) assigned to particular users can be shifted between physical hosts for several purposes.

In our case, during a cloud service, as the vehicle moves along the roadside, it will switch between different RSUs. For the continuity of cloud service, the VM should be synchronously transferred between the respective roadside cloudlets. The migration process is referred to as VM migration [6]. This process creates overhead which can degrade the performance of the vehicular cloud [11].

It is also possible that during the cloud service, the vehicle moves to an area uncovered by RSUs, which means that the allocated resource is out of service. In such a case, unless we take special precautions, the entire work done is lost and we have the restart the entire process again, taking chances on another car, and so on until eventually the job is completed [7]. These job failures can reduce the efficiency of the vehicular cloud dramatically.

The occurrence of one of the two cases mentioned above, VM migrations and job failures, depends directly on the vehicular traffic factors affecting vehicles mobility (i.e. position of a vehicle from RSU in a given instance, speed of the vehicle, traffic density, etc.), which means that the study of vehicles mobility is a possible approach to address these issues.

III. VMM SCHEMES

In cloud systems, VM migration is a critical event, and it can be triggered for various goals. In vehicular clouds, mobility of resources is the main trigger of VM migration [9]. As mentioned before, as the vehicle moves along the roadside, it will switch between different RSUs. For the continuity of cloud service, the VM should be migrated to the destined roadside cloudlets. It is also possible that the vehicle heads
towards an area uncovered with RSUs, which will lead to further problems.

In order to show the effect of utilizing mobility prediction in vehicular VM migration, we compare between the following vehicular VM migration schemes in our simulation study:

1- Uniform VM Migration (UVMM): A random resource is initially chosen to start the job. VM migration is performed when required, where the destination RSU is the preceding RSU (i.e. the job is always continued on the same vehicle and no new resource is chosen for it). If the vehicle leaves the range of an RSU to an uncovered area before the job is done, the job fails and all progress is lost. A new resource is allocated randomly to start the job again. This scheme doesn’t include any intelligence, and it is introduced only as a benchmark [3].

2- Redundancy-based VM Migration (RVMM): Unlike the first approach, each job is assigned to two randomly chosen vehicles. Vehicle A is the master vehicle and vehicle B is the redundant one, and the job is started on both vehicles. When vehicle A leaves, the state of vehicle Bs VM is saved, a new vehicle C is randomly chosen as the redundant vehicle, the saved image is copied to C and the job is restarted on vehicles B and C, and so on. If the master vehicle leaves, and the redundant vehicle was unavailable (i.e. located in an uncovered area in the same instance), the job fails and starts all over again. This approach is originally introduced by [7] to ensure reliability for VCC in a car parking lot, and we used it for comparison with our proposed predicitive VM Migration scheme.

3- Predictive VM Migration (PVMM): In our proposed scheme, a mobility prediction model is used to ensure reliable resources management without overusing resources due to redundancy, and to decrease the overhead due to numerous VM migrations interrupting jobs execution. The model predicts the lifetime of all available resources (i.e. the time through which a resource is available before leaving the covered area or a VM migration is required). The available resource with largest predicted lifetime \( \left(T_p \right) \) is chosen. After a suitable ratio of the predicted lifetime, a pre-planned VM migration is done, the available resource with largest predicted lifetime is chosen to be the destination, and so on. This lifetime utilization ratio (U) is adjusted to make sure that the migration is done before the vehicle leaves the current RSU to another one or to an uncovered area, as the predicted lifetime may be larger than the true lifetime. This approach ensures the accomplishment of jobs with the least number of job failures and VM migrations; which decrease the time overhead and improves the performance of the vehicular cloud notably.

IV. VEHICULAR TRAFFIC MODEL

In order to simulate a vehicular cloud environment, a realistic model for traffic flow is a necessity. It is crucial to build a model for traffic flow that captures the characteristics of real traffic, yet sufficiently simple to be implemented. Our simulation environment is based on Nagel-Shreckenberg CA discrete model for traffic simulation [12]. Our previous works [13], [14] had shown that the Nagel-Shreckenberg model is efficient in capturing traffic flow properties despite its simplicity, compared to more sophisticated vehicular traffic simulation tools.

The Nagel-Shreckenberg model is defined on a one-dimensional array of L cells, each of 7.5 m length, and with closed boundary conditions (i.e. the modeled road is a single-lane ring). Each cell may either be occupied by one vehicle, or it may be empty. Each vehicle has an integer velocity (i.e. number of cells covered in a time step) with values between 0 and \( \text{v}_{\text{max}} \). For an arbitrary configuration, one update of the system consists of the following four consecutive steps, which are performed in parallel for all vehicles [12]:

1. Acceleration: if the velocity \( v \) of a vehicle is lower than \( \text{v}_{\text{max}} \) and if the distance to the next car ahead is larger than \( v + 1 \), the speed is advanced by one \( [v \rightarrow v + 1] \).
2. Slowing down (due to other cars): if a vehicle at site \( i \) sees the next vehicle at site \( i + j \) (with \( j \leq v \)), it reduces its speed to \( j - 1 \) \( [v \rightarrow j - 1] \).
3. Randomization: with probability \( p \) the velocity of each vehicle is decreased by one \( [v \rightarrow v - 1] \).
4. Car motion: each vehicle is advanced \( v \) sites.

These rules are repeated each time step of the simulation. By the end of the simulation, we have a two-dimensional matrix expressing traffic characteristics through definite road length and simulation time. In our configuration, maximum speed \( (\text{v}_{\text{max}}) \) is 4 cells/time-step, which is equivalent to 110 km/h. Deceleration probability (p) is 0.2, and traffic density (\( \rho \)) is 0.15 vehicle/cell.

V. MOBILITY PREDICTION MODEL

In our proposed PVMM scheme discussed in section 3, it is required to predict the lifetime of available resources (i.e. the time during which the vehicle is located in the same cloudlet). In literature, many models were introduced for time predictions in various applications, including in vehicles arrival prediction, which is similar to our application [15]. ANN is one of most popular used algorithms in vehicles arrival prediction [16]–[19].

Machine Learning (ML) has powerful models, called Hybrid Models. These models are integration between two time prediction algorithms. In [20], R. Jeong and L. Rilett combined Linear Regression (LR) with ANN. As a result, they found that ANN Models outperformed historical data-based-model and regression model in terms of estimation precision. In this work, we use a hybrid model of LR aided by ANN to predict resources lifetime as mentioned earlier. The back-propagation algorithm is applied to measure the weights, map features with observations through minimizing the error and predict the observations.

Our model has three main layers: i- Input layer, which consists of 6 inputs. ii- Hidden layer, which contains two layers each with 200 neurons. iii- Output layer, which is the predicted resource lifetime. We have changed the output of the NN to be linear instead of sigmoid function, so as to change the NN from limited 0 and 1 classification into continuous linear regression. With tuning the hyper parameters of the NN such
as number of epochs, learning rate, number of hidden layers and number of neurons in each layer, we have reached an acceptable performance of data fitting.

Our training dataset was collected from simulation results, based on the vehicular traffic CA model discussed in section 4. 54000 samples were divided into 80% for training and 20% for testing. We used the 20% to estimate the performance of the model and how well it fits the data. Dataset is fed to the LR aided by ANN model. Dataset features are explained briefly as the following:

1- Resource lifetime: the time taken by a chosen vehicle from the instance of observation before it leaves its current cloudlet area. (The targeted output of the predictive model)
2- Vehicles position: the position of the chosen vehicle from the start of the current cloudlet area at the instance of observation. (A vehicle at the start of the cloudlet area is more likely to have a larger lifetime before it leaves the cloudlet)
3- Vehicles speed: the speed of the chosen vehicle at the instance of observation.
4- Number of vehicles ahead in the same cloudlet area at the instance of observation.
5- Average speed of vehicles ahead in the same cloudlet area at the instance of observation (A vehicle at a congested area is more likely to decelerate and have a larger lifetime).
6- Number of vehicles ahead in the next cloudlet area at the instance of observation.
7- Average speed of vehicles ahead in the next cloudlet area at the instance of observation.

The performance of our model is evaluated by four methods. The first method is R-squared (coefficient of determination), which is interpreted as the proportion of the variance in the dependent variable that is predictable from the independent variable [21]. Hence, a closer value of R-squared to one means better linear regression. R-squared is calculated by the following equation:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$  \hspace{1cm} (1)

Where:

$SS_{res}$: The regression sum of squares, also called the explained sum of squares.
$SS_{tot}$: The total sum of squares (proportional to the variance of the data).

The second method is Pearson’s correlation coefficient ($\rho$), which is a famous performance parameter for linear regression. It determines how well the data is fitted by the model. The value of ($\rho$) is between -1 and 1. If it is, 1 then the model perfectly fits the data, but as an over fit. If ($\rho$) is zero, then the model does not fit the data. Finally, if it is -1, then the model over fits the data but with negative slope.

The third method is MAPE (Mean Absolute Percentage Error) which represents the average percentage difference between the observed value and the predicted value. Smaller value of MAPE means that the model predicts more accurately. It is described by the following equation:

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - y_0|}{y_0} \times 100\%$$  \hspace{1cm} (2)

Where:

$y_i$: Predicted value.
$y_0$: Observed value.
n: The number of data considered.

The fourth method is the famous RMSE (Root Mean Square Error):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - y_o)^2}$$  \hspace{1cm} (3)

Table 1 contains the results of our performance analysis:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R^2$</td>
<td>0.856</td>
</tr>
<tr>
<td>MAPE</td>
<td>17.45</td>
</tr>
<tr>
<td>RMSE</td>
<td>4.1</td>
</tr>
<tr>
<td>$\rho$</td>
<td>0.935</td>
</tr>
</tbody>
</table>

The following figures are various representations for the performance of the model. In figure 2, the observations are plotted versus the predictions of the model. These observations and predictions are performed using the 20% test data. In figure 3, we plotted the histogram of prediction errors.

VI. SIMULATION STUDY

In order to compare between the performances of the three VM migration schemes discussed in section 3, a simulation scenario was implemented combining the vehicular traffic model and the mobility prediction model mentioned earlier. The aim of this simulation is to show the effect of mobility prediction on vehicular cloud performance. Thus, other conditions
affecting vehicular cloud performance were not taken into account for sake of simplicity (e.g. wireless communication overhead, VM workloads, heterogeneity of resources, etc.). The effect of these different conditions alongside resources mobility will be addressed in our future work.

Our simulation environment resembles a single-lane ring road, covered with five RSUs (i.e. five vehicular cloudlets), each cloudlet of 150 m length. One specific pre-known RSU is assumed to be out of service, which models the effect of vehicles moving to an area uncovered with RSUs, or leaving the whole road. The total simulation duration is 10 hrs. The simulation scenario is based on the following assumptions:

1- All vehicles in the simulation have eligible capabilities and resources to be allocated for a VM, and all resources are identical.
2- VM migration occurs when the vehicle leaves a section covered by an RSU, to a new area covered with another RSU.
3- The VM migration overhead ($T_m = 3$ s) is assumed to be constant for all migrations.
4- All jobs are identical, with a required time of ($T_j = 30$ s) to be successfully achieved.
5- Only one job is running at any instance, and when it is successfully accomplished another job is initiated.
6- Each job is assigned to one vehicle at a time, and the VM migration is only 1:1 migration. Load balancing on more than one vehicle is not allowed.
7- If a vehicle leaves the range of a RSU to an uncovered area before a migration occurs and in the absence of a redundant resource, the job is fails and all progress is lost. A new resource is allocated to start the job again.

For our proposed PVMM scheme discussed earlier in section 3, the lifetime utilization ratio ($U$) is found to be optimized at 0.6, after sweeping on its values through multiple simulation runs. This value was found to ensure the most use of resource lifetime while eliminating the effect of prediction error. This value can be improved by improving the performance of the mobility prediction model (i.e. decreasing prediction error), which will leverage the overall performance eventually.

![Fig. 3. Histogram of prediction error](image)

<table>
<thead>
<tr>
<th>Table II</th>
<th>Simulation Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
<td>Value</td>
</tr>
<tr>
<td>$T$ (total simulation duration)</td>
<td>10 h</td>
</tr>
<tr>
<td>$t_s$ (time step)</td>
<td>1 s</td>
</tr>
<tr>
<td>$L$ (cloudlet area length)</td>
<td>150 m</td>
</tr>
<tr>
<td>$V_{max}$ (maximum vehicle speed)</td>
<td>110 km/h</td>
</tr>
<tr>
<td>$L$ (cloudlet area length)</td>
<td>150 m</td>
</tr>
<tr>
<td>$p$ (vehicle deceleration probability)</td>
<td>0.2</td>
</tr>
<tr>
<td>$\rho$ (vehicular traffic density)</td>
<td>0.15 vehicle/cell</td>
</tr>
<tr>
<td>$T_j$ (job duration)</td>
<td>30 s</td>
</tr>
<tr>
<td>$T_m$ (VM migration overhead)</td>
<td>3 s</td>
</tr>
<tr>
<td>$U$ (PVMM utilization ratio)</td>
<td>0.6</td>
</tr>
</tbody>
</table>

The simulation, including traffic model and mobility prediction model, is implemented using MATLAB. Simulation parameters are summarized in table 2. The next section discusses simulation results depending on the described parameters.

VII. PERFORMANCE EVALUATION

As mentioned before, when a resource is out of service during job execution, the entire work done is lost and we have the restart the entire process again unless we take special precautions. The failure overhead affects the performance of vehicular cloud dramatically. In UVMM, job failures happen frequently due to vehicles moving towards the area uncovered with RSUs. On the other hand, failure overhead is reduced in RVMM because of resources redundancy. However, this approach is very resource-consuming as it uses double the resources needed for jobs accomplishment.

In PVMM, the mobility prediction model is utilized to choose resources which are less likely to depart the covered road. As mentioned in our simulation assumptions, the uncovered area is predetermined. So, it is possible theoretically to completely avoid job failures, by choosing a suitable value for the lifetime utilization ratio ($U$) as discussed earlier. However, in real-life implementation, job failures cannot be completely avoided due to randomness of resources availability. A comparison of failure overhead in proposed schemes is illustrated in figure 4.

Regarding VM migrations, resources are chosen randomly in UVMM scheme. Hence, the chosen resource can have a very short lifetime before a VM migration is required; leading to a large number of VM migrations. In RVMM, VM migration overhead is not affected because resources are chosen randomly as well, without considering resources lifetime. In our proposed PVMM scheme, there is a significant decrease in average VM migrations per job, as the mobility prediction model is utilized to choose resources with highest lifetime, which decreases the number of required VM migrations, as shown in figure 5.

Simulation results show that PVMM has the least overhead, followed by RVMM and UVMM respectively, as show in figure 6. Total overhead percentage is given by the following formula:
$$OH = \frac{T - N \times T_j}{T} \times 100\%$$

Where:
- $OH$: Total overhead percentage.
- $T$: Total simulation duration.
- $N$: Number of jobs accomplished during simulation.
- $T_j$: Job duration.

PVMM also has the best performance in terms of number of jobs accomplished during simulation accordingly, as shown in figure 7.

To summarize, simulation results show that our proposed PVMM scheme outperforms the two other VMM schemes, and it has been successful in leveraging the performance of vehicular cloud effectively, without overusing available vehicular cloud resources. Simulation results are summarized in table 3.

VIII. CONCLUSION

In this paper, we introduced a methodology for reducing the impact of vehicles mobility on vehicular clouds performance, using an efficient resources management scheme based on vehicles mobility prediction. An ANN mobility prediction model was used to make the vehicular cloud capable of taking pre-planned procedures, to reduce the negative impact of sudden changes in vehicles locations on vehicular cloud performance.

A simulation study was introduced to compare between the performance of our proposed resources management scheme and other resources management approaches. The simulation environment is based on Nagel-Shreckenberg CA discrete model for traffic simulation.

The concept of utilizing mobility prediction model in VCC resources management has shown its effectiveness in leveraging the performance of vehicular cloud without overusing the available resources. Simulation results showed that our

---

TABLE III
SIMULATION RESULTS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>UVM</th>
<th>RVMM</th>
<th>PVMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failure overhead (s)</td>
<td>5508</td>
<td>1490</td>
<td>173</td>
</tr>
<tr>
<td>Average VM migrations per job</td>
<td>1.20</td>
<td>1.21</td>
<td>1.00</td>
</tr>
<tr>
<td>Total overhead percentage (%)</td>
<td>24.5</td>
<td>14.5</td>
<td>9.8</td>
</tr>
<tr>
<td>Number of jobs accomplished</td>
<td>906</td>
<td>1026</td>
<td>1085</td>
</tr>
</tbody>
</table>
proposed resources management approach outperformed other approaches included in our simulation study.

Our future work includes extending our study to include more aspects and conditions affecting vehicular clouds performance (e.g. wireless communication overhead, VM workloads, heterogeneity of resources, etc.). Furthermore, we target presenting more realistic simulation scenario based on network simulation tools, instead of abstractions and assumptions used in this work; such as job duration and migration overhead.

REFERENCES


