Architecture and Channel Aware Task Offloading in Opportunistic Vehicular Edge Networks

Mohammad Pasha
Information Technology Dept.
MJCET
Hyderabad, India
muhammed.pasha@gmail.com

Khaleel Ur Rahman Khan
Computer Science Engg. Dept.
ACE
Hyderabad, India
khaleelrkhan@gmail.com

Abstract **- Vehicular Edge Computing architectures are projected recently that support expedient task offloading. However, in vehicular environments the character of the wireless channel and convenience of resources is unpredictable. Effective utilization of vehicular resources remains a challenge. To beat this challenge, the current work proposes Architecture and Channel Aware Task Offloading in Vehicular Clouds (ACATOVC) protocol for efficient task offloading in vehicular networks. Results obtained by in depth simulations are conferred to gauge and compare its performance with existing protocols.**

Keywords - vehicular cloud computing, mobile edge computing, vehicular ad-hoc networks, computation offloading, medium access control.

I. INTRODUCTION

The idea of execution tasks remotely is preponderantly tailored from mobile cloud computing (MCC) owing to the inherent limitations of mobile devices in processing-heavy tasks associated to an extent avoid battery drain [1]. however vehicular networks don't face this challenge. These rather will facilitate overcome the requirement of counting on the remote servers for process and storage by utilizing services provided by potential nodes and road-side instrumentation. Further, these vehicles will assist in distributed and coordinated execution of tasks that are associated with effectively managing numerous transport activities on roads. Planned evacuation throughout road hold up and accidents are often self-addressed exploitation vehicular networks [2].

Many alternatives connected domains conjointly consider distributed process and access of information [1]. Hence, considering the financial investment in information access, augmented latency and bandwidth demand, and conjointly the high burden of energy usage with 4G/LTE property, the native resource-rich cloudlet would be a decent various to the traditional remote servers if wireless access is correctly managed. As a new advantage vehicle may also give context primarily based info by collaborating in playing distributed computations victimization sensory information [3]. Moreover, future applications of transport networks embody autonomous transport systems, platooning, planned evacuations throughout accidents or different disasters, virtual stoplight implementation at intersections [4].

Due to the assorted capabilities and options of those resource-rich mobile nodes, they're being termed as transport Cloud like a Cloud of resource-rich servers. one in all the vital challenges in transport cloud computing is that the choice of surrogate vehicles that may fulfill shopper requests for information storage, computation offloading or providing processed sensory information.

A mobile computing platform referred to as vehicular Cloud (VC) is pictured attributable to the rise in process, storage and communication capabilities of vehicular nodes. it's recently being remarked as vehicular Edge Computing [5]. to enhance the on-road safety and different services varied infrastructural services are needed. The under-utilized resources in vehicles may be exploited to satisfy this want. The forefront challenges to beat in these situations are economical resource utilization and task offloading.

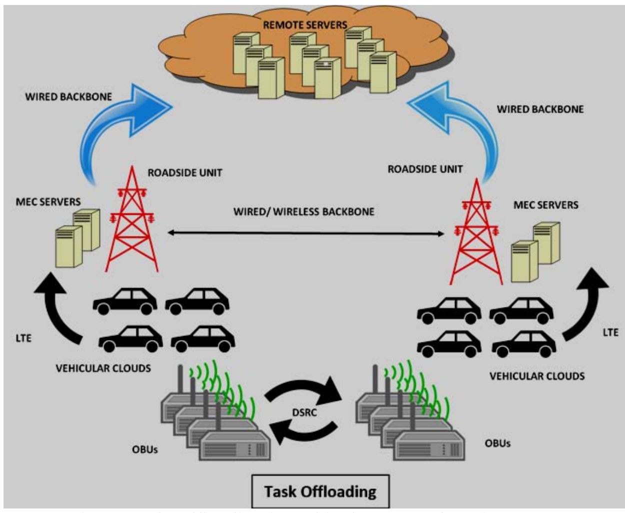

Fig. 1. Task Offloading in Vehicular Networks using OBU.

II. RELATED WORK

The following section provides an outline of the various aspects of vehicular cloud computing that are dealt since its beginning. broadly speaking classifying vehicular cloud computing provides services within the kind of stationary resource centers or dynamic resource centers [6]. additionally, the roadside units will give edge computing facilities to those resource centers. From the recent studies, we tend to determine the potential solutions these vehicular clouds offer for the on-road user. we tend to additionally discuss the challenges highlighted and their solutions projected by completely different works. Authors in [7] initial bestowed opportunities and challenges in exploiting computing resources in vehicular ad-hoc networks through a stratified design for cloud-based vehicular networks that gives sharing of process resources, storage resources among vehicles.

Authors in [8] projected a metric referred to as opportunist coverage quantitative relation and derived it as an aggregative operate. It additionally bestowed a vehicle selection algorithmic rule, a framework for recruiting vehicles implementing an urban vehicular sensing platform. Authors in [9] conferred a survey of VANET enabled services and claim there's an excellent potential to enable varied functionalities by offloading computation intensive blocks from the vehicles to different entities like RSUs. It additionally provides a classification of computations offloading in VANETs as V2V based mostly} and V2I based. Authors in [10] investigated the implementation of computation offloading in vehicular environments, proposing a framework to support it. The offloading algorithms were modelled exploitation MVC model however neither the vehicular network nor the communication network model was used. Authors in [11], [12] targeted on programming and allocation of procedure tasks of an application for distributed process revealing vital problems with the real-time execution of cooperative applications in wireless networks. Authors in [13] explored the computation offloading mechanism to fulfil the temporal order constraint of time period tasks with programming the execution order for independent tasks. Authors in [14] enforced and evaluated an example of CWC that employs an algorithmic rule to reduce the makespan of a group of computing tasks. Authors in [15] proposed a computation resource allocation scheme for a vehicular Cloud system and bestowed a best higher cognitive process scheme is to maximize the long-run expected total reward of the VCC system. Authors in [16] explored and bestowed "opportunistic impromptu cloudlet service" (OCS) treated as an intermediate mode to yield higher flexibility and value move alter an additional energy-efficient and intelligent strategy for computation offloading through the utilization of a commercial ad-hoc cloudlet.

Authors in [17] presented a model to produce the analytical estimates for the provision of procedure resources within the vehicular cloud employing a free-flow and a queuing-up traffic model to see the practicableness and attainable situations for assignment or migration of procedure tasks to vehicles. Authors in [18] designed a system to alter seek for mobile cloud server exploitation RSUs to act as cloud directories. Authors in [19] bestowed an algorithmic rule to model the protection resource allocation for VCC. Authors in [20] thought-about a fog computing paradigm exploitation bus networks to increase margin cloudlets. It additionally bestowed an allocation strategy of the margin cloudlet to dump the computation tasks to bus fog servers.

Authors in [24] proposed a reliable task-scheduling model in vehicular Cloud Computing setting to reduce execution time and satisfy job deadlines by formulating a MILP optimization problem. Authors in [21] proposed information as a Service paradigm for approved users outside of the VC exploitation gathered information from mounted sensors on the vehicles and constructs a stable variable size vehicular clusters referred to as vehicular clouds exploitation on vehicles quality and their distributions on the road. Authors in [22] addressed Intermittent property and non-seamless wireless access of remote clouds and for disaster response management and military operations, employing a fast and versatile impromptu offloading and recommended an approach is to offload computation to near mobile devices, and consequently kind an atmosphere referred to as Mobile Device Cloud (MDC).

Authors in [10] investigated the implementation of computation offloading in vehicular environments, proposing a framework to support it. The offloading algorithms were shapely exploitation MVC model however neither the vehicular quality network nor the communication network atmosphere was thought-about. Authors in [24] planned a reliable task-scheduling model in vehicular Cloud Computing atmosphere to reduce execution time and satisfy job deadlines by formulating a MILP optimization problem. however, the planned model relies on map-reduce that is supposed for data-intensive applications. this can be an inefficient model for the case of vehicular networks as a result of massive data transmissions sessions don't seem to be possible in realistic situations due to random access mechanism utilized by Wireless local area network standards like 802.11p. Instead, the task offloading edges solely computation-intensive applications. thence an acceptable model has to be supported virtual machine primarily based design. during this case solely a quick set of variables are necessary to invoke operations on the surrogate nodes.

III. Problem Definition and Formulation

A. Network Model

An opportunistic vehicular ad-hoc edge network consists of variety of vehicular nodes, modeled by an undirected communication graph G (V, E). to ascertain an immediate communication between any 2 nodes, the gap between them must be within their radio transmission range [31]. The planned system consists of client and surrogate nodes. client nodes, that generate task offloading requests and Surrogate nodes, give the computation service. The task process capability of the surrogate node, μs is denoted in units of mips per second|unit of measurement|unit} (million instructions per second).

B. Resource Modeling

The surrogates give task computations for client vehicles. Moreover, the requested resource is created accessible to the client on request supported a two-phase reservation method. once the surrogate receives an invitation it holds the resource for alittle amount of time (50 ms). If the

client doesn't send the task inside the stipulated amount the resource is marked idle once more.

C. Task Model

A job consists of tasks. A client is liable for maintaining a queue for the generated tasks. The tasks are inserted as they're submitted in FCFS order. every task within the task queue at the client node is denoted by where tid , task Id; T, time constraint of task; and C, quantity of computations to be processed; At every surrogate node, the queuing delay refers to the waiting time once the task is placed at the top of the queue till the instant that the task is processed.

D. Problem Formulation

Using the network and task models we present the calculations for the task completion time. the entire task completion time for execution a task ti consists of communication time, the computation time and therefore the queuing delay of the task within the surrogate queue.

$$TaskCompletionTime_i = T_{tx} + T_{rx} + T_{exec} + T_{queue} \quad (1)$$

Where

$T_{tx} = T_{rx} = D_i / B_{c,s}$ Where Di is the task data and Bc,s is the available shared bandwidth between the client and the surrogate;

$T_{exec} = C_i / \mu_s$ Where Ci is the computations requested by the client and μs is the computation capacity of the surrogate;

T_{queue} is the queuing delay of the task.

In vehicular networks the maximum time that can be allocated for completion of task is dependent on the link duration time between the client and surrogate. The link duration is derived from the relative velocity of the vehicles.

Thus we have

$$T_{link(i,j)} = (v_i - v_j) / d_{(i,j)} \quad (2)$$

Where

- $T_{link(i,j)}$ is the link duration time between vehicle i and vehicle j;
- vi and vj are instantaneous velocities of vehicle i and vehicle j;
- d(i,j) is the instantaneous distance between them.

Thus to successfully offload tasks to surrogates the following inequality needs to be realized.

$$2*(D_i / B_{c,s}) + (C_i / \mu_s) + T_{queue} \leq (v_i - v_j) / d_{(i,j)} \quad (3)$$

With the following constraints

Task completion time < task deadline (4)

μs >> Ci (5)

Di < Wave Short Message Packet payload (6)

d(i,j) = min (D(i, S)) where D(i, S) contains distance values to all surrogates. (7)

vi - vj = min(vi – vS) where vi – vS is relative velocity of client to all surrogates (8)

- Constraint (4) is critical for a task to be with success offloaded to satisfy its deadline. Otherwise, it's thought-about unsuccessful.
- Constraint (5) ensures that a task is often executes at a surrogate with high computation capability.
- Constraint (6) states that the task connected information being sent is inside the bounds of the Wave Short Message Packet payload.
- Constraint (7) and Constraint (8) ensures that the chosen surrogates provide increased link time period and reliable task transmission.

From the above equation we infer the subsequent

- The task completion time is dominated by the task execution time and therefore the task queuing delay. It may be reduced by choosing surrogates with high computation capability and with the smallest amount queuing delay.
- Though the task transmission time is a smaller portion of the task completion time it influences the responsibleness of the scheduled task. Hence, to pick extremely reliable surrogates the nodes with the longest link time period ought to be selected . just in case of wireless impromptu network the nodes with least relative speed and least distant to the consumer node are often the foremost reliable surrogates.
- The selection of surrogate nodes with stable links to the consumer is vital in extremely dynamic networks like VANETs. we used the quality issue parameter [32] obtained from the relative speed of the surrogate nodes. It provides increased link time period for task offloading. it's given by

$$Mf = \frac{(1+r)^{-w} + (Mf_{pre} * (N-1))}{N} \quad (9)$$

Where mf is current quality factor for a surrogate, Mfpre is previous quality factor, r is the relative speed, N is current node density, and w could be a weight factor for emphasizing tiny distinction in speeds, initialized to zero.3. the value of mf ranges from zero to one.With its most value it specifies a stable link time period between the client and surrogate. each client records and updates mf for all its surrogate vehicles.

Using the expression to estimate the task completion time a completely unique task offloading algorithm named architecture and Channel Aware Task Offloading in vehicular Clouds (ACATOVC) is projected as represented thoroughly within the next section. as a result of the solution to MILP is takes non-polynomial time to execute we propose a heuristic for finding it.

IV. ARCHITECTURE AND CHANNEL AWARE TASK OFFLOADING IN VEHICULAR CLOUDS (ACATOVC)

The planned algorithmic rule consists of 2 parts specifically resource discovery part and surrogate choice phase. The planned resource discovery protocol is broadcast protocol to exchange each point similarly as task connected info from the native neighborhood of a node. The gathered details are accustomed build a local knowledge-base at every node.

To accurately measure the task coordinated universal time between a client and a surrogate pair we introduce a parameter known as the one-hop latency for a node pair. it's measured by Medium Access control (MAC) latency of the received message. to search out the mac latency of the received message we log the time at that the application layer inserts a message for transmission within the mac queue. once this message is received by the receiver we calculate the one-hop latency as

One-hop latency = Time-stampapplication (sender) – current-time (receiver) (10)

Where Time-stampapplication (sender) is that the time at that the packet was inserted by the application layer in to the mac queue.

Next we provide the main points regarding the ACATOVC protocol exploitation heuristics. When a client node generates tasks for offloading it broadcasts a task offload request. Surrogate nodes among its range reply with the details of the computation capability and task queue length of backlogged tasks. The client receives the main points of the resources offered at all approachable surrogates. It calculates the calculable task completion time using equation (1). Surrogates that provide viable task process (i.e. task completion time is lesser than link lifetime) are marked as potential surrogates. The probable surrogates are sorted consistent with their distance to client. The surrogate with the smallest amount distance is chosen as target surrogate.

V. PERFORMANCE EVALUATION

The performance of ACATOVC algorithm compared with that of QARTS [24]. Two variations of the proposed protocol are also implemented. The first only considers the hardware profiles of the surrogates while offloading tasks. The second considers the task queue at the surrogates along with the processing capability. In these both cases the worst case transmission time of a task in 802.11p networks is used as communication time instead of one-hop latency.

E. Calculation of Maximum Task Transmission Time in WAVE based Vehicular Networks

In case of WAVE based vehicular networks maximum transmission latency is derived as follows. The transmission time for a WAVE packet is given by

TABLE I. A HEAURISTIC METHOD TO OFFLOAD TASKS USING ACATOVC

Input: (1) Knowledge-base at node u built from Vehicular Resource Discovery Protocol, (2) Task queue J
Output: Offloading result
Function: Offloading the tasks to surrogate node in the Opportunistic Vehicular Edge Computing
Notations and meanings: S : list of all surrogates, Sc: list of surrogates satisfying task completion time criteria, Sl: list of surrogates satisfying link lifetime criteria, Slc: list of surrogates satisfying link lifetime and task completion time criteria, MAX_COMM_TIME: maximum communication time allocated for task Offloading. It includes both task transmission time and task result receiving time.
Procedure body: *while ( ) do { // check task queue J* *extract task ti from J;* *initialize surrogate sets S, Sc, Sl and Slc to null,* *MAX_COMM_TIME as 2*one-hop latency;* *S is initialized as all the surrogate nodes in the Knowledge-base;* *for each surrogate s from S{* *calculate the link lifetime;* *add surrogate s to list Sl ;}* *for each surrogate s from S{* *calculate the estimated task completion time using task execution time, task queuing delay and one-hop task transmission time;* *add surrogate s to list Sc ;}* *for each surrogate s from Sc and Sl {* *if (surrogate s Є Sc & s Є Sl)* *add surrogate s to selected surrogates list Slc ;}* *for each surrogate s from Slc {* *Sort surrogates according to distance}* *Select surrogate with least distance for offloading task}*

$$T_{tx} = T_{preamble} + T_{sym} * \left\lceil \frac{16+l+B_W}{N_{DBPS}} \right\rceil + T_{signal} \qquad (10)$$

Where l is the payload in bits, Bw is the Medium Access Control layer throughput in Mbit/s and NDBPS is the number of data bits per OFDM symbol [33].

In worst case, with retransmission limit to 7 attempts and CWmin =15 and CWmax=1023, we obtain the maximum transmission time as:

$$T_{tx\_max} = 8 \text{ x } (T_{tx} + T_{ack}) + T_{slot} \text{ x } (15+31+63+127+255+511+1023) \qquad (11)$$

Where Ttx is transmission time, Tack is time to receive acknowledgment and Tslot is the slot time. Substituting the values for the variables according to 802.11p we obtain the value of maximum transmission time around 50 ms.

The algorithms are implemented using vehicles in network simulation framework [34] which couples OMNET++ [35] and SUMo [36] for realistic mobility modeling. To evaluate the performance we consider that the client nodes generate 2 to 8 jobs per minute. These jobs are partitioned into 10 to 20 tasks each [24]. Thus we vary the number of tasks from 20 to 160 tasks for different experiments. Each task has a computation requirement of 1000 MIPS to 2000MIPS.

F. Simulation Setup

We consider a one-dimensional vehicular network formed on a 2-Lane highway which follows the Level of Service concept of transport management as shown in Table II is considered. To evaluate the effect of the vehicular movement on the offloading decision we make use of the Gaussian exponential mixture of mobility model proposed in [37].

TALE II. LEVEL OF SERVICE ON HIGHWAYS

LOS	Quality	Speed (kmph)	V/C	Level of Comfort
A	Free-flow	80	0.6	High
B	Reasonable	70	0.7	Reasonable
C	Near	60	0.8	Low
D	Medium	50	0.85	Absent

We evaluate the proposed scheme to access the effect of (a) varying the network size and vehicular mobility, (b) varying the number of jobs offloaded. We evaluate the Average Job Execution Time and Successful Job Completion Rate for offloaded tasks.

TABLE III. SIMULATION PARAMETERS

Parameter	Value
Vehicular Network Parameters	
Warm-up distance	1000meters
mac1609_4.txPower	10mW
Tx_Range	100 mts approx.
mac1609_4.bitrate	6Mbps
Vehicular Density	20, 25, 30, 35, 40, 45, 50 veh/km
Car_Following_Model	Krauss
Vehicle_InterArrival	Gaussian_Exponential_Mixture
Offloading schemes	Architecture Aware Task Offloading Utility Aware Task Offloading Architecture and Channel Aware
Task Parameters	
Jobs/ min	2 to 8
No. of tasks per job	10 to 20
Task Computation amount (MIPS)	Uniform (1000, 2000)

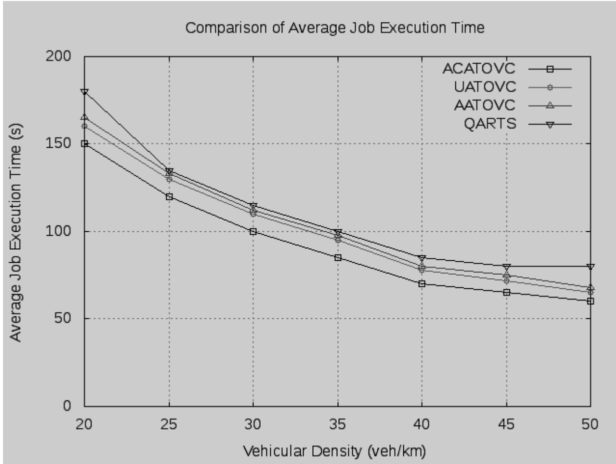

Fig. 2.Comparison of Average Job Execution Time for varying Vehicular Density.

Fig. 2 shows the performance of the proposed algorithms for varying vehicular densities. The job arrival rate is fixed at 6 jobs/ min. With 20 veh/km the network is sparse resulting in lesser number of surrogate nodes are available. This requires more amount of time for tasks to execute. As the vehicular density increases the amount of time required for task execution decreases. But when the vehicular density increases beyond 50 veh/km, the wireless channel becomes congested. Hence offloading tasks becomes inefficient due to transmission errors. The solution for this situation can be cooperative access to wireless channel as is available Time Division Multiplexed Systems. Presence of a Roadside Unit can also be used to provide the cooperation among competing nodes. The AATOVC and UATOVC algorithms consider the worst case transmission time when offloading tasks. This leads to lesser number of tasks being executed when compared to ACATOVC.

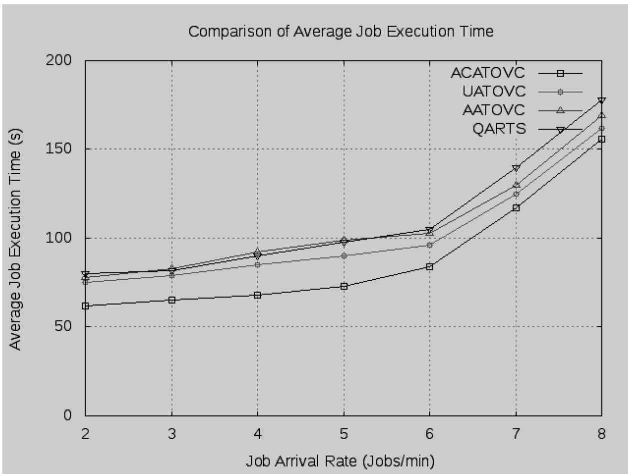

Fig. 3. Comparison of Average Job Execution Time for varying Job Arrival Rate.

Fig. 3 shows the performance of the proposed algorithms for varying job arrival rate. The vehicular density is fixed at 40 veh/ km. As the number of tasks offloaded increases the Average Job Execution Time also increases linearly. The AATOVC and UATOVC algorithms consider the worst case transmission time when offloading tasks. This leads to lesser number of tasks being executed when compared to ACATOVC.

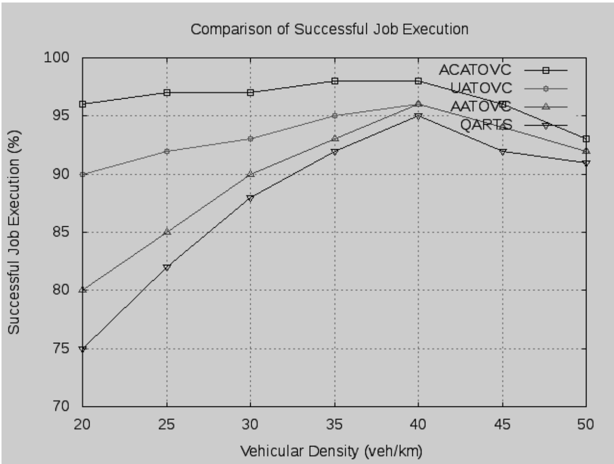

Fig. 4. Comparison of Successful Job Execution (%) for varying Vehicular Density.

Fig. 4 shows the performance of the proposed algorithms for varying vehicular density. The job arrival rate is fixed at 6 jobs/ min. As the number of tasks offloaded increases the Successful Job Execution (%) also increases linearly until the vehicular density reaches 40 veh/ km. Due to further increase in vehicular density the successful job execution percentage falls up to 93% for ACATOVC algorithm.

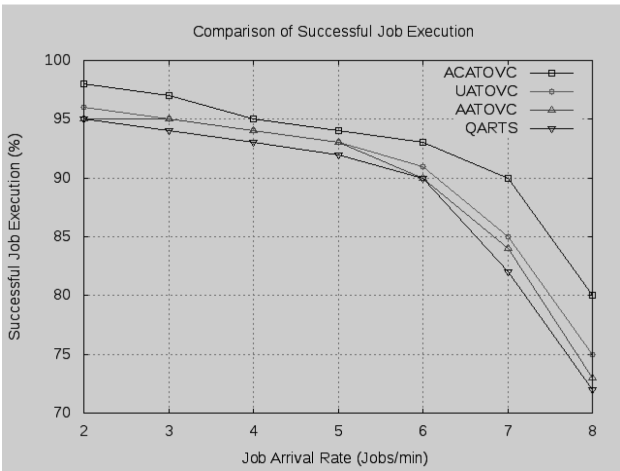

Fig.5. Comparison of Successful Job Execution (%) for varying Job Arrival Rate.

Fig. 5 shows the performance of the proposed algorithms for varying job arrival rate. The vehicular density is fixed at 40 veh/ km. As the number of job requests increases the Successful Job Execution (%) decreases. This is owing to the fact that with more number of job requests being flooded into the network it competes with the task response being sent by the surrogates. Hence for job arrival rate of 8 jobs/min successful job completion percentage falls to 80%.

VI. Conclusion

An efficient task offloading method was conferred for vehicular networks to support Edge Computing. The projected method considers task offloading from multiple client nodes to near surrogate nodes, satisfying the various tasks constraints whereas considering the wireless channel dynamics. One important purpose to notice that it's simply not enough that a specific surrogate is chosen strictly primarily based fulfilling the task's deadline constraint. The link life between the client and also the surrogate ought to be of enough length for the client to send the task work to the surrogate. The result of the delay that happens because of competition at the mac layer ought to be taken into consideration once offloading a task. The projected algorithm outperforms existing protocols in terms of Average Task Execution time and proportion of successful Completed tasks.

REFERENCES

[1] Fernando, N., Loke, S.W. and Rahayu, W., 2013. Mobile cloud computing: A survey. Future generation computer systems, 29(1), pp.84-106.

[2] Uzcátegui, R.A., 2009. Universidad Nacional Experimental Politécnica "Antonio José de Sucre" Guillermo Acosta-Marum. Georgia Institute of Technology "WAVE: A Tutorial" TOPICSIN Automotive Networking.

[3] Abdelhamid, S., Hassanein, H.S. and Takahara, G., 2014. Vehicle as a mobile sensor. Procedia Computer Science, 34, pp.286-295.

[4] Chen, L. and Englund, C., 2016. Cooperative intersection management: a survey. IEEE Transactions on Intelligent Transportation Systems, 17(2), pp.570-586.

[5] Hu, Y.C., Patel, M., Sabella, D., Sprecher, N. and Young, V., 2015. Mobile edge computing—A key technology towards 5G. ETSI White Paper, 11(11), pp.1-16.

[6] Whaiduzzaman, M., Sookhak, M., Gani, A. and Buyya, R., 2014. A survey on vehicular cloud computing. Journal of Network and Computer Applications, 40, pp.325-344.

[7] Zhao, D., Ma, H., Liu, L. and Li, X.Y., 2015. Opportunistic coverage for urban vehicular sensing. Computer Communications, 60, pp.71-85.

[8] Chaâri, R., Ellouze, F., Koubâa, A., Qureshi, B., Pereira, N., Youssef, H. and Tovar, E., 2016. Cyber-physical systems clouds: A survey. Computer Networks, 108, pp.260-278.

[9] Liu, K., Ng, J.K., Lee, V.C., Son, S.H. and Stojmenovic, I., 2016. Cooperative data scheduling in hybrid vehicular ad hoc networks: Vanet as a software defined network. IEEE/ACM transactions on networking, 24(3), pp.1759-1773

[10] Li, B., Pei, Y., Wu, H., Liu, Z. and Liu, H., 2014, August. Computation offloading management for vehicular ad hoc cloud. In

International Conference on Algorithms and Architectures for Parallel Processing (pp. 728-739). Springer, Cham.

[11] Datla, D., Chen, X., Tsou, T., Raghunandan, S., Hasan, S.S., Reed, J.H., Dietrich, C.B., Bose, T., Fette, B. and Kim, J.H., 2012. Wireless distributed computing: a survey of research challenges. IEEE Communications Magazine, 50(1).

[12] Alfaqawi, M.I., Habaebi, M.H., Siddiqi, M.U., Islam, M.R., Khan, S. and Datla, D., 2016, January. Adaptive load balancing algorithm for wireless distributed computing networks. In Intelligent Systems Engineering (ICISE), 2016 International Conference on (pp. 256-261). IEEE.

[13] Toma, A. and Chen, J.J., 2013, March. Computation offloading for real-time systems. In Proceedings of the 28th Annual ACM Symposium on Applied Computing (pp. 1650-1651). ACM.

[14] Arslan, M.Y., Singh, I., Singh, S., Madhyastha, H.V., Sundaresan, K. and Krishnamurthy, S.V., 2012, December. Computing while charging: Building a distributed computing infrastructure using smartphones. In Proceedings of the 8th international conference on Emerging networking experiments and technologies (pp. 193-204). ACM

[15] Zheng, K., Meng, H., Chatzimisios, P., Lei, L. and Shen, X., 2015. An SMDP-based resource allocation in vehicular cloud computing systems. IEEE Transactions on Industrial Electronics, 62(12), pp.7920-7928.

[16] Chen, M., Hao, Y., Li, Y., Lai, C.F. and Wu, D., 2015. On the computation offloading at ad hoc cloudlet: architecture and service modes. IEEE Communications Magazine, 53(6), pp.18-24.

[17] Zhang, T., Robson, E. and Boukerche, A., 2016. Design and analysis of stochastic traffic flow models for vehicular clouds. Ad Hoc Networks, 52, pp.39-49

[18] Mershad, K. and Artail, H., 2013. Finding a STAR in a Vehicular Cloud. IEEE Intelligent transportation systems magazine, 5(2), pp.55-68.

[19] Ye, T., Su, Z., Wu, J., Guo, L., Li, J. and Li, G., 2016. A Safety Resource Allocation Mechanism against Connection Fault for Vehicular Cloud Computing. Mobile Information Systems, 2016

[20] Ye, D., Wu, M., Tang, S. and Yu, R., 2016, June. Scalable fog computing with service offloading in bus networks. In Cyber Security and Cloud Computing (CSCloud), 2016 IEEE 3rd International Conference on (pp. 247-251). IEEE.

[21] Stanica, R., Fiore, M. and Malandrino, F., 2013, June. Offloading floating car data. In World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a (pp. 1-9). IEEE

[22] Mtibaa, A., Harras, K.A. and Fahim, A., 2013, December. Towards computational offloading in mobile device clouds. In Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on (Vol. 1, pp. 331-338). IEEE

[23] Shojafar, Mohammad, Nicola Cordeschi, and Enzo Baccarelli. "Energy-efficient adaptive resource management for real-time vehicular cloud services." IEEE Transactions on Cloud computing (2016).

[24] Adhikary, T., Das, A.K., Razzaque, M.A., Almogren, A., Alrubaian, M. and Hassan, M.M., 2016. Quality of service aware reliable task scheduling in vehicular cloud computing. Mobile Networks and Applications, 21(3), pp.482-493.

[25] Turcanu, I., Salvo, P., Baiocchi, A. and Cuomo, F., 2015, November. DISCOVER: a unified protocol for data dissemination and collection in VANETs. In Proceedings of the 12th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks (pp. 25-32). ACM.

[26] Liu, K., Ng, J.K., Lee, V.C., Son, S.H. and Stojmenovic, I., 2016. Cooperative data scheduling in hybrid vehicular ad hoc networks: Vanet as a software defined network. IEEE/ACM transactions on networking, 24(3), pp.1759-1773.

[27] Ghazizadeh, P., Olariu, S., Zadeh, A.G. and El-Tawab, S., 2015, June. Towards fault-tolerant job assignment in vehicular cloud. In Services Computing (SCC), 2015 IEEE International Conference on (pp. 17-24). IEEE.

[28] Ye, T., Su, Z., Wu, J., Guo, L., Li, J. and Li, G., 2016. A Safety Resource Allocation Mechanism against Connection Fault for Vehicular Cloud Computing. Mobile Information Systems, 2016.

[29] Li, B., Pei, Y., Wu, H., Liu, Z. and Liu, H., 2014, August. Computation offloading management for vehicular ad hoc cloud. In International Conference on Algorithms and Architectures for Parallel Processing (pp. 728-739). Springer, Cham.

[30] Orsini, G., Bade, D. and Lamersdorf, W., 2015, October. Computing at the mobile edge: designing elastic android applications for computation offloading. In IFIP Wireless and Mobile Networking Conference (WMNC), 2015 8th (pp. 112-119). IEEE.

[31] Wang, C., Li, Y. and Jin, D., 2014. Mobility-assisted opportunistic computation offloading. IEEE Communications Letters, 18(10), pp.1779-1782

[32] Lee, W.I., Pyun, J.Y., Lee, Y.S. and Lee, S.W., 2013. Relative velocity based vehicle–to–vehicle routing protocol over ad–hoc networks. International Journal of Ad Hoc and Ubiquitous Computing, 12(1), pp.14-22

[33] Klingler, F., Dressler, F. and Sommer, C., 2015, December. IEEE 802.11 p unicast considered harmful. In Vehicular Networking Conference (VNC), 2015 IEEE (pp. 76-83). IEEE

[34] Sommer, C., German, R. and Dressler, F., 2011. Bidirectionally coupled network and road traffic simulation for improved IVC analysis. IEEE Transactions on Mobile Computing, 10(1), pp.3-15.

[35] Varga, A. and Hornig, R., 2008, March. An overview of the OMNeT++ simulation environment. In Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops (p. 60). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[36] Krajzewicz, D., Hertkorn, G., Rössel, C. and Wagner, P., 2002. SUMO (Simulation of Urban MObility)-an open-source traffic simulation. In Proceedings of the 4th middle East Symposium on Simulation and Modelling (MESM20002) (pp. 183-187)

[37] Gramaglia, M., Serrano, P., Hernández, J.A., Calderon, M. and Bernardos, C.J., 2011, June. New insights from the analysis of free flow vehicular traffic in highways. In World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a (pp. 1-9). IEEE