Cluster Based Distributed Service Discovery in Internet of Vehicle

Nishant Sharma, Naveen Chauhan, and Narottam Chand

Abstract—**As automobile sector is witnessing a paradigm shift from Internal Combustion Engine (ICE) based vehicles to smart electric vehicles, a new concept of connected vehicles has emerged that is able to provide a number of services to its users. Internet of Vehicles has now transformed to an indispensable platform of information exchange among vehicles, city infrastructure, drivers and other connected entities. Due to highly dynamic nature of Internet of Vehicles, there is need of a dynamic service discovery infrastructure that can cater to Internet of Vehicle specific challenges. This paper is a step ahead in that direction to provide a distributed service discovery protocol that facilitates service discovery and service selection for Internet of Vehicles environment. We have proposed a clustering based service discovery approach that makes use of neighbor awareness to find neighboring vehicles. Proposed protocol has been validated by running extensive simulations and results show the improvement in terms of query success rate, transmission rate and transmission cost by a considerable margin.**

Keywords—**Internet, Connected vehicles, Internet of Vehicles, Service discovery.**

I. INTRODUCTION

We are witnessing an era of industrial transformation. Devices and machines around us are becoming more efficient as well as intelligent with passage of time. With high speed internet around us, this has led to an era of Internet of Things (IoT), where devices can communicate and exchange messages with each other to provide a set of services to its users [1]. "Internet of Things" term was first coined by Kevin Ashton in a lecture at MIT in 1999 [2]. IoT is already being used extensively in many sectors such as healthcare, smart home, smart city and many more [3]. Devices connected to internet, which were in the range of 10 billion in 2018 is expected to increase to 64 billion by 2025. A good chunk of these devices include vehicles that has given rise to an entire new paradigm called "Internet of Vehicles" or IoV.

IoV provides a platform for vehicles to exchange and communicate with each other in order to provide certain safety and other conventional services to its intended users [4]. IoV has evolved itself from a conventional Vehicular Ad-hoc Network (VANET) to a full-fledged network of heterogeneous devices of which vehicles form a major chunk.

VANET is considered as a limited network of moving vehicles, Road side units and users that are interconnected to provide a set of services within its limitations. Once a vehicle gets out of range of VANET, it gets disconnected and cannot avail its services. Other limitations of VANET include limited computation capability, frequent disconnections and dynamic topology. These limitations have resulted in limited real world use of VANET. IoV is an integration of VANET with IoT architecture to create a scalable network of vehicles, users, road side units and other heterogeneous small networks to provide different services [5]. Such services provided by a sophisticated IoV environment require a platform between service provider and service requestor. A user requiring a service shall search for it and concerned service provider should disseminate particular service to that user. This paves the way for development of a service discovery system that acts as a medium between requestor and provider. These services require a communication between service provider and service requester in order to perform dissemination and discovery processes. This lookup mechanism that enables user to locate services in IoV environment is known as a Service Discovery protocol or SDP [6]. In other words, we can define SDP as a network protocol that enables interested users to locate particular services offered by service providers.

There has already been a lot of research in the area of service discovery in Mobile Ad-hoc networks or MANETs. Unfortunately, it has little meaning and usability in context of IoV as MANET network are quite different from IoV in terms of certain characteristics such as mobility and topology [7]. An efficient and reliable SDP for IoV is need of the hour for IoV to achieve its full potential. In current scenario, most SDP's are focused on MANETs and cannot be directly implemented in IoV scenario [8]. Thus, it remains an open challenge to develop a SDP for highly dynamic IoV environment. SDP can be designed either based on a directory or it can have a directory less architecture [9]. In a directory less structure, as the name suggests, there is no need of maintaining a directory of services. In directory based architecture, a service directory is needed to coordinate service discovery process. This directory stores services provided by service providers and is either maintained in centralized form or is distributed among several nodes. Limitations of a centralized directory are quite obvious as it has a single point of failure. If the node that maintains the directory fails, whole system would come to halt. Therefore, it is recommended to maintain a distributed directory. Distributed

directories can be formed using two approaches: backbone-based approach and cluster-based approach. In backbone-based approach, nodes in the network form a backbone and it is responsible for handling the requests and advertisement of services to backbone constituents. Cluster-based approach forms cluster of a group of nodes and one cluster head is selected. Cluster head is then responsible for requests coming from the nodes in cluster and advertisement of services to these nodes. We propose a cluster-based distributed service discovery protocol (CDSD) for Internet of Vehicles that is responsible for sanctioning service discovery and selection of relevant services from vast IoV environment. CDSD generates low overhead, has less transmission cost and less waiting time for queries to be processed when compared to existing service discovery protocols.

Section II describes related work done in the field of service discovery. Section III describes proposed CDSD protocol. Section IV deals with performance analysis of CDSD and results thus obtained via simulations.

II. RELATED WORK

Service Discovery as a problem has been discussed and addressed as an optimization problem by A. Chakraborty et al. in [10]. M. Russo has addressed service discovery in mobile nodes in [11], however most of the research in this field is addressed to static nodes. D. Mascarenas et al. in [12] have proposed a Dual Beacon Discovery (2BD) protocol that is designed for Wireless Sensor Networks with mobile nodes. It considers a multi-path and mobile pattern for nodes. Distributed Service Discovery Protocol (DSDP) is also proposed in [13] by C. Hwang et al., where a hierarchical network structure has been designed for optimizing update cost. In this solution, nodes in lower hierarchical regions can move freely without the need of constantly sending update to coordinators. A. M. Khedr et al. in [14], have discussed about a sensor web that shall act as a middleware between mobile nodes and applications. Network users ranging from a cell phone to a connected vehicle can run sensor web as applications to pull service requests. M. Parhi et al. proposed a Natural Language Query Processing technique for service discovery in sensor networks [15]. Liu L. et al. propose a service discovery framework with a distributed and decentralized scheme [16]. Two bi-dimensional lists are created where one lists contains list of peers who answered to query and other list contains peers that did not answer the query. It also uses a routing algorithm to determine next node to receive query message based upon their capability to answer queries in past. Li J et al. in [17] make use of semantic descriptions and locate services based upon QoS demand of requester. Authors have also devised an efficient trust based propagation mechanism for service discovery. Author in [18] has proposed a service discovery framework for Internet of things based upon swarm intelligence. Service discovery starts with a query request to local server. Local server checks for similarity value in the query and sends it to neighboring server with highest similarity. Load management is one of the merits of this approach as server content is reorganized and distributed on number of servers. However, performance of this approach

depends upon how well descriptors and meta-data vectors are used.

As can be seen from discussed related work, most of research even in service discovery is limited to area of sensor network. Very limited existing research is there in IoT or IoT related areas to take inspiration for service discovery in Internet of Vehicles. In [19], authors have proposed a Service Discovery Protocol (SDP) for a WSN based IoT, where objective is to maximize energy efficiency without having an effect on request success rate. Ali H. Ahmad in [20] has devised a secure service discovery protocol for Internet of Things based upon secured broker based service discovery approach. This approach takes into consideration context awareness and QoS along with trust factor while delivering services to consumers. Authors in [21] have proposed a search engine based discovery framework that makes use of a central directory to store and index resources or services.

Unlike previously discussed research, CDSD works with mobile vehicular nodes with location information in IoV and does not look at it as optimization problem. CDSD uses a simple yet efficient method for network monitoring and service directory update instead of using a complex scheme to handle mobility problem as done in [13] so as to keep costs in check. CDSD uses a shared partial service directory integrated with service advertisement among moving vehicles in IoV environment. In [22], author has proposed Cluster Heads (CHs) that coordinate update process, but has not specified election process of CHs. However, in our proposed approach, we make use of hexagonal grid for network topology and we have proposed an election process that determines the monitoring vehicle for coordinating service discovery in each cell.

III. CLUSTER BASED DISTRIBUTED SERVICE DISCOVERY IN INTERNET OF VEHICLES (CDSD)

In this section, we discuss draft of proposed CDSD for Internet of Vehicles. Our proposed system model includes information related to neighbors in vicinity, service information and service directory. Figure 1 shows a lightweight 3 layered architecture of our proposed approach.

A. Proposed System Model

We define vehicular nodes in our proposed system as set of n nodes denoted by $V_i = \{V_1, V_2, V_3, \ldots, V_n\}$. Each vehicle is able to interact with each other and exchanges related information to provide different services and is responsible for neighbor discovery in its vicinity in IoV network. Vehicular node include its location information as well as neighboring and advertisement related information.

Vehicular node V_i is defined as; $V_i = <Loc, NT, AT, ST>$

where Loc denotes location of Vehicle which it gets through GPS. Vehicles get information about their neighboring vehicles and their recent location and stores it in neighbor table denoted as NT. Advertisement information is maintained in advertisement table denoted as AT. Any advertisement that is currently live in the network is stored in AT. These

advertisements are pushed onto the network by other hosts on IoV. Vehicular nodes also maintain a list of services they provide in services table. Service table includes information about type of service, frequency of service and time till when service will be available. Our system contains a comprehensive full service directory of whole network, but is distributed among all nodes as each vehicular node V_i contains a part of this service directory of its neighbor nodes. Implementation of CDSD is a three step approach.
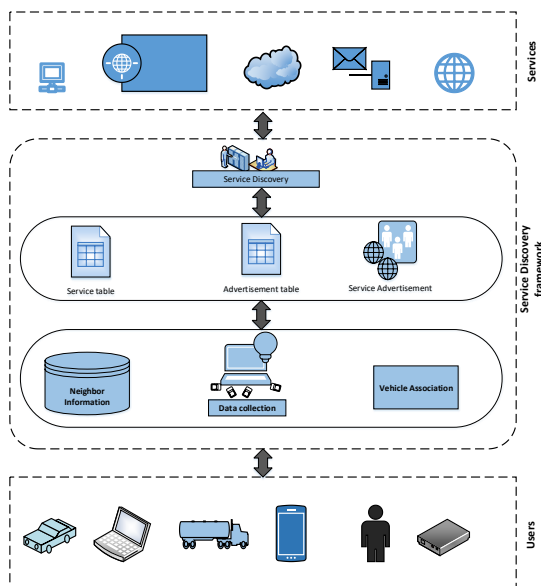

Fig. 1. 3-layered service discovery architecture

A.1 Neighbor awareness and network topology

Our routing approach in CDSD is based upon greedy geographical routing. In this approach, each node relays the message to its nearest node. To maintain topology awareness in each node, we make use of a five-step process that involves construction of a hexagonal grid; collection of information from neighboring vehicles; monitor-vehicle election; and creating and maintenance of routing table.

Hexagonal grid: An example of a hexagonal grid is demonstrated in figure 2. For vehicular nodes with assumed equal transmission radius as R, length of each side of hexagon is, $r = \frac{R}{\sqrt{13}}$.

Each hexagon in the structure represents a hex cell and each vehicle Vi with its position coordinates (x_i, y_i) is associated to a hex cell. Vehicle Vi is linked to its corresponding hex cell using a vehicle association algorithm. A vehicle Vi determines its hex-cell by calculating (a,b) as $a = \frac{x}{\alpha}$ and $b = \frac{x}{\beta}$, where $\alpha = \frac{3*r}{2}$ and $\beta = \frac{\sqrt{3}*r}{2}$.

If (a + b) comes out to be even, vehicle V_i is linked to nearest of (a,b) and (a + 1, b + 1). Else if (a + b) comes out to be odd, then V_i is linked to nearest of (a + 1, b) and (a, b + 1). Figure 3 describes our vehicle hex association approach. Using this

approach, we can successfully link each vehicle to its corresponding hex cell in a way that each vehicle is linked to some hex cell.
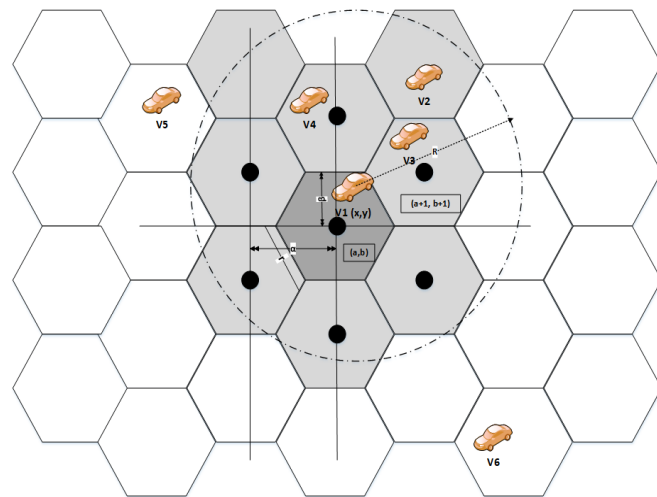

Fig. 2. Hex-cell representation in hexagonal grid structure

A.2 Information collection

Once each vehicle is linked to a hex cell, we collect vehicular information such as vehicle ID, its location and hex cell to which it belongs. Vehicles that share same hex cell are able to share their information to neighboring vehicles via broadcasting messages.

A.3 Monitor-vehicle election

Each hex cell is now assigned a monitoring vehicle that stores information of its current cell and also of adjacent cells. Selection of a monitoring vehicle is based upon algorithm 2 and we explain the approach below.

We introduce a mobility function M that will be used to select monitoring vehicle in each cell. So, consider all vehicles in a cell (a,b) as $Vi, where\ i = 1\ to\ N$. Our calculations are based upon these n vehicles within the cell. Position difference between a vehicle Vi and other N vehicles within cell is calculated as, $D_V = \sum_{n=1}^{N} \sqrt{(x_v - x_n)^2 + (y_v - y_n)^2}$. Next, we calculate speed difference between Vi and other vehicles as, $S_V = \sum_{n=1}^{N} |S_v - S_n|$. Acceleration difference between Vi and other vehicles is calculated as, $A_V = \sum_{n=1}^{N} |a_v - a_n|$. Taking all three functions in account, now we can calculate our mobility function M as, $M_v = \alpha \frac{D_v}{Max(D_n)} + \beta \frac{S_v}{Max(S_n)} + \gamma \frac{A_v}{Max(A_n)}$.

α, β and γ in the equation are special coefficients and α + β + γ = 1. These coefficients can be adjusted in respect to each other keeping in account different traffic conditions. In good traffic conditions within city limits, vehicle drive at similar speed thereby distance factor should be taken into account more. In such case, α should be more than β and γ. In case of highways, vehicle usually drive at higher speed, hence increasing β than other two factors. In case of stop and go traffic, where vehicles
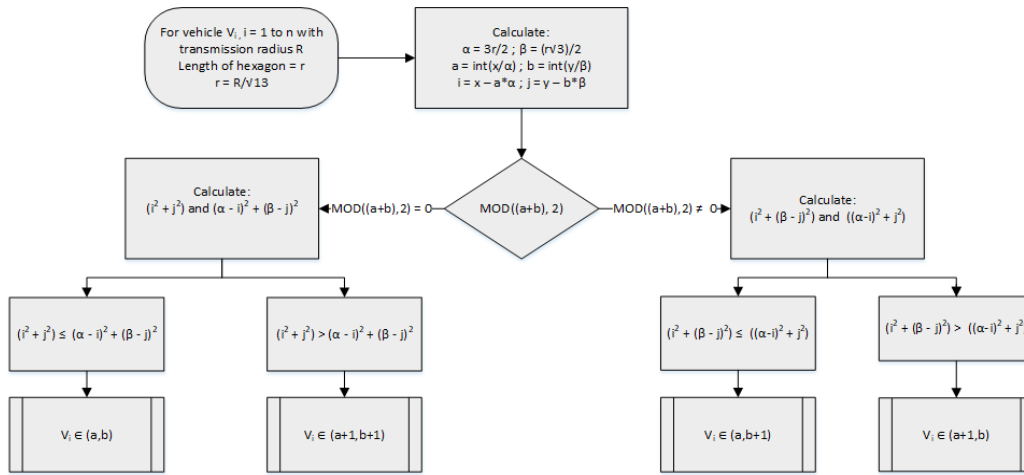
Fig. 3. Vehicle hex-cell association algorithm

accelerate at different rate, value of γ shall be higher.

Election of monitoring vehicle is based upon the notion that a vehicle that is less likely to change hex cell in near future should be selected as monitoring vehicle. Thus calculated mobility function Mv provides us with vehicle that has lower relative mobility in comparison to other vehicles in the hex cell. Algorithm 1 explains selection of monitoring vehicle among vehicles within a hex cell.

Monitoring vehicle thus selected is assigned for data routing, whereas rest of the vehicles register with monitoring vehicle using a REGISTER message. If any change occurs to a vehicle with respect to its service state or its location, it is sent via message to monitoring vehicle. Monitoring vehicle knows about total number of nodes in its corresponding cell as well as adjacent cells as well as various service being provided by them.

When a vehicle changes its cell, it inquires about monitoring vehicle of corresponding cell. If it does not get a response, it announces itself as monitoring vehicle of that cell by broadcasting LEADER message.

```
Input: N vehicles in a hex cell
Output: Monitoring vehicle of Hex cell
1 Set M_Min = +∞;
2 For each vehicle V do
3       Calculate mobility function M_v ;
4       If M_v < M_Min then
5       M_Min = M_v ;
6       D = v;
7       End
8 End
9 Return v;
```

Algorithm 1. Monitoring vehicle election algorithm

In the next step, after a monitoring vehicle is elected, monitoring vehicle broadcasts its location and hex cell number to its neighbors. When a vehicle receives such message, it updates its neighbor information. Thus, monitoring vehicle forms a routing table containing information about vehicles in adjacent cells that will be further required in data routing. In data routing, when a vehicle has a packet to be sent, it checks if the destination vehicle is within its transmission range. If it is found to be within transmission range, packet is sent directly.

However, in case when destination is out of range, a relay node is selected from routing table and forward packet to relay node. Same process is repeated at relay node till packet reaches its destination.

B. Service Advertisement

In our proposed approach, we have described a hexagonal cell structure to group our vehicles based upon their location and electing a monitoring vehicle among them in each cell. CDSD maintains a distributed directory of services in each cell using hex structure. Monitoring vehicles have comprehensive information about services available in their respective cells. A vehicle providing a specific service can register with its respective monitoring vehicle thus allowing each vehicle to maintain list of services available in its cell.

Our approach works on pull-push service model strategy. A vehicle that intends to provide a service pushes its services to advertise to its neighboring vehicles. In addition, monitoring vehicle pushes services provides by its cell to monitoring vehicles of adjacent cells. So, when one of the vehicle needs a particular service, instead of searching for service provider, it pulls information about any node with service advertisement or a node that provides service directly. When a vehicle shifts from one cell to next, it pulls information of monitoring vehicle in that cell. At any time, each vehicle is either in: 1) Idle State, 2) Service Advertisement, 3) Request Update.

In idle state, a vehicles is totally idle in IoV network although it is connected. It neither sends nor receives any advertisements. Vehicle also does not update and information in directory while in idle state. A vehicle gets into SERVICE ADVERTISEMENT state upon changing its hex cell. Each vehicle in this state sends service that it can provide as service advertisement to its respective hex cell fellows. Each monitoring vehicle in this state
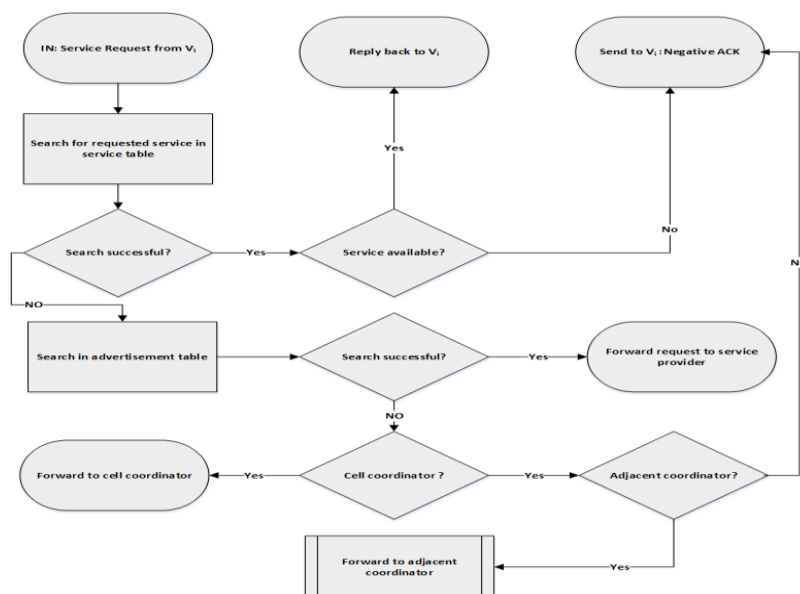
Fig. 4. Service discovery flowchart representation

does same things and sends service advertisement tables of its cell to monitoring vehicles of adjacent hex cells. In REQUEST UPDATE state, a vehicle that has moved from its hex cell updates its current cell address. It pulls information from its neighboring vehicles and requests updates on service information.

Service Discovery

To understand mechanism of service discovery in CDSD, let us take an example of set of n vehicles {Vi, Vj, Vk, …}. When a vehicle from this set Vi intends to invoke a service from other vehicles, it sends a request to one of the neighboring vehicle in its cell. Vehicle that received the request then checks if it can provide required service in the following manner. Figure 4 describes our service discovery approach in flowchart.

Vehicle that received the request first checks if service requested is in its service table. If it is available, it replies back to requesting vehicle. However, if it is not able to find it in service table, it forwards the request to monitoring vehicle. In a manner here it provides a partial service of forwarding other vehicle's request. Now monitoring vehicle looks for the service in its advertisement table. If its search returns successful and it finds a matching service as requested, service request is then forwarded to the vehicle that advertised the service. If no matching service is found in the table, monitoring vehicle sends back a message with negative acknowledgement to the node which first requested this service. Monitoring vehicle now forwards service request to next adjacent monitoring vehicles.

IV. PERFORMANCE EVALUATION

We have proposed a distributed cluster based approach for service discovery in Internet of Vehicles network. In this section, we aim to evaluate performance of our proposed protocol by running multiple rounds of simulation in varying conditions to validate our proposed technique. Results obtained from these simulations have been compared with results of Distributed Service Discovery Protocol (DSDP) and Scalable Geographic Service Protocol (SGSP). We use Simulation of Urban Mobility (SUMO) tool for simulating vehicle mobility [23]. SUMO is widely used and very popular traffic simulation package that also enables us to simulate vehicles at different speeds, mark one way or no traffic roads and many other functionalities. In SUMO, every street is assigned minimum and maximum speeds according to the digital map, and a functionality defines whether the street is a plain street, a source street or a sink street. Vehicles are generated randomly on source streets and they exit when they reach a sink street. Vehicle traces are stored in log file, which are further imported for simulations.

We have imported real world map from open street maps to SUMO with real highways and street roads [24]. SUMO then calculates length of lanes on its own. Using VEINS tool, SUMO can be integrated with OMNET++ that has been used for running further network simulations [25]. SUMO and OMNET are connected via TCP socket in veins. Veins_inet subproject is included in veins itself and it allows us to use all features of inet framework for veins simulation. Using inet framework, we have access to many network protocols such as IPV4 stack, ad-hoc network protocols to be used in simulation. Vehicular nodes thus generated from SUMO act as nodes in OMNET and vehicular movement in reflected as mobile nodes moving in OMNET. Mobility parameters used in SUMO model are described in table 1. Performance metrics that has been used for evaluation purposes have been summarized below.

Transmitted message count: Transmitted message count or discovery process. It accounts for all messages transmitted including queries, update requests and corresponding response. TMC shows total number of messages transmitted for service.

Query success rate: Query Success rate or QSR is an important metric that is used by almost every researcher to evaluate a service discovery technique. It is calculated by dividing number of messages received by number of total

transmitted messages. QSR can be represented as

$$QSR = \frac{Number\ of\ recieved\ messages}{Total\ transmitted\ messages} \times 100\%.$$

Transmission cost per query: Transmission cost per query or TCPQ takes into account all transmitted messages during both discovery as well as service delivery process of a service request.

TABLE 1.
SUMO MOBILITY PARAMETERS

Total lane length	70 kilometres
Total edge length	50 kilometres
Average vehicle speed	55 kmph
Number of Vehicles	100

V. SIMULATION RESULTS

In the first round of simulation, we analyze performance of our proposed approach CDSD by varying maximum speed of vehicles in the network. We repeat simulation five times for each round and then calculate their mean values for better results. We deploy 100 vehicles in a region of 5 x 5 km² with a fixed transmission range of 100 meters. Vehicles are deployed randomly at different source and sink points in the network and they send query message at interval of 10 seconds for a service they themselves are not providing. For 5 rounds of simulation, we set maximum speed of vehicles from 20 kmph to 100 kmph. As can be seen from Figure 5, Transmitted Message Count (TMC) increases whereas Query Success Rate (QSR) decreases slightly with increase in maximum speed of vehicle. This is due to the fact that with increase in speed, vehicle will more frequently change cells and will require to update its neighbor and service information more frequently. Due to this, some queries may fail as information may not be updated during that interval. Similarly, as the number of transmitted messages increase, transmission cost per query (TCPQ) also increases.

For our second round of simulation, we analyze how CDSD performs when transmission radius of vehicle is varied. As in previous simulation, we again deploy 100 vehicle in a region of 5 x 5 km². In this test, transmission radius is varied from 50 meters with an increment of 25 meters up to 175 meters. As transmission radius is increased, a vehicle is able to cover more area thus increasing number of neighbor nodes. With increase in neighboring vehicles, number of forwarding messages reduce. As can be seen in figure 6, TMC decreases with increase in transmission radius of vehicle nodes. This also reduces update messages that are sent when vehicle changes its cell address, thereby decreasing TCPQ. Also, with more neighboring vehicles and increase in number of vehicles in same cell, QSR increases because chances of required service being found in same cell are more neighboring nodes and hence more chances of answering a query. Transmission cost or TCPQ also decreases with number of vehicles increasing as can be seen in figure 7.

In our final round, we compare performance of our proposed approach CDSD with two existing protocols DSDP and SGSP. These protocols are designed for fulfilling service discovery requirements for mobile nodes in wireless sensor networks.

DSDP is a distributed service discovery protocol for sensor nodes that supports mobility of nodes. SGSP is Scalable Geographic routing with service provisioning for Mobile Ad-hoc Networks (MANET). Using same SUMO model that we used till now, vehicular nodes are deployed in a region of 5 x 5 km². A query is generated by a randomly selected vehicle every 10 seconds. Transmission radius of vehicles are set to 100 meters. We analyze these three approaches on same parameters that we used till now.

Looking at the graph of Query Success Rate (QSR) in figure 8, we can see that DSDP and SGSP have similar QSR. Service queries in these approaches may fail as service coordinator changes its position and moves farther away from the region. In our proposed approach CDSD, cell information is shared between all nodes in same hex-cell. Every vehicle is aware of services provided by every vehicle in same cell, thereby reducing chances of a failed query.

Transmitted Message Count (TMC), as seen from figure 8 increases for all three approaches as number of nodes increase. SGSP approach leads in number of messages transmitted as it frequently broadcasts service advertisements at global as well as local level on node movement. In DSDP approach, TMC is higher than proposed CDSD as DSDP uses hierarchal service coordination among its coordinators thereby increases total number of messages. In CDSD, we have used hex cells to form clusters of neighboring vehicles that reduces advertisement messages. CDSD also has less Transmission Cost per Query (TCPQ) than other two approaches as in our approach, each vehicle share a partial service directory that contains services being provided in current and also neighboring hex cells. Due to use of hex cell structure in CDSD, service discovery process is also faster. CDSD reduces number of messages transmitted in IoV network and also regulates message transmission. Only adjacent vehicles are sent advertisement message, which is then shared by vehicles in their respective hex cell. Messages in CDSD are shared first to adjacent cells and in later stages, messages are shared to next level of adjacent cells. So, when a vehicle changes its location to another cell, it associates with a new hex cell. Due to these features of CDSD, it outperforms other service discovery protocols.

Query Success Rate (QSR) is also higher in CDSD as each hex cell shares a partial service directory to store information of current and its neighboring cells. This approach drastically increases probability of finding and locating requested service by a vehicle.

VI. CONCLUSION

We have proposed a distributed service discovery protocol for Internet of Vehicles that uses a hex cell structure to group vehicles within a region. A monitoring vehicle is selected in each cell which is responsible for coordination with other monitoring vehicles in respective cells. This approach is highly suitable for a dynamic network such as IoV as vehicles are constantly on move and topology changes dynamically. As we can see from simulation results, we have developed a service discovery protocol with less transmission costs, less transmitted messages and higher query success rate. We have compared our
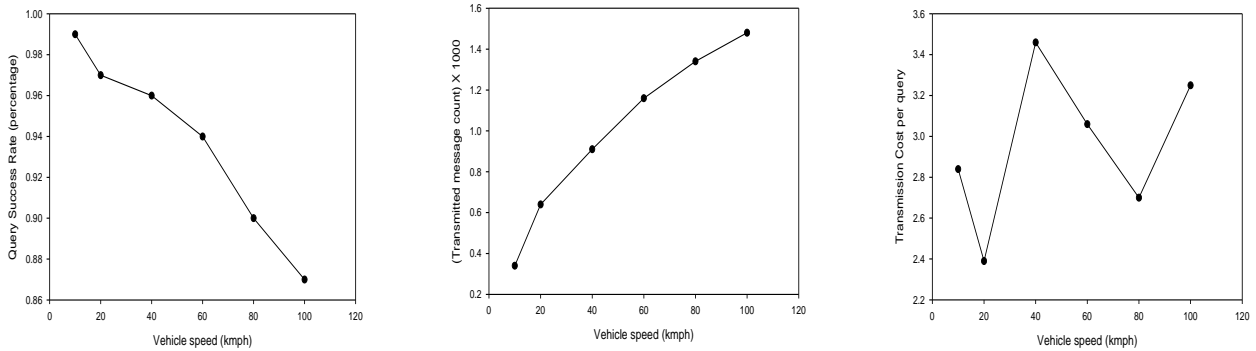
Fig. 5. Performance of CDSD with varying vehicle speed in terms of a) Query Success Rate; b) Transmitted message count; c) Transmission cost per query
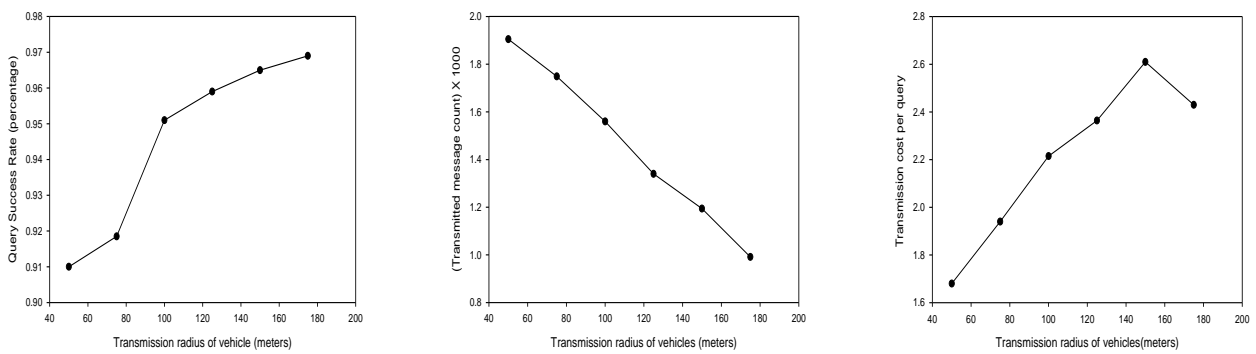

Fig. 6. Performance of CDSD with varying transmission radius of vehicle in terms of a) Query Success Rate; b) Transmitted message count; c) Transmission cost per query
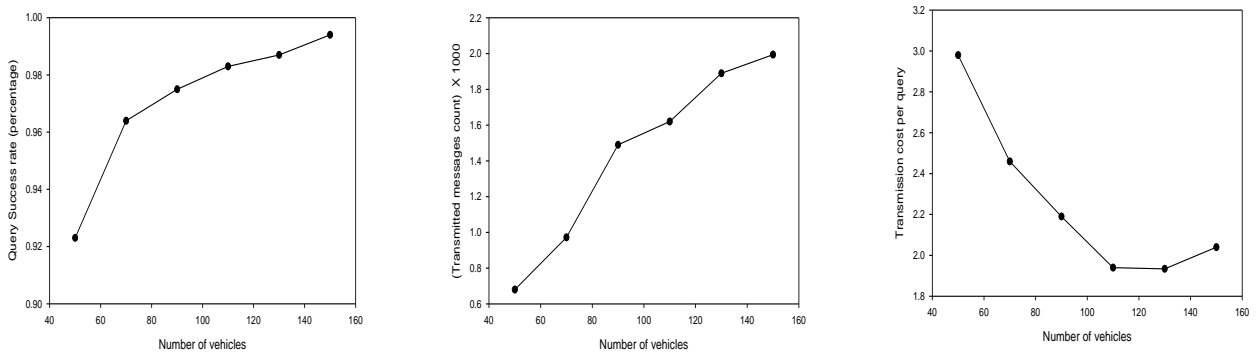

Fig. 7. Performance of CDSD with varying number of vehicles in terms of a) Query Success Rate; b) Transmitted message count; c) Transmission cost per query
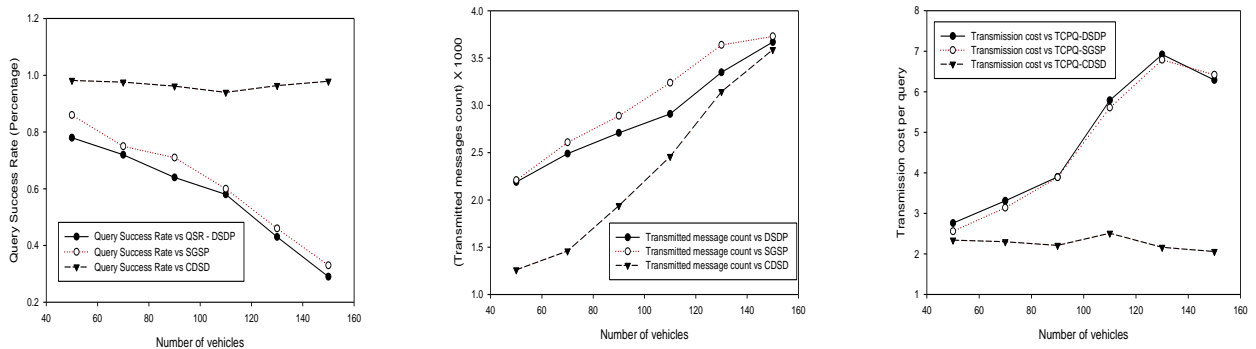

Fig. 8. Comparison of CDSD with DSDP and SGSP in terms of a) Query success rate; b) Transmitted message count; c) Transmission cost per query

approach with existing protocols and we can see from results that CDSD outperforms other approaches by quite a margin.

REFERENCES

[1] O. Bello, S. Zeadally: *Intelligent Device-to-Device Communication in the Internet of Things*, IEEE Systems Journal, vol. 10, no. 3, pp.1172-1182, 2016, doi:10.1109/JSYST.2014.2298837.

[2] D. McFarlane: *The Origin of Internet of Things*, http://www.redbite.com/the-origin-of-the-internet-of-things/. [Accessed: 19-Dec-2020].

[3] S. Hanif, A. M. Khedr, Z. Al Aghbari, D. P. Agrawal: *Opportunistically exploiting Internet of Things for wireless sensor network routing in smart cities*, J. Sens. Actuator Netw., vol. 7, no. 4, p. 46, 2018, doi: 10.3390/jsan 7040046.

[4] F. Cunha, L. Villas, A. Boukerche, G. Maia, A. Viana, R. A. F. Mini, A. A. F. Loureiro: *Data communication in VANETs: Protocols, applications and challenges, Ad Hoc Networks*, volume 44, 2016, pages 90-103, ISSN 1570-8705, doi:10.1016/j.adhoc.2016.02.017.

[5] Wu, Z. Yang, K. Li, B. Iov: *"Internet of Vehicles and Applications*, Internet of Things 2016, pp. 299- 17, ISBN 9780128053959, doi:10.1016/B978-0-12 805395-9.00016-2.

[6] Q. Wei, Z. Jin: *Service discovery for Internet of Things: A context awareness perspective*, in Proc. 4th Asia–Pacific Symp. Internetware, Qingdao, China, Oct. 2012, Art. no. 25, doi:10.1145/2430475.2430500.

[7] F. Yang, S. Wang, J. Li, Z. Liu, Q. Sun: *An overview of Internet of Vehicles*, in China Communications, vol. 11, no. 10, pp. 1-15, Oct. 2014, doi: 10.1109/CC.2014.6969789.

[8] K. Abrougui, R. W. N. Pazzi: *Towards a Balanced and Reliable Localization of Services in Heterogeneous Vehicular Ad Hoc Networks*, pp. 63–69, 2011, doi: 10.1145/2069105.2069116.

[9] L. Albraheem, M. AlRodhan, AbduAllah Aldhlaan: *Toward Designing Efficient Service Discovery Protocol in Vehicular Networks*, Internet of Vehicles – Technologies and Services First International Conference, IOV 2014 Beijing, China, September 1-3, 2014 Proceedings, doi: 10.1007/978-3-319-11167-4.

[10] A. Chakraborty, K. Lahiri, S. Mandal, D. Patra, M. K. Naskar, A. Mukherjee: *Optimization of service discovery in wireless sensor networks*, in Proc. WWIC in Lecture Notes in Computer Science, vol. 6074, 2010, pp. 351–362, doi: 10.1007/978-3-642-13315-2_29.

[11] M. Russo: *Service discovery for ip-based wireless sensor networks*, M.S. thesis, Dept. Inf. Technol., Inst. Inf. Technol., Chennai, Indian, Oct. 2012. [Online].Available:http://uu.divaportal.org/smash/get/diva2:562852/FU LLTEXT01.pdf.

[12] D. Mascarenas, E. Flynn, C. Farrar, G. Park, M. Todd: *A mobile host approach for wireless powering and interrogation of structural health monitoring sensor networks*, IEEE Sensors J., vol. 9, no. 12, pp. 1719–1726, Dec. 2009, doi: 10.1109/JSEN.2009.2030706.

[13] C. Hwang, E. Talipov, H. Cha: *Distributed geographic service discovery for mobile sensor networks*, Comput. Netw., vol. 55, no. 5, pp. 1069–1082, 2011, doi: 10.1016/j.comnet.2010.09.015.

[14] A. M. Khedr, W. Osamy, D. P. Agrawal: *Perimeter discovery in wireless sensor networks*, J. Parallel Distrib. Comput., vol. 69, no. 11, pp. 922–929, 2009, doi: 10.1016/j.jpdc.2009.08.002.

[15] M. Parhi, B. M. Acharya, B. Puthal: *Discovery of sensor Web registry services for WSN with multi-layered SOA framework*, in Proc. 2nd Int. Conf. Comput. Commun. Technol. (ICCCT), Allahabad, India, Sep. 2011, pp. 524–530, doi: 10.1109/ICCCT.2011.6075183.

[16] J. Li, Y. Bai, N. Zaman, V. C. M. Leung: *A decentralized trustworthy context and QoS-aware service discovery framework for the internet of things*, IEEE Access, vol. 5, pp. 19154-19166, 2017, doi: 10.1109/ACCESS.2017. 2756446.

[17] E. Rapti, C. Houstis, E. Houstis, A. Karageorgos: *A bio-inspired service discovery and selection approach for IoT applications*. In: 2016 IEEE International Conference on Services Computing (SCC), pp. 868–871 (2016), doi: 10.1109/SCC.2016.126.

[18] A. Forestiero: *A smart discovery service in internet of things using swarm intelligence*", pp. 75–86, Springer, Cham (2017), doi: 10.1007/978-3-319-71069-3_6.

[19] T. Dang, P. Minet, P. Bellot, C. Mozzati, E. Livolant: *Industrial IoT: Mobility support and service discovery for industrial process monitoring*, in Internet of Things: Novel Advances and Envisioned Applications. Cham, Switzerland: Springer-Verlag, 2016.

[20] A. H. Ahmad, N. M. Omar, H. M. Ibrahim: *Secured service discovery in IoT*, Journal of communications, Vol. 14(1), pp. 40-46, doi: 10.12720/jcm.14.1. 40-46.

[21] S. K. Datta, R. P. F. Da Costa, C. Bonnet: *Resource discovery in Internet of Things: Current trends and future standardization aspects*, IEEE 2nd World Forum on Internet of Things (WF-IoT), Milan, Italy, 2015, pp. 542-547, doi: 10.1109/WF-IoT.2015.7389112.

[22] Z. Chen, N. Chen, L. Di, J. Gong: *A flexible data and sensor planning service for virtual sensors based on Web service*, IEEE Sensors J., vol. 11, no. 6, pp. 1429–1439, Jun. 2011. doi: 10.1109/JSEN.2010.2095839.

[23] Krajzewicz, Daniel & Hertkorn, George & Feld, Christian & Wagner, Peter. (2002), *SUMO (Simulation of Urban MObility); An open-source traffic simulation*, 4th Middle East Symposium on Simulation and Modelling (MESM2002), pp. 183-187.

[24] http://www.openstreetmap.org/. [Accessed:25-Dec-2020].

[25] C. Sommer, R. German and F. Dressier: *Bidirectionally coupled network and road traffic simulation for improved IVC analysis*, IEEE transaction on mobile computing, vol. 10 (1), pp 3-15, January 2011, doi: 10.1109/TMC. 2010.133.

Nishant Sharma received his M.Tech degree in Computer science and engineering from Punjabi University, Patiala in 2015. Currently, he is doing his research in area of Internet of Vehicles under supervision of Dr. Naveen Chauhan from National Institute of Technology, Hamirpur. His main research interests include Internet of Things and Vehicular networks. He has published his research work in journals as well as in different international conferences.

Naveen Chauhan is Associate Professor at Department of Computer Science and Engineering, NIT Hamirpur. He has received his Ph.D. (Computer Science \& Engineering) from NIT Hamirpur in 2012. His research interest includes Mobile Wireless Networks with special emphasis towards Mobile Ad hoc Networks, Sensor Networks, Vehicular Ad hoc Networks, Internet of Things, Internet of Vehicles and their Security aspects. He has excellent research contributions in these areas and has published research articles in large number of Journals (including SCI and Scopus indexed Journals). He is reviewer for various journals of national and International repute, such as: Springer Journal of Wireless Networks, Journal of Optical Communication, Wiley Journal of Communication System. He is also handling externally sponsored R\&D project in the field of, "Post Disaster Communication Recovery using Integrated Mobile Ad Hoc Network". He has got international exposure by visiting countries like USA, UK and UAE and presented research papers in international conferences held in these countries.

Narottam Chand Kaushal received his PhD degree from IIT Roorkee in Computer Science and Engineering. Previously he received MTech and BTech degrees in Computer Science and Engineering from IIT Delhi and NIT Hamirpur, respectively. Presently he is working as associate professor, Department of Computer Science and Engineering, NIT Hamirpur. Dr. Narottam is an active industry consultant for entities like BHEL and Kangra Central Cooperative Bank. His current research areas of interest include mobile computing, mobile ad hoc networks and wireless sensor networks. He has published more than 150 research papers in International/National journals & conferences, guided six PhDs and guiding few more in these areas. He is member of IEEE, ISTE, CSI, International Association of Engineers and Internet Society.