# A Modified Fault Tolerant Location-Based Service Discovery Protocol for Vehicular Networks

Saeed Fathi Ghiri[1] and Morteza Rahmani and Hassan Almasi[2]

[1]Department of Computer Engineering, Azad University of Ghir-o-Karzin,Ghir,Iran
saeedfathi641@gmail.com

[2] Department of Electrical Engineering, Zanjan University, Zanjan,Iran
rahmanim@znu.ac.ir and hassan_almasi@znu.ac.ir

## ABSTRACT

*In the recent years, advances in Vehicular networks have attracted special attraction of researchers. Lately two types of applications have gain popularity: Road safety and Driving comfort. Reliable data transformation in the city environment is hard to accomplish due to presence of noise and obstacles. In addition transient or permanent faults of vehicles or roadside routers (road components) are unavoidable, so we need a fault tolerant algorithm to overcome such failures. Although utilizing fault-tolerant techniques cause to more efficiency and reliability in service discovery for vehicle networks, there are many few service discovery algorithms that have considered fault- tolerant techniques. In this paper we have improved one of these algorithms which is named Fault-Tolerant Location-Based Vehicular Service Discovery Protocol (FLocVSD) in order to being more reliable.*

## KEYWORDS

*fault tolerant, vehicle network, service discovery, location-based*

## 1. INTRODUCTION

Vehicular networks and their applications have gained great attention from research community and vehicle industry during recent years. Two main applications which have gained popularity lately are *Road safety* and *Driving comfort* [1,2]. Road safety applications deal with enhancement of driver and passenger safety on roads and Deriving comfort applications deal with such services that makes driving more enjoyable and more comfort. Due to these applications are provided by some service providers that generally are not known by others, derivers or passengers who want to utilize such applications should discover a suitable service provider. Service discovery is very important for suitable work of these applications. Failure of any road components is unavoidable during service discovery process. Thus, this is evident that we need to design a fault-tolerant service discovery protocol to overcome such failures and present a dependable service discovery.

Researchers have proposed some service discovery protocols but most of them have high overload and considerable poor performance in vehicular networks [3,4]. Raychoudhury et al. proposed a distributed directory- based service discovery protocol for mobile ad hoc networks. Their protocol is based on the selection of the top K nodes considered as directories for rich resources [5]. They have based their idea on the construction of multiple quorums and the replication of the registration of services among the members of the quorum. Their purpose is increasing the service availability and ensuring the fault tolerance of their proposed service discovery protocol.

DOI : 10.5121/acij.2012.3404

Another proposed protocol is in [6] and it is a location-based services over vehicular ad hoc networks using car-to-car communication. Their protocol provides time-sensitive information about the traffic conditions and the available services on the roadside, however. Their protocol has poor performance with presence of failure.

Reference [3] has proposed efficient and scalable infrastructure based context-aware and Location-Based Vehicle Service Discovery Protocol (LocVSDP) for vehicle ad hoc networks, but their algorithm does not work properly with presence of failure in road components and the performance is considerably degraded.

Our proposed protocol is based on FTLocVSDP [7] which is proposed by Abrougui et al., recently. They have considered fault tolerance features in their work and it has a good performance at most of conditions.

The reminder of this paper is organized as follow. In section II, we would describe the system model. Section III, presents our modified fault tolerant location based vehicular service discovery protocol and in section IV, we conclude this paper.

## 2. SYSTEM MODEL

In this work, we assume that our system uses wireless network and its traffic is based on Manhattan traffic model. Communication between road components is base IEEE 802.11 standard.

We have two kinds of road components in our model:

***Roadside Router (RR):*** we assume that RRs have two or more radio interfaces in order to send and receive data and use an efficient diverse channel. They use a Connection Less Approach in Street environment (CLA-S) [8] as their packet transfer protocol.

***Vehicle:*** Our model is not based on any predetermined or specific kind of vehicles. Any vehicle that is equipped with wireless interface can use our model.

We assume RRs are randomly distributed in the streets and there is not any predefined structure. Although, the configuration of RRs usually is such that in the congested areas we have more RRs to make routing more reliable, but it is not restriction for our model. This assumption helps us to apply this model to any existing network without any change in its structure. Our model is a cluster based model in other words, each RR is belonged to one or more clusters. The RRs that are in communication range of each other form a cluster and if a RR is not in communication range of any other RR, it forms a cluster by itself.

Any road component and any connection link between them can fail at any time during the service discovery. The connection links can fail because of interfering noise in the link medium, or presence of any temporary or permanent obstacle which disconnects a connection. Road components can fail due to loose of their electricity power, or failure in their devices such as processor, memory or other part of them, or failure in their software. these faults can influence the performance of service discovery and even can cause failure of whole service discovery process if it not fault tolerant.

## 3. The Description of Modified Fault Tolerant Location-Based Vehicular Service Discovery Protocol:

This protocol is based on a cluster-based infrastructure of RRs, where the routers are distributed near service providers, in congested road sections and in areas with intermittent vehicle connection. RRs are randomly distributed in the VANet, and RRs in the communication range of each other form a cluster. Our proposed MFTLocVSDP tolerates vehicular network component failures during the phases of the enhanced LocVSDP [3].

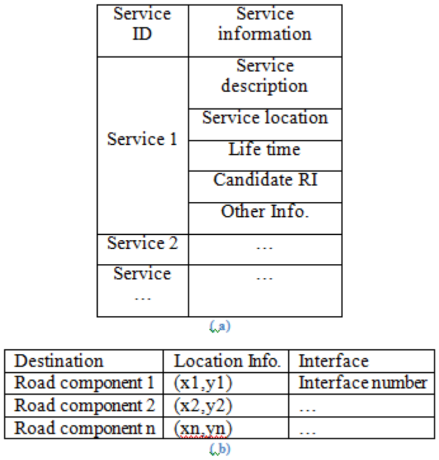
It is assumed that the requester (driver or passenger) determines the specifications of his/her desired service. These specifications are the region of interest (RI) and the desired type of service. Only services which match the desired criteria will be returned.

Another assumption is that our system utilizes a cluster-based infrastructure. Infrastructure that relies on clusters is very convenient for location-based applications, where clusters are formed such that RRs can cover all sites in their neighbourhod. A cluster can contain one or more RRs. A cluster is formed by RRs that are near each other and can communicate with each other, or an RR that is far from the others and could ‘t communicate with others, can form a cluster by itself. Vehicles establish the communication between clusters.

In the following, we describe the four modified phases of LocVSDP: (a) Service advertisement; the service providers advertise themselves by sending a message to the neighboring RRS. (b) Service request propagation; the drivers or passengers broadcast their request and other road components propagate this request. (c) Leader election and service reply generation; the clustered RRs in the RI elect a RR as leader to be responsible of sending a unique service reply message, and (d) Service reply propagation; the unique service reply which is generated by the leader is forwarded through intermediate road components to the service requester.

## 3.1. Service Advertisement Phase

In this phase, service providers advertise themselves through advertisement messages. This protocol is designed such that it can handle three types of services efficiently: (i) fixed services; (ii) moving services; and (iii) migratory services. Fixed services refer to those services that have a predetermined location, and their position does not change over time. Examples of fixed services could be restaurants with their menus, gas stations with their prices, or parking with available spots. Moving services refer to those services that provided by the vehicles on roads. The location of these services depends on the location of the moving vehicle. Examples of this type of service could be file sharing, music sharing, or even movie sharing between drivers or passengers. Migratory services are named to those services that have a fixed location, but they are provided by moving vehicles. When vehicles are moving around the fixed location, they provide the service; when they are far from the fixed location, the service migrates to provider



| Service ID | Service information |
|---|---|
| Service 1 | Service description |
| | Service location |
| | Life time |
| | Candidate RI |
| | Other Info. |
| Service 2 | … |
| Service … | … |

(a)

| Destination | Location Info. | Interface |
|---|---|---|
| Road component 1 | (x1,y1) | Interface number |
| Road component 2 | (x2,y2) | … |
| Road component n | (xn,yn) | … |

(b)

Figure 1 Formats of routing and service tables. (a) Service table format. (b) Routing table format.

vehicles close to the fixed location. Examples of this type of service could be traffic conditions, or sightseeing services where the location of the service is fixed but the providers are moving. In our protocol, a service advertisement message is received by neighboring road vehicles or RRs that are in the transmission range of the sending service provider. Only RRs will handle the advertisement messages after their reception by the integrated module. The integrated module receives the advertisement message, separates service information from the routing information and processes each type of information differently. Each kind of information is processed by its associated module. The service information is handled by the service module, which updates the service table by adding the service information if it is new or, if it is not new, by modifying the existing service information. This table is illustrated in fig.1.a. On the other hand, the routing information is processed by the routing module and it uses the received information for updating the routing table. This table is shown in fig.1.b.

## 3.2. Service Request Propagation Phase

In this phase, a driver or passenger looking for a service located in a specific area that we call a region of interest (RI) generates and broadcasts a service request message. The request message comprises the current location and speed of the sending vehicle. Road components in the transmission range of the requester receive the request message. This message is processed differently depending from whether the receiving component is a RR or a vehicle. If the receiving component is a RR and it is inside the RI, the RR initiates the leader election phase. If the receiving RR is outside the RI, it checks if it is closer to the center of the RI than the sending component. If it is closer, then the RR broadcasts the request message. But, if a vehicle receives the request and it receives the same message from a neighboring RR, it concludes that the message is processed by a RR and ignores the request message. If the vehicle has not received the same message from a neighboring RR, and it is inside the RI, then it broadcasts the request message. If the road vehicle is the requested service provider it sends a reply message to the requester. If the receiving road vehicle is not inside the RI but closer than the sending component to the RI, then it broadcasts the request message. This way, the request messages are handled principally by RRs and that road vehicles permit the connectivity between service requesters, service providers and RRs. This scheme guarantees that if there is no RR inside a RI, the service request message reaches the service provider and all service providers' information inside a RI are returned to the requester.

While there is no faulty component in the service request message path the service request reaches to its associated service provider, but by presence of failure this scenario is far from the truth.

We assume that each RR can be aware of its adjacent RR or its link failure by using a failure detection algorithm which is proposed in [7]. Each RR that detects failure in its adjacent and wants to forward a service request sets a *force to broadcast flag* high. Any road component that receives a message, checks its "force to broadcast" flag, if it is in high state, it broadcast the message without even its distance to RI is larger than the sender. It guaranties that the service request message reaches the associated service provider [7].

## 3.3. Leader Election and Service Reply Generation Phase

The purpose of the leader election phase is to connect all RRs inside the RI in a spanning tree and avoiding sending many service replies to the service requester if there are many service providers offering the same service inside the RI. The leader election phase starts when a RR inside a RI receives a request message. First, the RR sets its state to *leader election state*. Second, the RR generates an election message for the request that contains the current RRID, its geographic location, and its distance to the center of the RI. Third, the RR broadcasts the election message in its range. A RR inside the RI that receives the election message enters the *leader election state*, generates an election message with its information, and broadcasts the message. A road vehicle inside the RI that receives the election message and does not receive it again from another RR, rebroadcasts the message. A service provider that receives the election message sends a local reply message to the RRID stored in the message using the CLA-S routing protocol. The clustered RRs inside the RI exchange election messages. This allows each RR inside the RI to learn the distance of each of its neighbors from the center of the RI. After a predetermined period of time, each RR can determine whether it is a leader or a follower in the spanning tree of a connected RR's cluster. A RR changes its state to the leader state *leader state* if its distance to the center of the RI is the smallest one among its neighbors. All other RRs in the RI change their states to *follower state*. Then, the leaves in the spanning tree starts ending their local replies to their parents containing information about the requested service provider if it exists in their ranges. All the local replies are propagated to the leader RR. Many clusters of RRs could exist in one RI, therefore for every cluster inside a RI, a spanning tree is constructed and a leader is elected.

The leader election phase can be fault tolerant by adding a "force to broadcast flag" to the packet which each RR sends during the leader election phase. In other words when a RR detects a failure it sets the flag high and broadcast it. The other road components receiving this message check this flag if it is high, they broadcast it, otherwise they do as they supposed with respect to distance to RI [7].

This phase of this algorithm which is proposed in [7] fails in this scenario: assume that a RR in the RI is elected as leader because it is the nearest RR to the center of the RI. The other RRs in the RI send their local service replies to their parents and they add their local reply to it and forward it to the leader. The leader receives all local replies and aggregates it as a unique service reply message. If at this time the leader experiences a failure, its aggregated message is never sent and any of the road components cannot be informed, because they are not waiting for any reply.

We propose a modified algorithm which can tolerate this kind of failure. This algorithm is described bellow:

When a RR receives a service request message, if it is not in leader election phase for the same request, it changes its state to leader election state, adds its distance from center of the RI to the message and depending on it detects a failure in it's a adjacent it sets or resets the "force broadcast flag" and forward it. All RRs in the RI exchange the leader election message and finally, the RR with the shortest distance to the center of RI is determined as leader and the others as followers. The followers with respect to their distance are divided in two categories: parents and leafs. Then they send their local reply messages. First leafs send their local replies to their correspondent parents. Then patents add their local replies two them and forward them to the leader. The leader forms a unique aggregated message and broadcast it. The parents wait for a limited predefined period and if any of them did not heard a broadcasted service reply message with the same ID as service request ID they assume that a failure has occurred and start leader election phase again. If the failure was a transient one they elect the previous leader one more time, but if it was a permanent fault the elect another RR which is the nearest to the center of RI with respect to the remaining RRs. Finally, the elected leader sends a unique aggregated service reply to the requester.

Another way to overcome such failures is repeating service request for some more times, for examples three times, but this way causes more traffic on the network and wastes network's resources. We assume both of this ways simultaneously in order to guaranty that the service discovery is properly performed.

### 3.4. Aggregated service reply generation and propagation phase

Every elected leader inside the RI receives all the local replies from its descendants. The elected leader aggregates all this local replies and their service providers' information and generates a unique aggregated service reply message containing this information in its cluster. The leader sends this service reply message to the service requester through the CLA-S routing protocol. The information of the RI for which the service reply is generated is included in the service reply. This way, intermediate roadside routers during the service reply message propagation can cache the information of the service providers for the requested RI and can use this information to answer future requests.

As mentioned before, this phase does not work properly with presence of failure. Each RR that detects failure in its adjacent and wants to forward a service reply sets a "*force to broadcast flag"* high. Any road component that receives a message, checks its "force to broadcast" flag, if it is in high state, it broadcast the message even its distance to RI is larger than the sender. If the flag is not high the road component forwards it to the service requester if it is nearer to service requester. This guaranties that the service reply message reaches the associated service requester [7].

## 3. CONCLUSIONS

In this paper, we presented our modified fault tolerant location based service discovery protocol MFTLocVSDP for vehicular networks which relies on a cluster-based infrastructure. We discussed the MFTLocVSDP behavior with the presence of roadside router failures and wireless link failures. We illustrated that the leader in the FTLocVSDP can be a single point of failure for a specific time. Our algorithm is a modified version of the FTLocVSDP and we proposed a scheme to overcome such failure in the leader.

## REFERENCES

[1] C. H. Gan and Y. B. Lin, "Push-to-talk service for intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 391–399, Sep. 2007.

[2] F. J. Martinez, C. K. Toh, J. C. Cano, C. T. Calafate, and P. Manzoni,"Emergency services in future intelligent transportation systems based on vehicular communication networks," *IEEE Intell. Transp. Syst. Mag.*,vol. 2, no. 2, pp. 6–20, Summer 2010.

[3] K.,Abrougui, A.,Boukerche and R.W.N., Pazzi. "Design and evaluation of context-aware and location-based service discovery protocols for vehicular networks". IEEE Trans- actions on Intelligent Transportation Systems 2011;23:717–35.

[4] K.,Abrougui, A.,Boukerche and R.W.N., Pazzi." Context-aware and location-based service discovery protocol for vehicular networks". In: Proceedings of the 6th ACM symposium on performance evaluation of wireless ad hoc, sensor, and ubiquitous networks. , ACM; 2009. p. 93–100.

[5] V.Raychoudhury ,J. Cao ,W. Wu , W. Lai , C. Chen, J Ma . "K-directory community: reliable service discovery in MANET". In: Pervasive and mobile computing; 2010.

[6] M.D. Dikaiakos , A. Florides, T. Nadeem, and L. Iftode, . "Location-aware services over vehicular ad-hoc networks using car-to-car communication". IEEE Journal on Selected Areas in Communications 2007;25(8):1590–602.

[7] K.,Abrougui, A.,Boukerche and H. Ramadan, "Performance evaluation of an efficient fault tolerant service discovery protocol for vehicular networks", Journal of Network and Computer Applications, In Press, Corrected Proof, *Available online 7 November 2011.*

## Authors

**Saeed Fathi Ghiri,** received the B.Sc.degree in Electrical Engineering from the Lorestan University, Khoram Abad, Iran in 2010 and M.Sc. degree in Electrical ngineering from the Zanjan University, Zanjan, Iran, in 2012. He has been working on fault-tolerant protocols for wireless sensor networks and mobile computing, neural networks and computer vision. His research interests include medical image processing, computer networks.

Morteza Rahmani, received the M.Sc. degree in Electrical Engineering from the Zanjan University, Zanjan, Iran, in 2012. He has been working on image processing, computer networks, neural networks and robatic. He is working now on Smart Surveillance systems specifically automatic traffic control.

**Hasan Almasi**, received the B.S.E.E. Islamic Azad University (South Tehran Branch), Tehran, Iran, in 2010, and the M.S.E.E. from the Electrical Department of Zanjan University in 2012. Since 2010 he has activities on analog electronic circuits and neural networks. Analog Electronic and computer networks is also among his research areas. His research interests include deep-submicron CMOS architectures and circuits for frequency synthesizers, transmitters, receivers and computer networks.