Valentyna Osiievska, Yuriy Kulakov.

SDN-BASED ROUTING TECHNIQUE IN INTELLIGENT VEHICLE NETWORKS

Annotation. The article addresses traffic engineering (TE) issue in software-defined vehicular networks (SDVN). A modified method of TE is proposed, considering the peculiarities of SDN technology. This method allows us to reduce the time complexity of the generation routing information in the network switch, significantly.

Keywords: traffic engineering (TE), intelligent vehicle networks (IVN), software-defined vehicular networks (SDVN), Vehicle ad Hoc Network (VANET)

Relevance of research topic. Relevance of research topic. Recently, due to the ever-growing number of vehicles on the roads vehicular ad hoc networks are becoming one of the most promising research fields. VANET establishes a mobile network between moving vehicles by treating each vehicle as a node in the network. Communicating with each other these nodes form an Intelligent Vehicle Network (IVN), which is one of the essential elements of smart cities.

Setting the research issue. Compared to static or low-speed moving nodes in a traditional wireless network, vehicles move faster and more unpredictably, resulting in frequent changes to VANET network settings. Most well-known routing methods are not effective for VANET networks. In this regard, the topical task is traffic engineering methods considering the peculiarities of VANET networks. One of the promising approaches to solving this problem is the integration of VANET networks with software-defined networks (SDN).

Analysis of the last scientific researches and publications. The quality of service on a VANET network mainly depends on routing protocols. Focusing on the specifics of VANET, many protocols and routing algorithms are considered [1], taking into account all its special features. Many approaches are being considered to obtain the optimal protocol taking into account the parameters of Quality of Service (QoS) [2], such as security, low collision, and interference [3]. In the article [4], materials for reviewing routing protocols that are based on several metrics are presented. The protocol of special distance vectors on demand (AODV) is most popular in VANET networks [5]. Like all distance vector routing protocols, the AODV protocol has a relatively long redirection time. The wave routing algorithm proposed in [6] provides the ability to form many paths from intermediate nodes to the final one simultaneously. Thus, we obtain a reduced time complexity of the formation of a given set of paths, in contrast to the already known methods of forming a plurality of paths. The wave algorithm is most effective in the presence of a central network control device, such as for example, SDN technology [7]. The paper [8] describes a study of the integration of SDN and VANET technologies.

Defining of uninvestigated parts of the general issue. Since the traffic engineering, taking into account, the features of the VANET, remains relevant today, in this work, we propose a modification of the algorithm for generating routing information in switchers

The statement of basic materials. Routing information is generated directly in the SDN controller. In our work, the transport IVN will be presented in the form of a weighted graph $G(V, E, W)$, where: $V = \{v_i \mid i = 1, 2, \dots, n\}$ — set of vertices, representing intersections on the roads; $E = \{e_i\, j \mid j = 1, 2, \dots, k\}$ — is the set of edges representing the paths connecting the intersections in the given order; $W = \{w_j \mid j = 1, 2, \dots, m\}$ — is the set of weights of the edges representing the distance, time or any other metric for path characterize. During the experiment, we used the travel time of the path section as a metric. Obviously, to determine the weight of the whole path, you need to summarize the metrics of the sections that correspond to searching path. During the formation of traffic, for each vertex in the controller, a table of routes of the virtual switch (S_i) is created. These controllers contain information for the further formation of many paths. You can see an example of such a table in Table1 and Table 2. In more detail, we will talk about each value of the table during the description of the experiment. To fill the table with data, we have to search the shortest, intersected paths between the source and destination vertexes. For each such path, a new row with the actual information about its loading, metric, adjacent vertex in the direction of the destination vertex, etc. will be stored in the accordance routing table. To make the search process of disjoint paths the most optimal, in the proposed algorithm, the formation of routing information is carried out in the opposite direction from the destination to the source of the path. Another advantage of this method is the formation of optimal paths from intermediate vertices because this cancels the necessity of researching a path from an intermediate vertex to a destination

At the beginning of the path formation, a source (v_s) and destination vertices (v_d) are specified. For each of the vertices, a set of adjacent vertices is sequentially formed. Since the formation of paths starts from the end vertex, the set V_1 will contain this vertex, and the adjacent set V_2 will contain the vertices adjacent to at least one of the vertices of the previous level. Using these sets we can determine the direction vector of the path from the current to the end node and write the corresponding information in the table of the corresponding switch. All these switch tables are stored in the database. If source vertex is reached in the current set of the adjacent vertexes, routing formation stops, another way, a new set of vertices adjacent to previous was created and the information exchange between routing tables repeats.

The advantage of the proposed algorithm is that there is no need for reanalysis if it is necessary to simultaneously form paths for alternative vertices. Suppose we need to find a new optimal path from (v_{s1}) to (v_{d1}). If a switch table already exists for the source vertex of this path, this path will easily get from the routing table for the source vertex. Since all data is stored in the database, even if the formation of the previous path is not completed, routing data may be sufficient to construct the optimal path from (v_{s1}) to (v_{d1}). Such a modification significantly reduces the time complexity of path formation. Moreover, this method provides constant monitoring of the load of the entire system and dynamic formatting of information in the routing tables of the corresponding switches.

Experiments. As an example, we consider the formation of routing information when transmitting information from switch S_1 (v_1) to switch S_9 (v_9) of the transport network, of which is shown in Fig. 1. As a metric, we will consider the travel time of the path.

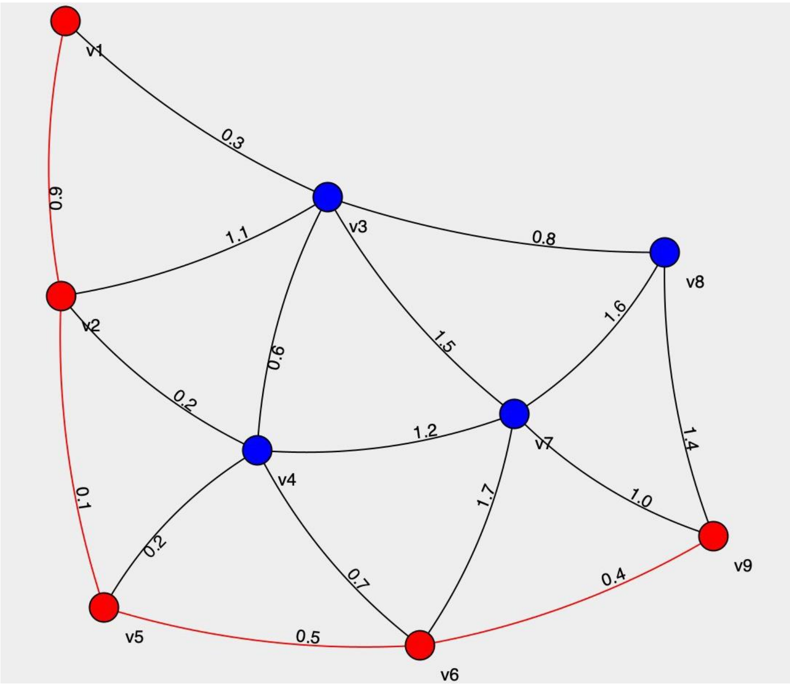

Fig. 1. Graph representation of the transport network

At the first stage, sets $V_1 = \{v_d\}$ and $V_2 = \{v_6, v_7, v_8\}$ are formed. Creating routing tables for all the vertices of set V_2 with their further filling by exchanging routing information between switches of the same level. If the start vertex and the destination vertex is not reached in level of V_i, new set of vertices have to be created V_{i+1}. Procedure of information exchanging repeats until source vertex will not be reached. As a result, new routing data is stored in the database and new tables are created.

Thus, we get many paths from the starting node to the final one and choose the optimal one from them, taking into account the load on the transport network. This path is represented by the red line in Fig. 1. Table 1 shows the routing table of switch S_1 (v_1), drawn from the database.

Table 1

Switch routing table S_1

Controller id	Destination	Adjacent vertex	Path weight	Path loading	Path
1	v9	v3	2.0	41.7714062500	v1 -> v3 -> v4 -> v6 -> v9
1	v9	v2	1.9	32.8476454293	v1 -> v2 -> v5 -> v6 -> v9

If in the process of forming many paths there is a need to search for another optimal path, then first step is to search for such path in the database. If any information was not found, the process of generating multiple paths starts again. The system load information is being updated. For example, it is necessary to form a path from the vertex v_4 to the vertex v_8. Routing data has already been found in the database and is ready for use. In Fig. 2. you can see two optimal paths performed in one act of analysis.

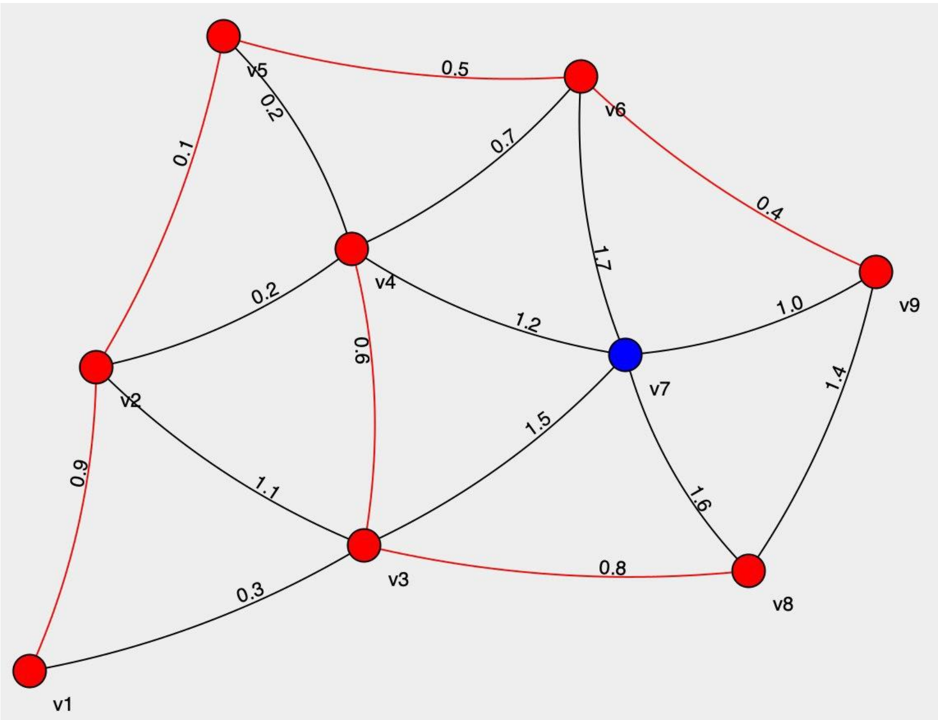

Fig. 2. Graph representation of the transport network with two optimal paths from source to destination

According to the table 2, we can see the routing data of switch routing table S_4 (v_4). You may notice that table 2 stores information collected as a result of the formation of many paths (the first 4 lines) from vertex (v_1) to vertex (v_9). The fifth line displays information about another path from the vertex v_4 to the vertex v_8. To determine this path, there was no need to re-analyze. If you take a closer look, then this path is contained in line 2 from which you can obtain the necessary information.

Table 2

Switch routing table S_4

Controller id	Destination	Adjacent vertex	Path weight	Path loading	Path
4	v8	v7	2.2	7.3615702479	v4 -> v7 -> v9
4	v8	v3	2.8	10.1315192743	v4 -> v3 -> v8 -> v9
4	v8	v6	1.1	33.9979338842	v4 -> v6 -> v9
4	v8	v5	1.1	45.237832874	v4 -> v5 -> v6 -> v9
4	v8	v3	1.4	8.938775510204	v4 -> v3 -> v8

Conclusions. Due to the number of vehicles on the roads is growing fast, vehicular ad hoc networks require much more research. Each participating vehicle is turned into a wireless router or node. The convergence of SDN with VANET can solve most current VANET problems, such as dynamic path reconfiguration, traffic balancing, and a number of other problems.

The proposed modification of the routing information generation algorithm in switches allows the path formation between different vertices at the same time. This way, we receive reduced time complexity, constant monitoring of the load of the entire system, and dynamically formatting information in the routing tables of the corresponding switches.

References

1. J. Kakarla, S. S. Sathya, B. G. Laxmi, et al, A survey on routing protocols and its issues in VANET. Int. Comput. J. Appl.28:, 38–44 (2011).

2. Mezher, A.M.; Igartua, M.A. Multimedia Multimetric Map-Aware Routing Protocol to Send Video-Reporting Messages over VANETs in Smart Cities. IEEE Trans. Veh. Technol. 2017, 66, 10611–10625. DOI: 10.1109/TVT.2017.2715719

3. Sottile, C.; Santamaria, A.F.; Marano, S. A reactive routing protocol for VANETs based on composite metric concept. In Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2014), Monterey, CA, USA, 6–10 July 2014; pp. 642–646. DOI: 10.1109/SPECTS.2014.6880005

4. Carolina Tripp-Barba Aníbal Zaldívar-Colado , Luis Urquiza-Aguiar and José Alfonso Aguilar-Calderón Survey on Routing Protocols for Vehicular Ad Hoc Networks Based on Multimetrics Electronics 2019, 8(10), 1177; https://doi.org/10.3390/ electronics8101177

5. Kulakov, Y., Kogan, A.: The method of plurality generation of disjoint paths using horizontal exclusive scheduling. Adv. Sci. J. 10, 16–18 (2014). https://doi.org/10.15550/ASJ.2014.10. ISSN 2219-746X

6. Wenfeng Xia, Yonggang Wen, Chuan Heng Foh, Dusit Niyato, Haiyong Xie A Survey on Software-Defined Networking IEEE Communication Surveys & tutorials, vol. 17, no. 1, 27-51 (2015) https://www.ntu.edu.sg/home/ygwen/ Paper/ XWF-CST-15.pdf

7. Lionel Nkenyereye , Lewis Nkenyereye, S. M. Riazul Islam, Yoon-Ho Choi, Muhammad Bilal and Jong-Wook Jang Software-Defined Network-Based Vehicular Networks: A Position Paper on Their Modeling and Implementation 2019, 19, 3788; doi:10.3390/s19173788 www.mdpi.com/journal/sensors

AUTORS

Valentyna Osiievska – student, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Politechnic Institute”.

E-mail: valentinaosievskaya@gmail.com

Kulakov Yurii (supervisor) – Doctor of Technical Sciences, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Politechnic Institute”.