Analysis of Time Monitoring Parameters in VANETs

Reema Sandhu[1], Mukesh Kumar[2]

[1]Research Scholar, NIILM University, Kaithal
[2]Professor, NIILM University, Kaithal

Abstract:
For vehicle communications a new effective certificateless aggregate signature system is being introduced. The proposed class scheme is primarily designed to secure vehicle communications in VANETs by drastically reducing signature verification time and helping to validate further messages in the stated time, thereby increasing them. The proposed scheme has much less computational cost in terms of checking signatures. This system would work effectively in networks that have restricted resources such as Ad-hoc vehicle networks. The research approach is integrated with the real time communication with satellite and base station in parallel. This is the key reason that the approach can take more time in the real time interaction and more packets transfer to the base station and satellite is giving more efficiency and security to the overall Intelligent Transportation Network for Internet of Vehicles.

Keywords: VANETs, Authentication, Latency.

1.Introduction

Latency

In a real transmission, there is packet loss, and then retransmissions and further latency in the transmission path. Latency is the period prerequisite to transfer a packet transversely in a network:

- Latency might be estimated from numerous points of view: round trip, one way, and so on.
- Latency might be affected by any component in the chain which is utilized to send information: workstation, WAN connections, switches, neighbourhood (LAN), server and at last it might be restricted, on account of huge organizations, by the speed of light.

In MATLAB, the Integration for Latency is done with the wireless and signal processing features used with the association of latency parameter. In addition, the mathematical formulation to address and evaluate the latency is done in the source whereby the packets transmission and delay are logged and analysed. Following equations or formulas are used for the network latency.

$$Network\ Latency = (Propagation\ delay) + (Serialization\ delay) \qquad eq.(1.1)$$

$$Propagation\ delay = \frac{(Distance)}{(Speed)} \qquad eq.(1.2)$$

$$Serialization\ delay = \frac{(Packet\ size\ (Bits)}{(Transmission\ rate\ (bps)} \qquad eq. (1.3)$$

$$\begin{aligned} &Propagation\ time \\ &\quad = (Frame\ Serialization\ time) + (Link\ media\ delay) + (Queueing\ delay) \\ &\quad + (Node\ processing\ delay) \qquad eq. (1.4) \end{aligned}$$

$$Frame\ serialization\ time = \frac{S}{R} \qquad eq. (1.5)$$

$$Link\ media\ delay = \frac{D}{p} \qquad eq. (1.6)$$

$$Queueing\ delay = \frac{Q}{R} \qquad eq. (1.7)$$

$$Node\ processing\ delay\ is\ normally\ specified\ or\ measured \qquad eq. (1.8)$$

$$Variable\ decoder: R \dashrightarrow Link\ data\ rate\ \left(\frac{bits}{second}\right) \qquad eq. (1.9)$$

$$Variable\ decoder: S \dashrightarrow Packet\ size\ (bits) \qquad eq. (1.10)$$

$$Variable\ decoder: D \dashrightarrow Link\ distance\ (meters) \qquad eq. (1.11)$$

$$Variable\ decoder: P \dashrightarrow Processing\ delay\ (seconds) \qquad eq. (1.12)$$

$$p: medium\ propagation\ speed\ \left(\frac{meters}{second}\right) \qquad eq. (1.13)$$

$$speed\ in\ copper\ is \dashrightarrow 210 * 10 ** 6 \qquad eq. (1.14)$$

$$speed\ in\ fiber\ is \dashrightarrow 300 * 10 ** 6 \qquad eq. (1.15)$$

$$\begin{aligned} &Q: Queue\ depth\ (bits) \dashrightarrow \\ &note\ if\ the\ link\ is\ not\ congested, there\ is\ no\ Queue\ depth \qquad eq. (1.16) \end{aligned}$$

Latency refers to the time it takes for data or a request to be conveyed from source to destination. Such latency explanations maybe because of saturating port of the network protocol, protocol failures, packet fragments, provider upstream outages, routing problems, etc. Packets queuing at any gateway in the process of travel are the most frequent source of latency. For latency of the network, it can be established as soon as the request from the sender to the recipient is processed and the recipient is needed. The entire trip from the browser to the server is also required. It is certainly preferred that you stay as close to 0 for this time, but you can play certain stuff to avoid latency on your website. If a channel's duration is low and bandwidth is low, therefore the performance is low.

If, however, the latency is low and the bandwidth high, higher efficiency and more effective link would be possible. Latency produces network congestion, thus reducing the volume of data that can be processed. Latency is an inherent feature of our networking which can be reduced, though not removed. Latency is a time delay indicator. In a network, latency tests the time taken to bring some data through the network to its destination. The time taken for information to arrive at and return to its destination is usually measured as a round trip delay. The delay in the journey is an essential measurement since a device using a TCP / IP network sends a small number of data to its place of business and then waits for a reception before transmitting it. The delay in the journey thus has a direct influence on network efficiency.
The smaller the bandwidth of a network, the quicker it is. A fast ping is a more sensitive link calculated in milliseconds (ms). Low latency and low bandwidth also mean low performance. Thus, although data packets can be supplied without delay, there can still be significant congestion because of a limited bandwidth.

Table 1.1: Latency: 100 Nodes

Simulation Attempt Scenario	Smart Middleware Architecture	Greedy Heuristics Approach	Improvements (Percentage)
1	2.5	2.9	7.41
2	2.1	2.4	6.67
3	2.3	2.6	6.12
4	1.9	2.1	5
5	2.4	2.6	4

Figure 1.1 shows the slighter values of Latency parameter in middleware architecture when contrasted with greedy heuristic approach. On evaluation of latency on 100 nodes, the improvements (in percentage) is 7.41, 6.67, 6.12, 5, 4 respectively in heuristic approach for different simulation attempts. This outcome displays that middleware approach is better when we compare it with the greedy approach.

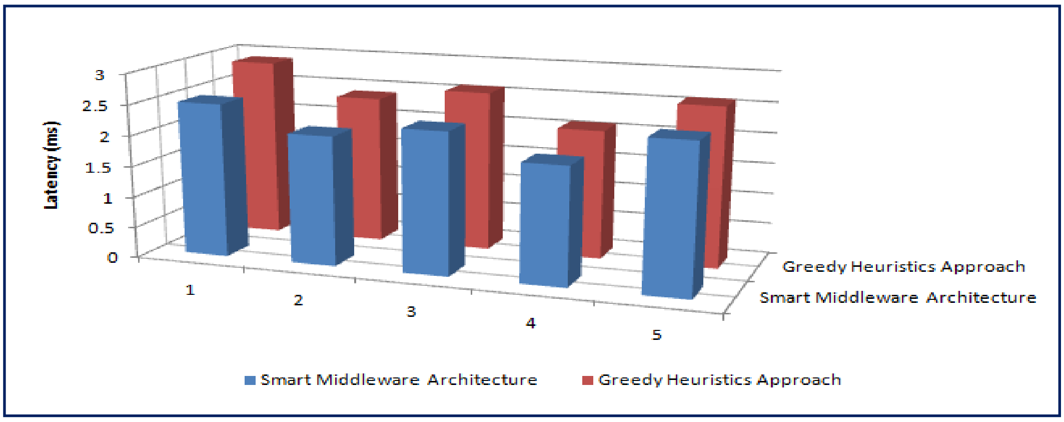

Figure 1.1: Latency: 100 Nodes

Table 1.2: Latency: 200 Nodes

Simulation Attempt Scenario	Smart Middleware Architecture	Greedy Heuristics Approach	Improvements (Percentage)
1	2.1	2.4	6.67
2	2.3	2.7	8
3	2.2	2.6	8.33
4	1.5	2.1	16.67
5	2.4	2.8	7.69

Figure 1.2 displays the value of Latency on 200 nodes. This result depicts that middleware approach is better as there is enhancements (in percentage) of 6.67, 8, 8.33, 16.67, 7.69 correspondingly in greedy heuristic approach for the changed simulation attempts. The outcome illustrates that smart middleware approach is improved in comparison with greedy approach.

$$For\ caluculation\ of\ improvements, the\ formula\ is\ as$$
$$\rightarrow \left(\frac{x-y}{(x+y)}\right) * 100,$$
$$x\ is\ middleware\ approach\ \ and\ y\ is\ greedy\ heuristic\ approach \qquad eq. (1.1)$$

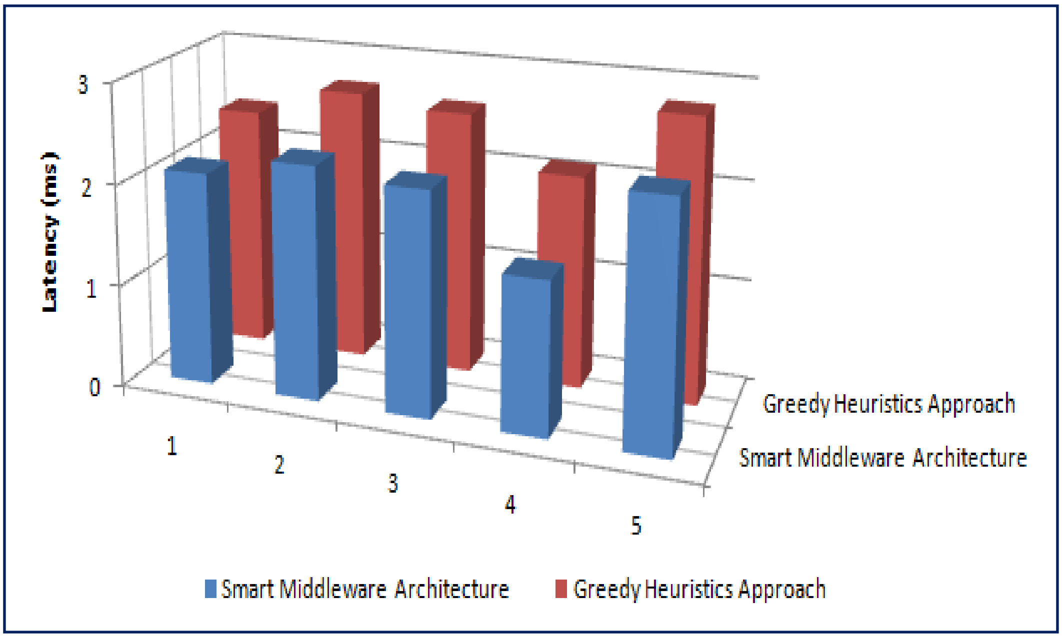

Figure 1.2: Latency: 200 Nodes

Table 1.3: Latency: 300 Nodes

Simulation Attempt Scenario	Smart Middleware Architecture	Greedy Heuristics Approach	Improvements (Percentage)
1	1.8	2.4	14.29
2	2.1	2.4	6.67
3	2.4	2.6	4
4	1.7	2.2	12.82
5	2.3	2.5	4.17

On evaluation of latency on 300 nodes, the improvements (in percentage) is 14.29, 6.67, 4, 12.82, 4.17 respectively in greedy heuristic approach for the diverse simulation attempts. Figure 1.3 displays that there is huge improvement in latency value in greedy approach on 300 nodes. The results are varying drastically but it displays enhancement for middleware approach on every simulation attempts.

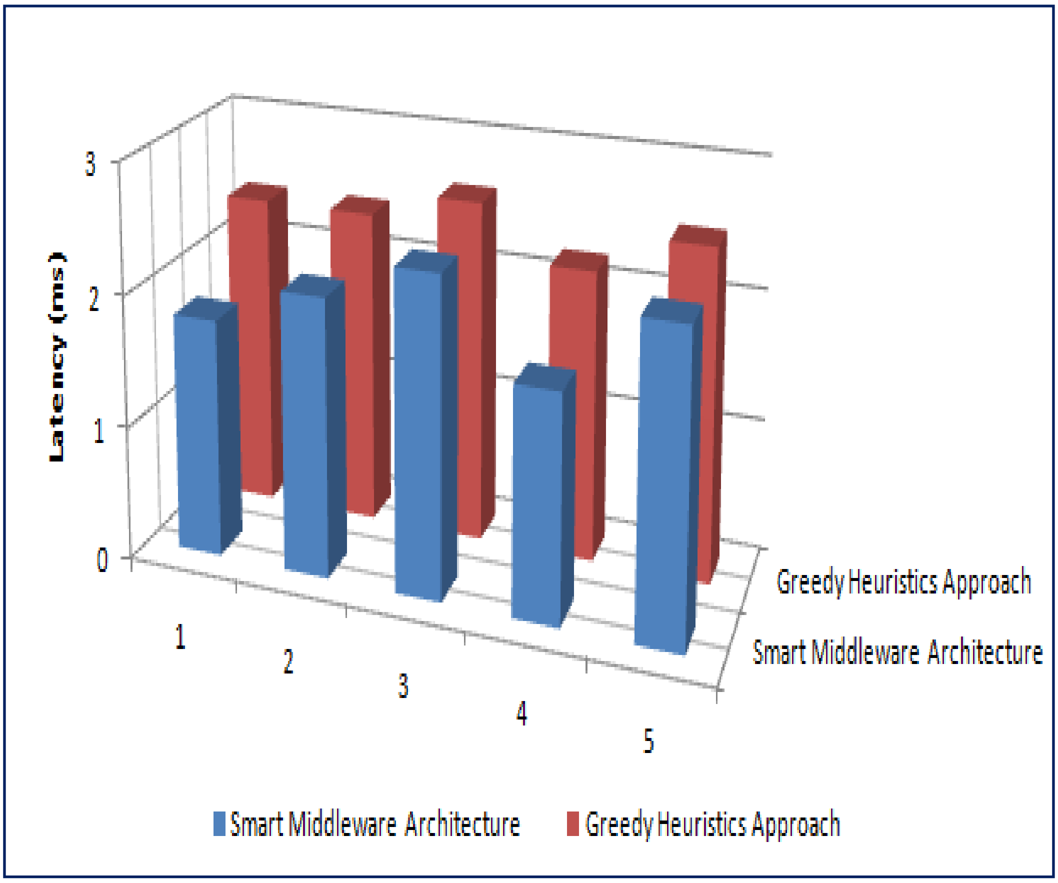

Figure 1.3: Latency: 300 Nodes

Table 1.4: Latency: 400 Nodes

Simulation Attempt Scenario	Smart Middleware Architecture	Greedy Heuristics Approach	Improvements (Percentage)
1	1.8	2.3	12.20
2	2.1	2.4	6.67
3	2.4	2.7	5.88
4	1.7	2.4	17.07
5	2.4	2.9	9.43

On evaluation of latency on 400 nodes, the improvements (in percentage) are 12.20, 6.67, 5.88, 17.07, 9.43 correspondingly in greedy heuristic approach as compared with smart middleware architecture for different simulation attempts. Figure 1.4 and Table 1.4 illustrates that smart middleware approach is very substantial in every single simulation attempts, when compared with greedy heuristic approach.

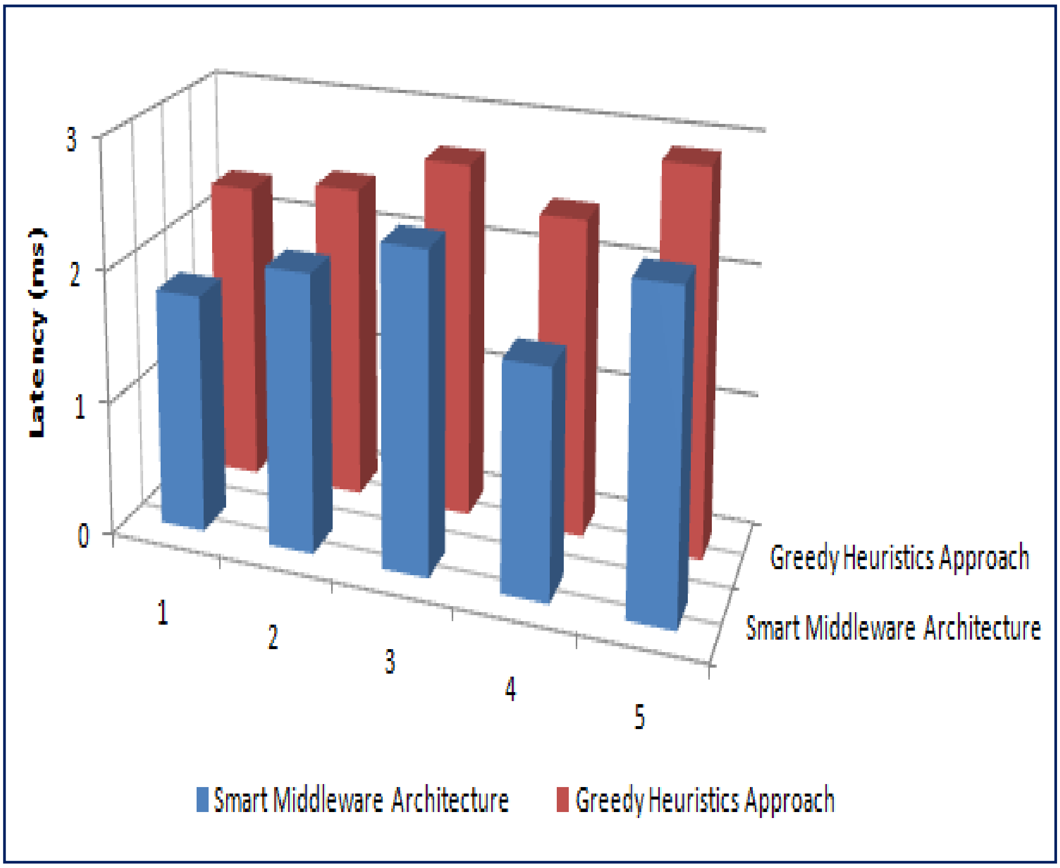

Figure 1.4: Latency: 400 Nodes

Table 1.5: Latency: 500 Nodes

Simulation Attempt Scenario	Smart Middleware Architecture	Greedy Heuristics Approach	Improvements (Percentage)
1	1.3	1.6	10.34
2	1.5	2.3	21.05
3	2.2	2.4	4.35
4	1.6	2.3	17.95
5	2.3	2.7	8

On estimation of latency on 500 nodes, the enhancements (in percentage) is 10.34, 21.05, 4.35, 17.95, 8 correspondingly in greedy heuristic approach as compared with smart middleware approach for the different simulation attempts. Figure 1.5 of Latency illustrates that for every simulation attempts on 500 nodes value, the middleware approach has reduced values of delay.

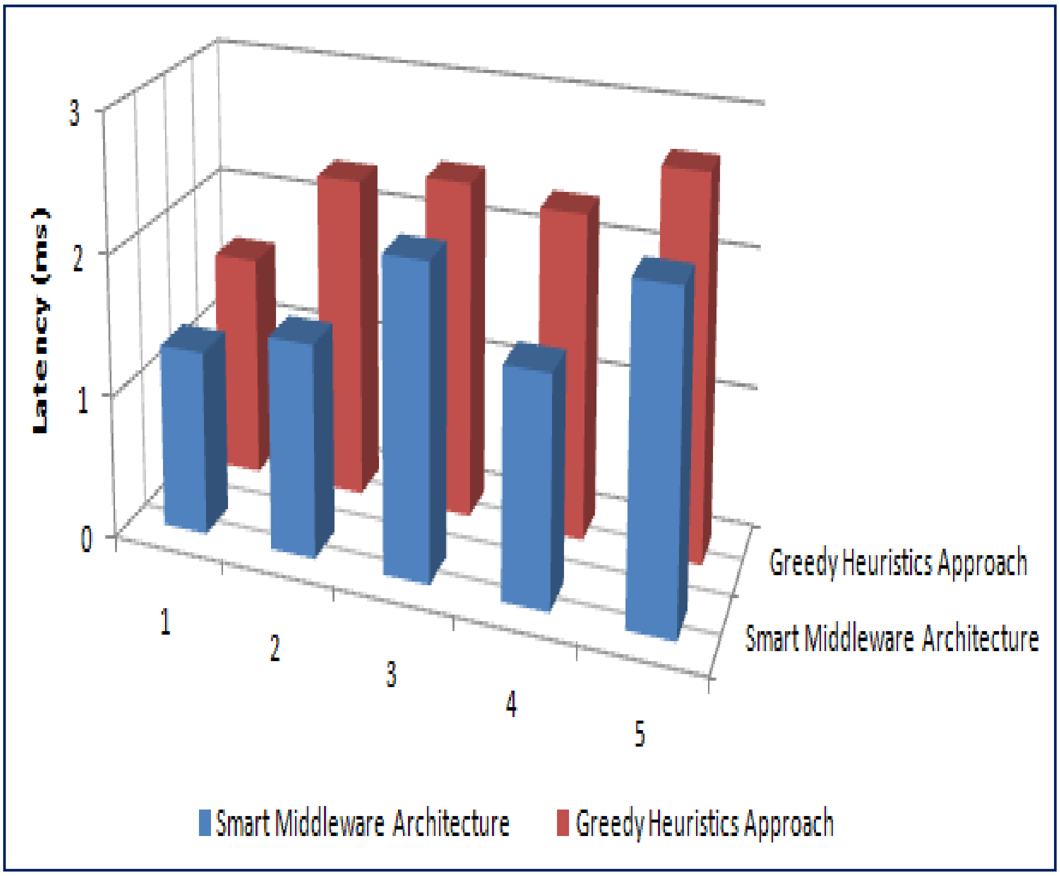

Figure 1.5: Latency: 500 Nodes

Conclusion

Performance parameters are selected for this purpose are latency These are standard performance metrics to Results for VANET smart middleware architecture are compared with greedy heuristic approach using the parameters stated above. It is observed that middleware approach performs better than the greedy heuristic approach with respect to all metrics. In other words, smart middleware approach achieves low latency, small power dissipation with varying number of nodes.

References

[1] Khamayseh, Y. M., BaniYassein, M., AbdAlghani, M., and Mavromoustakis, C. X., (2013), “NETWORK SIZE ESTIMATION IN VANETs,” Network Protocols and Algorithms, 5(3), pp. 136-152.

[2] Kumar, R., and Dave, M., (2013), “A Framework for Handling Local Broadcast Storm Using Probabilistic Data Aggregation in VANET,” Wireless Personal Communications, 72(1), pp. 315–341.

[3] Prabhakar, M., Singh, J. N., and Mahadevan, G., (2013), “Defensive mechanism for VANET security in game theoretic approach using heuristic based ant colony optimization,” International Conference on Computer Communication and Informatics, doi:10.1109/iccci.2013.6466118.

[4] Xie, X., Peng, M., Li, Y., Wang, W., and Poor, H. V. (2014). Channel Estimation for Two-Way Relay Networks in the Presence of Synchronization Errors. IEEE Transactions on Signal Processing, 62(23), pp. 6235–6248.

[5] Araujo, G. B., Queiroz, M., Duarte-Figueiredo, F. de L. P., Tostes, A. I. J., and Loureiro, A.F., (2014), "CARTIM: A proposal toward identification and minimization of vehicular traffic congestion for VANET," IEEE Symposium on Computers and Communications (ISCC), pp. 1-6.
[6] Chaubey, N.K., (2016), "Security analysis of Vehicular Adhoc networks (VANETs): A comprehensive study," International Journal of Security and Its Applications,10(5), pp.261-274.
[7] Urquiza-Aguiar, L., Tripp-Barba, C., and Aguilar Igartua, M., (2016), "A Geographical Heuristic Routing Protocol for VANETs," Sensors, 16(10), pp. 1-28.
[8] Bhuvaneswari, M., Jyothirbindhu V., and Paramasivan, B., (2017), "Statistical adaptive heuristic model supported driver intended destination and path prediction algorithm for vehicular ad hoc network," International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), Chennai, pp. 1-6.
[9] Faraj, M. F., Sarubbi, J. F. M., Silva, C. M., and Martins, F. V. C., (2018), "A Memetic Algorithm Approach to Deploy RSU s Based on the Gamma Deployment Metric," IEEE Congress on Evolutionary Computation (CEC), pp. 1-8.
[10] Hajlaoui, R., Moulahi, T., and Guyennet, H., (2018), "A Scatter Search Based Heuristic for Reliable Clustering in Vehicular Ad Hoc Networks," Artificial Intelligence Applications and Innovations, doi:10.1007/978-3-319-92007-8_43, pp. 507-519.
[11] Zhang, D., Zhang, T., and Liu, X., (2018), "Novel self-adaptive routing service algorithm for application in VANET," Applied Intelligence, 49(5), pp. 1866–1879.
[12] Al-Hashimi, A., Oza, P., Gerdes, R., and Cahantem, T., (2019), "The disbanding attack: Exploiting Human-in-the-loop control in Vehicular Platooning," ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Published by Springer Nature Switzerland AG 2019, pp. 163-183.
[13] Ghazzai, H., Khattab, A., and Massoud, Y., (2019), "Mobility and Energy Aware Data Routing for UAV-Assisted VANETs," IEEE International Conference of Vehicular Electronics and Safety (ICVES), pp. 1-6.
[14] Chahal, M., and Harit, S., (2019), "Optimal path for data dissemination in Vehicular Ad Hoc Networks using meta-heuristic," Computers and Electrical Engineering, 76 (2019), pp. 40–55.
[15] Nitti, M., Girau, R., Floris, A., and Atzori, L., (2014), "On adding the social dimension to the Internet of Vehicles: Friendship and middleware," 2014 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom). doi:10.1109/blackseacom.2014.6849025, pp. 134-138.
[16] Saravanaguru, R. K., and Thangavelu, A., (2014), "CoMiTe: Context Aware Middleware Architecture for Time-Dependent Systems: A Case Study on Vehicular Safety," Arabian Journal for Science and Engineering, 39(4), doi:10.1007/s13369-014-0955-4, pp. 2895–2908.