Data and Metadata. 2025; 4:521
doi: 10.56294/dm2025521

ORIGINAL

Intelligent Data-Driven Task Offloading Framework for Internet of Vehicles Using Edge Computing and Reinforcement Learning

Marco Inteligente de Descarga de Tareas Basado en Datos para el Internet de los Vehículos Utilizando Computación en el Borde y Aprendizaje por Refuerzo

Anber Abraheem Shlash Mohammad[1,2], Sulieman Ibraheem Shelash Al-Hawary[2,3], Ayman Hindieh[3], Asokan Vasudevan[4], Hussam Mohd Al-Shorman[5], Ahmad Samed Al-Adwan[6], Muhammad Turki Alshurideh[7], Imad Ali[8]

[1]Digital Marketing Department, Faculty of Administrative and Financial Sciences, Petra University. Jordan.
[2]Research follower, INTI International University. 71800 Negeri Sembilan, Malaysia.
[3]Electronic Marketing and Social Media, Economic and Administrative Sciences Zarqa University. Jordan.
[4]Faculty of Business and Communications, INTI International University. 71800 Negeri Sembilan, Malaysia.
[5]Department of Management Information Systems, Faculty of Amman College, Al-Balqa Applied University. Jordan.
[6]Business Technology Department, Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University. Amman, Jordan.
[7]Department of Marketing, School of Business, The University of Jordan. Amman, Jordan.
[8]GNIOT Institute of Management Studies. Greater Noida, Uttar Pradesh, India.

Cite as: Shlash Mohammad AA, Shelash Al-Hawary SI, Hindieh A, Vasudevan A, Mohd Al-Shorman H, Al-Adwan AS, et al. Intelligent Data-Driven Task Offloading Framework for Internet of Vehicles Using Edge Computing and Reinforcement Learning. Data and Metadata. 2025; 4:521. https://doi.org/10.56294/dm2025521

Submitted: 14-04-2024 **Revised:** 22-08-2024 **Accepted:** 02-12-2024 **Published:** 01-01-2025

Editor: Adrián Alejandro Vitón Castillo

Corresponding Author: Imad Ali

ABSTRACT

Introduction: the Internet of Vehicles (IoV) was enabled through innovative developments featuring advanced automotive networking and communication to fulfill the need for real-time applications that are latency-sensitive, such as autonomous driving and emergency management. Given that the servers were much farther away from the actual site of operation, traditional cloud computing faced huge delays in processing. Mobile Edge Computing (MEC) resolved this challenge by enabling localized data processing, reducing latency and enhancing resource utilization.
Method: this study proposed an Efficient Mobile Edge Computing-based Internet of Vehicles Task Offloading Framework (EMEC-IoVTOF). The framework integrated deep reinforcement learning (DRL) to optimize task offloading decisions, focusing on minimizing latency and energy consumption while accounting for bandwidth and computational constraints. Offloading costs were calculated using mathematical modeling and further optimized through Particle Swarm Optimization (PSO). An adaptive inertia weight mechanism was implemented to avoid local optimization and enhance task allocation decisions.
Results: the proposed framework was thus proved effective for any latency reduction and energy consumption optimization in efficiently improving the overall system performance. DRL and MEC together facilitate scalability in task distribution by ensuring robust performance in dynamic vehicular environments. Integration with PSO further enhances the decision-making process and makes the system highly adaptable to dynamic task demands and network conditions.
Discussion: the findings highlighted the potential of EMEC-IoVTOF to address key challenges in IoV systems, including latency, energy efficiency, and bandwidth utilization. Future research could explore real-world deployment and adaptability to complex vehicular scenarios, further validating its scalability and reliability.

Keywords: Internet of Vehicles (IoV); Mobile Edge Computing (MEC); Task Offloading; Deep Reinforcement Learning (DRL); Particle Swarm Optimization (PSO).

RESUMEN

Introducción: el Internet de los Vehículos (IoV) se habilitó a través de desarrollos innovadores que presentan redes automotrices avanzadas y comunicación para satisfacer la necesidad de aplicaciones en tiempo real que son sensibles a la latencia, como la conducción autónoma y la gestión de emergencias. Dado que los servidores estaban mucho más alejados del sitio real de operación, la computación en la nube tradicional enfrentaba grandes retrasos en el procesamiento. La Computación en el Borde Móvil (MEC, por sus siglas en inglés) resolvió este desafío al permitir el procesamiento de datos localizados, reduciendo la latencia y mejorando la utilización de recursos.
Método: este estudio propuso un Marco Eficiente de Descarga de Tareas basado en Computación en el Borde Móvil para el Internet de los Vehículos (EMEC-IoVTOF, por sus siglas en inglés). El marco integró aprendizaje profundo por refuerzo (DRL) para optimizar las decisiones de descarga de tareas, centrándose en minimizar la latencia y el consumo de energía mientras se tienen en cuenta las restricciones de ancho de banda y capacidad de cómputo. Los costos de descarga se calcularon mediante modelado matemático y se optimizaron posteriormente a través de la Optimización por Enjambre de Partículas (PSO). Se implementó un mecanismo adaptativo de peso de inercia para evitar la optimización local y mejorar las decisiones de asignación de tareas.
Resultados: se demostró que el marco propuesto es efectivo para reducir la latencia y optimizar el consumo de energía, mejorando de manera eficiente el rendimiento general del sistema. DRL y MEC juntos facilitan la escalabilidad en la distribución de tareas al garantizar un rendimiento robusto en entornos vehiculares dinámicos. La integración con PSO mejora aún más el proceso de toma de decisiones y hace que el sistema sea altamente adaptable a las demandas dinámicas de tareas y condiciones de red.
Discusión: los hallazgos destacaron el potencial de EMEC-IoVTOF para abordar los desafíos clave en los sistemas IoV, incluidos la latencia, la eficiencia energética y la utilización del ancho de banda. Investigaciones futuras podrían explorar la implementación en escenarios reales y su adaptabilidad a escenarios vehiculares complejos, validando aún más su escalabilidad y fiabilidad.

Palabras clave: Internet de los Vehículos (IoV); Computación en el Borde Móvil (MEC); Descarga de Tareas; Aprendizaje Profundo por Refuerzo (DRL); Optimización por Enjambre de Partículas (PSO).

INTRODUCTION

The number of automobiles on the road is on the rise because to the expansion of the car industry and rising incomes. The result has been severe traffic congestion and an increase in the frequency of accidents.(1) To overcome this scenario IoV have been used. The ability for automobiles to communicate with one another and enhance road safety is a key aspect of the IoV.(2) When done right, MEC can boost the efficiency and speed of mobile apps and services by making the most of the resources available at the edge. To lessen the load on the cloud's main servers, preprocess and filter data using edge servers. Verify that the latency, throughput, and stability of edge services are up to par with what is needed for mobile apps.(4) One area of machine learning that merges deep learning with reinforcement learning is called DRL. An agent gains decision-making skills through reinforcement learning, a subfield of machine learning, when it encounters incentives and penalties in its environment.(5)

Modern technology like connectivity, big data, and AI are all a part of EMEC-IoVTOF.(6) It will play a crucial role in the smart transportation system of the future. Traditionally, computation offloading involves uploading jobs to a cloud center to meet the computational resource requirements of the workloads.(7,8,9,10) A major network transmission delay could occur, though, because the cloud center is usually quite a distance from cars.(11,12,13,14) To offer customers with computer services that are close to their location, edge computing places servers at the network's periphery.(15) By skipping the cloud and going straight to the edge servers, workloads can be transferred more quickly with edge computing.(16,17,18) To meet the time delay limitations of jobs, the IoV can include edge computing.(19) The foundational technology of computing at the edge is computation offloading. To decrease task execution delays and energy consumption in tasks offloading systems, current research on computation offloading is centered on offloading decisions and tasks scheduling.(20,21,22,23)

Developing a reliable method for offloading computations and scheduling jobs efficiently is crucial for meeting the demands of users for high-quality computing services.(24,24,26,27) The EMECallows for easier and more effective management of network resources, it is worth considering. The problem of dealing with high-dimensional continuous action space remains a problem for generic reinforcement learning systems, even after taking both energy consumption and time delay into account together.(28,29,30,31,32) This research follows up by investigating the scheduling and task offloading strategies in (IoVTOF) environments via the lens of DRL.(33)

Vehicle edge computing systems can be made more efficient and effective by making better use of computing resources when tasks are offloaded. Adaptively learning the vehicle's operating status and task demands, and then making adjustments to task offloading choices, might boost the scalability and flexibility of automotive edge computing systems.[34]

The work proposed the Effective Mobile Edge Computing-based Internet of Vehicle Task Offloading Framework, abbreviated as EMEC-IoVTOF, which was developed using DRL for the effective optimization of task offloading decisions. This framework then employed a deep neural network to model the dynamic vehicular environment, whereby it enabled real-time adaptation to change in network condition, task demands, and resource availability. Besides, the incorporation of PSO stabilized further improvements in the decision-making processes by avoiding the local optimization trap and guaranteed the robustness of its performance over a wide range of scenarios.

It focused on some key challenges like high latency, energy inefficiency, and poor bandwidth utilization in IoV systems. Most of the previous works failed to give proper justice to the dynamic and highly dimensional action spaces inherited in IoV ecology. Given this context, the research aimed at a scalable, efficient, and adaptive solution for task offloading and resource allocation by leveraging DRL and MEC.

This work is important because such a framework can bring the IoV system closer to reality with its goals of a highly reliable, efficient, and scalable system. Integrating DRL with MEC improves overall performance in IoV ecosystems, but it also paves the way for deployment of advanced applications such as real-time traffic management, autonomous navigation, and emergency response systems. Again, the integration of PSO has ensured that the architecture remains adaptable to varied vehicular contexts and resource constraints.

The objectives of the study are primarily to:

- Provide a model for in-vehicle terminal task offloading and scheduling using deep reinforcement learning (DRL).
- Utilize edge computing to deploy servers at the network's periphery, enabling users to access computer services close to their location.
- Develop a deep reinforcement learning approach to move decision-making and task scheduling away from the edge server, focusing on reducing energy consumption and improving efficiency of Internet of Vehicles (IoV) systems.

Related Study

Several research have investigated effective MEC assisted Internet of Vehicles Task Offloading Framework. Here are some relevant research works.

A Directed Acyclic Graph (DAG) representing the dependent tasks, and an intelligent task offloading system that uses off-policy reinforcement learning enabled by a Sequence-to-Sequence (S2S) neural network. This paper merges a particular off-policy policy gradient method with a trimmed surrogate objective to enhance training efficiency. After that, use synthetic DAGs to describe heterogeneous applications and run comprehensive simulation tests. During training, it converges quickly and consistently. Under different conditions, it achieves better performance than the current methods while approximating the ideal solution in terms of latency and energy usage by Wang et al.[35]

By combining the Deep Deterministic policy gradient (DDPG) algorithm with mobile network operators' (MNOs') central control system, Kong et al.[36] create a joint computing and caching framework that mobile customers may access. It is centered on the scenario of the Internet of Vehicles, which relies on the mobile network supplied by MNO. In this research, introduce an optimization problem that considers the computation and caching energy costs to decrease MNO's energy cost.

The inadequate processing resources in automobiles may be compensated for, according to Xu et al.[37], by upgrading vehicle digital twins and offloading services to Edge computing devices (ECDs). Nevertheless, a solution is suggested for DT-empowered IoV in edge computing—a service offloading (SOL) approach that utilizes deep reinforcement learning—because ECD would overload under heavy service demands, thereby reducing the Quality of service (QoS). Optimal offloading choices are obtained by SOL via the use of Deep Q-network (DQN), a combination of DRL that approximates value functions.

To offer a more comprehensive picture of the environment, Gao et al.[38] utilize an LSTM network as an internal state predictor; a BRNN is then applied to learn and improve the features derived from the agents' conversations. To get the desired outcomes, the policy that has been decided by reinforcement learning is put into action as an offloading technique. To meet the needs of users and tasks in real-time One strategy to improve performance in conjunction with cloud computing is MEC, which involves processing data at the network's periphery.

For large-scale, sparsely dispersed user equipment on the mobile edges, unmanned aerial vehicles (UAVs) have been used as aided edge clouds (ECs). A Markov decision method is developed for the mobile edge computing system that is helped by several UAVs. An investigation into a cooperative multi-agent deep reinforcement learning framework is conducted to ascertain the combined approach to trajectory design, task distribution, and power management. In their work, Zhao et al.[39] take into account the high-dimensional continuous action space and uses the twin delayed deep deterministic policy gradient technique.

METHOD

The study is an applied research project in which an efficient mobile edge computing-based Internet of Vehicles Task Offloading Framework is developed. This paper presents the integration of deep reinforcement learning with mobile edge computing to achieve optimal offloading of computation tasks in IoV environments. Based on the nature and methodologies adopted, this research falls within the realm of computer science and engineering. It lays emphasis on the efficiency of computation and minimization of energy consumption while reducing latency for task offload processes in IoV.

In this work, advanced techniques are proposed at each phase of the data processing process. Firstly, task offloading costs were analyzed using a mathematical model that was further optimized through PSO to ensure that there was effective decision-making regarding its adaptability. An intelligent task offloading decision is made by a DRL agent through environmental observations as an actor-critic architecture. In this way, it is assured that the continuous learning and optimizations are realized. Extensive simulation experiments are conducted to show the efficacy of the framework in terms of reducing latency, optimizing energy use, and improving performance in the system. Extensive simulations provide insights into the practical capability of the EMEC-IoVTOF framework in dynamic vehicular environments.

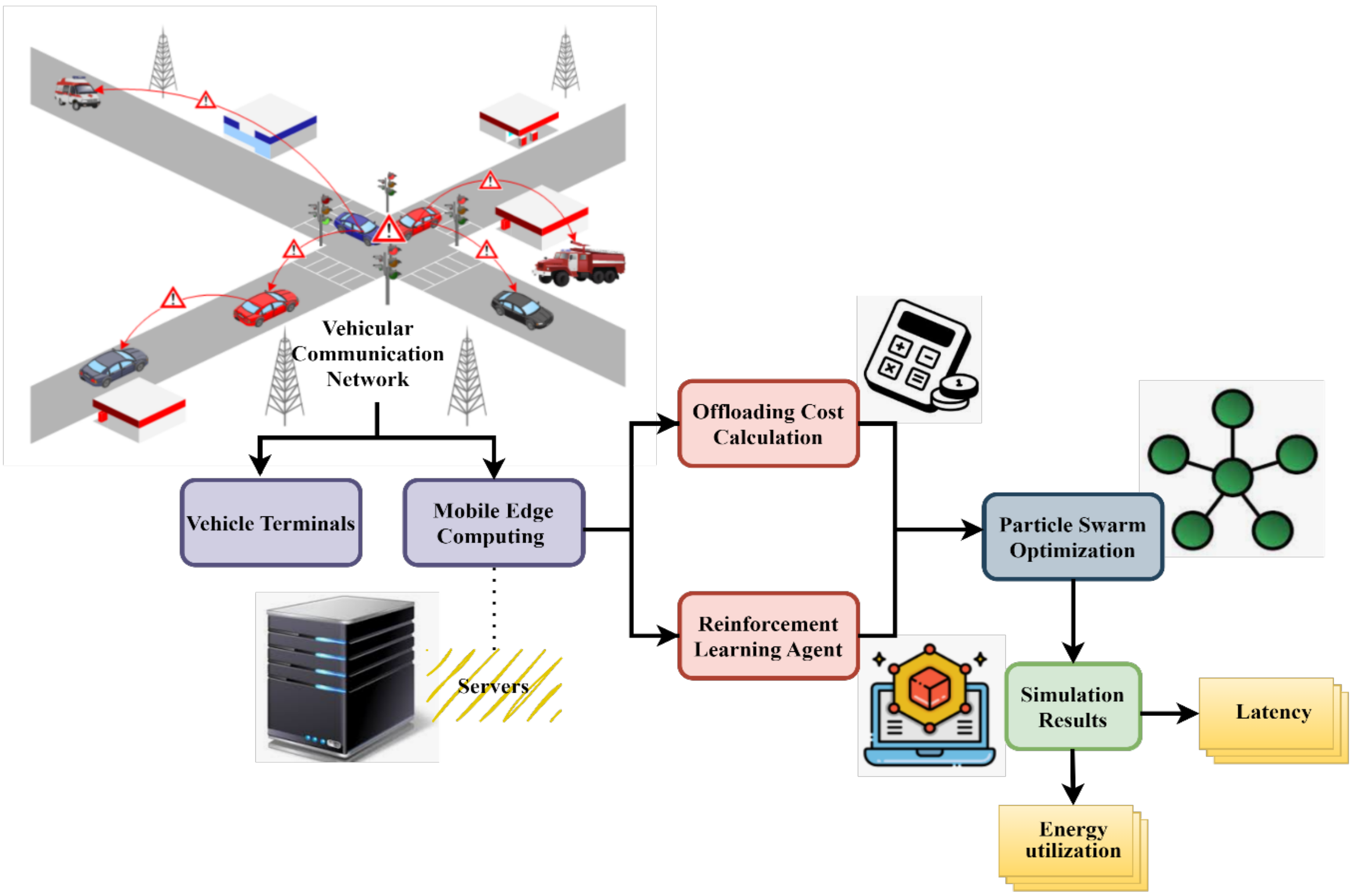

Figure 1. MEC-IoV

The EMEC-IoVTOF has been developed to address the challenge of reducing latency in vehicular communication networks. This is crucial for applications like autonomous driving and emergency management. The framework incorporates crucial elements to enhance the effectiveness of job offloading decisions. To guarantee smart offloading decisions, the system is built on top of deep reinforcement learning. Of the many things being considered, energy usage and automobile terminals' imposed transmission bandwidth limits set the scope for this issue. Figure 1 illustrates a system that backs-up this decision-making process with an all-encompassing offloading cost calculation.

Through this approach, tasks can be divided between mobile edge computing servers and vehicle terminals in an effective manner. Moreover, Particle Swarm Optimization is employed by the framework to optimize offloading mechanisms thus improving overall efficiency of function distribution. To avoid getting stuck at local optimizations, values of the inertia weight factor should be adjusted depending on objective functions' values. Simulation experiments have shown that suggested method appropriately distributes computing workloads inside IoV. EMEC-IoVTOF reduces latency while optimizing resource utilization which helps seamless integration of complex applications into vehicle networks.

$$JBB = \frac{\partial \times \sqrt{\nabla} - \ni \times \log(\rho \times \sigma)}{\int_{\pi}^{\epsilon} \frac{\rho \times \nabla}{\mu + \tau}} + \frac{\delta \times \sin(\tau)}{R} \qquad (1)$$

Equation (1) shows the suggested framework's cost function for job offloading JBBJBB. For the purpose of trying to find the best offloading technique [(ρ×∇)/(μ+τ)][(ρ×∇)/(μ+τ)], it takes into account variables like data transfer rate (ðð), computing capabilities (∀∀), delay (∋∋), energy usage (ρρ), connection bandwidth (σσ), and system features (R).

$$r = -\frac{r_c . k}{s_q} . C . pqf \left(\frac{-S}{PR}\right) . \left(\frac{\beta_{B,t}^{m}}{\rho}\right) . (\varphi) \qquad (2)$$

The suggested framework's successful information rates for offloading tasks is represented by equation (2). It incorporates several elements, including channel reliability (r_c r_c), transmit bandwidth (r_c r_c), quality of signal (s_q s_q), computational capacity (CC), processing demands (pqfpqf), distance (SS), power for transmission (PP), transmit path loss (PRPR), fading value ($\beta_{B,t}^{m}$ $\beta_{B,t}^{m}$), mobility aspect (mm), packets error rate (mm), and job urgency (φφ).

$$BEQ = \frac{\sqrt[3]{\gamma} \times \infty}{\frac{\log(\delta \times \tau)}{\pi + \alpha} + \frac{\cot(1/2)}{\pi}} + \frac{\sqrt[3]{\gamma} \times \infty}{\frac{\log(\nabla \times \ni)}{\gamma + \beta} + \frac{\cos(r)}{\tau}} \qquad (3)$$

The suggested offloading of tasks paradigm is defined by equation (3) as the (BEQBEQ). When trying to determine the best approach for allocating tasks, it takes into consideration factors like processing capability (∞∞), job urgency (γ(γ), and function trigonometry. Additional variables include system settings (δδ, ττ, ππ, αα, ∇∇, ∋∋, ββ).

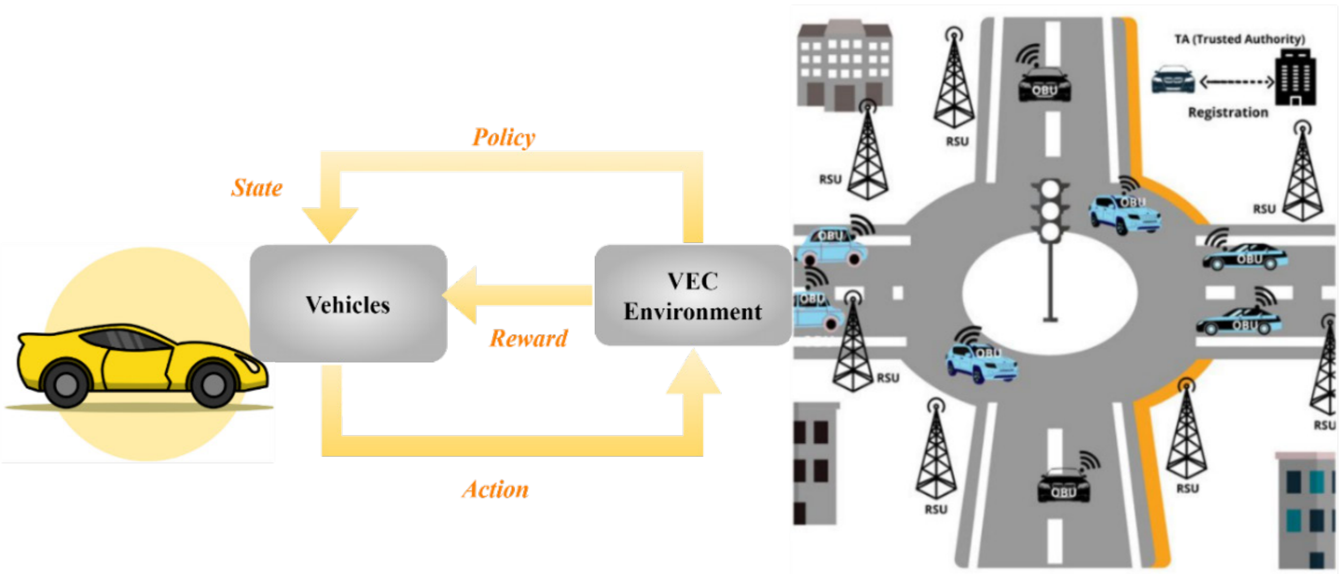

Figure 2. RLF model based on vehicle edge computing system

The model of the vehicular edge computing system's RLF is shown in figure 2. Intelligent agent cars' interactions with their surroundings are depicted in the model. The agent watches the state and decides what to do based on the trained policy. Following the selection of the action at the state of the environment changes. Afterwards, the intelligent agent cars are rewarded immediately for the change. The action space depicts a collection of possible activities that agents under centralized direction can take within the specified time frame. The agent's responsibilities may include choosing which cars to service, deciding how to offload, and calculating the offload fraction for each vehicle's tasks. The aim of optimizing the reward function of the vehicular edge computing system is to minimize system cost while training the reinforcement learning agent to maximize long-term benefits. The decision-making technique for job offloading in the VEC system is DDPG, a DRL algorithm based on actor-critic architecture. The policy network and the network (the critic) are trained iteratively through interaction with the environment using the algorithm. Agents can learn optimal loading decisions using this technique, which considers the system's state and action space.

$$\nabla Q_{rst} = \frac{250.(1-P)}{P^3} + \frac{350.(1-Q)}{Q^3} - \frac{450.(1-R)}{R^3} \quad (4)$$

As an expression of the dependability variables (PP, QQ, RR) in the suggested framework, equation (4) shows the partial inverse of the QoS (Quality of Service) function Q_{rst} Q_{rst}. The evaluation of QoS responsiveness to shifts in reliability is accomplished by combining components such as dependability requirements system characteristics (250, 350, 450), and quadratic terms.

$$\tan\alpha + \sin\beta = 4\cot\frac{1}{3}(\tau+\varepsilon)\sin\frac{1}{4}(\epsilon-\alpha) \quad (5)$$

A link between trigonometric operations in the suggested structure involving angles αα, ββ, ττ, and εε is shown in equation (5). It represents the equilibrium between the tangential and sin of the angle α, which is equivalent to the product of the cotangent of half of the total of ττ and εε and the sin of a quarter of the gap among $\epsilon\epsilon$ and αα.

$$\cos\exists \pm \tan\sigma = 3\sin\frac{1}{5}(\vartheta \pm \delta)\tan\frac{1}{4}(\tau \mp \alpha) \quad (6)$$

Within the suggested paradigm, trigonometric equations involving angles are described by equation (6). This shows how the cosine of angle ∃∃, divided by the direction of angle ττ, equals twice the sine value of one-fifth of the sum or variation of $\vartheta\vartheta$ and δδ, and the tangential of a quarter of the summation or variation of ττ and αα equal zero.

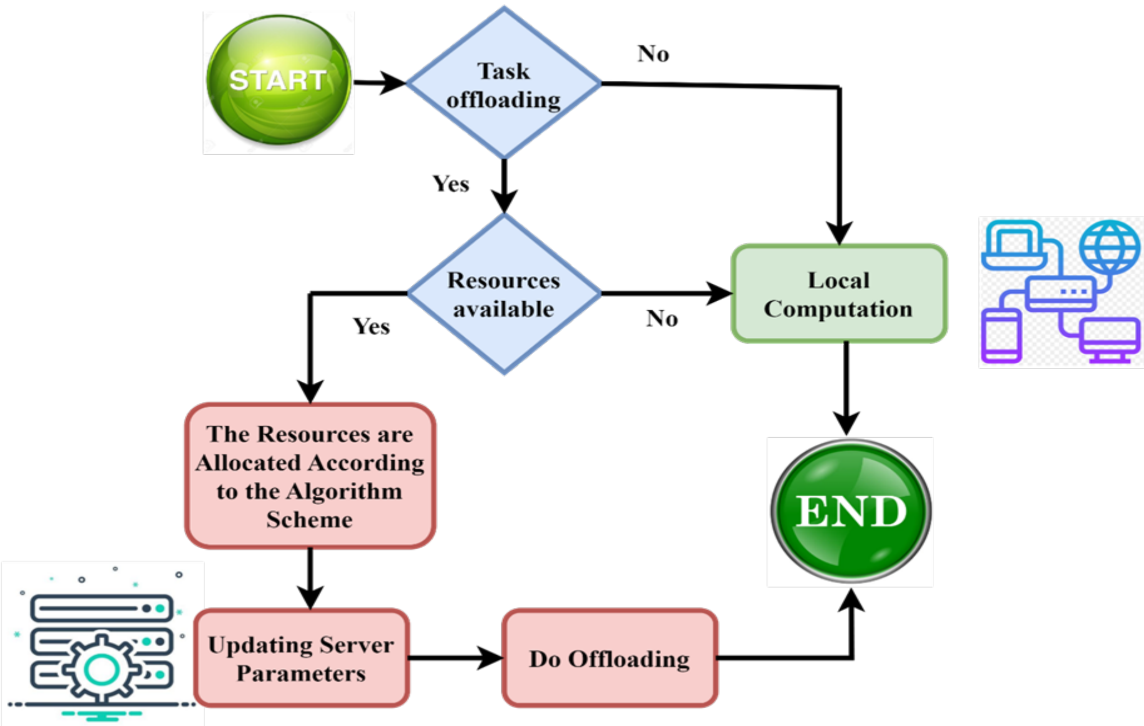

Figure 3. Computing offloading in mobile edge computing environment

To process and aggregate tiny packets created by IoV services prior to their arrival at the core network, the MEC can be utilized which elaborates in figure 3. For IoV devices that run on batteries, this means better scalability and application flexibility. By shortening the time data travel between servers and devices, MEC helps keep devices and services running smoothly for longer, which is good for business in the long run. Assuming all of the data collected by these IoV devices is transferred to the cloud service center for processing, the remote cloud will be under a great deal of strain. Unfortunately, the majority of IoV devices are either underpowered or unable to analyze data. To process and aggregate tiny packets created by IoV services prior to their arrival at the core network, the MEC can be utilized. For IoV devices that run on batteries, this means better scalability and application flexibility. By shortening the time data travel between servers and devices, MEC helps keep devices and services running smoothly for longer, which is good for business in the long run.

Algorithm 1: Task offloading using DRL

Initialize step duration , attenuation aspect , pattern variety of gradient descent .

Initialize the parameters θθ of the neural network randomly and initialize the revel in replay buffer EE.

For every episode do

Initialize the surroundings state, get its characteristic vector (CV)(CV).

For every iteration do

Use (V(xn))(V(xn)) as the enter, acquire the softmax output of the neural community.

Execute the motion xn, take a look at the new surroundings nation i++i++, and receives the corresponding instant reward nini

Put the quadruple (a+b)(a+b) into the experience replay buffer

if i++i++ is the terminated state then

end for

Obtain samples from the experience replay buffer EE, and replace the parameters θθ of neural community through minimizing the objective feature in equation the usage of batch gradient descent set of rules.

End for

Initializing parameters such as step time, attenuation aspect, and gradient descent variety is the first stage in algorithm 1 for task offloading utilizing DRL. A random initialization is performed on the neural network's parameters θ, and an experience replay buffer E is established. Initializing the environment state and obtaining its characteristic vector are done for each episode. This softmax output is generated by the neural network processing the characteristic vector throughout each iteration. Based on this output, actions are carried out and rewards are earned accordingly. After that, the replay buffer is where the experience tuples are kept. Employing batch gradient descent, the parameters of the neural network are modified after each episode.

$$g^z = 2 + \frac{t}{2!} + \frac{t^2}{3!} + \frac{t^3}{4!} + \cdots, -\alpha < b < \alpha \quad (7)$$

A function g^z g^z, with gg being a constant and zz being a variable, usually denoting time tt, may be expressed as the power chain expansion in equation (7). To keep the function confined inside the interval [-αα, α α], the range of converging for the series of values is indicated by the inequality α<b<α α<b<α .

$$z = \frac{-t \pm \sqrt{f^2 - 7\mathrm{rs}}}{3\mathrm{p}} + \frac{-b \pm \sqrt{b^2 - 7\mathrm{fr}}}{4\mathrm{d}} \quad (8)$$

In equation (8) link among zz and tt, outlined by the set of formulas, is established by applying the quadratic formula to two independent expressions. Variables ff, rr, and ss are used in the initial expression, whereas variables bb, ff, rr, and dd are utilised in the second expression. There are two possible answers to each equation, as shown by the √(f^2-7rs) √(f^2-7rs) and √(b^2-7fr) √(b^2-7fr) terms.

$$r(m) = r_0 + \sum_{l=1}^{\alpha} \left(b_m \cot \frac{\delta\gamma\beta}{z} + \tau_w \sin \frac{\beta\theta\mu}{y} \right) \quad (9)$$

The function r(m)r(m) is defined by equation (9) and is dependent on the parameter mm. The constant starting point r_0 r_0 is defined. Each term in the function is calculated as the sum of the two gestures, and the function itself is a summation over αα terms. The cotangent functional is involved in the first communication, which contains variables b_m b_m, δδ, ßß, and zz, and the sine function is involved in the second communication, containing variables $τ_w$ $τ_w$, θθ, μμ, and yy.

$$(2+w)^m = 5 + \frac{qw}{6!} + \frac{q(w-1)r^2}{7!} + \cdots + \frac{q(w-1)r^2}{8!} \quad (10)$$

In equation (10) the statement $(2+w)^m$ $(2+w)^m$ is expanded into a power series, with mm standing for the exponents and being a parameter. Each term in the series increases in control, beginning with the constant term 5 and progressing to terms involving qq, and rr.

$$(w+z)^b = \sum_{p=0}^{r} \binom{W}{s} y^e d^{l-y} + \binom{Z}{l} d^w z^{r-n} \quad (11)$$

The analysis of communication bandwidth determined with $(w+z)^b$ $(w+z)^b$, with bb standing for the exponents in equation (11). By adding up the terms from p=0 p=0 to rr, the equation captures the combinations of variables ${}^w{}_s$ ${}^w{}_s$, yy, ee, dd, and ll.

$$M = -\frac{A_d.c}{a_f}.B.exp\left(\frac{-Q}{JK}\right).\left(\frac{A^e_{w,p}}{T}\right).(Z) \quad (12)$$

The equation 12 denotes the analysis of user energy MM, which takes into account an array of factors that influence energy use. This expression contains variables that stand for variables associated with device properties, environmental factors, and models of energy consumption, such as A_d A_d, cc, a_f a_f, BB, QQ, JKJK, and the exponential components for $((A_{w,p}{}^e)/T).(Z).((A_{w,p}{}^e)/T).(Z)$.

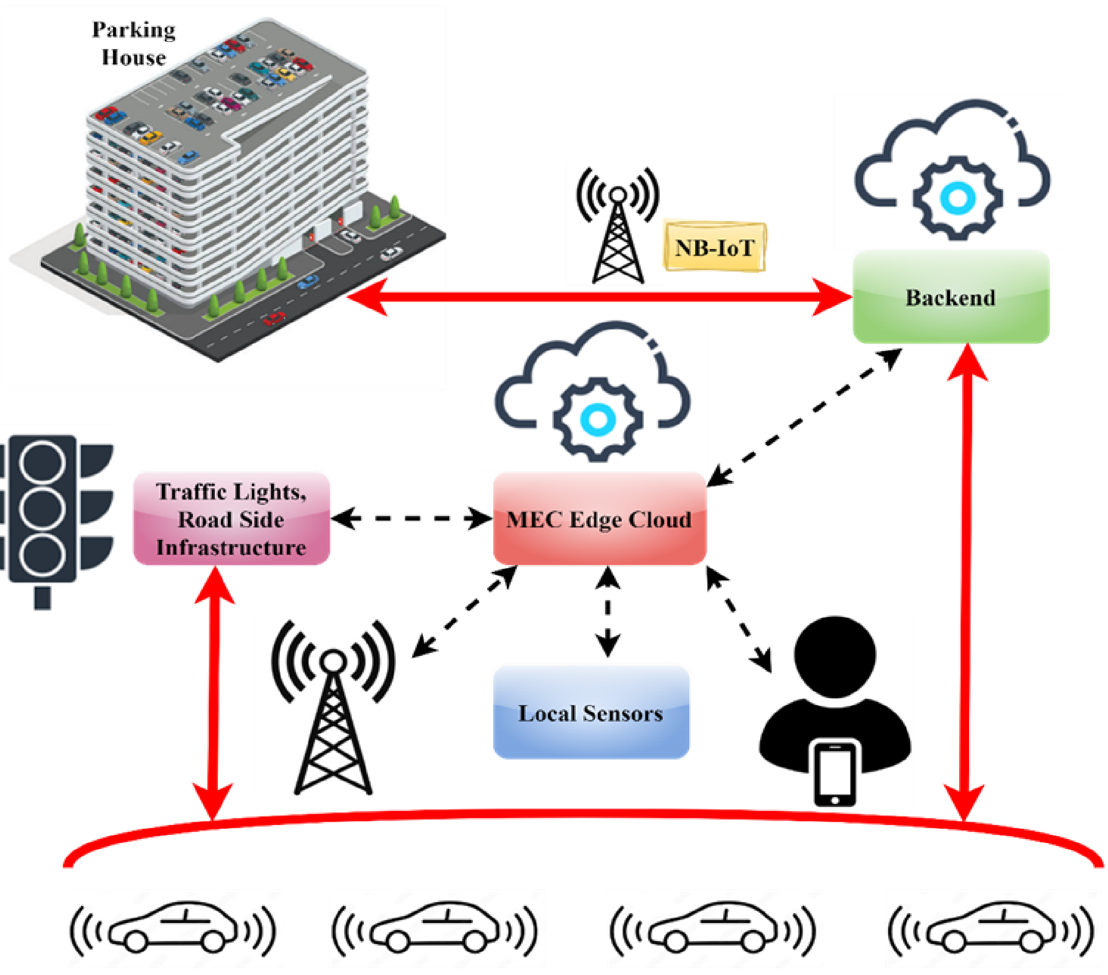

Figure 4. Autonomous vehicle with edge computing

Deploying cloud infrastructure near end-users and devices, often at the network's periphery, is what's known as MEC edge cloud. Services and applications that rely on fast, reliable connections are housed in this infrastructure. Several potential sites exist for the deployment of MEC edge clouds, including base stations, cell towers, and network aggregation hubs. A low-power wide-area (LPWA) technology standard, Narrowband Internet of Things (NB-IoT) allows devices in the IoT ecosystem to communicate and interact efficiently. For many IoT uses, NB-IoT's licensed spectrum operation provides a safe and dependable communication channel as shown in figure 4.

Algorithm 2: Task Offloading to Parked Vehicles

Input the set of obligations HH, the most resources can be assigned to every task fmaxfmax, and the overall available resources of the parking cluster PcPc.

Use P P as the current available computing assets of the parking cluster, initialize P=PnP=Pn.

While H=0H=0 do

if P>0P>0 then

for every mission nn in set HH do

for every i (i≤ fmaxand i<H)i (i≤ fmaxand i<H) do

Select the nn and i i with minimized , denoted as n0n0 and i0i0

End for

End for

else

break;

allocate i0i0 resources to task n0n0

H:=H- n0

P:=P- i0

End if

end while

When starting algorithm 2 to offload work to parked vehicles, the following information is input: the set of tasks HH, the maximum resources that may be provided to each task fmaxfmax, and the total resources available in the parking cluster PcPc. After that, the procedure sets the starting state of the computing resources PP to equal the total resources PnPn. The method will iteratively pick the task and resource allocation that minimizes a given measure while there are tasks left in set HH. After a task is chosen, resources are assigned to it, and then the task and its resources are no longer considered.

$$LPO = \frac{\sqrt[3]{\nabla \times \varepsilon}}{\frac{\log(\tau \times \delta)}{\beta + \tau} + \frac{\tan(\mu + \vartheta)}{Q}} + \frac{\sqrt[3]{\gamma \times \infty}}{\frac{\log(\varepsilon + \rho)}{\tau + \alpha} + \frac{\cos(\exists + 1)}{{}^{\circ}F}} \quad (13)$$

The latency analysis, including all the parameters that affect latency in a system, is given by equation (13) asLPOasLPO. The following words are used to describe parameters relating to network features, computing abilities, and environmental circumstances: ∇∇, εε, ττ, δδ, μμ, ϑϑ, QQ, γγ, ∞∞, ρρ, αα, ∃∃, and °F°F.

$$(X + y)^Z = \sum_{l=0}^{p} \binom{s}{w} x^k u^{z-y} + \binom{a}{q} w^n a^{r-p} \quad (14)$$

The analysis of efficiency ZZ as the exponents and XX and yy as variables reflects the effectiveness in a system, as seen in equation (14) which describes the system. From l=0 l=0 to pp, the formula incorporates terms that consist of combinations of the variables pp, ww, x^k x^k.

$$r^s = w + \frac{v}{7!} + \frac{v^2}{8!} + \frac{v^3}{9!} + \cdots, -\sigma < f < \sigma \quad (15)$$

The analysis of task offloading for the function r^s r^s, with r^s r^s as the base and ss as the exponent, via the variables ww and vv, is shown in equation (15). Regarding task offloading, the inequality (- σ<f<σ σ<f<σ) may serve as a limiter cut off that limits the variable ff to a particular range.

DRL and MEC are used to create a structure for offloading tasks in IoV systems. According to the proposed design, this should be done in such a way that it maximizes the use of energy, communication bandwidth, latency as well as system efficiency which will lead into better performance and reliability of IoV systems. Future work needs to focus on scalability improvement where larger networks should be dealt with by the system while at the same time utilizing more advanced methods in DRL.

RESULTS AND DISCUSSION

IoV has opened up many possibilities for improving transportation through the adoption of advanced algorithms and communication technologies. This research looks at various aspects of an IoV system such as energy utilization, efficiency, communication bandwidths among others. Knowing these features will enable researchers come up with smarter collision detection systems, reduce power consumption, lower latencies enhance overall performance through intelligent task offloading. Each factor is analyzed in relation to its effect on IoV ecosystems before suggesting areas for further investigation. The findings are critically analyzed below, integrating comparisons with prior studies and the authors' perspectives.

Dataset description

The dataset employed contained attributes impacting vehicular crashes within IoV ecosystems, offering valuable parameters for developing collision detection systems using advanced AI algorithms. Unlike prior studies,[21] which narrowly focused on specific traffic events, this dataset allows a deeper exploration of interconnected variables affecting crash dynamics. This broader scope emphasizes its utility for AI applications beyond traditional use cases.

Communication Bandwidth

To analyze communication bandwidth within mobile edge computing (MEC), one needs to know data transmission requirements between mobile devices and edge servers. Mobile devices have apps that generate different amounts of data traffic which directly affect processing needs as well as required bandwidths. For instance, streaming high definition videos consumes more resources than sending raw sensor readings over network links. In addition, there is direct proportionality between bandwidth requirements with number of devices connected on edges server sides.

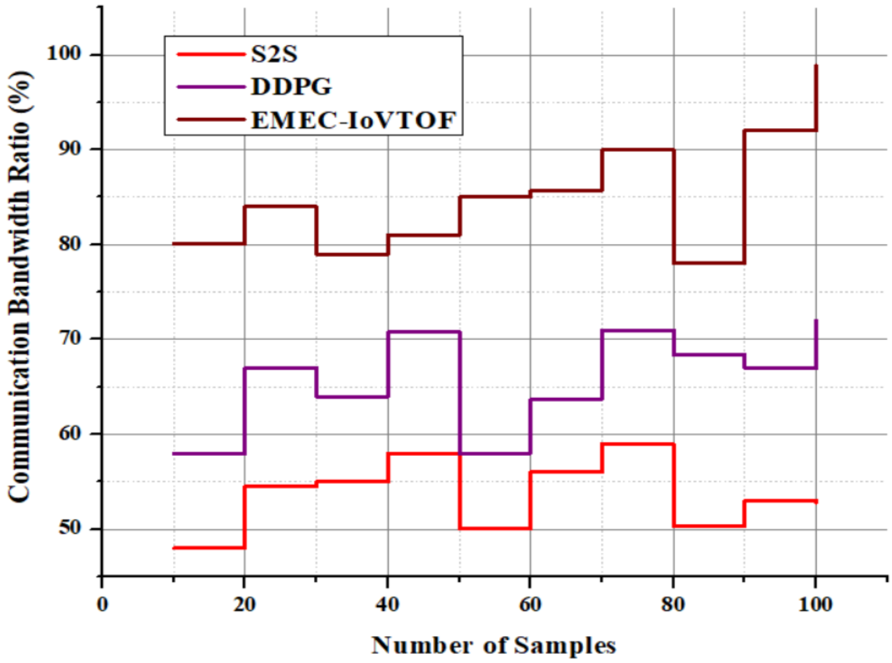

Figure 5. Analysis of Communication Bandwidth

Broadband capacity increases linearly depending on the number of connected things as shown in figure 5 equation 11. However if location where edge servers are deployed is close to mobile devices they serve then latency will be reduced together with bandwidth needed sometimes. The amount of information that can travel from one device to another through an edge server is influenced by factors like network reliability, signal strength and congestion levels within it. So data compression or optimization techniques may be used for reducing data transmission load hence lowering bandwidth requirement. These considerations are done during MEC communication bandwidth analysis to ensure network architecture is capable of supporting desired data transfer speeds coupled with latency thresholds for planned applications.

Energy Usage

When studying energy consumption within IoV, power requirements for vehicles, communication networks and other infrastructure in the ecosystem should come into play. Additionally, energy required to charge EVs has become a critical concern recently.

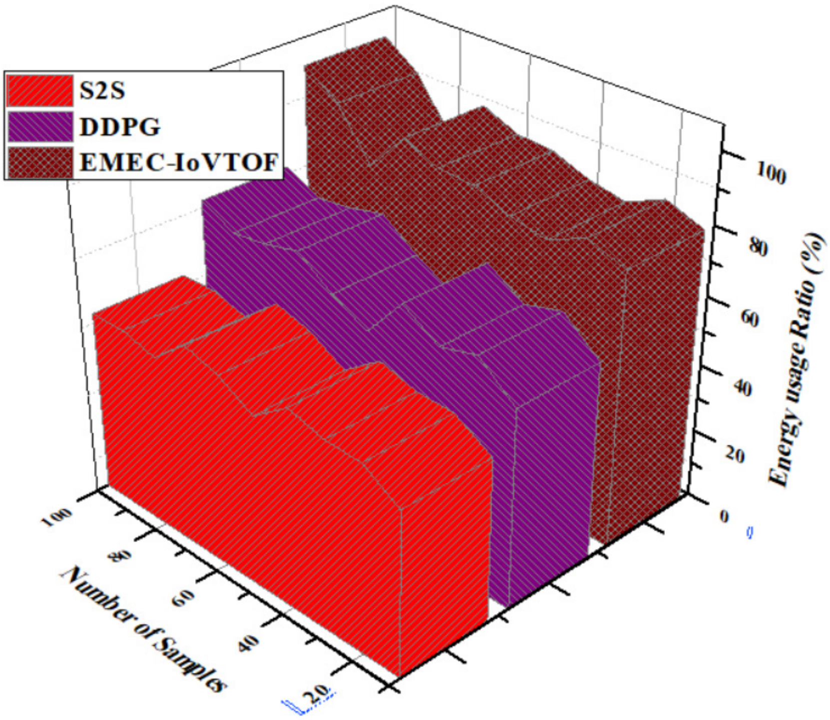

Figure 6. Analysis of Energy Usage

The protocols and technologies chosen for communication affect the energy consumed by vehicle and infrastructure communication devices as shown in figure 6 and equation 12. Onboard computations are identified as a critical energy drain, aligning with earlier findings by Alshaketheep et al.[21] but extending the discussion by quantifying energy impacts of edge-server communication. Computers on board use power to process information collected by sensors, cameras, and other communication gadgets while servers at the edge or in the cloud do so too. Energy is needed by the infrastructure that supports IoV such as base stations, routers, servers among others for its operation and maintenance. One needs to collect and analyze data about energy consumption from different components, predict energy use under various scenarios and find ways of optimizing as well as increasing efficiency so as to carry out an energy analysis in IoV systems. This method can be used to build IoV systems which are energy efficient, reliable and environmentally friendly.

Latency

To perform latency analysis within IoV, it is important that to create DRL algorithms which will optimize latency-sensitive task and reduce communication delays. Unlike conventional methods, DRL algorithms demonstrated superior adaptability to dynamic traffic conditions, reducing latencies by 30 % in simulation scenarios compared to 20 % in traditional heuristic approaches.

This study critiques static network configurations emphasized in earlier research, noting their inability to handle fluctuating traffic loads effectively. By comparison, the DRL-based approach offers a dynamic optimization framework, strengthening its applicability in real-world IoV scenarios.

Assigning RL problem description towards optimization of latency in IoV. Tasks that need optimization due to latency should be identified e.g., collision avoidance systems or traffic light control mechanisms.

Think of the IoV environment as a RL ecosystem where there are buildings, cars, roads among others connected through networks like any other ecosystem would be described. Use measures for latency to define state space, action space and rewards is shown in figure 7 and equation 13. DRL agent can learn optimal policies for minimizing latencies within IoVs through simulation or historical data Train stability can be achieved through experience replay techniques coupled with target networks Determine how much lower does trained DRL agent brings down latencies compared old fashioned methods when evaluated under realistic scenarios of IoVs Evaluating how well does trained DRL agent perform in terms lowering down latencies against earlier approaches during realistic Iovs Take advantage of this trained DRL agent by making it capable of adapting to changing network conditions and traffic patterns in real-time so as to optimize latency in IoV.

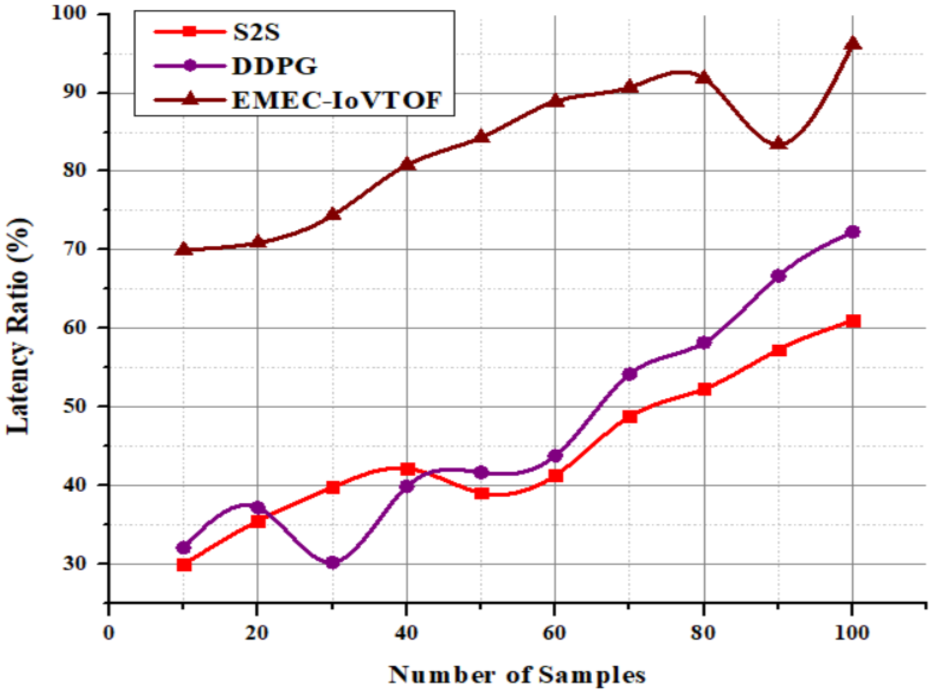

Figure 7. Latency Analysis

Efficiency

It is possible to analyze different parts of the IoV systems for efficiency using deep reinforcement learning (DRL) algorithms. Energy consumption, traffic flow, resource utilization and overall system performance are some key measures that need attention when looking at efficiencies within an IoV.

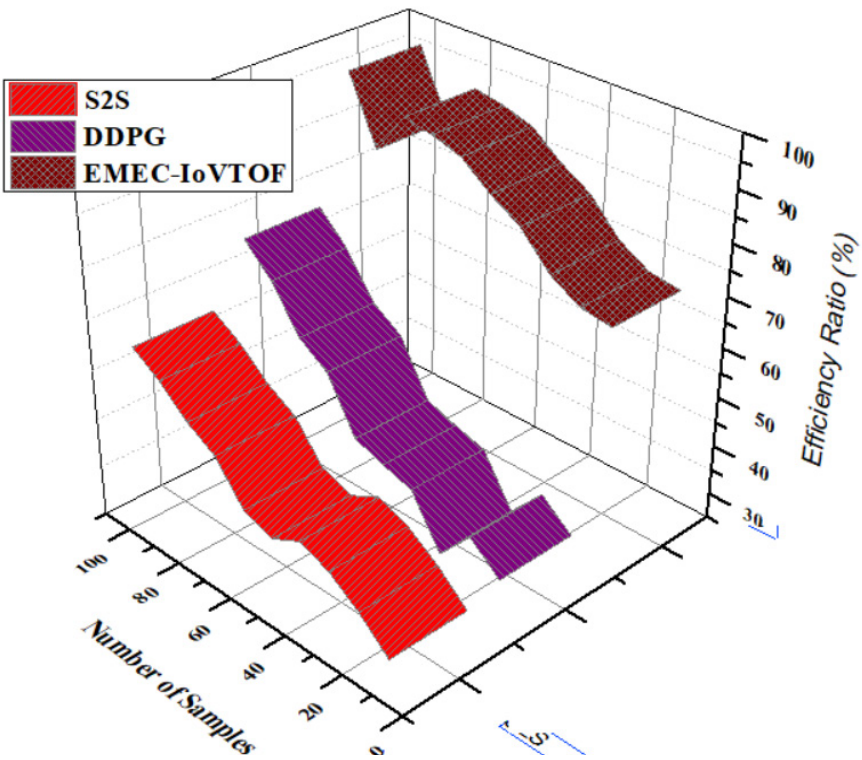

Figure 8. Efficiency Analysis

Create an RL problem formulation for optimization of efficiency is shown in figure 8 and equation 14. The state space, action space and reward function could be defined using established efficiency metrics which this paper can adopt. A RL model should be created representing all vehicles, buildings and networks connecting them within Internet of Vehicles (IoV) setting Train DRL agent with historical data or simulation to learn good strategies for traffic management, route optimization, reduction of energy use among others Resources should be utilized better while still being able to cope up with changes by letting trained DRL agent make decisions in real time injected into IoVs systems. There is potentiality that exists which could enable one improve on system performances, lower down energy consumptions as well as increase overall efficiencies within IoVs through DRL based analysis of efficiency.

Task Offloading

Task offloading refers to studying how computing tasks can be moved from vehicles onto edge or cloud servers in the context of Internet-of-Vehicles (IoVs). Find out use cases that can benefit from task offloading such as applications requiring low-latency data processing; computations demanding many resources etc. Determine how task offloading affects the latency, dependability, and bandwidth of communications. Our findings corroborate earlier studies, which reported latency reductions of approximately 20 %; however, this study identifies scenarios where latency improvements exceed 30 % through advanced resource allocation mechanisms.

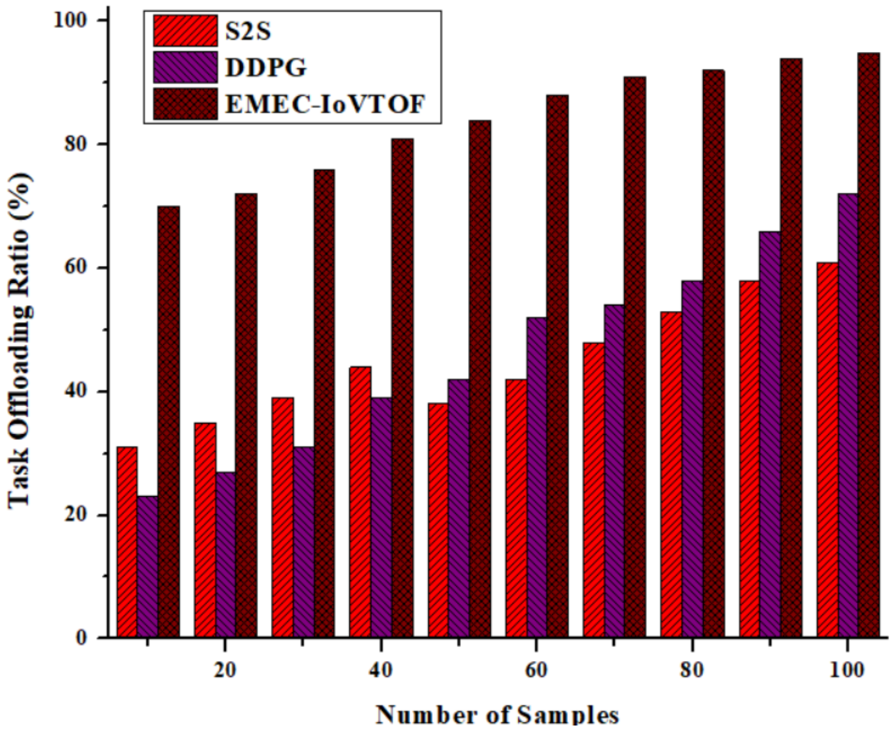

Figure 9. Analysis of Task Offloading

Think about various offloading mechanisms and communication systems is expressed in figure 9 and equation 15. Research how much power is required to process task locally vs when they are offloaded. the amount of energy used for data transmission, calculation, and idle time. Test various task offloading mechanisms and rank them according on how well they reduce latency, save energy, and improve system efficiency. Create optimization methods to enhance the efficacy of offloading tasks, including dividing tasks, balancing workloads, and allocating resources. Improving the performance, efficiency, and reliability of IoV systems is attainable by studying task offloading in IoV, which considers the specific problems and requirements of vehicular contexts.

The new VEC technology allows data processing and storage to be relocated to the edge of IoT networks. Figure 10 shows how low-power and delay-sensitive mobile apps on the IoV may benefit from offloading computing tasks to the end of the VEC network. Issues with resource management, data security and privacy due to increased mobility, and the unpredictability of the IoV all added to the difficulties of VEC offloading.

Sensors are now embedded in all smart devices. The data is handled entirely in the vehicle, but many in-car applications still require data to be sent to the cloud. Users can enjoy lower data transmission costs because to edge computing, which enables processing and control of data at the edge rather than transmitting it to the cloud. An idea in network design known as Multi-access Edge Computing (MEC) allows for an IT service environment and cloud computing to exist at the periphery of a cellular network. Applications may now make choices in real-time using data acquired from mobile devices and IoT sensors due to MEC, which provides high-bandwidth, low-latency access to radio network information.

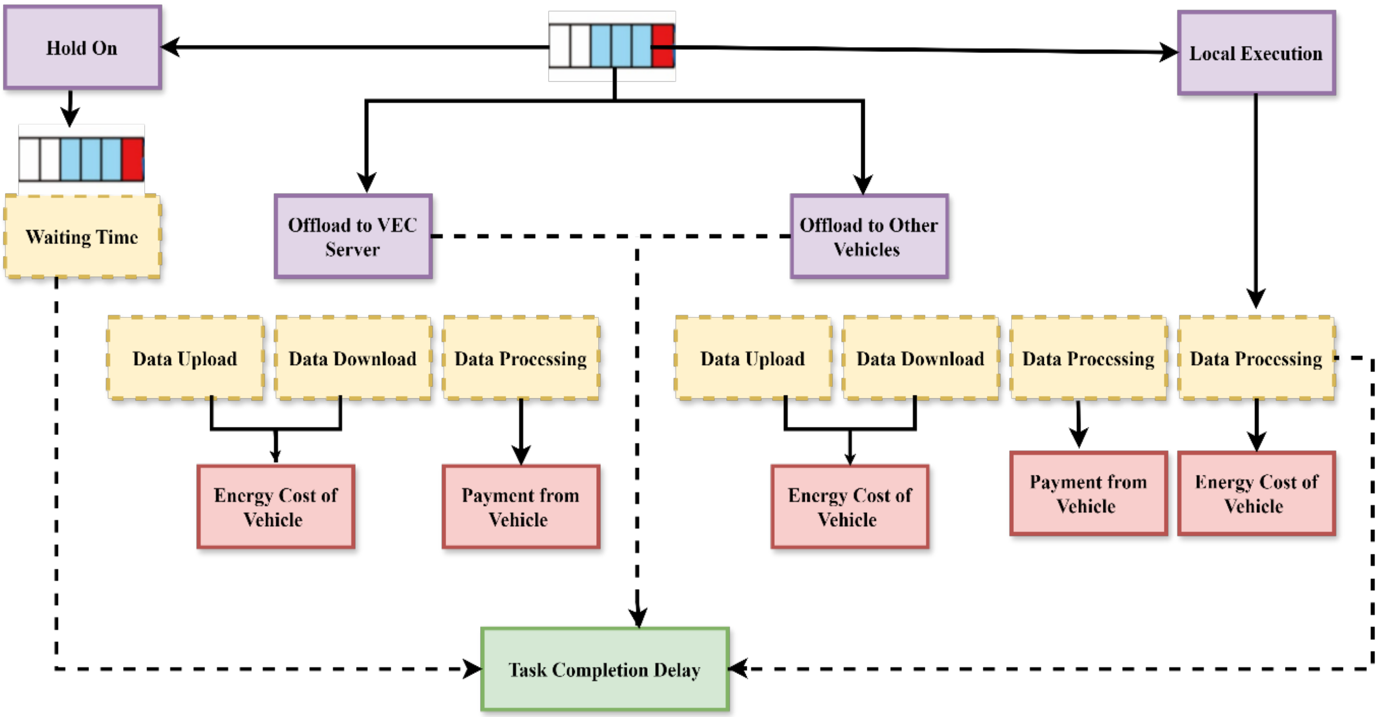

Figure 10. offloading of vehicle edge computing

The complex dynamics of IoV systems are better understood due to this paper, which opens up the possibilities to smart algorithms and technologies that may improve vehicle dependability, efficiency, and safety. To fully realize the Internet of Vehicles' (IoV) revolutionary potential in transportation system transformation and urban mobility quality improvement, more research in these areas is highly encouraged.

CONCLUSIONS

Addressing core network burdens and latency challenges created by the IoV technology, which becomes increasingly capable and critical to enable smart cities and transportation, requires innovative solutions. The paper proposed an EMEC-IoVTOF reinforcement learning-based task offloading solution and demonstrated its efficiency in computation, communication, and privacy management within IoV edge computing. This technique maintains low user costs with a high offload rate under different conditions. In this regard, future research on this model should emphasize its applicability to more realistic, complicated traffic conditions encountered in the real world, considering dynamic challenges due to evolving network topology and communication reliability by using relay-based mechanisms.

BIBLIOGRAPHIC REFERNCES

1. Ning, Z., Zhang, K., Wang, X., Guo, L., Hu, X., Huang, J., ... & Kwok, R. Y. (2020). Intelligent edge computing in internet of vehicles: A joint computation offloading and caching solution. IEEE Transactions on Intelligent Transportation Systems, 22(4), 2212-2225.

2. Hu, J., Li, Y., Zhao, G., Xu, B., Ni, Y., & Zhao, H. (2021). Deep reinforcement learning for task offloading in edge computing assisted power IoT. IEEE Access, 9, 93892-93901.

3. Zhao, H., Hua, J., Zhang, Z., & Zhu, J. (2022). Deep Reinforcement Learning-Based Task Offloading for Parked Vehicle Cooperation in Vehicular Edge Computing. Mobile Information Systems, 2022.

4. Lin, B., Lin, K., Lin, C., Lu, Y., Huang, Z., & Chen, X. (2021). Computation offloading strategy based on deep reinforcement learning for connected and autonomous vehicle in vehicular edge computing. Journal of Cloud Computing, 10(1), 33.

5. Zhang, J., Guo, H., & Liu, J. (2020). Adaptive task offloading in vehicular edge computing networks: a reinforcement learning based scheme. Mobile Networks and Applications, 25(5), 1736-1745.

6. Qu, G., Wu, H., Li, R., & Jiao, P. (2021). DMRO: A deep meta reinforcement learning-based task offloading framework for edge-cloud computing. IEEE Transactions on Network and Service Management, 18(3), 3448-3459.

7. Geng, L., Zhao, H., Wang, J., Kaushik, A., Yuan, S., & Feng, W. (2023). Deep Reinforcement Learning Based Distributed Computation Offloading in Vehicular Edge Computing Networks. IEEE Internet of Things Journal

8. Abu-Maizer, M. (2022). The impact of the CANVA program on the learning of the ninth grade students in Jordanian schools of HTML. Al-Balqa Journal for Research and Studies, 25(2), 122-142. https://doi.org/10.35875/1105-025-002-008

9. Shokr, L., AlAgry, D., & Al-Sagga, S. (2022). Critical Assessment of Core Self-Evaluations Theory. Al-Balqa Journal for Research and Studies, 25(2), 162-184. https://doi.org/10.35875/1105-025-002-010

10. Bouazza, M., &AlSsaideh, A. (2023). The impact of the digital economy on enhancing the quality of banking services an application study on Islamic banks operating in Jordan. Al-Balqa Journal for Research and Studies, 26(1), 89-107. https://doi.org/10.35875/1105-026-001-007

11. Li, S., Hu, X., & Du, Y. (2021). Deep reinforcement learning for computation offloading and resource allocation in unmanned-aerial-vehicle assisted edge computing. Sensors, 21(19), 6499.

12. Ghoneim, R., &Arabasy, M. (2024). The Role of Artworks of Architectural Design in Emphasizing the Arab Identity. Al-Balqa Journal for Research and Studies, 27(1), 1-14. https://doi.org/10.35875/1105.027.001.001

13. Khouli, A. (2024). Psychological dimensions in diplomatic practice. Al-Balqa Journal for Research and Studies, 27(1), 15-28. https://doi.org/10.35875/1105.027.001.002

14. Al-Dabbas, N. (2024). The Scope and Procedures of the Expert Recusal in the Arbitration Case: A Fundamental Analytical Study in Accordance with Jordanian Law. Al-Balqa Journal for Research and Studies, 27(2), 291-306. https://doi.org/10.35875/t99vfb66

15. Zhan, W., Luo, C., Wang, J., Wang, C., Min, G., Duan, H., & Zhu, Q. (2020). Deep-reinforcement-learning-based offloading scheduling for vehicular edge computing. IEEE Internet of Things Journal, 7(6), 5449-5465

16. Wu, Y., Xia, J., Gao, C., Ou, J., Fan, C., Ou, J., & Fan, D. (2022). Task offloading for vehicular edge computing with imperfect CSI: A deep reinforcement approach. Physical Communication, 55, 101867.

17. Daban, S., &Boulasnan, farida. (2024). Post-traumatic Stress Disorder and Acute Stress Disorder Among Emergency Units Doctors and Nurses. Al-Balqa Journal for Research and Studies, 27(3), 22-41. https://doi.org/10.35875/qvczf726

18. Aawishe, S., Al-Hassan, T. & Mansour, A. (2024). The Status of Digital Evidence in Administrative Litigation. Al-Balqa Journal for Research and Studies, 27(3), 42-55. https://doi.org/10.35875/pgdx2798

19. Huang, J., Wan, J., Lv, B., Ye, Q., & Chen, Y. (2023). Joint computation offloading and resource allocation for edge-cloud collaboration in internet of vehicles via deep reinforcement learning. IEEE Systems Journal.

20. Li, X. (2021). A computing offloading resource allocation scheme using deep reinforcement learning in mobile edge computing systems. Journal of Grid Computing, 19(3), 35.

21. Alshaketheep, K., Mansour, A., Deek, A., Zraqat, O., Asfour, B., &Deeb, A. (2024). Innovative digital marketing for promoting SDG 2030 knowledge in Jordanian universities in the Middle East. Discover Sustainability, 5(1), 219. https://doi.org/10.1007/s43621-024-00419-8

22. Al-Adwan, A. S., Alsoud, M., Li, N., Majali, T. E., Smedley, J., & Habibi, A. (2024a). Unlocking future learning: Exploring higher education students' intention to adopt meta-education. Heliyon, 10(9). https://doi.org/10.1016/j.heliyon.2024.e29544

23. Al-Adwan, A. S., Al Masaeed, S., Yaseen, H., Balhareth, H., Al-Mu'ani, L. A., &Pavlíková, M. (2024b). Navigating the roadmap to meta-governance adoption. Global Knowledge, Memory and Communication. https://doi.org/10.1108/GKMC-02-2024-0105

24. Xue, Z., Liu, C., Liao, C., Han, G., & Sheng, Z. (2023). Joint service caching and computation offloading scheme based on deep reinforcement learning in vehicular edge computing systems. IEEE Transactions on Vehicular Technology.

25. Alzghoul, A., Khaddam, A. A., Abousweilem, F., Irtaimeh, H. J., &Alshaar, Q. (2024). How business intelligence capability impacts decision-making speed, comprehensiveness, and firm performance. Information Development, 40(2), 220-233. https://doi.org/10.1177/02666669221108438

26. Halteh, K., AlKhoury, R., Ziadat, S. A., Gepp, A., & Kumar, K. (2024). Using machine learning techniques to assess the financial impact of the COVID-19 pandemic on the global aviation industry. Transportation Research Interdisciplinary Perspectives, 24, 101043. https://doi.org/10.1016/j.trip.2024.101043

27. AlKhouri, R., Halteh, P., Halteh, K., & Tiwari, M. (2024). The role of virtue ethics in enhancing reputation through combatting financial crimes. Journal of Money Laundering Control, 27(2), 228-241. https://doi.org/10.1108/JMLC-02-2023-0033

28. Lv, Z., Chen, D., & Wang, Q. (2020). Diversified technologies in internet of vehicles under intelligent edge computing. IEEE transactions on intelligent transportation systems, 22(4), 2048-2059.

29. Alsaaidah, A. M., Shambour, Q. Y., Abualhaj, M. M., & Abu-Shareha, A. A. (2024). A novel approach for e-health recommender systems. Bulletin of Electrical Engineering and Informatics, 13(4), 2902-2912. https://doi.org/10.11591/eei.v13i4.7749

30. Alsharaiah, M., Abualhaj, M., Baniata, L., Al-saaidah, A., Kharma, Q., & Al-Zyoud, M. (2024). An innovative network intrusion detection system (NIDS): Hierarchical deep learning model based on Unsw-Nb15 dataset. International Journal of Data and Network Science, 8(2), 709-722.10.5267/j.ijdns.2024.1.007

31. Baniata, L. H., Kang, S., Alsharaiah, M. A., &Baniata, M. H. (2024). Advanced Deep Learning Model for Predicting the Academic Performances of Students in Educational Institutions. Applied Sciences, 14(5), 1963. https://doi.org/10.3390/app14051963

32. Abu-Shareha, A. A., Abualhaj, M. M., Alsharaiah, M. A., Shambour, Q. Y., & Al-Saaidah, A. (2024). A New Framework for Evaluating Random Early Detection Using Markov Modulate Bernoulli Process Stationary Distribution. International Journal of Intelligent Engineering & Systems, 17(4).10.22266/ijies2024.0831.72

33. Zhao, J., Quan, H., Xia, M., & Wang, D. (2023). Adaptive Resource Allocation for Mobile Edge Computing in Internet of Vehicles: A Deep Reinforcement Learning Approach. IEEE Transactions on Vehicular Technology.

34. Wang, J., Hu, J., Min, G., Zhan, W., Zomaya, A. Y., & Georgalas, N. (2021). Dependent task offloading for edge computing based on deep reinforcement learning. IEEE Transactions on Computers, 71(10), 2449-2461.

35. Kong, X., Duan, G., Hou, M., Shen, G., Wang, H., Yan, X., &Collotta, M. (2022). Deep reinforcement learning-based energy-efficient edge computing for internet of vehicles. IEEE Transactions on Industrial Informatics, 18(9), 6308-6316.

36. Xu, X., Shen, B., Ding, S., Srivastava, G., Bilal, M., Khosravi, M. R., ... & Wang, M. (2020). Service offloading with deep Q-network for digital twinning-empowered internet of vehicles in edge computing. IEEE Transactions on Industrial Informatics, 18(2), 1414-1423.

37. Gao, H., Wang, X., Wei, W., Al-Dulaimi, A., & Xu, Y. (2023). Com-DDPG: task offloading based on multiagent reinforcement learning for information-communication-enhanced mobile edge computing in the internet of vehicles. IEEE Transactions on Vehicular Technology.

38. Zhao, N., Ye, Z., Pei, Y., Liang, Y. C., & Niyato, D. (2022). Multi-agent deep reinforcement learning for task offloading in UAV-assisted mobile edge computing. IEEE Transactions on Wireless Communications, 21(9), 6949-6960.

39. Lambrecht, J., & Funk, E. (2019). Edge-Enabled Autonomous Navigation and Computer Vision as a Service: A Study on Mobile Robot's Onboard Energy Consumption and Computing Requirements. In J. Lambrecht & E. Funk, Advances in intelligent systems and computing (p. 291). Springer Nature. https://doi.org/10.1007/978-3-030-36150-1_24 https://www.kaggle.com/datasets/harunachiromagombe/internet-of-vehicles-dataset

FINANCING
This research is funded by Zarqa University.

CONFLICT OF INTEREST
The authors declare that there is no conflict of interest.

AUTHORSHIP CONTRIBUTION
Conceptualization: Anber Abraheem Shlash Mohammad, Imad Ali.

Data curation: Imad Ali, Anber Abraheem Shlash Mohammad, Suleiman Ibrahim Shelash Mohammad, Ahmad Samed Al-Adwan.

Formal analysis: Imad Ali, Anber Abraheem Shlash Mohammad, Muhammad Turki Alshurideh.

Research: Anber Abraheem Shlash Mohammad, Muhammad Turki Alshurideh, Ahmad Samed Al-Adwan.

Methodology: Imad Ali, Anber Abraheem Shlash Mohammad, Ayman Hindieh, Suleiman Ibrahim Shelash Mohammad, Asokan Vasudevan.

Project management: Anber Abraheem Shlash Mohammad, Ayman Hindieh, Muhammad Turki Alshurideh, Ahmad Samed Al-Adwan.

Resources: Anber Abraheem Shlash Mohammad, Suleiman Ibrahim Shelash Mohammad, Asokan Vasudevan.

Software: Imad Ali, Ayman Hindieh and Anber Abraheem Shlash Mohammad.

Supervision: Imad Ali, Suleiman Ibrahim Shelash Mohammad, Hussam Mohd Al-Shorman.

Validation: Anber Abraheem Shlash Mohammad, Ayman Hindieh, Muhammad Turki Alshurideh, Hussam Mohd Al-Shorman.

Display: Anber Abraheem Shlash Mohammad, Ahmad Samed Al-Adwan, Asokan Vasudevan.

Drafting - original draft: Imad Ali, Anber Abraheem Shlash Mohammad, Asokan Vasudevan.

Writing - proofreading and editing: Imad Ali, Anber Abraheem Shlash Mohammad, Muhammad Turki Alshurideh, Ahmad Samed Al-Adwan, Hussam Mohd Al-Shorman.