A Novel Trusted Intelligent Computing-aware Routing Service Offloading Method Based on MOPSO for Internet of Vehicles

Huiyong LI*, Furong WANG

School of Intelligent Manufacturing and Information, Jiangsu Shipping College, Nantong, 226010, China
lihuiyong@jssc.edu.cn (**Corresponding author*), wfr@jssc.edu.cn

Abstract: Due to the high-speed displacement of vehicles, various computing resources in the Internet of Vehicles have such characteristics as limited communication bandwidth, unstable network connections, and dynamic changes in network topology. Therefore, establishing a trusted service offloading location and supplying consumers with dependable and low-latency services in a resource-constrained mobile edge computing system is still a significant difficulty. This paper proposes a "device-edge-cloud" collaborative trusted edge computing-aware network model, and presents an intelligent computing-aware routing service offloading method based on multi-objective particle swarm optimization for this system model. First, a trustworthiness model for data transmission across distributed computing resources in the Internet of Vehicles environment is proposed. Then, the literature overview points out that the trustworthiness of computing resources is relative, dynamic, reflexive, symmetrical, and not transitive. Based on the trustworthiness model, a multi-dimensional QoS attribute model for computing resources is established, and the scheduling problem for computing resources is abstracted into a multi-objective optimization problem. Finally, an intelligent computing-aware routing scheduling method based on a multi-objective particle swarm optimization algorithm is proposed for solving the task scheduling problem in the Internet of Vehicles environment. Simulation results show that in comparison with the random scheduling algorithm and the greedy scheduling algorithm, the MOPSO scheduling algorithm is significantly better with regard to the reliability of calculation results and communication cost.

Keywords: Trusted computing, Internet of vehicles, Computing-aware networks, MOPSO, Edge computing.

1. Introduction

In recent years, with the continuous development of the Internet of Things technology, the Internet of Vehicles (IoV), which integrates technologies such as Internet of Things technology, cloud computing, mobile computing, and edge computing, continues to develop (Arooj et al., 2022; Cho et al., 2022; Ghafoor et al., 2020; Jain et al., 2023; Jafar & Hamad, 2023; Jayapal et al., 2023; Jung et al., 2022; Kotapati et al., 2023; Lu & Tettamanti, 2021; Praneeth et al., 2021; Rai et al., 2023; Simon et al., 2021; Tan & Loh, 2022; Vikruthi et al., 2022; Zhang et al., 2022). Based on vehicle wireless communication technology (Vehicle to Everything, V2X), the Internet of Vehicles computing system can connect cloud computing units, vehicle computing units and Road-side Units (RSU) and other computing units into an organic computing resource pool (Zhou et al., 2020; Qureshi et al., 2021). Common computing services in the Internet of Vehicles include various information services such as autonomous driving, path planning, collision warning, in-vehicle entertainment, augmented reality, and image rendering (Lu et al., 2019; Cui et al., 2022). These services are generally completed in the computing resource pool of the Internet of Vehicles utilizing distributed computing technologies.

In the computing resource pool of the Internet of Vehicles, cloud computing resources, on-board computing resources, and roadside computing resources have their own characteristics. Cloud computing resources use virtualization technology to establish a large-capacity computing resource pool, so that various services can obtain the required computing resources, storage resources, and power resources. Cloud computing resources can meet the needs of computing-intensive business processing, but many services in the Internet of Vehicles have high real-time requirements, and the transmission delay from the terminal to the cloud cannot meet the service performance requirements in many cases. The computing resources of the on-board computing unit and the roadside computing unit belong to edge computing resources. Many calculations with high real-time requirements in the Internet of Vehicles are generally directly calculated through Mobile Edge Computing (MEC) (Lu et al., 2022; Kong et al., 2022; Zhang et al., 2022). However, although edge computing solves the problems of bandwidth shortage, network congestion, and excessive delay caused by uploading massive data to the cloud computing center in the Internet of Vehicles, it also makes computing resources present a trend of ubiquitous deployment, which inevitably creates a "Computing island" effect. On the one hand, edge computing nodes do not perform effective collaborative processing tasks, and the computing resources of a single node cannot meet the

https://doi.org/10.24846/v32i3y202310

computing resource requirements of super-large computing-intensive computing tasks, and still cannot solve new types of business that are both computing-intensive and time-delay sensitive - the problem of ultra-low latency requirements. On the other hand, although some edge computing nodes are overloaded and unable to effectively process computing tasks, due to the unbalanced network load, some computing nodes are bound to remain idle, resulting in insufficient computing resources for the edge network use. Therefore, in order to efficiently and collaboratively utilize these heterogeneous computing resources, the academic community has proposed a distributed system-based computing and network fusion network architecture - Computing-aware Networking (CAN) (Wang et al., 2021; Du et al., 2022). A computing-aware network aims to connect and coordinate various computing tasks of the cloud, edge and terminal through the network, and accomplish the deep integration and collaborative perception of computing and network, as well as the on-demand scheduling and efficient sharing of computing resources.

Computing-aware routing and computing resource allocation are key issues in the research of computing-aware networks. Many scholars have proposed many effective computing-aware routing algorithms and computing resource allocation strategies for this problem (Han et al., 2021; Wang et al., 2021). Based on various attributes of computing resources in the computing network, these strategies match various distributed computing tasks in the computing-aware network to computing resources in the computing resource pool and complete specified computing tasks.

Due to the high-speed operation of vehicles in the Internet of Vehicles system, in comparison with ordinary computing network systems, the Internet of Vehicles system based on computing networks features limited communication bandwidth, unstable network connections, dynamic changes in network topology, and heterogeneous distributed computing resources. The network connections between computing resources in these computing resource pools are intermittent. Therefore, the trustworthiness of computing resources becomes an important factor in the distribution of distributed tasks. How to determine the trusted service offloading location, form trusted computing routing, and provide users with reliable and low-latency services in a resource-constrained mobile computing network system such as the Internet of Vehicles is a huge challenge.

Xu et al. (2021) designed a distributed service offloading method D-SOAC that combines deep learning and deep reinforcement learning. Liu et al. (2020) proposed a computing-aware routing scheduling strategy based on the Floyd algorithm to solve the problem of intelligent task scheduling. These resource scheduling algorithms cannot solve the multi-dimensional optimal scheduling problem of computing resources in the Internet of Vehicles. The multi-objective particle swarm optimization algorithm is an efficient and fast multi-objective optimization algorithm (Cui, Meng & Qiao, 2022; Han et al., 2021).

Aiming at the gaps in the existing research on the Internet of Vehicles with computing, this paper first provides a trustworthiness model for data transfer across dispersed computing resources in the Internet of Vehicles environment. The remainder of this paper is as follows. Section 2 presents that the trustworthiness of computing resources is relative, dynamic, reflexive, symmetrical, and not transitive. In Section 3, based on the trustworthiness model, a multi-dimensional QoS attribute model of computing resources in the Internet of Vehicles is established, and the scheduling problem of computing resources is abstracted into a multi-objective optimization problem. In Section 4, an intelligent computing-aware routing scheduling method based on a multi-objective particle swarm optimization algorithm is proposed to solve the task scheduling problem in the Internet of Vehicles. Section 5 presents and discusses the simulation results, which show that the proposed scheduling algorithm is effective and has the best overall performance. Finally, Section 6 sets forth the conclusion of this paper.

2. Overview of Internet of Vehicles Systems Based on a Computing-aware network

2.1 Internet of Vehicles System Based on a Computing-Aware Network

An IoV system based on a computing-aware network mainly includes: on-board computing equipment, roadside computing units, base stations, edge gateways, edge servers, and cloud computing centers. In the Internet of Vehicles environment, computing resources can be divided into three categories: cloud computing resources, roadside computing resources, and on-board computing resources.

In the IoV system based on the computing-aware network, the user service is generally initiated by the vehicle terminal and transmitted to the roadside computing unit through the wireless link. Edge gateways are mainly responsible for routing control and data forwarding. In actual network systems, edge gateways can be deployed in base stations or roadside computing units, and data transmission can be performed between routing nodes through real-time dynamic links. The edge server is an edge application platform with hardware infrastructure as a virtualized resource, and is mainly responsible for providing computing resources and storage resources to process user services. Data transmission is performed between edge computing nodes and routing nodes through fixed links. The cloud computing center has sufficient computing resources and storage resources, and is a large server cluster deployed far away from users. There are fixed links between edge computing nodes and cloud computing nodes for data transmission. Throughout the system, cloud computing resources, roadside computing resources, and on-board computing resources constitute a large distributed computing resource pool.

2.2 Distributed Computing Task Scheduling Model in Computing-based Vehicle Networking

Distributed computing task scheduling model in the computing Internet of Vehicles is as follows: in the **M**×**N** distributed computing task scheduling model composed of **m** tasks and **n** computing resources, the distributed computing task scheduling problem can be described as a quadruple, that is **Dis** = (**T**, **R**, **O**, **Θ**). Among them, **T** is a computing task set composed of m computing tasks, **R** is a vehicle networking computing resource set composed of **n** computing resources, **O** represents the scheduling optimization objective function of the scheduling system, and **Θ** represents the scheduling algorithm.

For **T** = (**TASK**,<), **TASK** = ($task_1$,$task_2$,…,$task_n$) is a set of executable tasks. The sign < indicates a partial order relationship on **T**, which is used to illustrate the priority relationship between tasks, that is, if $task_1$<$task_2$, this means that the execution of $task_1$ must start before $task_2$.

2.3 Distributed Computing Trusted Model in Computing-based Vehicle Networking

In the computing-based vehicle networking system, due to the high-speed operation of the vehicle, the communication method between the on-board computing resources and other computing resources is mainly wireless communication, and its quality is affected by the dynamic communication environment and varies from time to time. The network topology of computing-based vehicle networking is also changing in real time. Therefore, characterizing and assessing the changes in the trustworthiness of the transmitted data produced by changes in the communication environment between computing resources in the computing-based Internet of Vehicles is a very important topic. In this paper, this problem is referred to as the trustworthy attribute among computing resources in the computing-based Internet of Vehicles system. The model and measurement method for this attribute are given below.

Definition 1: In the computing Internet of Vehicles environment, computing resource A transmits data M to computing resource B, then the probability that B correctly receives M is the trustworthiness of computing resource A to computing resource B, denoted by $Tr_{A\to B}$.

When computing resource A directly sends data to computing resource B without passing through other routes,

(1) When computing resource A and computing resource B communicate through memory, data bus or wired communication, $Tr_{A\to B} = 1$.

(2) When computing resource A and computing resource B have no communication connection, $Tr_{A\to B} = 0$.

(3) Cheng et al. (2014) pointed out the direct impact of communication radius and communication loss on communication quality in wireless communication. When computing resource A and computing resource B use wireless

communication, let the communication distance be d, the communication radius be R, and the path loss coefficient be a (2<a<6), then:

$$\mathrm{Tr}_{A\to B}=\begin{cases}1-0.5\left(\dfrac{d}{R}\right), d<R\\ 0.5\left(2-\dfrac{d}{R}\right)^{2a}, R\le d<2R\\ 0, d\ge 2R\end{cases} \quad (1)$$

According to the above definitions and calculation formulas, the trustworthiness of computing resources in the computing Internet of Vehicles has the following properties:

Theorem 1. The trustworthiness of computing resources in the computing Internet of Vehicles is relative: $Tr_{A\to B} \neq Tr_{A\to C}$.

Explanation: The trustworthiness of a computing resource is not an absolute measure, but a relative measure. The trustworthiness of computing resources can only be determined after the sender and receiver are determined. The trustworthiness of a computing resource relative to different objects is not necessarily the same.

Proof (proof by contradiction): Computing resources A, B and C are distributed computing resources in the Internet of Vehicles environment, it is assumed that $Tr_{A\to B} = Tr_{A\to C}$.

∴ In the wireless communication environment, one can obtain: communication distance $d_{A\to B}=d_{A\to C}$, and path loss coefficient $a_{A\to B}=a_{A\to C}$.

∴ The communication environment of the Internet of Vehicles is equidistant and stable.

∵ This conclusion contradicts the fact that the communication environment of the Internet of Vehicles is dynamically changing and unstable.

∴ The assumption in the question does not hold.

∴ $Tr_{A\to B} \neq Tr_{A\to C}$

Q.E.D.

Theorem 2. The trustworthiness of computing resources in the Internet of Vehicles is dynamic: $Tr_{A\to B}(t) \neq Tr_{A\to B}(t')$.

Explanation: The trustworthiness of a computing resource relative to another object is not fixed, but changes with changes in the communication environment (for example: different routing options, changes in communication signal strength, etc.).

Proof (proof by contradiction): Computing resources A and B are distributed computing resources in the Internet of Vehicles environment, assuming $Tr_{A\to B}(t) = Tr_{A\to B}(t')$.

∴ In the wireless communication environment, according to formula 2, one can obtain: communication distance $d_{A\to B}(t) = d_{A\to B}(t')$, and path loss coefficient $a_{A\to B}(t) = a_{A\to B}(t')$.

∴ The location of computing resources in the communication environment of the Internet of Vehicles is fixed.

∵ This conclusion contradicts the fact that computing resources in the communication environment of the Internet of Vehicles are constantly changing dynamically.

∴ The assumption in the title is not valid.

∴ $Tr_{A\to B}(t) \neq Tr_{A\to B}(t')$

Q.E.D.

Theorem 3. The trustworthiness of computing resources in the Internet of Vehicles is reflexive:

$Tr_{A\leftrightarrow A} = Tr_{A\to A} = Tr_{A\leftarrow A}$

Explanation: For a computing resource relative to itself, since the data is stored in the memory and not transmitted through the communication line, the trustworthiness is 1 at this time. If a computing task is always calculated by one computing resource without data interaction with other computing resources, then this computing mode is also called centralized computing.

Proof: Assuming that computing resource A is a distributed computing resource in the Internet of Vehicles environment, the intermediate result data running on computing resource A is exchanged through the stack or queue in the memory.

∴ According to Theorem 3 for the trustworthiness calculation method, the following can be obtained: $TrA\to A = 1$ and $TA\leftarrow A = 1$,

∴ $Tr_{A\leftrightarrow A} = Tr_{A\to A} = Tr_{A\leftarrow A}$

Q.E.D.

2.4 Distributed Computing Resource Model of the Internet of Vehicles that Adds Trustworthiness

The topological graph model is a regularly used paradigm for explaining distributed computing resources.

The distributed computing resource model of the Internet of Vehicles with added trustworthiness can be expressed as a weighted undirected topology graph **RG** = (**V**, **E**, **W**, $\mathbf{W}_{tr}$), where **V** is the computing resource set, E is Undirected edge set, **W** is the weight set, and **W**tr is the trustworthiness set between computing resources. In this model, node v represents a computing resource. Among the other vectors, the computing resource weight **W**(v) = (speed, cost, energy, stability, success) is a five-dimensional vector, respectively representing the computing speed, computing cost, computing energy consumption, computing stability and the computing success rate attribute of the computing resource v. The trustworthiness weight between computing resources can be expressed as: $W_{tr}(v_i,v_j) = trust_{vi\text{-}vj}$.

3. Task Scheduling Optimization Model

3.1 Concepts of Multi-Objective Optimization Problems

The mathematical formula of the multi-objective optimization problem can be expressed as follows:

$$minF(x)=[F_1(x),F_2(x),\ldots,F_n(x)] \tag{2}$$

where F(x) is the objective function or fitness function, x represents the decision vector of n decision variables, and the formula gives the situation where all objective functions are minimized.

In single-objective optimization problems, the goal is to find the best possible solution. However, for multi-objective optimization problems, if the objective functions conflict with each other, there is not only one optimal solution, but a group of optimal solutions. In order to represent this group of optimal solutions, Pareto optimal solutions are used.

Pareto dominant: Considering a minimization problem, a decision vector X_a is said to dominate another decision vector Xb, which happens, among k objective functions, if and only if $F1(x_a) \leq F2(x_b)$ is always established, where i = 1,2,…k, and there exists at least one constant $Fj(x_a) < Fj(x_b)$, where j=1,2,…k.

Pareto optimal set: Feasible decision vector X* represents Pareto optimal solution, if there is no feasible decision vector X_i so that $F(x_i)$ dominates F(x*), this Pareto optimal decision vector set is called Pareto optimal set, this means that every solution in this set is of equal importance and is a good compromise with regard to the intended goals. The trade-off curve of the target space obtained from the Pareto optimal set is called the Pareto front.

3.2 Multi-Objective Optimization Model for Computing-based Scheduling in the Computing Internet of Vehicles

In the computing-based Internet of Vehicles system, the routing controller can perceive the data size for user services and the link status of the network (including the data transmission rate, jitter, and packet loss rate of the link, etc.). When performing distributed computing, it is generally necessary to compare various computing resources in the system and assign tasks to the optimal computing resources. The attributes of the computing resources of the Internet of Vehicles include: the trustworthiness of the computing resource, computing speed, computing cost, computing energy consumption, computing stability and computing success rate. Therefore, the purpose of this paper is to present an intelligent computing-aware routing service offloading method. The goal is to achieve a scheduling algorithm with the smallest computing time for computing tasks, the most reliable computing results, the lowest computing energy consumption, the cheapest computing costs, and the highest computing success rate. This is a typical multi-objective optimization problem.

The distributed computing task scheduling model of the Internet of Vehicles system is based on the computing-aware network and the distributed computing resource model of the Internet of Vehicles that adds trustworthiness. The intelligent computing-aware routing service offloading method can be abstracted into a multi-objective optimization model.

The objective function is:

$$\begin{cases} \max F_{trust} = \prod_{i=1}^{n} trust_{vi-vi+1} \\ \max F_{speed} = \sum_{i=1}^{n} \frac{1}{speed_{vi}} \\ \min F_{\text{cost}} = \sum_{i=1}^{n} \text{cost}_{vi} \\ \min F_{energy} = \sum_{i=1}^{n} energy_{vi} \\ \max F_{stability} = \prod_{i=1}^{n} stability_{vi} \\ \max F_{success} = \prod_{i=1}^{n} success_{vi} \end{cases} \tag{3}$$

s.t.

$$\begin{cases} 0 < trust_{vi-vi+1} < 1 \\ 0 < speed_{vi} < 1000 \\ 0 < \text{cost}_{vi} < 100 \\ 0 < energy_{vi} < 100 \\ 0 < stability_{vi} < 1 \\ 0 < success_{vi} < 1 \end{cases} \tag{4}$$

4. Computing-Aware Routing Algorithm Based on Particle Swarm Optimization

Since the multi-objective optimization problem constructed above is a nonlinear programming problem and is an NP-hard problem, the optimal solution cannot be obtained using traditional optimization methods. Therefore, a new swarm intelligence algorithm is considered in this section for solving the problem.

4.1 Multi-Objective Particle Swarm Optimization Algorithm

The notion of particle swarm optimization algorithm is derived from the studies on the predatory behavior of birds. Each individual in the group finds the optimal solution through mutual cooperation and information sharing and interaction. The specific idea of the Multi-Objective Particle Swarm Optimization (MOPSO) algorithm is as follows: by assuming that a massless random particle simulates the individual in the bird swarm, each random particle has three main attributes: speed, direction and state. Speed represents the movement speed of the random particle in the particle swarm. Direction indicates the moving direction of the random particle, and state indicates the state of the random particle at that moment. In the process of optimization, each random particle looks for the optimal solution for the state attribute in the specified solution space, and shares the relevant information on the individual optimal value with other random particles in the particle swarm, and finds the optimal solution in the particle swarm. The optimal individual extremum is used as the current global optimal solution gbest of the entire particle swarm. In the process of the next iteration, all random particles in the particle swarm adjust their speed, direction and state according to their current individual extremum pbest and the current global optimal solution gbest of the entire particle swarm to calculate the next individual state optimum.

The multi-objective particle swarm optimization algorithm can find the approximate optimal solution for the optimization target through the possibility search in the defined solution space, and it has excellent performance in solving NP-hard problems and combinatorial optimization problems. Therefore, this paper adopts MOPSO to solve the optimal solution of the above optimization problem.

4.2 Intelligent Computing-Aware Routing Algorithm Based on MOPSO

The optimization process of the multi-objective particle swarm optimization algorithm can be divided into the following steps:

Step 1:

A group of random particles is initialized within the scope of the solution space, and the state of the particle swarm is initialized, and the state space of the particles includes multiple random variables. Therefore, the initial state of the particle swarm can be expressed as a K×N matrix, namely:

$$P \triangleq \{P_{k,n} \mid k \in K, n \in N\} \in R^{K \times N} \tag{5}$$

This is the weighted sum which represents the adaptive value in MOPSO:

$$fit_p = \{fit \mid k \in K\} \in R^{K \times 1} \tag{6}$$

Step 2:

Calculate the best adaptive value pbest for the individual:

$$pbest = \max fit_p = \max(\eta + \kappa P) \tag{7}$$

Step 3:

Evaluate and update the global optimal adaptive value gbest: if pbest < gbest, there is no need to update gbest; if pbest > gbest, set pbest = gbest, and record the current state when the global optimal solution is obtained.

Step 4:

Update the velocity and position of the particles, and solve the following equations for new adaptive values.

$$v_{i+1} = \omega_i \times v_i + c_1 \times rand() \times (pbest_i - x_i) \\ + c_2 \times rand() \times (gbest_i - x_i) \tag{8}$$

$$x_{i+1} = x_i \times v_{i+1} \tag{9}$$

The first of the above formulas represents the updated speed of the particles. Among them, $\boldsymbol{\omega_i} \times \mathbf{v_i}$ is called the memory item, which represents the influence of the magnitude and direction of the last speed, ω_i represents the inertia factor, and by dynamically adjusting ω_i one can obtain better optimization results than with fixed values. $\mathbf{c_1} \times$ **rand()** $\times$ ($\mathbf{pbest_i}$ - $\mathbf{x_i}$) is called the self-awareness term, which is a vector from the current point to the best point of the particle itself, indicating that the particle's action comes from its own experience, and c_1 represents the individual learning factor. $\mathbf{c_2} \times$ **rand()** $\times$ ($\mathbf{gbest_i}$ - $\mathbf{x_i}$) is called the group cognition item, which is a vector from the current point to the global best point, reflecting the cooperation and knowledge sharing among particles, and c_2 represents the global learning factor. The formula x_{i+1} = $x_{i \times vi+1}$ indicates that the i-th state of the particle swarm turns to the i+1-th state.

Step 5:

When the end conditions (time limit, number of iterations limit, error limit) are satisfied, then end; otherwise, continue the loop according to the above process.

5. Simulation and Analysis

In order to verify the effectiveness of the scheduling algorithm proposed in this paper, a typical distributed computing task of the Internet of Vehicles system is taken as an example for the following analysis. It shall be assumed that, in order to check whether all vehicles in the city have speeding behaviour, the monitor publishes the calculation task to the Internet of Vehicles system. Now using the distributed computing method, the real-time speed data for each car in the city will be collected in real time and distributed to the distributed computing resource pool of the Internet of Vehicles for calculation. It shall be assumed that a car Car611 is currently driving on a city road. The car C611 carries several sensors (including speed sensor, GPS, etc.), which periodically collect various real-time data.

It shall also be assumed that the current vehicle has several on-board computing units, and that there are several computing units on the roadside of the current road. These distributed computing resources of the Internet of Vehicles are as follows: (1) On-board computing resources: computing resources 29, 58, 33, 38, 89, and 41; (2) Roadside computing resources: computing resources 263, 613, 368, and 417; (3) Computing center computing resources: computing resources 1134 and 1519. These on-board computing resources and roadside computing resources communicate wirelessly through 5G. The attribute values of each computing resource of the Internet of Vehicles are shown in Table 1.

Suppose that the calculation task can be decomposed into a list of subtasks TaskList = {t_4, t_1, t_3, t_2, t_5, t_7, t_6, t_8, t_9}, and the number of calculations for each subtask is illustrated in Table 2:

Table 1. Attribute values for each computing resource in Internet of Vehicles

Type	Location	Speed	Cost	Energy	Stability	Success
Onboard	Car611	200	32	68	0.53	0.75
	Car1537	260	29	73	0.48	0.8
	Car956	180	36	65	0.57	0.79
Roadside	Roadside68	300	67	35	0.74	0.9
	Roadside47	320	56	43	0.82	0.92
Background	Computing Center	380	73	46	0.92	0.95

Table 2. Number of calculations of each subtask in the task list

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9
500	1300	700	2300	1500	2700	1400	3100	1900

Based on the computing scenarios and related parameter settings of the computing-based Internet of Vehicles mentioned above, the MOPSO-based intelligent computing-aware routing allocation algorithm for distributed computing task scheduling was used. In order to illustrate and analyse the effectiveness of the scheduling algorithm, the commonly used random scheduling algorithm (RSA) and the greedy scheduling algorithm (GSA) were introduced for comparison purposes.

When using the MOPSO algorithm, it is necessary to set the algorithm parameters. Reasonable parameter settings can better simulate the foraging behaviour of birds, and enable one to find the optimal solution with the least number of iterations. In this paper, after the experiments carried out, the main parameters of the MOPSO algorithm were set as follows: the number of iterations is 50, the number of particles is 300, the inertia factor $\omega = 0.5$, and the learning factor $c_1 = c_2 = 2$.

Computing resources are denoted by the letter c, for example, computing resource 41 is denoted as c_{41}. Computational tasks are denoted by the letter t. The random scheduling algorithm is denoted as RSA, the greedy scheduling algorithm is denoted as GSA, the intelligent computing-aware routing scheduling algorithm based on MOPSO is denoted as MOPSO, and the scheduling length is rendered in seconds. The scheduling results are shown in Table 3.

From the comparison of the above scheduling results, it can be seen that the scheduling length of the random scheduling algorithm is the longest, while the scheduling length of the GSA scheduling algorithm is the shortest, and the scheduling length of the MOPSO scheduling algorithm is close to that of the GSA scheduling algorithm. In order to further compare these three scheduling methods, the change of the values of computing resources for each dimension of the three employed algorithms during the calculation process related to the calculation task, as it is given in the Table 4.

Among the various parameters of the three scheduling algorithms in the above table, the number of calculations, calculation cost and computing energy consumption for the computing resources are summed when combined, and the computing stability and success rate for the computing resources are averaged.

In Table 4, the multiplication method is used to calculate the total trustworthiness across resources, and the communication cost is calculated by summation. It can be seen from the above comparison that the performance of each attribute of the GSA scheduling algorithm is close to that of the RSA scheduling algorithm, while the MOPSO scheduling algorithm is obviously superior to the RSA scheduling algorithm and the GSA scheduling algorithm in terms of the trustworthiness of the calculation results and the communication cost. Therefore, in the Internet of Vehicles environment with heterogeneous computing resources and complex communication, the MOPSO scheduling algorithm is effective and the comprehensive performance is optimal.

Table 3. Scheduling results for the three scheduling algorithms

RSA	Task	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9
	Resources	c_{29}	c_{33}	c_{263}	c_{41}	c_{368}	c_{58}	c_{1134}	c_{89}	c_{38}
	Start	0	0	0	0	0	5.01	12.79	4.69	21.92
	End	2.5	5	2.33	12.78	4.69	18.5	16.46	21.91	29.22
	length	**29.22**								
GSA	Task	t_2	t_1	t_3	t_5	t_4	t_7	t_6	t_8	t_9
	Resources	c_{1134}	c_{1519}	c_{368}	c_{417}	c_{263}	c_{368}	c_{1519}	c_{1134}	c_{1519}
	Start	0	0	0	0	0	7.68	3.43	4.70	12.86
	End	3.42	1.31	2.19	4.69	7.67	12.05	10.53	12.85	17.85
	length	**17.85**								
MOPSO	Task	t_4	t_1	t_3	t_2	t_5	t_7	t_6	t_8	t_9
	Resources	c_{368}	c_{417}	c_{1134}	c_{1519}	c_{263}	c_{368}	c_{417}	c_{1134}	c_{368}
	Start	0	0	0	0	0	7.2	3.43	5.01	13.17
	End	7.19	1.56	1.84	3.42	5	11.57	11.86	13.16	19.1
	length	**19.1**								

Table 4. QoS values of computing resources for each dimension of the three scheduling algorithms

	Task name	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	total
RSA	Computing resources	c_{29}	c_{33}	c_{263}	c_{41}	c_{368}	c_{58}	c_{1134}	c_{89}	c_{38}	
	Number of calculations	500	1300	700	2300	1500	2700	1400	3100	1900	15400
	Computing cost	80	145	156.11	460.08	262.64	432	268.64	619.92	211.99	2636.38
	Calculate energy	170	365	81.55	830.7	201.67	918	169.28	1119.3	533.63	4389.13
	Computing stability	0.53	0.48	0.74	0.57	0.82	0.53	0.92	0.57	0.48	0.627
	Computing success rate	0.75	0.8	0.9	0.79	0.92	0.75	0.95	0.79	0.8	0.828
GSA	Task name	t_2	t_1	t_3	t_5	t_4	t_7	t_6	t_8	t_9	total
	Computing resources	c_{1134}	c_{1519}	c_{368}	c_{417}	c_{263}	c_{368}	c_{1519}	c_{1134}	c_{1519}	
	Number of calculations	1300	500	700	1500	2300	1400	2700	3100	1900	15400
	Computing cost	249.66	95.63	122.64	262.64	513.89	245.28	519.03	595.68	365	2969.45
	Calculate energy	157.32	60.26	94.17	201.67	268.45	188.34	327.06	375.36	230	1902.63
	Computing stability	0.92	0.92	0.82	0.82	0.74	0.82	0.92	0.92	0.92	0.867
	Computing success rate	0.95	0.95	0.92	0.92	0.9	0.92	0.95	0.95	0.95	0.934
MOPSO	Task name	t_4	t_1	t_3	t_2	t_5	t_7	t_6	t_8	t_9	total
	Computing resources	c_{368}	c_{417}	c_{1134}	c_{1519}	c_{263}	c_{368}	c_{417}	c_{1134}	c_{368}	
	Number of calculations	2300	500	700	1300	1500	1400	2700	3100	1900	15400
	Computing cost	402.64	87.36	134.32	249.66	335	245.28	472.64	595.68	332.64	2855.22
	Calculate energy	309.17	67.08	84.64	157.32	175	188.34	362.92	375.36	255.42	1975.25
	Computing stability	0.82	0.82	0.92	0.92	0.74	0.82	0.82	0.92	0.82	0.844
	Computing success rate	0.92	0.92	0.95	0.95	0.95	0.92	0.92	0.95	0.92	0.928

6. Conclusion

In view of such characteristics as limited communication bandwidth, unstable network connections, dynamic changes in network topology, and heterogeneous distributed computing resources among various computing resources in the Internet of Vehicles, this paper proposes a “device-edge-cloud” trusted collaborative edge computing-aware network model of Internet of Vehicles. In this model, the trustworthiness of data transmission is highlighted, and a trustworthiness model for data transmission across dispersed computing resources in the Internet of Vehicles environment is provided. A literature overview points out that the trustworthiness of computing resources is relative, dynamic, reflexive, symmetrical, and not transitive. Based on this system model, an intelligent computing-aware routing service offloading method based on multi-objective particle swarm optimization was designed. The simulation results show that the proposed scheduling algorithm is effective and has the best overall performance.

Acknowledgements

This work was supported by the 2021 Basic Science Research Program of Nantong, Jiangsu, China (No.: JC2021127), and the Science Foundation of Jiangsu Shipping College, China (No.: HYKY/2021B07), High-level Talents Research Start-up Fund supported by Jiangsu Shipping College.

REFERENCES

Arooj, A., Farooq, M. S., Akram, A., Iqbal, R., Sharma, A. & Dhiman, G. (2022) Big data processing and analysis in internet of vehicles: architecture, taxonomy, and open research challenges. *Archives of Computational Methods in Engineering*. 29(2), 793-829. doi: 10.1007/s11831-021-09590-x.

Cheng, H. J., Huang, X.B. & Xiong, N.X. (2014) Minimum-Energy broadcast algorithm for wireless sensor networks with unreliable communications. *Ruan Jian Xue Bao/Journal of Software*. 25(5), 1101-1112.

Cho, M. K. & Chun, Y.H. (2022) High-Precision Position Protocol for Vehicle to Pedestrian using 5G Networks. *Journal of System and Management Sciences*. 12(1), 241-253. doi: 10.33168/JSMS.2022.0117.

Cui, G., Zhang, W., Xiao, Y., Yao, L. & Fang, Z. (2022) Cooperative perception technology of autonomous driving in the internet of vehicles environment: A review. *Sensors*. 22(15), 5535. doi: 10.3390/s22155535.

Cui, Y., Meng, X. & Qiao, J. (2022) A multi-objective particle swarm optimization algorithm based on two-archive mechanism. *Applied Soft Computing*. 119, 108532. doi: 10.1016/j.asoc.2022.108532.

Du, Z., Li, Z., Duan, X. & Wang, J. (2022). Service Information Informing in Computing Aware Networking. In: *2022 International Conference on Service Science (ICSS), 13-15 May, 2022, Zhuhai, China.* IEEE. pp. 125-130.

Ghafoor, K. Z., Kong, L., Zeadally, S., Sadiq, A. S., Epiphaniou, G., Hammoudeh, M. & Mumtaz, S. (2020) Millimeter-wave communication for internet of vehicles: status, challenges, and perspectives. *IEEE Internet of Things Journal*. 7(9), 8525-8546. doi: 10.1109/JIOT.2020.2992449.

Han, F., Chen, W. T., Ling, Q. H. & Han, H. (2021) Multi-objective particle swarm optimization with adaptive strategies for feature selection. *Swarm and Evolutionary Computation*. 62, 100847. doi: 10.1016/j.swevo.2021.100847.

Han, X., Zhao, Y., Yu, K., Huang, X., Xie, K. & Wei, H. (2021) Utility-Optimized Resource Allocation in Computing-Aware Networks. In: *2021 13th International Conference on Communication Software and Networks (ICCSN), 4-7 June, 2021, Chongqing, China*. IEEE. pp. 199-205.

Jafar, Z. & Hamad, A.H. (2023) Performance evaluation of a multi organizations secure internet of vehicles based on hyperledger fabric blockchain platform. *Ingénierie des Systèmes d'Information*. 28(3), 703-709. doi: 10.18280/isi.280320.

Jain, A., Nandan, D. & Meduri, P. (2023) Data export and optimization technique in connected vehicle. *Ingénierie des Systèmes d'Information*. 28(2), 517-525. doi: 10.18280/isi.280229.

Jayapal, P. K., Muvva, V. R. & Desanamukula, V. S. (2023) Stacked Extreme Learning Machine with Horse Herd Optimization: A Methodology for Traffic Sign Recognition in Advanced Driver Assistance Systems. *Mechatronics and Intelligent Transportation Systems.* 2(3), 131-145. doi: 10.56578/mits020302.

Jung, I. H., Lee, J .M. & Hwang, K. (2022) Advanced Smart Parking Management System Development Using AI. *Journal of System and Management Sciences*. 12(1), 3-62. doi: 10.33168/JSMS.2022.0105.

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. doi: 10.1109/TII.2022.3155162.

Kotapati, G., Ali, M. A. & Vatambeti, R. (2023) Deep Learning-Enhanced Hybrid Fruit Fly Optimization for Intelligent Traffic Control in Smart Urban Communities. *Mechatronics and Intelligent Transportation Systems*. 2(2), 89-101. doi: 10.56578/mits020204.

Liu, Y., Peng, M., Shou, G., Chen, Y. & Chen, S. (2020) Toward edge intelligence: Multiaccess edge computing for 5G and Internet of Things. *IEEE Internet of Things Journal*. 7(8), 6722-6747. doi: 10.1109/JIOT.2020.3004500.

Lu, H., Liu, Q., Tian, D., Li, Y., Kim, H. & Serikawa, S. (2019) The cognitive internet of vehicles for autonomous driving. *IEEE Network*. 33(3), 65-73. doi: 10.1109/MNET.2019.1800339.

Lu, J., Chen, L., Xia, J., Zhu, F., Tang, M., Fan, C. & Ou, J. (2022) Analytical offloading design for mobile edge computing-based smart internet of vehicle. *EURASIP Journal on Advances in Signal Processing*. 2022(1), 44. doi: 10.1186/s13634-022-00867-2.

Lu, Q. & Tettamanti, T. (2021) Impacts of Connected and Automated Vehicles on Freeway with Increased Speed Limit. *International Journal of Simulation Modelling*. 20(3), 453-464.

Praneeth, V., Kumar, K. R. & Karyemsetty, N. (2021) Security: Intrusion prevention system using deep learning on the internet of vehicles. *International Journal of Safety and Security Engineering*. 11(3), 231-237. doi: 10.18280/ijsse.110303.

Qureshi, K. N., Din, S., Jeon, G. & Piccialli, F. (2021) Internet of vehicles: Key technologies, network model, solutions and challenges with future aspects. *IEEE Transactions on Intelligent Transportation Systems*. 22(3), 1777-1786. doi: 10.1109/TITS.2020.2994972.

Rai, A., Kundu, K., Dev, R., Nayak, S., Keshari, J. P. & Nandan, D. (2023) Special vehicle like ambulance recognition and security system using Mobility Accredit System. *Revue d'Intelligence Artificielle*. 37(2), 509-515. doi: 10.18280/ria.370228.

Simon, J., Trojanová, M., Hošovský, A. & Sárosi, J. (2021) Neural Network Driven Automated Guided Vehicle Platform Development for Industry 4.0 Environment. *Tehnički vjesnik*. 28(6), 1936-1942. doi: 10.17559/TV-20200727095821.

Tan, V. & Loh, Y.P. (2022). Dashboard camera view vehicle license plate compliance verification. *Journal of Logistics, Informatics and Service Science*. 9(4), 37-50. doi: 10.33168/LISS.2022.0404.

Vikruthi, S., Archana, M. & Tanguturi, R. C. (2022) Shadow detection and elimination technique for vehicle detection. *Revue d'Intelligence Artificielle*. 36(5), 753-760. doi: 10.18280/ria.360513.

Wang, K., Yu, X., Lin, W., Deng, Z. & Liu, X. (2021) Computing aware scheduling in mobile edge computing system. *Wireless Networks*. 27, 4229-4245. doi: 10.1007/s11276-018-1892-z.

Xu, X., Fang, Z., Qi, L., Dou, W., He, Q. & Duan, Y. (2021) A deep reinforcement learning-based distributed service offloading method for edge computing empowered internet of vehicles. *Chinese Journal of Computers.* 44(12), 2382-2405.

Zhang, D., Cao, L., Zhu, H., Zhang, T., Du, J. & Jiang, K. (2022) Task offloading method of edge computing in internet of vehicles based on deep reinforcement learning. *Cluster Computing*. 25(2), 1175-1187. doi: 10.1007/s10586-021-03532-9.

Zhang, L. Y., Duan, X. K., Ma, J., Zhang, M., Wen, Y. & Wang, Y. (2022) Mechanism of Road Capacity under Different Penetration Scenarios of Autonomous Vehicles. *International Journal of Simulation Modelling*. 21(1), 172-183.

Zhou, H., Xu, W., Chen, J. & Wang, W. (2020) Evolutionary V2X technologies toward the Internet of vehicles: Challenges and opportunities. *Proceedings of the IEEE*. 108(2), 308-323. doi: 10.1109/JPROC.2019.2961937.