International Journal of Computational Intelligence in Engineering (IJCIE)
Vol. 01, No. 01, March 2026, pp. 16-28
ISSN: XXXX-XXXX, DOI: 10.5281/zenodo.19513313

Coalition based Game-Theoretic Routing Technique for Delay Tolerant Networks with Cost and Congestion Optimization

Shyamala G[1*], Pallavi GB[2], Latha N.R[3], Rajesh I S[4], Sumit Gupta[5]

[1]Associate Professor, Department of CSE, BMS College of Engineering, Bengaluru, India. Email: shyamala.cse@bmsce.ac.in
[2]Associate Professor, Department of CSE, BMS College of Engineering, Bengaluru, India. Email: pallavi.cse@bmsce.ac.in
[3]Associate Professor, Department of CSE, BMS College of Engineering, Bengaluru, India. Email: latha.cse@bmsce.ac.in
[4]Associate Professor, Dept. of AI & ML, BMSIT&M, Bengaluru, India. Email: rajeshaiml@bmsit.in
[5]Research Head, R&D Department, DeepCognix AI Labs, Bangalore, Karnataka, India. Email: sumit@deepcognix.com

Article Info

Article history:

Received Jan 11, 2026
Revised Mar 14, 2026
Accepted Mar 20, 2026

Keywords:

Delay Tolerant Networks
Congestion Control
Game Theory
Network Performance
Wireless Communication

ABSTRACT

Delay Tolerant Networks (DTNs) routing is a difficult but very important task, mostly because of the intermittent connection with the network as well as the necessity to effectively schedule the data packets and choose the best transmission path. These problems are also aggravated by the frequent network cut-offs. This paper is aimed at solving these problems by suggesting a new communication approach which is coalition building between network nodes. In this work, utility functions are developed that reflect three important elements of the performance of DTN capacity, cost, and congestion. Cost-based utility model uses a variety of parameters such as the connectivity status, availability of the gateway, distance of transmission and overloading of the node. To further improve the efficiency of routing, we propose a stochastic game-theoretic model which allows making adaptive and intelligent decisions when forwarding packets. Also, a congestion control scheme is formulated, with a special utility function in a game-theoretic model. Experiments show that the suggested method is a lot more efficient than the current routing protocols. In particular, the approach attains an average overhead of 52, as compared with 77 in the case of Plague, 66.5 in the case of PROPHET and 60.75 in the case of Schedule-PROPHET, which shows that the network efficiency has been significantly improved.

Corresponding Author:

Shyamala G
Associate Professor, Department of CSE, BMS College of Engineering, Bengaluru, India.
Email: shyamala.cse@bmsce.ac.in

1. INTRODUCTION

Delay Tolerant Networks (DTNs) are a type of communication network that is created to facilitate data transfer in such an environment where it is impossible to provide continuous connectivity. It is through these networks that communication between mobile nodes is made possible through the use of a store-carry-forward mechanism in which the data packets are temporarily stored and carried by the nodes until a favorable forwarding opportunity occurs.

Applications of DTNs include opportunistic data sharing, mobile data offloading, satellite communications and disaster recovery. They can be especially useful in the dynamic setting such as stadiums, shopping malls, and airports where many mobile users interact, and sometimes disconnection is a typical phenomenon [1].

Although there has been a tremendous progress in the research of DTN, the optimal network performance is still a complicated issue. The correct routing or more precisely the selection of the optimum route during data transmission with minimum delay and overhead is one of the determinants of overall performance. Additionally, it is important to have consistent communication channels since any disruption may result in packet loss and higher latency. Although, the security issue has been considered in previous studies, this study aims at enhancing the performance measures like delay, channel capacity, communication delay, and reduction of overheads [2].

One of the main complexities in DTNs is due to the actions of individual nodes. These nodes will tend to be rational and self-interested agents that will pursue their own goals like saving energy. Consequently, one node might not forward the information of others, thus, creating ruined communication routes as well as compromised network performance [3]. With this egoistic approach, it is hard to maintain cooperative communication.

Game theory is an effective technique to model interactions between nodes to deal with this problem. The model of rational players in a strategic game allows the design of incentive mechanisms that will encourage cooperation by treating each node as a rational player in the game. Game theory assists in creation of equilibrium tactics where the nodes will behave in a manner that will not only favor them but also enhance the overall network performance.

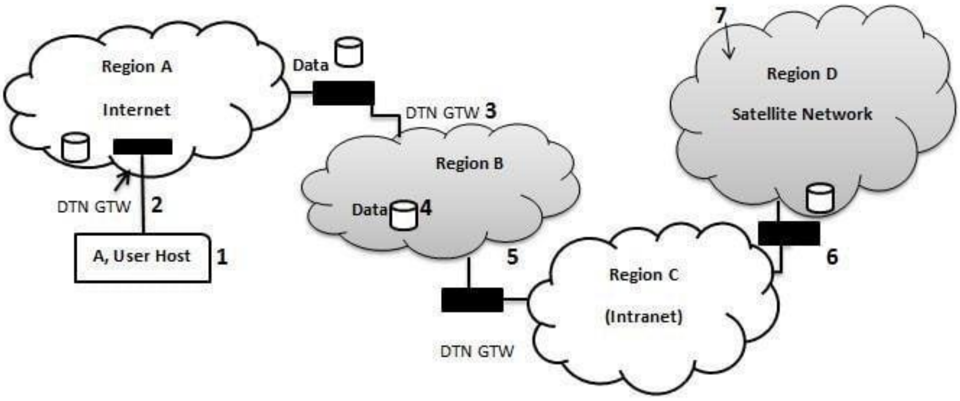

Fig. 1. General architecture of DTNs

Game theory has continued to receive more and more interest in many fields of science as a viable tool of decision making in competitive and uncertain situations. Here, the participants (or players) have chosen a strategy that affects the overall result and the payoff of the outcome is linked to each of the possibilities. The aim of every player is to maximize the payoff with other players also acting rationally [4].

It should not be overlooked though that the game-theoretic models are founded on rules and assumptions that are not always very realistic in capturing the complexities in reality. Although this is a limitation, game theory is still a useful model especially where the classical mathematical models are not applicable, as a result of uncertainty about the behavior of other players in the game. It has been effectively applied in networking applications in solving issues to do with routing, allocation of resources and collaboration amongst distributed nodes [5].

Here, we concentrate on two-hop routing scheme in DTNs and offer a framework which uses rewards to motivate nodes to cooperate. Here, the source node pays incentives to the possible relay nodes which then determine whether to forward the message. The success of a relay is affected by aspects like the number of nodes that already have the message and anticipated benefit of collaboration.

Extending this concept, we come up with a generalized routing approach which combines techniques of coalition formation and game theory to solve the connectivity and routing problems in DTNs. This work has made major contributions as follows:

- Creation of a communication structure that is based on coalition to enhance connectivity.
- Utility function designed in terms of capacity, cost and congestion parameters.
- Implementation of a parametric cost model using network conditions like distance, availability of gateway and node load.
- Incorporation of a stochastic game model in order to optimize transmission of packets.
- Demonstration of a congestion-aware routing scheme based on game-theory.

2. LITERATURE REVIEW

Mao et al. [6] explored the use of the Internet of Things (IoT) in untrustworthy and dynamic mobile wireless systems, specifically in Delay Tolerant Networks (DTNs). They suggested an improved routing mechanism which is based on the Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) to address routing issues in such networks. Their design enhances the message delivery rates with low overhead and decreases the delay in transmissions, which is appropriate in the IoT-enabled DTN applications.

Rahimi et al. [7] were interested in enhancing the routing performance through reducing the probability of choosing non-optimal relay nodes. They proposed a fuzzy logic method of determining the forwarding capability of the neighboring nodes. The approach selects the most appropriate node to forward to by assigning a probability score to each node according to various criteria, thus improving on routing decisions by using a greedy selection strategy.

Wang et al. [8] analyzed DTNs with dynamic topology transformation, small buffer capacity and high mobility of nodes. Although multi-copy routing techniques have been suggested to enhance the performance of delivery, it resulted in excessive use of buffers and congestion. To address this problem, the authors proposed a weight-based buffer management approach which groups messages into priority queues, i.e., urgent messages, high-weight messages, and low-weight messages so that the node resources are utilized more effectively.

In a different study, Wang et al. [9] investigated data dissemination approaches in Vehicular Sensor Networks (VSNs) which have the problems of high mobility and frequent disconnections. They suggested the Encounter Utility Rank (EUR) routing scheme that combines three important strategies: utility-based replication, lifetime-based replication and social rank-based replication. Using the encounter data of the past and patterns of social interactions, the protocol enhances the forwarding decision and enhances the overall network performance.

Wang et al. [10] also emphasized the relevance of DTNs in the difficult conditions like battlefields where the electromagnetic noise, infrastructure absence and node mobility generate significant communication challenges. They came up with a mobility-aware scheduling model to deal with these challenges called Aggregation and Spray Mobility Model (ASMM). The model maximizes message replication by taking into account limited copies of data and biased allocation that allow making more efficient routing choices under restricted circumstances.

Zekkori et al. [11] surveyed the combination of IoT and DTN routing protocols and discovered the main issues, such as the lack of connectivity and the inefficient routing of data. In order to solve these problems, they came up with a Quality of Service (QoS)-sensitive DTN routing protocol which integrates the merits of both flooding and forwarding techniques. This hybrid solution enhances the reliability of delivering data and the efficient communication between heterogeneous IoT devices.

Guo et al. [12] proposed a routing algorithm known as Location-Aided Controlled Spraying (LACS) in order to improve routing in DTNs. The proposed scheme has two stages: a single-copy forwarding stage and a controlled spraying stage. The single-copy phase applies a semi-Markov process (SMP) to capture both time and space properties, and the spraying phase can adjust replication strategies dynamically according to the recent node encounters, thus enhancing efficiency in delivering.

The routing protocol suggested by Wu et al. [13] was a vehicular DTN, and it was oriented on the Anycast communication between vehicles and cloud infrastructure. They apply a Q-learning-based algorithm to determine the probability of multi-hop delivery and use an adaptive data-replication mechanism. This combination allows to achieve high data delivery rates and reduce network overhead, and is suitable in intelligent transportation systems.

Table 1. Comparative analysis of recent works in the research field

Author	Method/Protocol	Key Technique	Strengths	Limitations
Mao et al. [6]	PRoPHET-based Scheduling Routing	Probabilistic routing using encounter history and transitivity	Improved delivery ratio, reduced delay, low overhead	Performance depends on accurate encounter history; less effective in highly dynamic environments

Rahimi et al. [7]	Fuzzy Logic-based Routing	Node selection using fuzzy inference for forwarding probability	Better decision-making, reduces irrational node selection	Increased computational complexity; requires parameter tuning
Wang et al. [8]	Weight-based Buffer Management	Priority-based message queue (urgent, high, low weight)	Efficient buffer utilization, reduced congestion	Does not fully eliminate buffer overflow under heavy traffic
Wang et al. [9]	Encounter Utility Rank (EUR)	Utility, lifetime, and social rank-based replication strategies	Improved delivery using social and historical data	High overhead due to multiple replication strategies
Wang et al. [10]	ASMM Routing Model	Mobility-aware scheduling with limited data copies	Efficient data distribution in dynamic environments	Requires accurate mobility modeling; complex implementation
Zekkori et al. [11]	QoS-based DTN Routing	Hybrid flooding and forwarding strategy	Improved QoS, better reliability in heterogeneous networks	Increased overhead due to partial flooding
Guo et al. [12]	LACS Routing Algorithm	Location-based controlled spraying with SMP modeling	Efficient use of spatial-temporal information	Dependency on location accuracy; higher computation cost
Wu et al. [13]	Q-learning-based Routing	Reinforcement learning for multi-hop delivery prediction	Adaptive routing, high delivery ratio, low overhead	Requires training time; performance depends on learning convergence

3. METHODOLOGY

3.1 System Overview

This section outlines the proposed solution for improving DTN communication efficiency. We consider a DTN model in which a single destination is served as s_i, one destination d_i and n relay nodes are present. Wireless networking interfaces are mounted on relay nodes, allowing them to connect with other mobile nodes. The message that requires to be transmitted to the destination is generated by the source node. The node distribution is sparse and isolated is taken into account. In this situation, communication takes place. When two nodes are in contact range with one another. According to the suggested scheme, if a source node produces a message, a general incentive mechanism is enabled to create a packet distribution competition for the duration of the message. We form coalitions out of several parties C_j, to produce the packet, each alliance competes with the others. The source node makes multiple copies of the packet using relays in an effort to deliver it. Every copy includes a number of different facts about the letter, such as a time stamp that indicates the message's age, after which the message can be discarded. Since constant communication between nodes cannot be assured, we remove the feedback process that allows relays and source nodes to recognize the distribution information about messages in these networks.

The ρ_l denotes the cost of electricity for the j^{th} in a given active time interval as a relay node $[0, \tau]$. The probability of efficient message delivery during this activity's duration can be seen as:

$$\delta^{s}_{success}(|n^*|) = 1 - \prod_{k \in \{n\}} Q^k_\tau \tag{1}$$

Where n^* denotes the relay nodes that are currently involved, Q^k_τ iis the likelihood of a node failing to send a packet to its intended destination, $1 - Q^k_\tau$ is the probability of a node successfully delivering a duplicate of a message to its destination within the defined time frame τ. The Q^k_τ relies on the distribution of intervals λ, given as:

$$Q^k_\tau = \lambda \tau e^{-\lambda \tau} \tag{2}$$

Now, the source node assigns a reward r for relay nodes which are denoted as a certain number and can be used by nodes to transmit their own message. Along with this, a reward r_l is also created, which is determined by the form of network device present. Due to the energy-saving process, nodes with increased battery capacity are chosen to receive the reward, while nodes with lower battery capacity will be excluded from the reward collection. Assuming that the source node will identify node types, which will help us determine the reward category based on the system type l.

Then once the message is sent either from the source node or the relay node, neighboring nodes make a coalition group and work together to send the message to the destination node. This procedure aids in the distribution of packets and the efficient use of electricity. As a result, we present a model for forming coalitions to reduce energy usage, congestion, and packet distribution.

3.2 Proposed Coalition Process for DTN

For n relay nodes, we show a suggested solution based on coalition formula. To improve packet distribution, we think of energy that is heterogeneous costs and control in this phase. An alliance game is what we call it G in the form of pair (N, U) where N denotes the set of players, U is the coalition value and empty subset $C \subseteq N$ is known as coalition. The role of coalition in the game under consideration is determined by the coalition meaning. The primary goal of these strategies is to determine which coalition to meet in order to optimize results. The payoff function is defined as $U(C_j, \pi_i)$. The coalition system is made up of different coalitions made up of various players as $\Pi = \{\pi_1, \pi_2, \pi_3, \dots \pi_n\}$. These coalitions cannot always succeed in achieving the network's desired cooperation. As a result, any coalition's chances of success are slim C_j can be calculated based on the performance. It's described as:

$$P_{succes}^{C_j,\pi_i} = \left(1 - \prod_{i \in C_j} Q_i\right) . \prod_{k \in B} \left(1 - P_{success}^{C_k}\right) \tag{3}$$

We describe the wasted energy based on this probability calculation $\rho_j^{C_i,\pi_i}$ to receive and send data packets in accordance with $P_{succes}^{C_j,\pi_i}$ when the message is route to its intended recipient. This can be mentioned as follows:

$$\rho_l^{C_j,\pi_i} = \begin{cases} E_r + E_w.\tau \quad if\ T > \tau \\ E_r + E_{tran}.P_{succes}^{C_j,\pi_i} + E_w.T \quad if\ T \le \tau \end{cases} \tag{4}$$

We let homogeneous and heterogeneous types of nodes, which can be assured as utility function, since DTNs generally consist of various types of nodes such as cars, cell phones, laptops, and so on. The utility function for homogeneous devices (those with the same transmission technology) can be written as:

$$U(C_j, \pi_i) = r.P_{succes}^{C_j,\pi_i} - \rho^{C_j,\pi_i} \tag{5}$$

On the other hand, we imagine a heterogeneous scenario in which we take into account M classes of various types of instruments in the DTN. Each device class has its own set of rules that contains N_l number of relay nodes as $N = \sum_l^N N_l$. The function can be denoted as:

$$U(C_j, \pi_i) = r_l.P_{succes}^{C_j,\pi_i} - \rho_l^{C_j,\pi_i} \tag{6}$$

Moreover, we find a congestion control mechanism and present a utility feature for a congestion control game model. We presume the contact connection is intact l_{ij} between node i and j nodes are within contact range, a link can be created. The congestion game can be defined as follows:

$$G = \{N, N_r, (A_i)_{i \in N}, (C_i)_{i \in N}\} \tag{7}$$

Where N is the amount of nodes, $N_r = N \cup N_g$denotes the node source and gateways N_g, A_i is an action chosen by the player to pick a forwarder is denoted as $A_i = \{j \in N_r, i \neq j\ , dist(i,j) \le R\}$and C denotes the cost function. The following cost parameters are used to estimate the cost of each player:

- Predicting the communication connection between two nodes is the first cost factor n_i and n_j which is given as:

$$C_i^1 = f(\tau_{ij}) = \frac{\tau_{max} - \tau_{ij}}{\tau_{max}} \tag{8}$$

- The selection among the considered neighbors is the second factor to consider which is given as:

$$C_i^2 = \begin{cases} 0, & if\ j\ is\ a\ gateway \\ C_0, & otherwise \end{cases} \tag{9}$$

- Third parameter is to determine the range between interacting nodes as follows:

$$C_i^3 = g(d_{ij}) = \begin{cases} 0, & if\ j\ is\ a\ gateway \\ \frac{d_{ij}}{R}, & otherwise \end{cases} \tag{10}$$

- Fourth factor is when the same node is chosen by another adjacent node, this is referred to as node overloading:

$$C_i^4 = h(n_j) = \frac{n_j}{n} \tag{11}$$

The total amount for each player in the specified time slot can be calculated using these parameters τ can be expressed as:

$$C_i(a_i, a_{-i} = \alpha C_i^1(a_i) + \beta C_{-i}^2 + \gamma C_i^3(a_i) + \delta C_i^4(a_i, a_{-i}) \tag{12}$$

In order to further improve the performance, we present a stochastic game model to adopt the efficient packet transmission.

This game will be studied using a discrete-time model. The message is generated by the source at constant 0 with a deadline of instant $\tau + 1$. It is presumed that receiving the message from the main source and transmitting it to the destination each require one time, requiring a relay to arrive at the destination ahead of schedule τ in order to get the point. When a relay comes into contact with a source, it has the option of accepting or rejecting the message. If the message has been accepted by the relay, it can opt to keep it or place it in each time slot after it reaches its destination or the message's deadline has passed. As a consequence, the set includes the possible decision epochs for each relay $\{0, 1 \dots \tau - 1\}$. Each relay must make several decisions over stages, and the cost is determined by its own reactions as well as the actions of the other. Each goal is to keep the anticipated cost of playing in the game to a minimum. Each relay our model is conscious of its own state but is unaware of the state of the others. Furthermore, it has no way of knowing whether or not the packet has been sent to the intended destination. As a consequence, our game is a partial-information stochastic game. We'll now go through some context details on these types of games. These games are characterized by:

- τ: time
- $\mathcal{R} = \{1,2, \dots N\}$ set of players
- $\mathcal{E}_j, j \in \mathcal{R}$ state space of relay j. We expressed by X_j^n the capacity of player j at time n.
- $\mathcal{A}_j, j \in \mathcal{R}$ action of relay j. We expressed by A_j^n the action did by player j at time n.
- $\mathcal{E} := \bigotimes_{j \in \mathcal{R}} \mathcal{E}_j.$
- $\mathcal{A} := \bigotimes_{j \in \mathcal{R}} \mathcal{A}_j.$
- $\mathcal{B}_j: \mathcal{E}_j \times \{0,1, \dots, \tau - 1\} \to \mathcal{D}(\mathcal{A}_j)$where $\mathcal{D}(\mathcal{A})$ is a list of probability measures based on $\mathcal{A}$. The set $B_j\ (t)$ is the list of some strategies available to relay j at every time instants. In other words, the element $\sigma_n^j(x)$ is the set of acts' probability distribution $\mathcal{A}_j$ used by j to put its action when it is in x at time n.
- $P_j, j \in \mathcal{R}$ on the space of its state-action pairs, the transfer probability matrix of relay j.
- $\mathcal{E}_0$: state space of the packet. This value may be 0 or 1, indicating whether or not the packet has been sent. We use the word by X_n^0 the state of the time n.

- $g_j : \mathcal{E}_j \times \mathcal{A}_j \times \mathcal{E}_0 \rightarrow \mathbb{R}, j \in \mathcal{R}$: cost function for relay j.

Initially, it is a partially observable stochastic game since each relay only knows its own state and not the states of the others. Since there isn't enough detail, the concepts of Markov strategies and Markov equilibrium aren't relevant in this situation. The action of a relay in a given state is determined by the state of other relays that this relay is unaware of. The probability distribution over another relays' states will be determined by their previous behavior. This implies that to measure its own behavior in a given state, a relay would need to keep track of the previous actions of others. Since the behavior in the state is based on the state of the other, which is unknown, the likelihood of arriving in that state is determined by previous acts.

A policy σ is said to be an equilibrium if

$$\beta_j\left(\sigma^{-j}; x_0^0, b_{-j}\right) = \sigma^j, \forall_j \tag{13}$$

The following are the values for various parameters in our model.

- **State and action spaces**

Every relay's state can take one of five different values:

Table 2. State and action set

Value	Significance	Action Set
0	Relay that does not have the packet	ϕ
$\boldsymbol{m_s}$	Relay meets the source	$(accept, reject)$
1	Relay has the said packet	$(drop, keep)$
$\boldsymbol{m_d}$	Relay meets the destination	ϕ
2	Relay then quits the game	ϕ

The relay has no non-trivial operation in states 0 and 2. It is in state 0 as it awaits the source, and state two as it has already exited the game.

- **Transition matrix**

We'll use i.i.d. contact times for the contact phase, which keeps track of the relay's contacts. As a result, in order to make a decision, a relay just needs to be aware of the current situation of the communication mechanism. We will see what happens next p be the likelihood that a relay will arrive at its destination at the next time stage, and q be the likelihood of it colliding with the source Each relay's state is determined by a Markov chain of time-homogeneous transfer probabilities are determined by the action taken in each state, and is provided by:

$$P_j = \begin{array}{c} \\ 0 \\ m_s \\ m_d \\ m_d \\ 2 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & m_s & 1 & m_d & 2 \end{array} \\ \begin{bmatrix} 1-q & q & 0 & 0 & 0 \\ \mathbb{1}_{\text{reject}} & 0 & \mathbb{1}_{\text{accept}} & 0 & 0 \\ \mathbb{1}_{\text{drop}} & 0 & (1-p)\mathbb{1}_{\text{keep}} & p\mathbb{1}_{\text{keep}} & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \tag{14}$$

Below figure 2 depicts the Markov chain's transformation diagram.

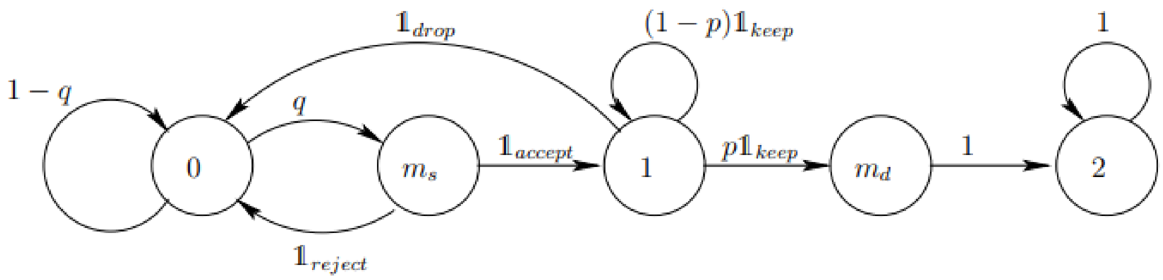

Figure 2. Transition diagram for the Markov chain.

- **State of the packet**

The packet's condition can be either 0 (not been delivered) or 1 (been delivered). The probability of switching between these two is determined by the relays' current state.

$$P(X_{n+1}^0 = 1 \mid X_n^0 = 0, \mathbf{X}_n) = P((\cup_{j\in\mathcal{R}}\{X_n^j = 2\} = \emptyset) \cap (\cup_{j\in\mathcal{R}}\{X_n^j = m_d\} \neq \emptyset)),$$
$$P(X_{n+1}^0 = 1 \mid X_n^0 = 1, \mathbf{X}_n) = 1. \quad (15)$$

- **Cost function**

The relay's one-step cost is determined by its state and the step it takes. Furthermore, when it arrives at its destination (state m_d) expense is determined by whether or not another relay has already shipped the packet. As a result, for all other arguments, and is 0.

$$g(m_s, accept, \ .) = C_r$$
$$g(1, keep, \quad .) = C_s \quad (16)$$
$$g(m_d, \quad ., \quad 0) = R - C_d$$

3.3 Game with Two Relays

Now we'll look at the network with two relays. We'll focus on threshold laws, such as those enacted by the federal government such $\sigma_n^j(m_s) = accept\ if\ n \leq \theta_1$ and reject $otherwise$, and $\sigma_n^j(1) = drop\ if\ n \geq \theta_2$ and keep $otherwise$. The threshold θ_2 possible that it'll be determined by the time of the source's meeting. We will demonstrate that if one relay uses a threshold policy, the other relay's best-response is also a threshold type policy.

As a consequence, we'll presume that one of the two relays, for instance let relay 2, works under a threshold policy. That is to say, they exist θ_1^2 and $\theta_2^2 > \theta_1^2$:

$$(\sigma_n^2(m_s) = \begin{cases} \text{accept} & \text{if } n \leq \theta_1^2, \\ \text{reject} & \text{if } n > \theta_1^2, \end{cases} \quad (17)$$

And

$$(\sigma_n^2(1) = \begin{cases} \text{keep} & \text{if } n \leq \theta_2^2, \\ \text{drop} & \text{if } n > \theta_2^2, \end{cases} \quad (18)$$

4. RESULTS AND DISCUSSION

We go through the entire experimental setup, the results for different conditions, and a comparative review to show how the proposed solution outperforms the competition. We use several parameters to assess the efficiency of the proposed solution, as shown in Table 3 below.

Table 3. Simulation parameter for game theory simulation

Parameter Name	Parameter Value
Simulation City	Helsinki city
Simulation Area	4500mx3400m
Pedestrians	0.5–1.5 m/s (moving speed)
Automobiles	2.7–13.9 m/s (moving speed)
Trams	7–10 m/s
Transmission range	10 m
Buffer size	2-20 MB
Message size	600-1000 KB
Number of copies	8

The proposed simulation is carried out with the help of ONE, a Java-based simulator that is open source. To implement the routing protocols, we can use this method to produce different real-world traces and different forms of movement models. It also accommodates a number of mobility situations. Automobiles travel along the shortest road, while trams travel along the Route Map Movement model. Pedestrians, cars, and trams ride at speeds of 0.6–1.5 m/s, 2.8–13.9 m/s, and 7–10 m/s, respectively. The delivery pace, average delay, and overhead are all variables to consider are all factors that we consider when evaluating results. The distribution rate is calculated as the percentage of delivered packets divided by total packets: $Delivery\ Rate = \frac{N_{delivered}}{N_{Total\ Packets}}$, the total delay is the total time it takes for a packet to arrive at its destination, and it can be calculated as follows: $average\ delay = \frac{\sum_{i=1}^{N_{delivered}} delay_i}{N_{delivered}}$ and the it is measured as the ratio of relayed packets to delivered packets as $overhead = \frac{N_{relayed} - N_{delivered}}{N_{delivered}}$. By changing the transmission range, amount of nodes, and buffer size, you can create a unique experience, we can assess efficiency.

4.1. Varying the Transmission Range

In this part, we evaluate the suggested approach's efficiency by depending on the transmission range from 4 to 10 meters. The buffer area for pedestrians and cars is 5 meters, and for trams it is 50 meters. Figure 3 depicts the relative success in terms of delivery pace.

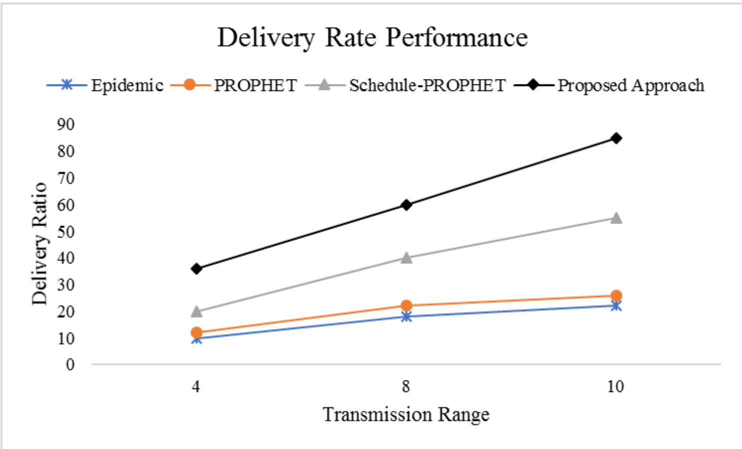

Figure 3. Delivery rate performance

We contrasted the output of plague, PROPHET, and Schedule-PROPHET with transmission ranges of 4-10m in this experiment. Using Disease, PROPHET, Schedule-PROPHET, and Suggested Approach, the average packet transmission rate is 16.6, 20, 38.33, and 60.33.

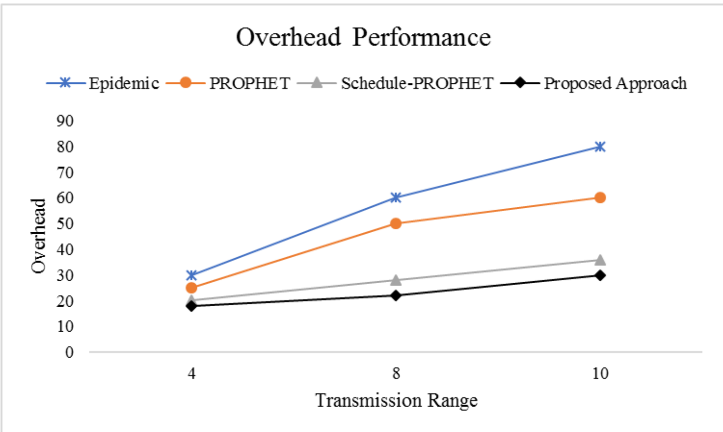

Figure 4. Overhead performance

In the same way, we evaluated overhead efficiency, as shown in figure 4. Using the three performance, we get average overhead output of 56.66, 45, 28, and 23.33. This experiment demonstrates that the suggested model performs better over a large transmission range. The distribution rate increases as the transmission distance is expanded.

4.2. Varying the Number of Nodes

We examine a total of 650 nodes in this case, which are distributed randomly in a 2D geographical region. For Pedestrians, Automobiles, and Trams, we grant buffer space of 5M, 5M, and 50M, respectively. The transmission distance is set to 10 meters, and the message interval is set to 5 seconds. Figure 5 depicts the delivery output for a variety of node counts.

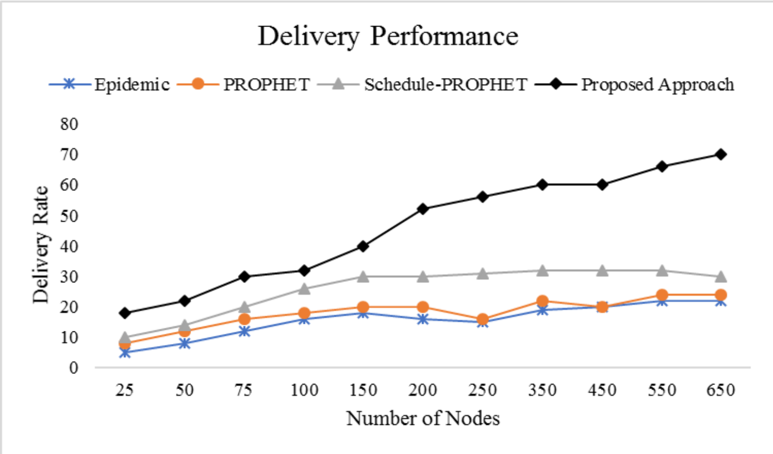

Figure 5. Delivery performance for varied number of nodes

Using Epidemic, PROPHET, the total delivery rate is 15.72, 18.18, 26.09, and 46. The PROPHET Schedule and Proposed Approach is used to assess the efficiency of the same experimental in terms of delay. The comparative output for this experiment is shown in Figure 6.

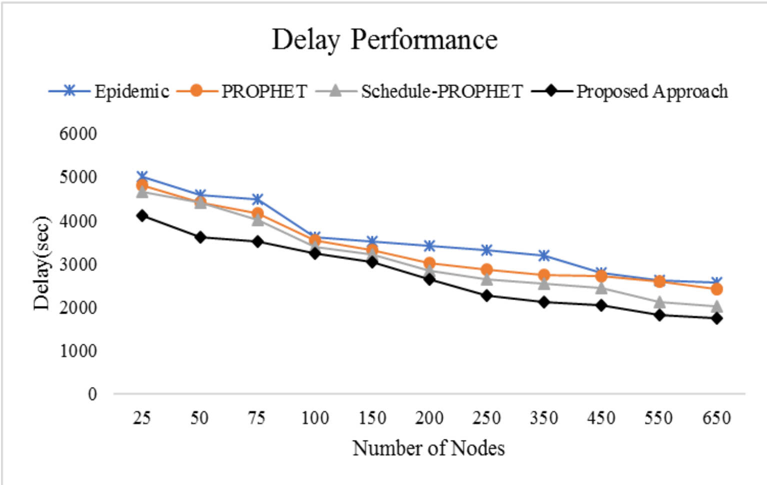

Figure 6. Delay performance for varied number of nodes.

The packet delay is higher in low-density node distributions due to congestion, while as the number of nodes rises, the packet transmission delay decreases. Using the three approach performance, we obtained average delays of 3553.27, 3321.72, 3118.18, and 2745.54, respectively.

4.3. Varying the Buffer Size

With this, we calculate output in terms of total arrival, average latency, and overhead by varying the buffer size. We'll use 100 nodes in this experiment, and the packet generation time will be set to 5 seconds. Since a small buffer may only accommodate a certain number of packets, incoming packets can be lost due to some memory problems in the network buffer. Table 4 displays the buffer assignments for various node groups.

Table 4. Buffer assignment for different types of nodes

Scenarios	Pedestrians	Automobiles	Trams
S1	2M	2M	10M
S2	8M	8M	40M
S3	12M	12M	60M
S4	18M	18M	90M

Below given figure 7 depicts the delivery output for different buffer counts. Using the three performances approach, the average performance is 28.25, 36.5, 41.5, and 48.5.

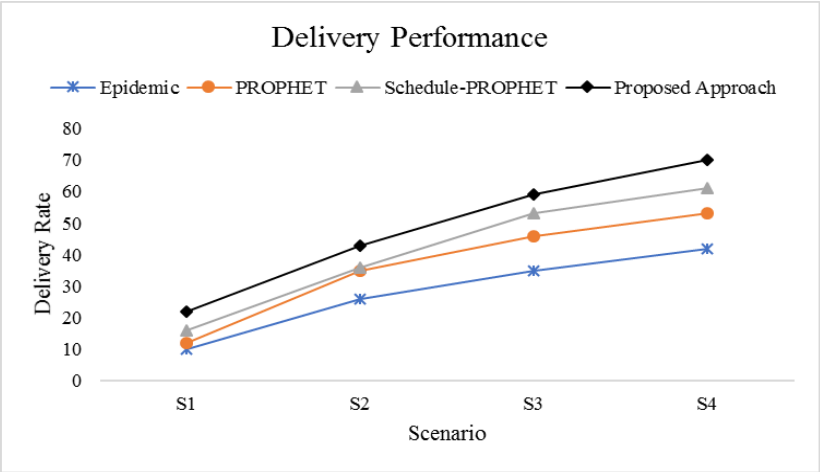

Figure 7. Delivery Performance

Figure 8 depicts a similar comparative study when it comes to average delay. The packet delivery speed was increased as the amount of buffers increases as well, resulting in a reduction in packet delivery delay.

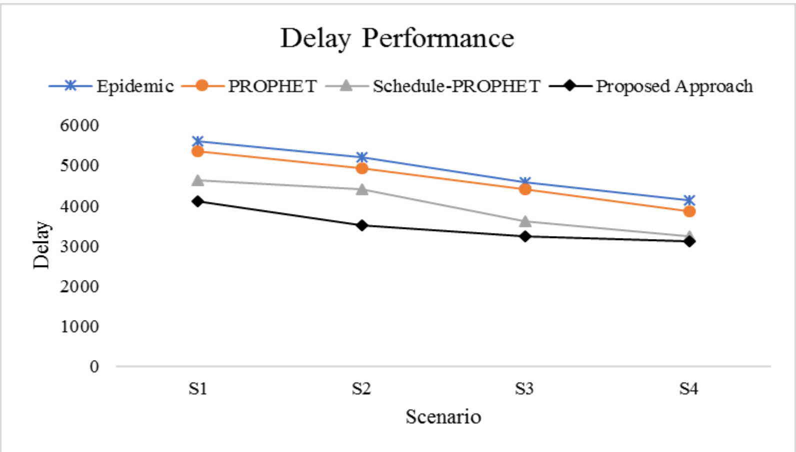

Figure 8. Delay performance

The average output for Outbreak, PROPHET, Schedule-PROPHET, and Proposed Solution is 4877.5, 4630, 3975, and 3491.25, respectively. We also test the overhead efficiency for different buffer sizes.

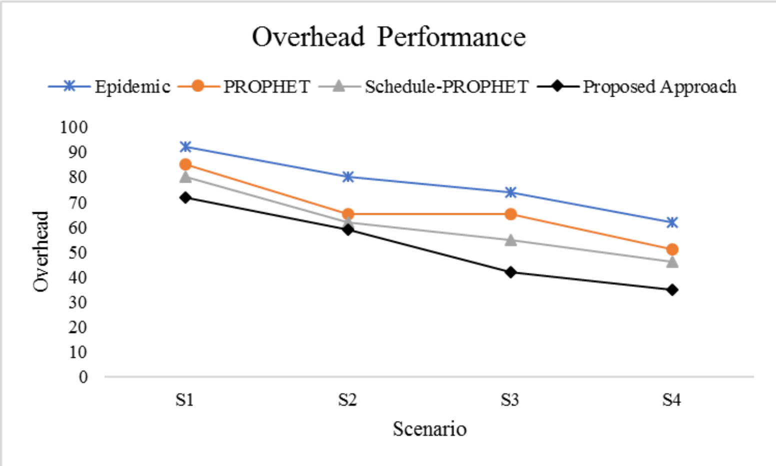

Figure 9. Network overhead performance

The comparative study in terms of network overhead for four different scenarios is shown in Figure 9. The proposed method outperforms other approaches by achieving an average overhead performance of 52, while Plague, PROPHET, and Schedule-PROPHET achieve 77, 66.5, and 60.75, respectively, for their average overhead performance.

5. CONCLUSION

This paper has provided an extensive overview of the routing strategies in Delay Tolerant Networks (DTNs) and especially intelligent and adaptive strategies that are based on game-theoretic principles. The environments that DTNs work in are very dynamic and intermittently connected and efficient routing becomes a challenging endeavor. The literature demonstrates that conventional routing algorithms, such as probabilistic, fuzzy logic-based, and replication-based routing algorithms have greatly enhanced the rates of messages delivery and minimized delays. Nevertheless, they have a tendency of being limited by a large overhead, un-efficient use of resources, and lack flexibility to quickly changing network environments. The comparative study shows that approaches such as PRoPHET, fuzzy-based routing, and Q-learning models have been effective in decision-making, but they are yet to achieve trade-offs between delivery performance, congestion control and computational complexity. Furthermore, selfish node behavior, the capacity of the buffer, and unpredictable mobility patterns are other problems which enhance the challenge of reliable communication in DTNs. Game theory incorporation of routing design is one such promising direction to overcome these challenges. The approach to making nodes rational agents and integrating utility-based decision models makes possible the promotion of cooperation, better resource allocation, and a more efficient overall network. Furthermore, the integration of various variables, including capacity, cost and congestion into joint utility functions facilitates better and adaptive routing decisions. To sum up, the next generation of DTN routing protocols must be concerned with hybrid, multi-objective optimization techniques which utilize game theory, machine learning, and context-awareness mechanisms to provide robust, scalable and efficient communication in real-world situations.

CONFLICT OF INTEREST STATEMENT

No conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] S. El Alaoui and B. Ramamurthy, “EAODR: A novel routing algorithm based on the modified temporal graph network model for DTN-based interplanetary networks,” *Computer Networks*, vol. 129, pp. 129–141, 2017, doi: 10.1016/j.comnet.2017.09.017.

[2] B. Kang, F. Malute, O. Bagdasar, and H. Choo, “DETN: Delay-efficient tolerant network for Internet of Planet,” *IEEE Sensors Journal*, vol. 20, no. 7, pp. 3696–3705, 2020, doi: 10.1109/JSEN.2019.2958612.

[3] V. Singh and G. L. Saini, “DTN-enabled routing protocols and their potential influence on vehicular ad hoc networks,” in *Soft Computing: Theories and Applications*. Singapore: Springer, 2018, pp. 367–375, doi: 10.1007/978-981-10-5699-4_36.

[4] P. Du and M. Gerla, “An evolutionary multi-player game model for two-hop routing in delay tolerant networks,” in *Proc. IEEE 14th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, 2017, pp. 108–116, doi: 10.1109/MASS.2017.22.

[5] F. Semsarieh and N. Derakhshanfard, “Relay node selection based on adaptive neuro-fuzzy inference system in delay tolerant networks,” *International Journal of Networked and Distributed Computing*, vol. 14, no. 1, p. 5, 2026, doi: 10.2991/ijndc.k.251230.001.

[6] Y. Mao, C. Zhou, Y. Ling, and J. Lloret, “An optimized probabilistic delay tolerant network (DTN) routing protocol based on scheduling mechanism for Internet of Things (IoT),” *Sensors*, vol. 19, no. 2, p. 243, 2019, doi: 10.3390/s19020243.

[7] S. Rahimi and M. A. J. Jamali, “A hybrid geographic-DTN routing protocol based on fuzzy logic in vehicular ad hoc networks,” *Peer-to-Peer Networking and Applications*, vol. 12, no. 1, pp. 88–101, 2019, doi: 10.1007/s12083-018-0642-4.

[8] H. Wang, H. Wang, G. Feng, and H. Lv, “NWBBMP: A novel weight-based buffer management policy for DTN routing protocols,” *Peer-to-Peer Networking and Applications*, vol. 11, no. 5, pp. 917–923, 2018, doi: 10.1007/s12083-017-0600-x.

[9] T. Wang, Y. Zhou, X. Wang, and Y. Cao, “A social-based DTN routing in cooperative vehicular sensor networks,” *International Journal of Cooperative Information Systems*, vol. 27, no. 1, p. 1741003, 2018, doi: 10.1142/S021884301741003X.

[10] H. Wang, L. Song, G. Zhang, and H. Chen, “Timetable-aware opportunistic DTN routing for vehicular communications in battlefield environments,” *Future Generation Computer Systems*, vol. 83, pp. 95–103, 2018, doi: 10.1016/j.future.2018.01.017.

[11] E. A. Abdellaoui Alaoui, H. Zekkori, and S. Agoujil, “Hybrid delay tolerant network routing protocol for heterogeneous networks,” *Journal of Network and Computer Applications*, vol. 148, p. 102456, 2019, doi: 10.1016/j.jnca.2019.102456.

[12] H. Guo, X. Wang, H. Cheng, and M. Huang, “A location aided controlled spraying routing algorithm for delay tolerant networks,” *Ad Hoc Networks*, vol. 66, pp. 16–25, 2017, doi: 10.1016/j.adhoc.2017.06.003.

[13] C. Wu, T. Yoshinaga, D. Bayar, and Y. Ji, “Learning for adaptive anycast in vehicular delay tolerant networks,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 4, pp. 1379–1388, 2019, doi: 10.1007/s12652-018-0725-1.