Message Matching-based Artificial Bee Colony based Behaviour Detection in Delay Tolerant Networks

Srimathi. M
PG Scholar
Department of CSE
Sri Ramakrishna Engineering College
Coimbatore

Karthigha. M
Assistant Professor
Department of CSE (PG)
Sri Ramakrishna Engineering College
Coimbatore

ABSTRACT

Delay Tolerant Networks (DTNs) is a promising current research field that have captured a lot of attention. In DTN, an end-to-end path is not assured and packets are transmitted from a source node to a destination node by the use of store-carry-forward based routing. In case of DTN, a source node or an intermediary node accumulates the packets in its buffer and holds them at the time of moving around in the network. In the majority of routing protocols, each node is necessary to truthfully pass on information. On the other hand, in certain scenarios few nodes possibly will break this principle, and pass on information in a greedy manner with the intention of maximizing their individual gain. The majority of the current security mechanism doesn't depend on the practice of swarm intelligence based process for selection of untrusted nodes and recognizes the behaviour of illegal nodes. With the aim of solving this complication, this work employs the practice of swarm intelligence method, an Artificial Bee Colony (ABC) based greedy behaviour, in order to match the messages where a smart Mobile Trusted Module (MTM) is launched to supervise the forwarding progression of messages buffered in the node for the purpose of preventing the greedy behaviour. The performance of the proposed protocol was assessed in terms of average relay time, average message delivery ratio and reward ratio. Results confirm that the proposed protocol can onsiderably enhance the average message delivery ratio, diminished delay time and condensed packet overhead ratio by properly selecting the threshold values of forwarding counter and threshold hop counter.

Keywords:

Delay tolerant networks, Greedy behaviour, Mobile trusted module, Artificial Bee Colony (ABC)

1. INTRODUCTION

As an emerging research area, delay tolerant networks (DTNs) have attracted a lot of attention. In a DTN, a continuous end-to-end connection from a source to a destination is not available because the network is frequently partitioned, and thus messages are relayed in a store–carry–forward fashion. DTNs are usually used to implement communications in some extremely challenging environments where traditional networks may not work, such as interplanetary internet (IPN) [2], vehicular disruption-tolerant networks, US navy seaweb [3], and sparse ad hoc networks.

In delay tolerant network (DTN), however, an end-to-end path is not guaranteed and packets are delivered from a source node to a destination node via store-carry forward based routing [3-5]. In DTN, a source node or an intermediate node stores packets in buffer and carries them while it moves around. These packets are forwarded to other nodes based on predefined criteria and finally are delivered to a destination node via multiple hops. A lot of attention has been paid to DTN for possible uses in disconnected network environments, especially for extreme cases such as interplanetary communications [3] and disaster scenarios [6]. One of the representative DTN routing protocols is epidemic routing protocol [7]. As the name implies, a source node forwards a message to all the neighbour nodes whenever it contacts neighbour nodes, like "epidemic."

This simple routing protocol is very powerful even when the buffer size of nodes is sufficient. However, if the buffer size is not sufficient, especially as in mobile nodes, epidemic protocol generates significant message overhead and the performance degrades. In order to solve the message overhead problem of the epidemic protocol, several schemes have been proposed, such as Spray and Wait protocol [8] and PRoPHET Protocol [9]. In these protocols, the total number of message copies present in a network is limited by a certain number or message forwarding is carried out only when a certain condition is met.

In most routing protocols of DTNs, each node is required to honestly carry out a specific routing protocol. Originating from the greedy nature and potential benefit, some nodes are likely to relay messages in a way that maximizes their own benefit. For example, greedy based artificial nee colony algorithm the bee nodes may give a message forwarding priority to each other. The greedy bee behaviour impairs the benefit of honest nodes, and undermines the routing fairness of DTNs. Furthermore, this misbehavior decreases the average message delivery ratio of the network. This paper a message matching schema is performed based on the behaviour of bees in the ABC algorithm for DTNs differs from selfish behaviours where selfish nodes are unwilling to cooperate with members in a DTN. The contribution of this paper is summarized as follows:

1. Introduce a trusted sublayer containing a smart mobile trusted module to the trusted bundle layer model in order to enhance the security of the DTNs architecture.

2. To simplify the ABC based detection and reduce the resources overhead in a DTN node, a message matching-based detection method built on the new trusted bundle layer model is proposed. Experimentation work discusses the effect of these misbehaviours on the routing fairness and the average message delivery ratio of the network.

The remainder of this paper is organized as follows. In Section 2, related works are surveyed. In Section 3, the detailed algorithm of the proposed Message matching-based artificial bee colony based behaviour is described. In Section 4, numerical examples are presented using simulation from the aspect of average message delivery ratio, reduced delay time and reduced packet overhead ratio. Finally, conclusions and further works are drawn in Section 5.

2. RELATED WORK

Li et.al [10] says that Nodes in disruption-tolerant networks (DTNs) usually exhibit repetitive motions. Several recently proposed DTN routing algorithms have utilized the DTNs' cyclic properties for predicting future forwarding. The prediction is based on metrics abstracted from nodes' contact history. However, the robustness of the encounter prediction becomes vital for DTN routing since malicious nodes can provide forged metrics or follow sophisticated mobility patterns to attract packets and gain a significant advantage in encounter prediction. Here it examines the impact of the black hole attack and its variations in DTN routing. And introduce the concept of encounter tickets to secure the evidence of each contact. The scheme is, nodes adopt a unique way of interpreting the contact history by making observations based on the collected encounter tickets. Then, following the Dempster-Shafer theory, nodes form trust and confidence opinions towards the competency of each encountered forwarding node [10].

To handle misbehaving nodes, reputation-based schemes were first considered by previous studies [11, 12]. Marti et al. proposed a dynamical measurement method [11] which introduces two kinds of special nodes, watchdogs and path rates. Watchdogs are used to identify misbehaving nodes, and the function of pathraters is to help routing protocols to avoid misbehaving nodes. Later, Buchegger and Boudec [12] presented a similar approach called CONFIDANT. Argue that these reputation-based schemes may not be suitable for the greedy behaviour because it is impractical to detect or identify greedy nodes in an extremely challenging environment. Furthermore, how to propagate the reputation of misbehaving nodes throughout a large delay network is still an open issue.

The credit-based incentive schemes [13] provide another solution, which adopt credits (virtual coins) to stimulate selfish nodes to cooperate rather than detect misbehaving nodes. The credit-based incentive schemes are similar to an electronic cash system, where a virtual bank (VB) takes charge of credits management and cash clearance. To compensate resources consumption, a source pays a certain number of credits to intermediates. If the credits of a source are not enough to send a message, the source has to get more credits by relaying the messages of other nodes.

Tit-for-tat based incentive schemes [14], where a node autonomously lowers the quality of service to misbehaving neighbours, and fully cooperates with honest neighbours. These mechanisms may temporarily isolate misbehaving nodes, but they still suffer a bootstrapping problem. In [15], Shevade et al. developed an incentive-aware routing protocol that incorporates ‘generosity’ and ‘contribution’ to address the above issue. However, this approach needs to generate a set of candidate paths from a source to a destination, and then approximates the message delivery ratio using the linear programming optimization. Unfortunately, it is difficult to determine a path before routing since DTNs follow the opportunistic routing.

In proactive routing, routes are computed automatically and independently of traffic arrivals. Most Internet standard routing protocols and some ad-hoc protocols such as DSDV (Destination Sequenced Distance Vector) and OLSR (Optimized Link-State Routing) are examples of this style [16]. In a DTN, these protocols are capable of computing routes for a connected sub graph of the overall DTN topology graph. They fail when asked to provide paths to nodes which are not currently reachable. Despite this drawback, proactive network-layer routing protocols may provide useful input to DTN routing algorithm by providing the set of currently-reachable nodes from which DTN routing may select preferred next hops.

MaxProp [17] attempts to transfer all messages not held by the other node, when it is in communication range. The protocol uses acknowledgments to clear the remaining copies of a message in the network when it is received by the destination node. When nodes discover each other, MaxProp exchanges messages in a specific priority order, taking into account message hop counts and the delivery likelihood to a destination based on previous encounters. New packets are assigned higher priority, and the protocol attempts to avoid reception of duplicate packets. In the Resource Allocation Protocol for Intentional DTN (RAPID) [18], routing packets are opportunistically replicated until a copy reaches the destination node. The protocol models DTN routing as a utility-driven resource allocation problem. The routing metric is a per-packet utility function. When nodes are in communication range, RAPID replicates the packet that results locally in the highest increase in utility.

In [19] and [20], studies of the effects of cooperation in DTNs are presented. The authors in both studies considered three routing protocols and studied the performance of these routing protocols in a non-cooperative environment, in terms of delivery delay and transmission overhead [20] and in terms of message delivery performance [19]. Both works are close to ours, but have considered more routing protocols, more metrics, and different types of misbehaviour.

3. PROPOSED ARTIFICIAL BEE COLONY BASED BEHAVIOUR DETECTION METHODOLOGY

First describe a network model, and then discuss ABC which may happen in DTNs. Formalize a DTN as a set of mobile devices held by individuals. Here, use a node N_i to denote a mobile device. Each node has constrained wireless network resources, such as limited buffer, restricted battery power, and low network bandwidth. Because of these constrained resources, the node is difficult to construct a continuous end-to-end connection. Source S transmits messages to a destination D through n intermediates after a large propagation delay. Additionally, to prevent unauthorized nodes from accessing the DTN, assume that there exists a management authority referred to as registration authority (RA). In this section, first propose a basic message matching-based detection (MMBD) method which aims to detect and inhibit the honey bee behaviour, and then reduce the computation and resources overhead in a DTN node. The work is based on a bilinear pairing over elliptic curves. Let G be an additive cyclic group, and G_T be a multiplicative cyclic group with the same prime order q. P is a generator of G. Assume that e is a bilinear map, e: G × G → G_T, which satisfies the following properties:

Bilinearity: For a, b ∈ Z_q, $e(a_P, b_P) = e(P,P)^{ab}$.

Non-degeneracy: $e(P, P) \neq 1G_T$.

Computability: There is an efficient algorithm to compute e(R, S) for any $R \in G$ and $S \in G$.

At the initial stage, the registration authority in the network generates system parameters {q, G, G_T, e, P} and a hash function: $H : \{0, 1\}^* \rightarrow G$, which are preloaded in each DTN node. A node Ni randomly chooses s_i as its private key that corresponds to a public key, $PK_i = s_iP$. Let Enc(∗) be an encryption algorithm, and Dec(∗) be the corresponding decryption algorithm. Accordingly, an MTM (M_i) in Ni possesses an independent private key λ_i and a public key, P $M_i = \lambda_iP$. Sign(∗) and Sigm(∗) are signature functions of N_i and M_i, respectively.

Information collection

When a connection opportunity approaches, nodes exchange some relay information between neighbours, which includes a connection event, routing information, and a relay request of messages. A connection event means that neighbours have successfully constructed a channel connection and are preparing to deliver messages. The connection event will be handed over all the layers in a node. The routing information contains neighbours' knowledge on the network environments, such as encounter history, node mobility track, and reputation. The node will use the routing information to make a routing decision. A relay request describes the basic information of messages, such as IP addresses, endpoint IDs, TTL, sizes of messages, and time stamp. However, a greedy node can still misbehave as described if the MTM in the node fail to get the relay information. Thus, the MTM must obtain the relay information prior to the upper layers as follows:

The PHY layer perceives the approaching neighbours, and then a connection event is generated and handed overall the layers.

When nodes N_i and N_j have constructed a connection, N_i generates routing information and a relay request (R) which are encrypted by M_i,C = Enc$_{PMj}$(R).

M_j in N_j decrypts the ciphertext C, R = Dec$_{\lambda j}$ (C). M_j saves the routing information, and records the relay request if these messages are successfully received by the node.

MTM loads the same routing protocol with its host in advance. The current routing protocols in DTNs are not complex, and they can be implemented by an MTM. The MTM is driven by connection events. When a connection event (*CE*) arrives at the trusted sublayer, the MTM runs the routing protocol according to collected routing information and relay requests above, and then generates next forwarded message via the available connection. Meanwhile, the basic information (*P_n*) of this message is abstracted by the MTM, *P_n*: *source IP_destination IP_endpoint IDs_sizes_time stamp*.

One of the most important swarm-based algorithms is Artificial Bee Colony (ABC) algorithm. ABC suggests the intellectual searching behaviour of a honeybee swarm. In ABC algorithm, the dependency of artificial bees contains of three major groups of bees: employed bees, onlookers and scouts. Each and every bees waiting on the dancing area to choose best trusted node in the path for routing is called as onlooker bee and one going to the Information collection source visited by it before is named employed bee. The other kind of bee is scout bee that carries out random investigation for discovering new trusted node in the routing protocol. In initial step of ABC randomly generates initial population of size SN, where SN (total number of nodes in the DTN during routing protocol) denotes the size of population. Each trusted node solution $\mathrm{x_i}, (\mathrm{i} = 1,2, \ldots . \mathrm{SN})$ is a D-dimensional vector. After initialization of nodes in the DTN, each population has a number of nodes positions is subjected to maximum number of cycles, $\mathrm{C} = 1, 2, \ldots, \mathrm{MCN}$, to complete trusted node selection process.

ABC [21] is easy way to develop and solve many optimization problems with only fewer controls of parameters [22]. Employed bees visit the food source position for trust and gathers information about mobile trust nodes basic information (*P_n*) of this message is abstracted by the MTM, *P_n*: *source IP_destination IP_endpoint IDs_sizes_time stamp*. Employed bees have memory, so they know the places they have visited before and the trusted nodes are selected. Each and every nodes waiting on the nest area to choose which node is trusted node in DTN is known as onlooker bee. An onlooker bee performs the global investigation of trusted node selection for and updates global results. Scout bees do a random search for each node in the DTN. A scout bee discovers the new nodes areas which are not focused by the employed bees. These three steps are continued until a termination criterion is satisfied.

The position of each node in the DTN is updated and the nectar amount of trusted host node is selected based on the fitness of the associated solution.

$$fit_i = \frac{1}{1 + fit_i} \tag{1}$$

Here fitness is considered as follows,

$$H(P_b) \overset{?}{} H(P_n) \tag{2}$$

It indicates that the host is honest if Eq. (2) holds, so the MTM will sign *P* with its private key (*Sig_m (P)*) and send it as the message format. Otherwise, the MTM believes that the host is a bee node, and refuses to sign *P*. An artificial onlooker bee chooses a trusted host node source depending on the probability value p_i, calculated by the following expression,

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \tag{3}$$

Once it is trusted a candidate food position from the old one in memory, the ABC uses the following expression,

$$v_{ij} = x_{ij} + \theta_{ij}\left(x_{ij} - x_{kj}\right) \tag{4}$$

where $\mathrm{k} \in \{1, 2, \ldots, \mathrm{SN}\}$ and $\mathrm{j} \in \{1, 2, \ldots, \mathrm{D}\}$ are randomly chosen indexes, $\phi_{ij} \in [-1, 1]$. As can be seen from (6), as the difference between the parameters of the $z_{i,j}$ and $z_{k,j}$. The value of predetermined number of cycles is an important control parameter of the ABC algorithm, which is called ''limit'' for abandonment. Assume that the abandoned source is $\mathrm{x_i}$ and $\mathrm{j} \in \{1,2, \ldots \ldots . \mathrm{D}\}$, then the scout discovers a new food source to be replaced with χ_i. This operation can be defined as in (5)

$$\chi_i^j = \chi_{min}^j + rand(0,1)\left(\chi_{max}^j - \chi_{min}^j\right) \tag{5}$$

After each candidate source position v_{ij} is produced and updated to bee results. Otherwise, the old one is retained in the memory. Totally, ABC algorithm employs four different trusted node selection processes: (1) a global probabilistic selection process in equation (4). (2) a local probabilistic by the employed bees and the onlookers depending on the visual information described by (5) a local selection called greedy selection process for trusted node selection is carried out by onlooker and employed bees corresponds to the fitness function. Otherwise, the bee keeps the present selected nodes are keep in the memory. (4) A random selection of nodes process carried out by scouts.

Algorithm 1- Artificial Bee Colony (ABC) optimization

```
Initialize the population of solutions $\mathrm{x_i}, \mathrm{i} = 1, \ldots \ldots . . \mathrm{SN}$, each population as number of nodes
Evaluate the population
Set cycle = 1
Repeat
Produce new trusted node solutions $\mathrm{v_i}$ for the employed bees by using (5) and evaluate them best trust
Apply the greedy selection process for the employed bees
Calculate the probability values $\mathrm{P_i}$ by (4)
Produce the new trusted node selection solution $\mathrm{v_i}$ from the solutions $\mathrm{X_i}$ depending on $\mathrm{P_i}$
Apply the greedy selection process
```

```
    Determine the abandoned nodes solution for the scout, if exists, and replace it with a new randomly produced solution $\chi_i^j$ by (5)
    Memorize the best solution achieved so far
    cycle = cycle + 1
    until cycle = MCN
```

In the optimized MMBD method, an MTM assumes that its host is honest during a period T_j, and grants a trust certificate (a signature on T_j) to the host. If the host relays messages as the routing protocol, the MTM will update the trust certificate at the next period T_{j+1}, and vice versa. The messages relayed by the host without a trust certificate will not be received by honest nodes. Thus, the trust certificate can restrain the greedy behaviours of the node, which is similar to a single signature in the basic MMBD method. The optimized detection method based on the trust certificate is presented as follows:

Algorithm 2:Optimized MMED algorithm

```
Let the current period be $T_j$
Store the available connection opportunities in a queue CO;
Collect the routing information saved in RI
Record the relay request in a queue RR
While size (CO) > 0 do
$l \leftarrow Getconnection(CO);$
$P \leftarrow Compute\ candidate(I, RR, RI);$
Add (Q,p);
If is period ($T_j$) then
If $HS_{j-1} == True$ && $F_j == False$ then
$Cert_j \leftarrow Sigm(T_j);$
Sendto upperlayer($Cert_j$)
$F_j \leftarrow True$
$F_{j+1} \leftarrow false$
End if
End if
End while
$P_n \leftarrow Getcandidate(Q);$
$P \leftarrow wait;$
$P_b \leftarrow extract(P);$
sendtoPHY(P)
if notmatch($P_b, P\_n$)
$Hs_j \leftarrow False;$
End if
Variables
$Hs_{j-1}: It\ is\ true\ if\ the\ host\ is\ honest\ during\ T_{j-1}$
$F_j$ update flag of trust certificate
Macros:
Sendtoupperlayer(Cert$_j$) :Submit a trust certificate on $T_j$ to the upper layers;
Getconnection(CO):get a specific connection from CO;
$Compute\ candidate(I, RR, RI)$ :Generate a new candidate message from RR;
Add(Q,P) :add P to the end of Q;
Sig$_m$(p):MTM signs for P;
Getcandidate(Q):get a candidate from Q;
Wait ():Wait for a message P from the upper layers;
NotMatch(P$_b$, P$_n$) :Check whether P$_b$ is not consistent with P$_n$
Extract(P):Extract the information of the message P
SendtoPHY (P) :Send P to PHY layer.
```

4. OVERHEAD AND PERFORMANCE EVALUATION

Adopt the opportunistic network environment (ONE) simulator which is a powerful simulation tool. ONE supplies a lot of dominant DTN routing protocol implementations, such as MaxProp, SprayAndWait, Prophet, and Epidemic. The simulations choose Epidemic and Prophet as routing protocols, and use the default setting of ONE_1.4.0 where each node has the same resources configuration as described in Table 1.

Table 1:Simulation parameters

Parameter	Value
Simulation time	12 hours
Number of nodes	100 nodes
Transmission range	50 m
Transmission speed	250 KB/s
Mobile speed	0.5-1.5 m/s
Mobility model	Map based mobility model
Application protocol	Ping application
Routing protocol	Epidemic, Prophet
Message size	500 KB
Message generation interval	100 s
Combination signature period	600 s

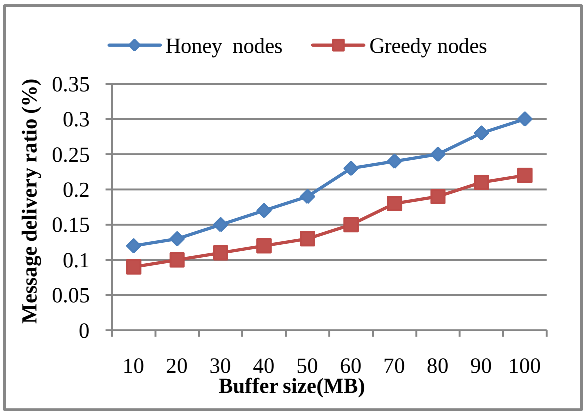

Fig.1. The average message delivery ratio comparison of greedy nodes with honey nodes in MMBD

Fig.1 illustrates the result, the average message delivery ratio of greedy nodes reaches 0.21, and the corresponding ratio of honest nodes is 0.31. Furthermore, the delivery ratio increment of greedy nodes from 10 MB to 120 MB is also higher than honest nodes. As shown in Fig.1, honey nodes have higher average message delivery ratio than greedy nodes.

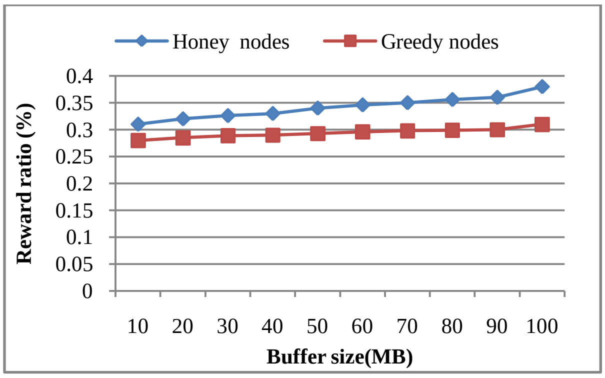

Fig.2. The average reward ratio comparison of greedy nodes with honey nodes in MMBD

Most credit-based incentive schemes, such as SMART, require that nodes fairly relay all the messages. To gain more credits with fewer resources in Greedy Behaviour II, greedy nodes break the SMART scheme, and only relay a message whose hops are less than 3. Therefore, these nodes can gain more credits than honest nodes since the rewarded credits are decided by length (hops) of a message delivery path. Use a reward ratio to evaluate this kind of misbehavior. The reward ratio is the proportion of total rewarded credits to the number of relayed messages. As illustrated in Figure. 2, the average reward ratio of greedy nodes is degraded into 0.312 which is similar to 0.326 of honest nodes in MMBD.

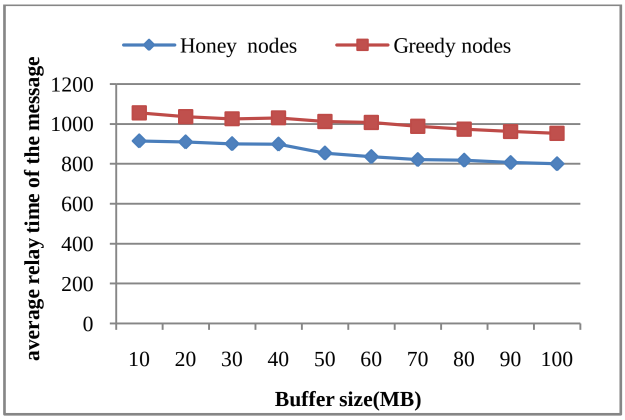

Fig.3. The average relay time of the message comparison of greedy nodes with honey nodes in MMBD

Contrarily, honest nodes relay messages according to the encounter probability in Prophet, and don't concern the relay time. So, the average relay time in honest nodes is larger than greedy nodes, and it reaches over 1180 s. This result indicates that honest nodes consume more resources than greedy nodes per relaying a message. In our method, it is difficult for greedy nodes to implement this misbehavior. The average relay time in greedy nodes nearly equals the time in greedy nodes as shown in Figure 3 and they are 1048 s and 989 s, respectively.

5. CONCLUSION AND FUTURE WORK

A delay tolerant network (DTN) is a store and forward network where end-to-end connectivity is not assumed and where opportunistic links between nodes are used to transfer data. During this data transfer process in DTN, require each node to honestly relay messages, but a few greedy nodes violate this principle in order to maximize their own benefit. This artificial bee colony (ABC) algorithm detects the misbehaviour nodes and breaks the routing fairness and decreases the message delivery ratio of DTNs. It has been concluded that the ABC algorithm can be efficiently used for by monitoring the forwarding sequence of messages inside a node in the mobile trusted module and proposed a message matching-based detection method, which uses an MTM to detect misbehaviour nodes. Therefore, it requires less computation time and fewer resources than the trusted computing group attestation. As future work, we plan on evaluating the impact of misbehaving nodes in different conditions, like different types of nodes, different scenarios, and other types of node behaviours. It is interesting to test different scenarios, as it has influence on the nodes contacts pattern. Another issue that affects contact patterns are mobility models. Finally, intend to propose new mechanisms to make routing more robust in the presence of misbehaving nodes. Future Enhancement will focus on the extension of the work to other kinds of networks and reduces the bandwidth of the nodes for malicious detection.

6. REFERENCES

[1] K. Fall, A delay-tolerant network architecture for challenged internets, in: Proceedings of ACM SIGCOMM, August 2003, pp. 27–34.

[2] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, E. Travis, H. Weiss, Interplanetary Internet (IPN): Architectural Definition, http://www. ipnsig.org/reports/memo-ipnrg-arch-00.pdf.

[3] J.A. Rice, R.K. Creber, C.L. Fletcher, P.A. Baxley, K.E. Rogers, D.C. Davison, Evolution of Seaweb underwater acoustic networking, in: Proceedings of IEEE Conference of the Oceans on Information Systems and Sciences, September 2000, pp. 2007–2017.

[4] Z. Zhang, “Routing in intermittently connected mobile ad hoc etworks and delay tolerant networks: overview and challenges,” IEEE Communications Surveys and Tutorials, vol. 8, no. 1, pp. 24–37, 2006.

[5] P. R. Pereira, A. Casaca, J. J. P. C. Rodrigues, V. N. G. J. Soares, J. Triay, and C. Cervell´o-Pastor, “From delay-tolerant networks to vehicular delay-tolerant networks,” IEEE Communications Surveys & Tutorials, vol. 14, no. 4, pp. 1166–1182, 2012.

[6] Y. Cao and Z. Sun, “Routing in delay/disruption tolerant networks: a taxonomy, survey and challenges,” IEEE Communications Surveys and Tutorials, vol. 15, no. 2, pp. 654–677, 2013.

[7] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, “Performance modeling of epidemic routing,” Computer Networks, vol. 51, no. 10, pp. 2867–2891, 2007.

[8] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Efficient routing in intermittently connected mobile networks: the multiple- copy case,” IEEE/ACM Transactions on Networking, vol. 16,no. 1, pp. 77–90, 2008.

[9] A. Lindgren, A. Doria, E. Davies, and S. Grasic, “Probabilistic routing protocol for intermittently connected networks,” IETF RFC 6683, August 2012.

[10] F. Li, A. Srinivasan and J. Wu, “Thwarting Blackhole Attacks in Disruption-Tolerant Networks using Encounter Tickets,” in Proc. of IEEE INFOCOM'09, 2009.

[11] S. Marti, T. Giuli, K. Lai, M. Baker, Mitigating routing misbehavior in mobile ad hoc networks, in: Proceedings of ACM MobiCom, 2000, pp. 255–265.

[12] S. Buchegger, J. Boudec, Performance analysis of the CONFIDANT protocol: Cooperation of nodes-fairness in dynamic ad-hoc networks, in: Proceedings of IEEE/ACM Workshop on MobiHoc, 2002, pp. 226–236.

[13] H. Zhu, X. Lin, R. Lu, X. Shen, D. Xing, Z. Cao, An opportunistic batch bundle authentication scheme for energy constrained DTNs, in: Proceedings of IEEE INFOCOM, March 2010, pp. 605–613.

[14] V. Srinivasan, P. Nuggehalli, C. Chiasserini, R. Rao, Cooperation in wireless ad hoc networks, in: Proceedings of IEEE INFOCOM, 2003, pp. 808–817.

[15] U. Shevade, H. Song, L. Qiu, Y. Zhang, Incentive-aware routing in DTNs, in: Proceedings of IEEE ICNP, October 2008, pp. 238–247.

[16] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In ACM Mobicom, Aug. 1998.

[17] J. Burgess, B. Gallagher, D. Jensen and B. Levine, “MaxProp: Routing for Vehicle-Based DisruptionTolerant Networks,” In Proc. IEEE INFOCOM, Apr. 2006.

[18] A. Balasubramanian, B. Neil Levine and A. Venkataramani, “DTN routing as a resource allocation problem,” In Proc. ACM SIGCOMM, Aug. 2007.

[19] A. Panagakis, A. Vaios and I. Stavrakakis, “On the Effects of Cooperation in DTNs,” In Proc. 2nd International Conference on Communication Systems Software and Middleware (COMSWARE), pp.1-6, Jan. 2007.

[20] A. Keranen, M. Pitkanen, M. Vuori and J. Ott, “Effect of non-cooperative nodes in mobile DTNs,” In Proc. 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp.1-7, Jun. 2011.

[21] N.Suguna and K.G.Thanushkodi, “An Independent Rough Set Approach Hybrid with Artificial Bee Colony Algorithm for Dimensionality Reduction”, American Journal of Applied Sciences 8 (3): 261 – 266, 2011.

[22] Li Bao and Jian-chao Zeng, Comparison and Analysis of the Selection Mechanism in the Artificial Bee Colony Algorithm, Proc. IEEE Ninth International Conference on Hybrid Intelligent Systems, 2009