Wireless Pers Commun (2013) 73:149–168
DOI 10.1007/s11277-012-0960-4

Secure Multi-copy Routing in Compromised Delay Tolerant Networks

Eyuphan Bulut · Boeslaw K. Szymanski

Published online: 15 December 2012

Abstract Routing in delay tolerant networks (DTNs) is challenging due to their unique characteristics of intermittent node connectivity. Different protocols (single-, multi-copy, erasure-coding-based etc.) utilizing store-carry-and-forward paradigm have been proposed to achieve routing of messages in such environments by opportunistic message exchanges between nodes that are in the communication range of each other. The sparsity and distributed nature of these networks together with the lack of stable connectivity between source destination pairs make these networks vulnerable to malicious nodes which might attempt to learn the content of the messages being routed between the nodes. In this paper, we study DTNs in which malicious nodes are present, to which we refer to as *compromised DTNs*. We discuss and analyze the effects of presence of malicious nodes on routing of messages in compromised DTNs. We propose a two period routing approach which aims at achieving the desired delivery ratio by a given delivery deadline in presence of malicious nodes. Our simulation results with both random networks and real DTN traces show that, with proper parameter setting, the proposed method can achieve delivery ratios which surpass those reached by other algorithms by a given delivery deadline.

Keywords Delay tolerant networks · Security · Routing

An initial version of this work is published in [34].

E. Bulut (✉)
2200 E President George Bush Hwy, Richardson, TX 75081, USA
e-mail: ebulut@cisco.com

B. K. Szymanski
Społeczna Akademia Nauk, ul. Sienkiewicza 9, Łódź 90-113, Poland

B. K. Szymanski
Network Science and Technology Center Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180, USA

1 Introduction

Delay tolerant networks (DTNs) are wireless networks in which at any given time instance, the probability that there is an end-to-end path from a source to destination is low. There are many examples of such networks in real life including wildlife tracking sensor networks [1], military networks [2] and vehicular ad hoc networks [3]. Since the standard routing algorithms assume that the network is connected most of the time, they fail in routing of packets in DTNs.

In DTNs, there is a sporadic connectivity between nodes. Therefore, to route the messages towards destination, *store-carry-and-forward* paradigm is utilized. In other words, if a node has a message copy but it is not connected to (i.e. not in the range of) another node, it stores the message until an appropriate communication opportunity arises. As the node encounters other nodes in the network, it assesses the benefit of the encountered node for delivery and either passes the message to it or not. The passing can take two forms: forwarding in which the sending node does not preserve the copy of the message and copying in which it does.

Several routing algorithms utilizing this paradigm have been proposed for DTNs based on flooding and erasure coding techniques. Since flooding based schemes suffer from huge overhead of bandwidth and energy consumption due to redundant transmissions, controlled flooding algorithms that use limited number of copies for each message have been developed. Also, single-copy based algorithms in which messages are forwarded towards the nodes which are predicted to have higher probability of meeting with destination have been proposed.

Even though there have been numerous routing algorithms proposed for DTNs in the literature, very few of them consider the security, trust and privacy issues in their designs. However, DTNs are very vulnerable to possible malicious node behavior because of their low node density and lack of stable end-to-end paths between source destination pairs. In this paper, we focus on compromised delay tolerant networks in which malicious nodes[1] are present. We discuss and analyze the effects of such malicious nodes on efficient routing of messages in compromised DTNs. We also propose a two period routing approach which aims to increase delivery ratio of uncompromised messages by a given delivery deadline in such compromised DTNs.

The remaining of the paper is organized as follows. In Sect. 2, we give background on related work. In Sect. 3, we describe the network model and the corresponding assumptions. In Sect. 4, we discuss and analyze the effects of malicious node behavior on routing under different trust models and network environments. We also elaborate on our two period routing approach. Next, in Sect. 5 we discuss the application of proposed algorithm in real DTN traces. In Sect. 6, we validate our analysis results and evaluate the performance of proposed algorithm through simulations with random and real DTN traces. Finally, we end up with conclusion in Sect. 7.

2 Related Work

2.1 DTN Routing Algorithms

Routing algorithms for DTNs can generally be classified as: single-, multi-copy (replication based) and coding based algorithms. In single-copy based routing [4,5], a message is forwarded to an encountered node if the delivery metric (computed depending on social relations [6,7], contact frequency [8] etc.) of that encountered node offers higher delivery

[1] We use attacker and malicious node interchangeably throughout the paper.

probability than the current carrier. In multi-copy based algorithms, multiple copies (limitless [9] or limited [10]) of the message are generated and distributed to other nodes (referred to as relays) in the network. Then, each of these nodes, independently of others, tries to deliver the message copy to the destination. In coding based algorithms (erasure coding [11,12] or network coding [13]), a message is converted into a large set of code blocks such that any sufficiently large subset of these blocks can be used to reconstruct the original message. As a result, a constant overhead is maintained and the network increases its robustness against packet drops when the congestion arises. However, those algorithms introduce computation as well as communication overhead resulted from coding, forwarding, and reconstructing of code blocks.

All of the above algorithms try to achieve average high delivery ratio for messages in different ways. They have advantages and disadvantages over each other in different network environments. However, they all assume friendly network environments which might not be realistic in many real-life DTN scenarios.

2.2 Security of DTN Routing

Recently, some researchers have studied the security of DTN routing. In [14], Burgess et al. show that replication based DTN routing algorithms are intrinsically fault-tolerant and robust against a large number of attacks even without authentication mechanisms. On the contrary, in a more recent study [15], it has been shown that some specific combinations of attacks can reduce the delivery ratio remarkably.

In [16] and [17], encrypted encounter tickets are proposed to prevent claiming of forged encounter history by malicious nodes. However, these methods cannot detect packet drops in the malicious nodes. Moreover, to detect the blackhole nodes and prevent them from attracting data from the network, different reputation based mechanisms are utilized. In [18], a trusted third-party examiner node called ferry node (which moves around the network) is introduced. In [19,20] the history of packet exchange records between nodes is used and in [21] and [22], the feedback mechanisms are used to increase the reputation of nodes which previously had a role in the delivery of packets. Similarly, a trust based mechanism for encounter based routing is proposed in [23] and in [24].

All of the previous studies mentioned above attempt to secure routing by detecting the individual nodes behaving maliciously and preventing them from obtaining the messages in the network. They consider the malicious behavior only from the attacker's point of view and do not consider the trust among the current network members and their ability to collectively mistrust the attacker. Still, even the currently trustworthy nodes can be open to influence of malicious nodes which might appear in the network later. Moreover, the current approaches consider the messages to be successfully delivered even if they passed via malicious nodes (in single-copy based routing) or a copy of the message is obtained by any of them (in multi-copy based routing). Yet, exposure of the content of the messages to attackers (or unwanted nodes) often significantly lowers or negates the value of its delivery to the destination in many DTN applications (e.g. military, financial).

In this paper, we define the secure delivery as follows.

Definition 1 Secure delivery: The message is securely delivered to its destination if and only if the message is received by the destination before the deadline and before any attacker receives it.

Note that in multi-copy based routing, once the destination receives the message, it starts an epidemic like acknowledgment and informs other nodes carrying the message copy about

delivery. Then, these nodes delete the message copies from their buffer. As shown in [25], this epidemic like acknowledgment takes very short time compared to data delivery. Here, we assume that such acknowledgment is used and therefore the likelihood that an attacker will receive a copy of a message after the destination receives it is negligible.

Moreover, also note that the above definition of secure message delivery differs from the ones in previous work that basically consider only delivery of the message to its destination but not its exposure to attackers.

3 Network Model and Assumptions

DTNs are characterized using different mobility models. Random models (e.g. random direction, random waypoint), community-based models [26] and real DTN trace-driven models (e.g. zebranet [1]) are among the most popular ones. We analyze the effect of malicious nodes and coalition of nodes in the network with these malicious nodes on the secure delivery using a limited multi-copy based routing algorithm such as Spray and Wait [10]. Hence, we assume a network environment similar to the one described in [10].[2]

We assume that there are M nodes randomly walking on a $\sqrt{N} \times \sqrt{N}$ 2D torus according to a random mobility model (which makes the intermeeting time between two nodes exponentially distributed). Each node has a transmission range R and all nodes are identical. The buffer space at each node is assumed to be sufficiently large that no message is ever dropped because of lack of storage (this is practical since the proposed algorithm uses few copies of the message). The communication between nodes is assumed to be perfectly separable, that is, any communicating pair of nodes do not interfere with any other simultaneous communication (which is most often the case in DTNs due to sparse node density).

In Spray and Wait algorithm [10], each source node distributes a limited number of copies (L) of its message to other nodes in the network and wait for the delivery of one of them to the destination. Since a limited number copies (L) of the message during delivery is used and these copy counts are much smaller than the total node count in the network ($L << M$), as it is shown in [10] and [25], $1 - e^{-\lambda L t}$ is a good approximation of delivery probability by time t after the generation of the message at source node. If there are no malicious nodes in the network, source node can find the minimum number of copies [10] that it needs to distribute to other nodes to achieve a desired delivery rate (d_r) within a given time constraint (t_d) or delivery deadline by computing $L_{min} = \lceil (\ln(1 - d_r))/(-\lambda t_d) \rceil$, where λ is the rate of exponentially distributed intermeeting times of nodes.

However, delay tolerant network environments in real life may be hostile and due to the sparse network topology and intermittent connectivity, malicious nodes can easily attack the network and degrade the routing and delivery performance of these networks. These malicious nodes can even join the network for a short time and form coalition with the existing nodes. Moreover, it is also reasonable to expect that some nodes in such sparsely connected networks can be open to coalitions. Military based DTNs are a good example of such networks. Even though all actors (i.e. soldiers) initially follow only their commander, all may have a level of trustworthiness beyond which they may be convinced to cooperate with unauthorized people.

A high school network is another example. Students in the same class are more likely to be good friends with each other, so their relations are on average more trustworthy than

[2] In Sect. 5, we also discuss the application of proposed algorithm to real DTN traces where nodes have heterogeneous meeting behaviors.

the relations between the students in different classes. Consequently, the best strategy for a student to deliver its message to a specific student outside of her class is to propagate the message to the first classmate of that student met during the class break. However, if she let all students (including the ones out of her class who may not be in as good relationship with her as her classmates) carry the message, she might risk the secrecy of her message, as less trustworthy carriers might reveal the message to a teacher or public in general.

The objective of secure routing is to deliver the messages with a desired delivery ratio by the given deadline but without revealing the message content to malicious nodes. Thus, we discuss the ways of distributing the message copies to relay nodes based on their trustworthiness levels and propose a two period spreading algorithm in which initially the message is spread only to trusted nodes, and if this is not enough to reach the desired delivery rate by the given deadline, then by the start of second spraying period the message copies are also shared with more risky nodes.

4 Proposed Protocol and its Analysis

In this section, we discuss and analyze the secure delivery of messages in compromised DTNs where the nodes in the network might be open to coalition with malicious nodes. We first define the trust model used throughout the paper.

Definition 2 Trust model: The nodes are assumed to be trusted by the source, from whom they received the messages, with a probability of p_t that this trust is justified. Thus, when a node at this level of trust carrying a message copy meets the attacker, it gives the message copy to attacker node with probability $p = 1 - p_t$.

Next, we analyze different variants of trust distribution among the nodes and discuss the effect of different message distribution schemes on secure delivery.

4.1 Constant Trust Model

Here, assuming that all nodes are trusted by the source node with a constant probability of p_t, we analyze the effects of attackers on secure delivery and find out the L_{min} number of copies of a message needed to achieve a desired delivery ratio d_r by deadline t_d.

Theorem 1 *For a given d_r, t_d, λ (rate of exponentially distributed intermeeting time between nodes), n (number of attackers), and $p = 1 - p_t$, the minimum number of copies that must be distributed to the network is:*

$$L_{min} = \left\lceil \frac{\ln(1 - d_r(pn + 1))}{-\lambda t_d(pn + 1)} \right\rceil \tag{1}$$

Proof We first find the cumulative distribution function (cdf) of secure delivery when there are L copies of the message under the given network environment. Let X be the random variable (r.v.) representing the secure delivery. Then, cdf of X, $F_X(x)$, is:

$$F_X(x) = P(X \leq x) = \int_0^t L\lambda e^{-L\lambda x}(e^{-Lpn\lambda x})dx$$

Here, the first term ($L\lambda e^{-L\lambda x}$) shows the probability density function (pdf) of the meeting probability of any of the L nodes (carrying a message copy) with destination and the second

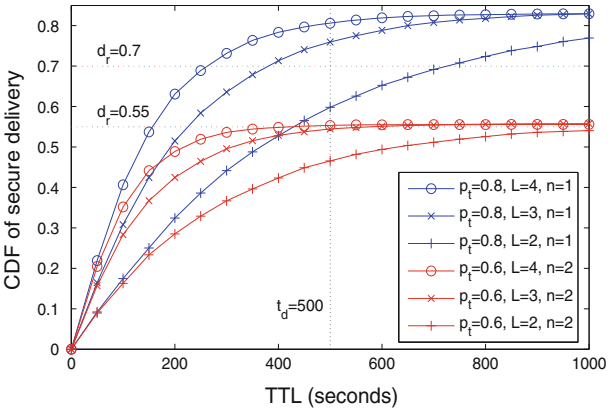

Fig. 1 Cumulative distribution function of secure delivery ratios with different L in different network settings

Table 1 Analysis results for maximum achievable secure delivery ratios with different constant trust probabilities and attacker counts

n\p	0.2	0.4	0.6	0.8	1.0
1	0.83	0.71	0.62	0.55	0.50
2	0.71	0.55	0.45	0.38	0.33
3	0.62	0.45	0.35	0.29	0.25

term $(e^{-Lpn\lambda x})$ shows the cdf of the non-meeting probability of any of these L nodes with any attacker node. This is a consequence of the definition of secure delivery, which requires the delivery of a message copy to destination before attacker gets it. Then:

$$F_X(x) = \int_0^t L\lambda e^{-L(pn+1)\lambda x} = \frac{1}{pn+1}\left(1 - e^{-L(pn+1)\lambda t}\right)$$

Thus, equation for L_{min} that can achieve d_r by t_d becomes Eq. 1:

Note that, in the above $F_X(x)$ formula, it is clear that the maximum value of $F_X(x)$ is $1/(pn+1)$ (which becomes $1/(n+1)$ when $p = 1$). Thus, if the deadline of delivery is not an issue, attacker count defines the maximum achievable delivery rate. Moreover, for a given network setting with constant p, whatever the number of copies distributed is, it may not be possible to reach d_r by t_d if $d_r > \frac{1}{pn+1}$. In Fig. 1, we plotted cdf of secure delivery ratios with different L values in two different network settings. For example, when there is a single attacker and $p_t = 0.8$ (or $p = 0.2$), the maximum reachable secure delivery ratio is 0.83. If d_r is beyond this point, it is not reachable. On the other hand, if $d_r \leq \frac{1}{0.2+1} = 0.83$, L_{min} is found using Eq. 1. For example, achievable $d_r = 0.70$ when $t_d = 500$ s, $L_{min} = 3$. Similarly, in the other network setting (when there are two attackers and $p_t = 0.6$), achievable $d_r < 0.55$ at the same $t_d = 500$, $L_{min} = 3$.

Table 1 shows the maximum achievable secure delivery ratios with different constant trust probabilities and attacker counts (n). In simulations section below, we describe the simulations that verified these results.

4.2 Group-Based Trust Model

The trust levels of nodes in a network may also be group-based, making the distribution of message copies more challenging. Then, the question is: "what should the spraying strategy be for a given group-based trust distribution of nodes in the network?".

A source node, having the objective of delivering its messages to their destinations without revealing them to attackers, may use one the following message copy distribution strategies:

- Fully Trusted Spraying (FTS): Source node sends the message copies to its fully trusted friends only. Even though this strategy makes the routing of messages completely secure, delivery delay might increase if only few nodes are trusted.
- Aggressive Spraying (AS): Source node sprays the message copies to nodes it encounters first. With this strategy, the number of message copy carriers increases quickly in the network, improving chances of delivery, but message copies may also be distributed to partially trusted or even untrusted nodes, increasing the probability of revealing the message to attackers.
- Trusted First Spraying (TFS): Source node distributes the message copies to the nodes in the network in the order of their trust levels. Thus, first the message copies are distributed to fully trusted friends. Once all trusted nodes have a message copy, message is copied to partially trusted nodes. Finally, after all trusted and partially trusted friends have the message copy, untrusted nodes are given the message copies.

Each of the above strategies might be advantageous compared to others in different network environments and with different delivery objectives. In addition to the above three simple strategies, we propose a fourth and a novel way of spraying:

- Two Period Spraying: The message copies are distributed to the network in two periods. In the first period, the source copies the messages only to trusted nodes. Then, if there will not be sufficient number of such nodes to reach the desired delivery rate by the given deadline, by the start of the second period, less trusted nodes (can be any node or nodes with p_t more than a threshold) are also given the message copies. In other words, source starts with secure spraying and switches to aggressive spraying (or limited aggressive spraying if a threshold is used) with the start of the second period. In the first period the message is routed only through trusted nodes but as it gets closer to delivery deadline, to reach the desired delivery rate by the deadline, the algorithm increases the number of nodes carrying a message copy in the network by giving a message copy to less trusted nodes that source node meets. However, this also increases the risk of message being compromised. Thus, the key point is 'how to decide the start of the second period?'

Next, we will show the performances of these algorithms on a sample network environment. Here, we used the first network setting introduced in simulation section in detail.

We generated three groups of nodes with different trust levels. Only 5 % of all nodes in a network are fully trusted by source node ($p_t = 1$) and they never pass the message to the attacker. Another 20 % of nodes are trusted with probability of $p_t = 0.7$, meaning that when they have message copies and the attackers meet with them, they give the message copy to attacker with probability $p = 1 - p_t = 0.3$. The remaining nodes in the network are untrusted ($p_t = 0$), thus if they have the message copy and meet the attackers, they always pass the message to the attacker ($p = 1$). Note that p_t defines the probability that node passes the message to the attacker during their meeting but it does not define probability of receiving of the message copy from nodes that carry it. The latter probability is defined by the spraying strategies defined above.

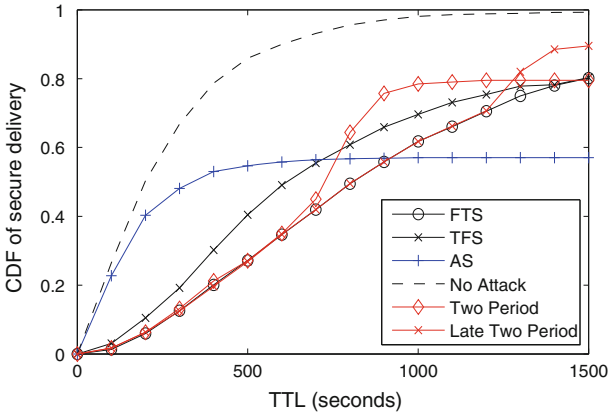

Fig. 2 Cumulative distribution function of secure delivery ratios in different spraying algorithms

Figure 2 shows the cumulative distribution function of secure delivery ratios (with respect to the time passed since the generation of messages at source nodes) when the aforementioned spraying algorithms are used. We considered a single attacker in the system. It is clear that when there is no attacker, the maximum delivery ratio is achieved. When there is an attacker, the delivery ratio of aggressive spraying increases fast but it can only reach the maximum which is around 0.5. The delivery ratio of FTS increases slowly, however, since the source node gives the copies only to nodes that do not give the message copy to attackers, the delivery ratio has potential of reaching the maximum value of 1. This algorithm might be preferable since it preserves privacy. However, if achieving a higher delivery ratio within a time constraint is an objective, it is not the best choice. Looking at the graph of TFS, we notice that the delivery ratio increases faster than the delivery ratio of FTS, but it eventually converges to a constant value since it risks the privacy of the message while using partially trusted nodes. Those nodes contribute to delivery ratio in earlier stages but since they might form coalition with attacker, in long term their benefit is lost. Note that the spanned delivery ratios by the plots of these three algorithms (TFS, FTS, AS) clearly indicate that each of these algorithms might be preferred depending on the given network parameters (desired delivery ratio, deadline). However, one can have a goal of achieving a delivery rate that cannot be achieved by any of these algorithms. For those cases, we propose to use two period spraying algorithm. Consider the delivery ratios achieved in two different runs of two period spraying algorithm. The only difference between these two runs is the start of second period of spraying. Clearly, the start of second period can be chosen according to the network parameters and desired routing output.

Theorem 2 *When there are L_t trusted nodes carrying the copy of the message in the first period and L_u partially trusted nodes with trust probability $p_t = 1 - p$ start to carry a message copy in second period (making in total $L_a = L_u + L_t$ nodes with a copy), to achieve a given d_r (with no t_d), the start of second period, t_2, must be larger than a constant, t_2^{min}, defined as:*

$$t_2^{min} = \frac{-\ln\left((1-d_r)(\frac{L_a}{npL_u}+1)\right)}{\lambda L_t}$$

Proof Let X_2 be the r.v. representing the secure delivery in two period spraying. It is clear that in the first period $F_{X_2}(x)$ grows with $1 - e^{-\lambda L_t x}$. But if the delivery does not happen in first period (with probability $e^{-\lambda L_t t_2}$) and second period starts, the pdf of secure delivery in second period is supported by L_a nodes towards delivery and risked by L_u partially trusted nodes. Thus, $F_{X_2}(x)$ in second period is:

$$F_{X_2}(x) = 1 - e^{-\lambda L_t t_2} + e^{-\lambda L_t t_2}(S) \quad \text{where}$$

$$S = \int_0^{x-t_2} L_a \lambda e^{-L_a \lambda x} \left(e^{-L_u np \lambda x}\right) dx$$

$$= \frac{L_a}{L_a + npL_u} \left(1 - e^{-(L_a + npL_u)\lambda(x-t_2)}\right)$$

In the above formula, it is easy to see that maximum delivery ratio that can be reached (when x goes to ∞) is $1 - e^{-\lambda L_t t_2} \left(\frac{npL_u}{L_a + npL_u}\right)$. Since this value must be larger than d_r, minimum value of the start of the second period can be derived as:

$$t_2 \geq \frac{-\ln\left((1-d_r)\left(\frac{L_a}{npL_u} + 1\right)\right)}{\lambda L_t}$$

□

Corollary 1 *For a given parameter set (L_t, L_u, t_2), the cdf of delivery rate in FTS is definitely better than the cdf of delivery rate in two period spraying after t_{max}, where:*

$$t_{max} = t_2 + \frac{\ln\left(1 + \frac{L_a}{L_u np}\right)}{\lambda L_t}$$

which can be easily proved by comparing the maximum achievable delivery ratio of two period spraying with the cdf of delivery ratio of FTS algorithm.

If there is a time constraint, t_d, and the goal is to achieve the maximum possible delivery rate (which is not achievable by FTS) with given L_u, then the start of the second period could be adjusted accordingly.

Theorem 3 *For a given delivery deadline, L_u and L_t, the optimal value of t_2 that gives the maximum delivery rate by t_d is t_2^{opt}, where:*

$$t_2^{opt} = t_d + \frac{\ln\left(\frac{L_t npL_u}{L_a(L_a + npL_u - L_t)}\right)}{\lambda(L_a + npL_u)} \tag{2}$$

Proof We first find $d'(t_2) = \frac{F_{X_2}(t_d)}{d(t_2)}$:

$$d'(t_2) = \lambda(e^{-\lambda L_t t_2}) \left[L_t \left(\frac{npL_u}{L_a + npL_u}\right) + (L_t - L_a - npL_u) e^{-\lambda(L_a + npL_u)(t_d - t_2)}\right]$$

Then, solving $d'(x) = 0$, we obtain:

$$x = t_d + \frac{\ln\left(\frac{L_t npL_u}{L_a(L_a + npL_u - L_t)}\right)}{\lambda(L_a + npL_u)}$$

Since, $d''(x) < 0$, $F_{X_2}(t_d)$ has local maximum at x, making $t_2^{opt} = x$. □

In addition to time constraint, if there is a desired delivery rate, d_r (again which is not achievable by FTS), and minimizing the average cost of the algorithm (average number of message copies sprayed to network) is also an objective, the start of second period and the number of untrusted nodes, L_u, that will carry a message copy in second period must be selected carefully.

Theorem 4 *For a given delivery deadline, t_d, and desired delivery rate, d_r, the optimal number of untrusted nodes that minimize the overall routing cost which still achieves d_r by t_d can be computed as in Algorithm 1.*

Proof Cost of the algorithm (i.e. average number of copies used) can be computed as:

$$\begin{aligned} c(L_t, L_u) &= L_t(1 - e^{-\lambda L_t t_2}) + (L_t + L_u)e^{-\lambda L_t t_2} \\ &= L_t + L_u e^{-\lambda L_t t_2} \end{aligned}$$

We first find the L_u value that achieves a secure delivery rate higher than desired d_r by t_d. Then, if the achieved delivery rate is much higher than d_r, we delay the start of second period as much as possible (without dropping delivery rate below d_r) because with constant L_t and L_u, the cost of the algorithm decreases with the increase of t_2. To find such t_2, we use binary search between t_2^{opt} and t_d. □

Finding the closed form of exact optimum t_2 that achieves d_r by t_d will be the subject of our future work.

Algorithm 1 Find Optimum Routing(L_t, p, n, d_r)

```
L_u = 1
Find t_2^opt for current L_u from Eq. 2
while (F_X2(t_2^opt) < d_r) do
    L_u = L_u + 1
    Find t_2^opt for current L_u from Eq. 2
end while
if (L_t + L_u e^{-λ L_t t_2^opt} > d_r) then
    Find exact t_2^{opt_exact} by binary search in [t_2^opt, t_d]
end if
opt_L_u = L_u; opt_cost = L_t + L_u e^{-λ L_t t_2^{opt_exact}}
```

Note that, the above algorithm finds L_u that gives the optimum cost when a given constant L_t nodes can not achieve d_r by t_d. However, if there are sufficient number of trusted nodes (L_t) to achieve these goals, only they are used without using untrusted ones.

4.3 Complex Functional Trust Models

The trust distribution of nodes may be more complex than the ones enumerated above. That is, all nodes of the network may be open to coalition with attackers, but with different probabilities (i.e. following a distribution function). Consider Fig. 3, where we plot three different trust level distributions of nodes in a network. These three plots represent the generalized views of majority of possible trust distributions when sorted in descending order. In a trust-prone network, most of the nodes have high p_t values, while in a distrust-prone network most of the nodes are open to coalition with malicious nodes. In between these two network types, there may also exist networks with linear trust distribution.

In a network with nodes having a typical functional trust model, AS algorithm defined above can be applied without no change. However, other algorithms will not be applicable

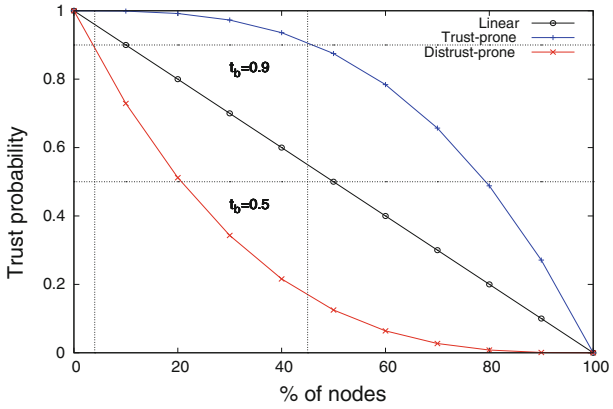

Fig. 3 Different trust level distributions

in the way they are described above. This is because there will not be any other node that the source node totally trusts (only source node has full trust ($p_t = 1$) to itself). Thus, in FTS, source cannot send a message copy to a node in the network. Similarly, in TFS, if the source node waits until the next node (without a message copy) whom it trusts the most among the remaining nodes to make a copy, it will take much longer to distribute the copies of the message, yielding low delivery rates with small TTL values.

To simplify complex trust models and also to make all algorithms applicable, we can convert a given functional trust model to a group based trust model using some approximations. Consider the trust-prone functional model in Fig. 3. The first 43 % of nodes have $p_t \geq 0.9$. They could be considered as a group of *most trusted nodes* having a p_t that is equal to the average of their original p_t values. Since, the rest of the nodes will have more varying p_t values, they could be divided into multiple groups (i.e. partially trusted and untrusted nodes). Then, the algorithms introduced in previous section could be applied. The previous formulas can also be used once the trust probability for each group (with k nodes) is set to the average by computing $p_t = \sum_{i=1}^{k} \frac{p_t^i}{k}$, where p_t^i shows the node i's p_t value.

Similar approximation can also be applied for linear and distrust-prone functional models. However, the division of nodes into groups has to be done appropriately to achieve a good approximation. Moreover, the limitations on the number of copies that are allowed to be distributed to each group of nodes should also be taken into account. For example, if the same trust boundary (t_b) of 0.9 is used for linear and distrust-prone networks, then 10 and 3 % of nodes will have $p_t \geq 0.9$. Assuming FTS will distribute L message copies to these nodes only, in distrust-prone network case, there may not even be L nodes with $p_t \geq 0.9$ if $3M/100 < L$, where M is the number of nodes in the network.

5 Application on Real DTN Traces

Recently, many projects focused on the deployment of real delay tolerant networks in several network environments (office [27], conference [28], city [29], skating tour [30]) using different mobile objects (humans [31], buses [32], zebras [1]). The collected trace data from these deployments demonstrated that the characteristics of DTNs and also the mobility of mobile

objects might be more complex than the random models. Thus, in this section, we discuss how the proposed algorithm can be applied in these heterogeneous network environments.

5.1 Online Behavioral Trust Computation

In the previous section, we assumed that the trust distribution of other nodes (to source node) is already computed or known by the source node. This is also practical if we consider the real life example of DTNs. For example, in a military network, a commander can compute the loyalty of soldiers (from the number of years they served, their previous accomplishments etc.) working in his region and would select the confidential message carrying soldiers accordingly. Similarly, in a high school network, a student can consider their classmates more trustworthy than the students in other classes. Moreover, the student can rank her classmates by making an assessment of her relations with them in the past.

Even though trust computation would change according to the context and environment in which the DTN is running, a good way of computing trustworthiness of users could be made using the previous relations between users and the malicious node. Consider high school network example. Different students could be considered malicious (i.e. the ones whom the source node does not want to learn the message content) for different students. Thus, a message's secure routing from source node (i.e. student) to destination node should be done through nodes who are considered to be trusted by source node and are not in good relationship with malicious nodes. Assuming that $f(i, j)$ shows the meeting frequency of two nodes, node i can decide how much node j is trusted (denoted as p_t^j) for a message to deliver to node k by computing:

$$p_t^j = \frac{f(j,k)}{f(j,k) + f(j,\forall a)} \tag{3}$$

where

$$f(j,\forall a) = \sum_{m\in 1}^{n} f(j,a_m)$$

and $a_1, \ldots, a_n$ is the list of all malicious nodes. Here, note that p_t^j for each node can also be computed in case of less likelihood of coalition with malicious nodes as:

$$p_t^j = 1 - \left(\frac{f(j,\forall a)}{f(j,k) + f(j,\forall a)}\right)^{\alpha}$$

where α is predefined constant and can be computed according to network environment and the relations of source node with other nodes.

5.2 Two Period Routing in Heterogeneous Network Environment

Once each node's p_t value is computed in online manner as the nodes interact with each other, a source node can decide the group of nodes which it can trust the most ($p_t \geq t_{b_1}$), partially ($t_{b_1} > p_t \geq t_{b_2}$ and the least ($t_{b_1} > p_t$) using two trust boundaries, t_{b_1} and t_{b_2} (more groups can also be formed by using multiple trust boundaries). Then, the routing algorithms could be used as it is described in complex functional trust model section. In two period routing, first the source will start distributing the message copies to the most trusted nodes. If there is no sufficiently many of such nodes and the distribution of more copies is allowed, then it can continue distribution with partially trusted nodes. If the deadline to reach a desired

delivery rate, d_r, is not expected to be met, then it starts AS or limited AS type of message copy distribution in which it also sends copies to less trusted nodes it encounters as long as the allowed copy quota is not exceeded.

6 Simulations

In this section, we describe simulations done to (1) validate the theoretical results we found in previous sections and (2) evaluate the performance of proposed two period routing algorithm. We used two different network settings. In the first one, we generated a more generic network consisting of nodes that move according to a random mobility model. We used this network setting specifically to validate our theoretical foundations. In the second one, we simulated real DTN traces that were collected through Haggle project [28]. The evaluation of proposed two period routing algorithm is performed in both network settings.

6.1 Network Settings

6.1.1 Random Model

We deployed $M = 100$ mobile nodes onto a torus of size 300 m by 300 m. All nodes are assumed to be identical and their transmission range is set to $R = 10$ m (note that these parameters generate a sparse delay tolerant network which is the most common case in practice). The movements of nodes are defined according to random walk model. The speed of a node is randomly selected from the range [4, 13] m/s and its direction is also randomly chosen. Then, each node goes in the selected random direction with the selected speed until the epoch lasts. Each epoch's duration is randomly selected from the range [8, 15] s. When nodes move according to this model with the given above parameters, the average intermeeting time between any pair of nodes is 480 s.

6.1.2 Real DTN Traces

We also generated a simulation setting by emulating one of the popular DTN traces which were collected during Haggle project [28]. The dataset consists of many traces from different experiments. We selected the Bluetooth sightings recorded between the iMotes carried by 41 attendants of Infocom 2005 Conference held in Miami. Devices were distributed on March 7th, 2005 between lunch time and 5 p.m. and collected on March 10th, 2005 in the afternoon.

6.2 Results

In the simulations, we generated 5,000 messages, each from a random source node to a random destination node every t seconds. To account for duration of experiments, we set $t = 20$ s for random network setting, but for Haggle traces, we set $t = 5$ s. All messages are assigned a time-to-live (TTL) value representing the maximum delay requirement.

We first start with comparison of theoretical values in Table 1 with simulation results. Figure 4 shows the maximum achieved secure delivery ratios in simulations with different network environments in the case of constant trust model. We used one to three attackers and different $p = 1 - p_t$ values in range [0, 1]. Comparing the results in this figure with the results in Table 1, we see a complete match. Moreover, we also looked at the maximum secure delivery ratios achieved when different trust boundary (t_b) values are used for limited

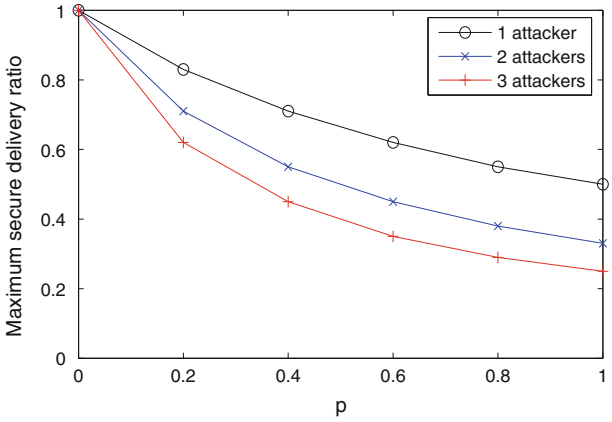

Fig. 4 Maximum secure delivery ratios achieved with different constant trust probabilities and attacker counts

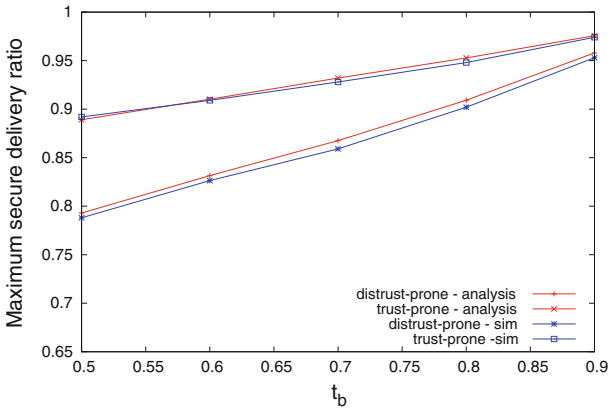

Fig. 5 Maximum secure delivery ratios achieved by limited AS with different trust boundary (t_b) values

AS algorithm in the case of functional trust distributions. Figure 5 shows the comparison of results computed by analysis and obtained from simulations. When AS algorithm distributes the copies only to nodes with $p_t \geq t_b$ (making it limited AS), as a result of approximation, the maximum achievable secure delivery rate becomes $1/(1+p_{avg})$, where $p_{avg} = 1 - \sum_{i=1}^{k} \frac{p_t^i}{k}$ and k is number of nodes satisfying $p_t \geq t_b$. Figure 5 shows the goodness of the approximation for different functional trust distributions with different t_b values (when there is $n = 1$ attacker).

In Figs. 6, 7 and 8, we show the secure delivery ratios achieved by different algorithms[3] in linear, distrust-prone and trust-prone network environments, respectively. In the case of linear and distrust-prone networks, we observe that the proposed two period algorithm can achieve significantly high delivery ratios than AS or FTS algorithms. For example, in Fig. 6, two period algorithm can achieve $d_r = 0.76$ at $t_d = 400$ s while the others can not. Similarly, in Fig. 7, two period algorithm can achieve $d_r = 0.68$ at $t_d = 400$ s while again the others fail to reach the same delivery rate by t_d. However, in trust-prone network case, the proposed algorithm

[3] We also show the results with no attacker for reference to the maximum achievable delivery ratios in a secure environment.

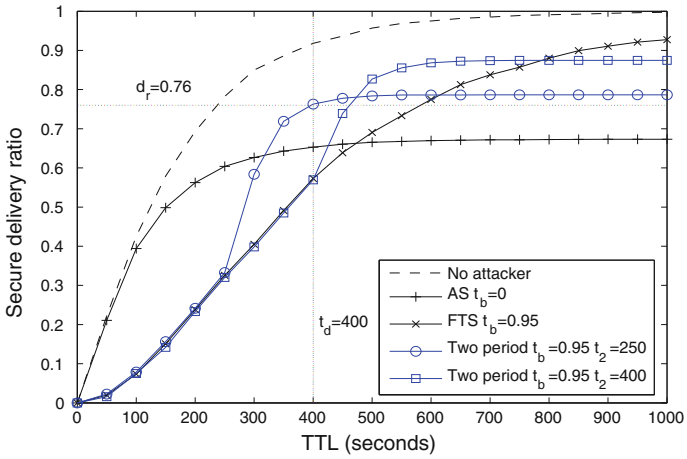

Fig. 6 Secure delivery ratio with linear trust distribution

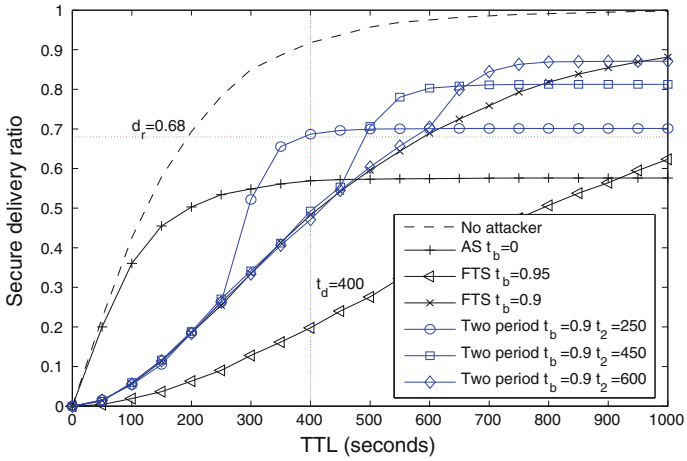

Fig. 7 Secure delivery ratio with distrust prone network

achieves only little increase over other algorithms. This is because in trust-prone networks, there are usually sufficiently many nodes with high p_t values. Source node distributes the message copies to them to get high delivery ratios.

Next, in Table 2, we show the comparison of analysis and simulation results for two period spraying algorithm. Assuming that source node has already given message copies to two ($L_t = 2$) trusted nodes (with $p_t = 1$) in the first period, using the analysis results we computed number of untrusted nodes (L_u) with $p_t = 0.4$ that also need to carry message copies by the start of second period to achieve a given delivery rate (d_r) by the given deadline (t_d). Then, we simulated two period routing algorithm for each set of parameters. The table shows that >90 % of messages can reach the desired delivery ratios by the given deadlines. The reason why some messages (<10 %) can not reach that delivery ratio is because we do not take into account the effect of spraying duration in our analysis. As the number of nodes

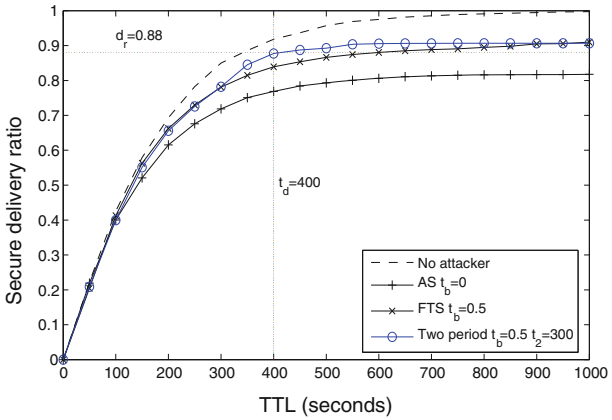

Fig. 8 Secure delivery ratio with trust-prone network

Table 2 Analysis versus simulation results for two period algorithm

Given		Analysis			Simulation
t_d (s)	d_r	L_{min}	t_2^{opt} (s)	Average cost	% of messages achieving d_r
500	0.6	1	315	2.27	100
600	0.85	2	445	2.32	94
700	0.90	3	560	2.45	91

carrying message copies increases the impact of spraying duration increases, however, the simulation results show quite good match with analysis results with the assumption we made.

For the simulations with real DTN traces, before generating messages, we also let the nodes move during a warm up period (1/5 of total data) and build their initial contact history. Each source node computes the trust of each node j (p_t^j) to itself using Eq. 3. Here, note that the warm up period lets the nodes have sufficient contact history to compute initial trust values. However, at each meeting of source node with other nodes, the computed trust values can change as new meeting history of other nodes is learned. We assume that as the nodes having a message copy meet the single attacker, they give the message copy to the attacker with probability $p = 1 - p_t$, where p_t is its trust to source node computed from the current contact history.

In Figs. 9, 10 and 11, we show the outputs of simulations with real DTN traces. In Fig. 9, we show the secure delivery ratios achieved by compared algorithms. Similar to the first network setting, two period algorithm can also achieve delivery rates which are not achievable by other algorithms in real DTN simulation setting as well. In addition to secure delivery ratio, we also plotted results with two different metrics. Average cost is measured by the average number of copies distributed per message during the simulation. The routing efficiency [7,33] is defined as the ratio of the secure delivery ratio to the average cost. From Figs. 10 and 11, we conclude that two period routing algorithm can maintain similar routing efficiency as FTS algorithm and higher routing efficiency than AS algorithm, while it can achieve higher delivery ratios than both of these algorithms.

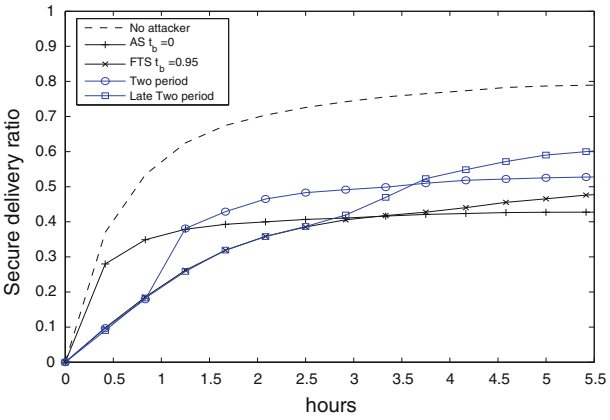

Fig. 9 Secure delivery ratio with real DTN traces

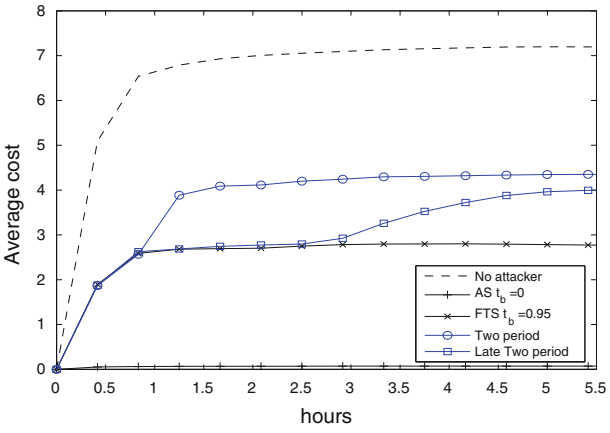

Fig. 10 Average cost with real DTN traces

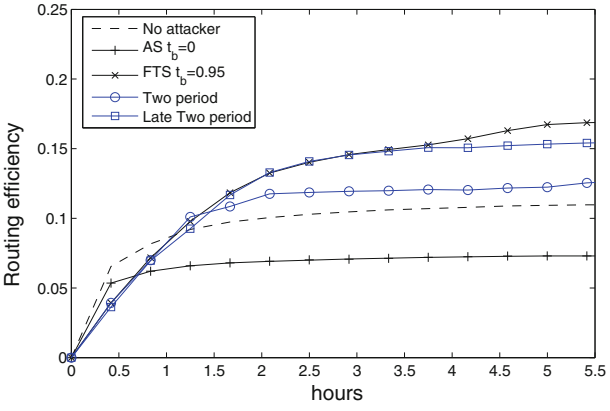

Fig. 11 Routing efficiency with real DTN traces

7 Conclusion

In this paper, we focused on the problem of routing in compromised DTNs in presence of malicious nodes. Assuming that, with certain probability, the nodes in the network are open to coalition with these malicious nodes, we discussed and analyzed several message distribution schemes in terms of secure delivery of messages. We also proposed a novel method of two period spraying in which routing of messages is risked when the remaining time to delivery deadline gets closer. By our initial simulations and analysis, we showed that two period spraying protocol achieves better delivery ratio at larger TTLs which can not be achieved by other methods. We believe that our secure delivery definition with the proposed two period spraying protocol will lead to a new studies of the routing problem in delay tolerant networks with limited trust between nodes (compromised DTNs).

References

1. Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L. S., & Rubenstein, D. (2002). Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet. In *Proceedings of the ACM ASPLOS*.
2. *Disruption tolerant networking*. http://www.darpa.mil/ato/solicit/DTN/.
3. Ott, J., & Kutscher, D. (2005). A disconnection-tolerant transport for drive-thru internet environments. In *Proceedings of the IEEE INFOCOM*.
4. Burgess, J., Gallagher, B., Jensen, D., & Levine, B. N. (2006). Maxprop: Routing for vehicle-based disruption-tolerant networking. In *Proceedings of the INFOCOM*, pp. 1–11.
5. Bulut, E., Geyik, S., & Szymanski, B. (2010). Efficient routing in delay tolerant networks with correlated node mobility. In *Proceedings of the 7th IEEE international conference on mobile ad-hoc and sensor systems (MASS)*.
6. Daly, E., & Haahr, M. (2007). Social network analysis for routing in disconnected delay-tolerant manets. In *Proceedings of the ACM MobiHoc*.
7. Bulut, E., & Szymanski, B. K. (2012). Exploiting friendship relations for efficient routing in mobile social networks. In Ivan Stojmenovic (Ed.), *IEEE transactions on parallel and distributed systems* (Vol. 23(12), pp. 2254–2265). New York: IEEE.
8. Lindgren, A., Doria, A., & Schelen, O. (2003). Probabilistic routing in intermittently connected networks. *SIGMOBILE Mobile Computing and Communication Review, 7*(3), 19–20.
9. Vahdat, A., & Becker, D. (2000). *Epidemic routing for partially connected ad hoc networks*. Duke University, Tech. Rep. CS-200006.
10. Spyropoulos, T., Psounis, K., & Raghavendra, C. S. (2008). Efficient routing in intermittently connected mobile networks: The multiple-copy case. *IEEE/ACM Transactions on Networking, 16*(1), 77–90.
11. Wang, Y., Jain, S., Martonosi, M., & Fall, K. (2005). Erasure coding based routing for opportunistic networks. In *Proceedings of the ACM SIGCOMM workshop on delay tolerant networking (WDTN)*.
12. Bulut, E., Wang, Z., & Szymanski, B. (2010). Cost efficient erasure coding based routing in delay tolerant networks. In *Proceedings of the ICC, South Africa*.
13. Lin, Y., Li, B., & Liang, B. (2008). Efficient network coded data transmissions in disruption tolerant networks. In *Proceedings of the IEEE INFOCOM*.
14. Burgess, J., Bissias, G. D., Comer, M. D., & Levine, B. N. (2007). Surviving attacks on disruption-tolerant networks without authentication. In *Proceedings of the Mobihoc '07*, pp. 61–70.
15. Choo, F. C., Chan, M. C., & Chang, E. (2010). Robustness of DTN against routing attacks. In *Proceedings of the second international conference on communication systems and networks (COMSNETS)*, pp. 1–10.
16. Li, F., & Wu, J. (2009). Thwarting blackhole attacks in disruption–tolerant networks using encounter tickets. In *Proceedings of the INFOCOM*, pp. 2428–2436.
17. Nelson, S. C., Bakht, M., & Kravets, R. (2009). Encounter-based routing in DTNs. In *Proceedings of the IEEE Infocom*, Rio De Janeiro, Brazil, pp. 846–854.
18. Ren, Y., Chuah, M. C., Yang, J., & Chen, Y. (2010). Muton: Detecting malicious nodes in disruption-tolerant networks. In *Proceedings of the WCNC*.
19. Ren, Y., Chuah, M. C., Yang, J., & Chen, Y. (2010). Detecting blackhole attacks in disruption-tolerant networks through packet exchange recording. In *Proceedings of the 1st workshop on D-SPAN (colocated with WoWMoM)*.

20. Li, Q., & Cao, G. (2012). Mitigating routing misbehavior in disruption tolerant networks. In *IEEE transactions on information forensics and security* (to appear).
21. Dini, G., & Duca, A. L. (2010). A reputation-based approach to tolerate misbehaving carriers in delay tolerant networks. In *Proceedings of the 15th IEEE symposium on computers and communications*, Italy.
22. Li, N., & Das, S. K. (2010). RADON: Reputation-assisted data forwarding in opportunistic networks. In *Proceedings of the MobiOpp*, pp. 8–14.
23. Chen, I. R., Bao, F., Chang, M. J., & Cho, J. H. (2010). Trust management for encounter-based routing in delay tolerant networks. In *IEEE global communications conference*, Miami, USA.
24. Adali, S., Escriva, R., Hayvanovych, M., Magdon-Ismail, M., Szymanski, B., Wallace, W., et al. (2010). Measuring behavioral trust in social networks.*In IEEE innternational conference on intelligence and security informatics (ISI 2010)*, pp. 150–152, Vancouver, BC, May 23–26, 2010.
25. Bulut, E., Wang, Z., & Szymanski, B. (2010). Cost effective multi-period spraying for routing in delay tolerant networks. In *IEEE/ACM transactions on networking*, Vol. 18.
26. Spyropoulos, T., Psounis, K., & Raghavendra, C. S. (2006). Performance analysis of mobility-assisted routing. In *MobiHoc*.
27. Srinivasa, S., & Krishnamurthy, S. (2009). CREST: An opportunistic forwarding protocol based on conditional residual time. In *Proceedings of the SECON*.
28. *A European Union funded project in situated and autonomic communications*. www.haggleproject.org.
29. Leguay, J., Lindgren, A., Scott, J., Friedman, T., Crowcroft, J., & Hui, P. (2006). *CRAWDAD data set upmc/content (v. 2006-11-17)*. Downloaded from http://crawdad.cs.dartmouth.edu/upmc/content.
30. Tournoux, P. U., Leguay, J., Benbadis, F., Conan, V., Amorim, M., & Whitbeck, J. (2009). The accordion phenomenon: Analysis, characterization, and impact on DTN pouting. In *Proceedings of the INFOCOM*.
31. Chaintreau, A., Hui, P., Crowcroft, J., Diot, C., Gass, R., & Scott, J. (2006). Impact of human mobility on the design of opportunistic forwarding algorithms. In *Proceedings of the INFOCOM*.
32. Zhang, X., Kurose, J. F., Levine, B., Towsley, D., & Zhang, H. (2007). Study of a bus-based disruption tolerant network: Mobility modeling and impact on routing. In *Proceedings of the ACM MobiCom*.
33. Pujol, J. M., Toledo, A. L., & Rodriguez, P. (2009). Fair routing in delay tolerant networks. In *Proceedings of the IEEE INFOCOM*.
34. Bulut, E., & Szymanski, B. (2011). On secure multi-copy based routing in delay tolerant networks. In *Proceedings of the first international workshop on privacy, security and trust in mobile and wireless systems (MobiPST 2011), in conjunction with ICCN*.

Author Biographies

Eyuphan Bulut (M'08) received the B.S. and M.S. degrees in computer engineering from Bilkent University, Ankara, Turkey, in 2005 and 2007, respectively. Then, in May 2011, he received his Ph.D. degree in the Computer Science Department of Rensselaer Polytechnic Institute (RPI), Troy, NY, USA. Now, he is with Mobile Internet Technology Group (MITG) of Cisco Systems in Richardson, TX, USA. His interests include design of protocols for wireless sensor and ad hoc networks such as routing protocols for delay-tolerant networks.

Boleslaw K. Szymanski (M'82 F'99) is the Claire and Roland Schmitt Distinguished Professor of Computer Science and the Director of the Social Cognitive Academic Research Center led by RPI. He received his Ph.D. in Computer Science from National Academy of Sciences in Warsaw, Poland, in 1976. He is an author and co-author of over 300 publications and an editor of five books. He is also an Editor-in-Chief of Scientific Programming. He is an IEEE Fellow and a member of the ACM for which he was a National Lecturer. His interests focus on parallel and distributed computing and networking.