DETECTING SELFISH NODE IN MANET USING COLLABORATIVE CONTACT BASED WATCHDOG

Mr. S. S. Ingole[1] Miss. Pallavi Bankar[2] Mr. R. S. Jamgekar[3]

[1]*Assistant Professor, SKNSCOE, Korti, Pandharpur, Solapur University, Solapur, India*

[2]*ME- CSE, SKNSCOE, Korti, Pandharpur, Solapur University, Solapur, India*

[3]*Assistant Professor, SKNSCOE, Solapur, Solapur University, Solapur, India*

[1]*Sumeet.ingole@sknscoe.ac.in*

2pallavibankar13@gmail.com

[3]*rs.jamgekar@gmail.com*

ABSTRACT*:- Wireless mobile ad hoc networks are dynamic networks, self-configuring in that nodes are free to move .Mobile ad-hoc networks (MANETs) assume that mobile nodes voluntary cooperate in order to works properly. So, this cooperation is a cost-intensive activity and that nodes can refuse to cooperate, leading to selfish node behaviour. So in this way, the overall network performance could be affected. The watchdog is used to well-known mechanism to detect a selfish node. The detection process is performed by watchdog can fail, generating false positives and false negatives that can induce to wrong operations. Moreover, the relying on local watchdogs alone can lead to poor performance when detecting selfish nodes, in term of precision and speed. This is specially important on networks with sporadic contacts, such as delay tolerant networks (DTNs), where sometime watchdog lack of enough time or information to detect the selfish nodes. The collaborative contact-based watchdog (CoCoWa) is a collaborative approach based on the diffusion of local selfish nodes awareness when a contact occurs, that information about selfish nodes is quickly propagated. The collaborative approach reduces the time and increases the precision when detecting selfish nodes.

Index Terms—*Wireless networks, opportunistic and delay tolerant networks, selfish nodes*

1 INTRODUCTION

Cooperative networking is a currently receiving significant attention as an emerging network design strategy for future mobile wireless networks. The successful cooperative networking can prompt the development of advanced wireless networks to cost-effectively provides services and applications in a contexts such as vehicular ad hoc networks (VANETs) or mobile social networks. Two basic technologies that are considered as the core for these types of networks are mobile ad-hoc networks and opportunistic and delay tolerant networks.

Mobile nodes are directly communicated with each other, if a contact occurs. This cooperation is a cost intensive activity for mobile nodes. So in the real world, nodes could have a selfish behavior, being reluctant to forward packets for others. Selfishness means that some nodes fail to forward other nodes packets to save their own resources.

The literature review provides two main strategies to deal with selfish behavior: a) motivation or incentive based approaches, b) detection and exclusion. First approach, tries to motivate nodes to actively participate in the forwarding activities.

The impact of node selfishness in MANETs has been studied in [7], [8], [9]. It is shown that, when no selfishness prevention mechanism is present, So the packet delivery rates are become seriously degraded, from a rate of 80 % when the selfish node ratio is 0 - 30% when the selfish node ratio is 50 %. The survey shows similar results: the number of packet losses is increased by 500 %, when the selfish node ratio increases from 0 – 40 %.

A more detailed study [7] shows that a moderate concentration of node selfishness (starting from a 20 %level) has a huge impact on the overall performance of Mobile Adhoc Networks, such as the average hop count, the number of packets dropped, the throughput, and the probability of reachability. In Delay Tolerant Networks, selfish nodes are seriously degrade the performance of packet transmission. So, detecting such nodes quickly and accurately is essential for the overall performance of the network. Previous works have demonstrated that watchdogs are the appropriate mechanisms to detect misbehaving and the selfish nodes.

Watchdog systems is overhear wireless traffic and analyze it to decide whether the neighbour nodes are behaving in a selfish manner. The watchdog detects a selfish node it is marked as a positive detection (or a negative detection, if it is detected as non selfish node). Though, watchdogs can fail that detection, generating false positives and false negatives that degrade the behavior of a system.

This paper introduces a Collaborative Contact-based Watchdog (CoCoWa) as a new scheme for detecting selfish nodes that combines the local watchdog detections and the dissemination of this information on the network. If one node has previously detected a selfish node and it can transmit this information to other nodes when a contact occurs. This way, nodes have second hand information about selfish nodes in a network. The goal is to reduce the detection time and to improve the precision by reducing effect of both false negatives and false positives.

2. LITERATURE SURVEY

Sybil attacker can create the more than one identity on a single physical device in order to launch a coordinated attack on the network or can switch the identities in order to weaken the detection process, thereby promoting lack of accountability in the network. In this paper, we propose a lightweight scheme to detect the new identities of Sybil attackers without using a centralized trusted third party or any extra hardware, such as directional antennae or geographical positioning system [1].

The Ad hoc networks rely on the cooperation of the nodes participating in the network to forward the packets for each other. A node may decide not to be cooperate to save its resources while still using the network to relay its traffic. If so many nodes exhibit this behavior, network performance degrades and cooperating nodes may be find themselves unfairly loaded. If a node is observes another node not participating correctly, it reports this observation to other nodes who then take action to avoid being affected and the potentially punish the bad node by refusing to forward its traffic[2].

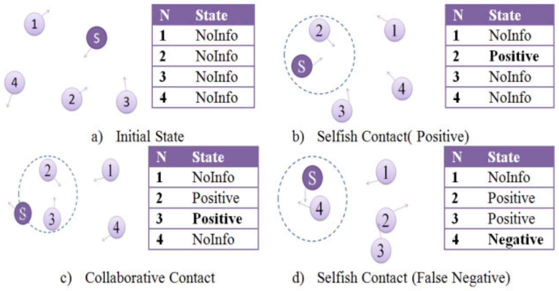
In this paper, describe the use of a self-policing mechanism based on reputation to enable mobile ad hoc networks to keep functioning despite the presence of misbehaving nodes. The reputation system in all nodes makes them detect misbehavior locally by observation and use of the second-hand information. Once a misbehaving node is detected it is automatically isolated from the network. So, we explain in particular how it is possible to use second-hand information while the mitigating contamination by the spurious ratings[3].

We can see, the problem of service availability in mobile ad-hoc WANs. We present a secure mechanism to stimulate end users to keep their devices turned on, to refrain from overloading the network, and to oppose the tampering aimed at converting the device into the "selfish". So, the our solution is based on application of the tamper resistant security module in each device and a cryptographic protection of message[4].

In this paper, we can see the each node have its own authority and tries to maximize the benefits it gets from the network. So, we assume that the nodes are not willing to forward packets for the benefit of the other nodes. This problem may be arise in civilian applications of the mobile ad hoc networks. So, in order to stimulate the nodes for the packet forwarding, we propose a simple mechanism based on a counter in each node. [5].

3. ARCHITECTURE OVERVIEW

The selfish node usually denies packet forwarding in order to save its own resources. Such type of behaviour implies that the selfish node is neither participates in routing nor relays data packets. A common technique to detect this selfish behaviour is network observe using local watchdogs. A node's watchdog consists on overhearing the packets transmitted and received by its neighbours in order to detect the anomalies, such as ratio between packets received to packets being re-transmitted. So, Byusing this technique, the local watchdog can generate a positive (or negative) detection in case the node is acting selfish (or not).



N	State
1	NoInfo
2	NoInfo
3	NoInfo
4	NoInfo

a) Initial State

N	State
1	NoInfo
2	**Positive**
3	NoInfo
4	NoInfo

b) Selfish Contact(Positive)

N	State
1	NoInfo
2	Positive
3	**Positive**
4	NoInfo

c) Collaborative Contact

N	State
1	NoInfo
2	Positive
3	Positive
4	**Negative**

d) Selfish Contact (False Negative)

Fig. 1: An example of how CoCoWa works. a) Initially all nodes have no information about the selfish node. b) Node 2 detects the selfish node using its own watchdog. c) Node 2 contacts with node 3 and it transmit the positive about the selfish node. d) The local watchdog of Node 4 fails to detect the selfish node and it generates a negative detection (a false negative)

The example of how CoCoWa works is outlined in Fig. 1. It is based on the combination of a local watchdog and the diffusion of information when a contact between pairs of nodes occurs. A contact is defined as an opportunity of transmission between a pair of nodes. Assuming that there is only one selfish node, above figure shows how initially no node has information about the selfish node. When node detects a selfish node by using its watchdog, it is marked as a positive, and if it detected as a non selfish node, it marked as a negative. After on, when this node contacts to another node, it transmit this information to it; so, from that moment on, both nodes are store information about this positive (or negative) detections. So, node can become aware about selfish nodes directly or indirectly, through a collaborative transmission of information that is provided by other nodes.

This scheme is the uncontrolled diffusion of positive and negative detections can produce the fast diffusion of wrong information, therefore, a poor network performance. For example, in figure 1, on the last state d), node two and three have positive detection and node four has negative detection (a false negative). Now, node one, which has the no information about selfish node, has several possibilities: if it contacts the selfish node it may be able to detect it; if it contacts a node two or three it can get positive detection; but if it contacts node four, it can get a false negative.

Figure 2 shows the functional structure of CoCoWa

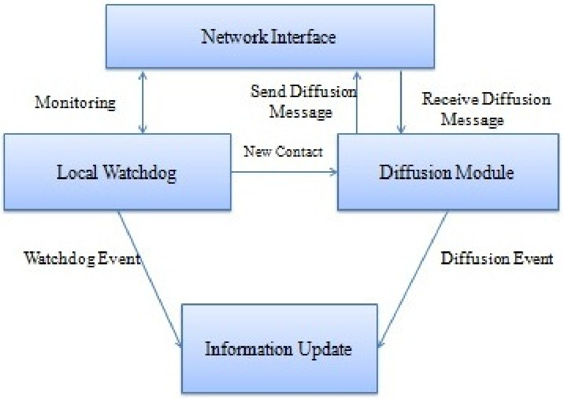


Fig. 2 :COCOWA Architecture

The Local Watchdog having two functions: the detection of selfish nodes and the detection of new contacts. Local watchdog can generate following events about the neighbor nodes: PosEvt (positive event) when the watchdog detects selfish node, NegEvt (negative event) when watchdog detects that node is not selfish, and NoDetEvt (no detection event) when watchdog does not have the enough information about a node. The detection of new contacts is based on neighbourhood packet overhearing; thus, when watchdog overhears packets from a new node it is assumed to be a new contact, and so it generates an event to the network information module.

A Diffusion module has two functions: the transmission as well as the reception of positive and negative detections. The key issue of our approach is the diffusion of information. The number of selfish nodes is low compared to the total number of nodes; the positive detections can always be transmitted with a low overhead. So, transmitting only positive detections is a drawback: false positives can be a spread over the network very fast. Thus, the transmission of negative detections is a necessary to neutralize the effect of this false positives, but sending all known negative detections can be troublesome, producing excessive messaging or fast diffusion of false negatives. Consequently, we introduce a negative diffusion factor g, that is the ratio of negative detections that are actually transmitted. This value ranges from 0 to 1. Finally, when diffusion module receives new contact event from the watchdog, it transmits message including this information to new neighbor node. When the neighbour node receives a message, it generates an event to the network a information module with the list of these positive (and negative) detections.

The updating or consolidating the information is the another key issue. This is a function of the Information Update module. A node have the following internal information about other nodes: NoInfo state, Positive state and Negative state. A NoInfo state means that it has no enough information about a node, a Positive state means it believes that node is selfish, and a Negative state means it believes that node is not selfish. The node have direct information (from local watchdog) and indirect information (from the neighbour nodes). CoCoWa is event driven, so the state of a node is updated when the PosEvt or NegEvt events are received from local watchdog and diffusion modules. In particular, these events update a reputation value ρ using following expression:

$$P=p+\Delta \quad \Delta=\begin{cases} +\delta & (\mathrm{PosEvt, Local}) \\ +1 & (\mathrm{Posevt, Indirect}) \\ -\delta & (\mathrm{NegEvt, Local}) \\ -1 & (\mathrm{NegEvt, Indirect}) \end{cases} \qquad \delta \geq 1 \qquad (1)$$

So, a PosEvt event increments reputation value while NegEvt event decrements it. Defining θ as a threshold and using reputation value ρ, and the state of the node changes to Positive if $\rho \geq \theta$, and to Negative if $\rho \leq -\theta$. Otherwise, the state is a NoInfo. The combination of δ and θ parameters allows is a very flexible and dynamic behaviour. First of all, if $\theta > 1$ and $\delta < \theta$ we need a several events in order to change the state. For example, starting from the NoInfo state, and if $\theta = 2$ and $\delta = 1$, at least a local and an indirect event is needed to change the state, but if $\theta = 1$, only one event is a needed. Second, we can give a more trust to the local watchdog or to a indirect information. For example, a value of $\delta = 2$ and $\theta = 3$, means that we needed one local event and one indirect event, or three indirect events, to change state. This approach can be compensating wrong local decisions: for example, a local NegEvt can be compensated by $2\delta + \theta$ indirect PosEvt events, and in the order to change from Positive to Negative states (or vice-versa) we need twice the events.

The advantages of the such type of strategy are twofold. First, with the threshold θ we can reduce the fast diffusion of the false positive and the false negatives. However, this can be produce a delay on the detection. Second, the decision about selfish node is taken by using the most recent information. For example, if a node had contact with selfish node a long time ago (so it had a Positive state) and now receives several NegEvt in a row from the other nodes, the state is updated to Negative.

The network information about nodes has an expiration time, so after some time without contacts it is updated. The implementation of such mechanism is straightforward. When the event is received, it is marked with a time stamp, so the given timeout an opposite event is generated, in a order to update the value of ρ.

4. SYSTEM MODEL

The network is modeled as set of N wireless mobile nodes, with C collaborative nodes, M is malicious nodes and S is selfish nodes ($N = C + M + S$). The goal is to obtain time and overhead that a set of $D \leq C$ nodes need to detect the selfish nodes in the network. The overhead is number of information messages transmitted up to the detection time. The following models evaluate the detection of single selfish node. The effect of having a several selfish nodes in network is easy to evaluate, and it does not required to a specific model. We assume that selfish nodes are not cooperative, we can analyse the impact of each selfish node on a network independently. In case of several selfish nodes ($S > 1$) on a network with N nodes, we can assume that $C = N - S$ are cooperative nodes.

4.1 The Model for the CoCoWa Architecture

The goal of this model is the behaviour of the different modules of our architecture (see figure 2).The local *watchdog* is modeled using three parameters: the probability of detection $\mathrm{p_d}$, the ratio of false positives p_{fp}, and the ratio of false negatives $\mathrm{p_{fn}}$. The first parameter, the probability of detection ($\mathrm{p_d}$), reflects the probability that, when a node contacts to the another node, the watchdog has enough information to generate a PosEvt or NegEvt event. So, this value depends on effectiveness of the watchdog, the traffic load, and the mobility pattern of nodes. For example, for Opportunistic Networks or DTNs, the contacts are sporadic and have low duration, this value is lower than for MANETs. Moreover, the watchdog can a generate false positives and false negatives. A false positive is when watchdog generates positive detection for a node that is not selfish node. A false negative is generated when selfish node is marked as negative detection. In order to measure the performance of a watchdog, these values can be expressed as a ratio or probability: p_{fp} is the ratio (or probability) of false positives generated when a node contacts a non-selfish node, and p_{fn} is the ratio (or probability) of false negatives generated when a node contacts a selfish node. By using the previous parameters we have model the probability of generating local PosEvt and NegEvt events when a contact occurs:

- PosEvt event: the node contacts with a selfish node and the watchdog detects it, with probability $\mathrm{p}_{\mathrm{d}}(1-p_{fn})$. Note down the false positive can also be generated with probability $p_d \cdot \mathrm{p}_{\mathrm{fp}}$.
- NegEvt event: the node contacts with a non-selfish node and detect it with probability $\mathrm{p}_{\mathrm{d}}(1-p_{fp})$. A false negative can also be generated when it contacts with the selfish node with probability $p_d \cdot \mathrm{p}_{\mathrm{fn}}$.

So, the diffusion module can be generate indirect events when a contact with neighbour nodes occurs. Nevertheless, a contact does not always imply collaboration, so we have model this probability of collaboration as pc. The degree of collaboration is a global parameter, and it used as a reflect that either a message with the information about the selfish node is lost, or such node temporally does not collaborate. In a real networks, full collaboration (p_c= 1) is almost impossible. Finally, the probabilities of generating the indirect events are the following:

- PosEvt event: When a contact with another node that has a Positive state of the selfish node with probability pc.
- NegEvt event: When a contact with another node that has a Negative state, being the probability $\gamma \cdot \mathrm{p}_c$. Note down a not all Negative states are to be transmitted, it depends on a diffusion factor γ.

The information update module is driven by previous local and indirect events. These event are update the reputation ρ about a node, and it is used to finally decide if node is a selfish or not by using the threshold θ.

4.2 Malicious Nodes and Attacker Model

The malicious nodes attempt to attack the CoCoWa system by generating a wrong information about the nodes. So, the attacker model is addresses the behaviour or the capabilities of such malicious nodes. A malicious node attack consists of trying to send a positive about a node that is not the selfish node, or a negative about the selfish node, with the goal of producing a false positive and a false negative on the rest of nodes. In order to do this, it must have the knowledge about how CoCoWa works. The effectiveness of this behaviour clearly depends on the rate and precision that malicious nodes can be generate wrong information. The malicious nodes are assumed to have communications hardware similar to the rest of a nodes, so they can be hear all neighbour messages in a similar range than the rest of nodes. Nevertheless, the attacker could uses a high-gain antennas to increase its communications range and thus disseminate false information in a more effective manner.

Regarding the diffusion of information on the network, our approach is does not assume any security measures, such as a message cyphering or a node authentification. Nevertheless, if these measures exist, the effect of the malicious nodes in CoCoWa will be very reduced or even non-existent. The diffusion module can also accept messages from every node, including from malicious ones. So, we assume that the malicious nodes can be active, and use this information in order to generate wrong positives/negatives about other nodes. Nevertheless, we assume that malicious nodes cannot impersonate other nodes and do not a collude with other malicious nodes . Another problem is the Sybil attack. Since the malicious nodes can be create and control more than one identity on single physical device, it can have a serious impact on CoCoWa.

A behaviour of malicious nodes is modeled from the receiver perspective, which is a based on probability of receiving wrong information about given node when a contact with a malicious node occurs (that is, it receives a Negative about selfish node, and a Positive about other nodes). We denote this behaviour as the maliciousness probabilityp_m.

Following give the details of several aspects that can affect this probability:

1) The reception of the information, considering that not all contacts produce this reception. This aspect is similar to the collaboration degree, but an increase of communication range of the malicious nodes will increase information reception.
2) The malicious node does not have information about all nodes; so, in order to send a positive/negative about a node, they must have contacted to this node previously or have received a message from other nodes.
3) Another issue is to consider the proper generation of wrong information, for example when the receiving a positive of a node that is not the selfish node. From the receiver point of view, a perfect malicious node will always provide wrong information. In such case, malicious node, in order to send the wrong information, must know the state of each node. In other words it must have been a perfect local watchdog.

Summing up, above parameter reflects the average intensity or the effectiveness of the attack of the malicious nodes.

4.*3 The Model for the Detection of Selfish Node*

In this section we introduce an analytical model for evaluating the performance of the CoCoWa. The goal is to obtain the detection time of selfish node in a network. This model takes into account the effect of false negatives. The false positives does not affect on the detection time of the selfish node, so p_{fp} is not introduced in this model.

Using λ as the contact rate between the nodes, we can model the network by using a 4D Continuous Time Markov chain (4DCTMC). For modeling purposes, the collaborative nodes are divided into two sets: a set with D destination nodes, and a set of E = C − D as intermediate nodes. The destination and the intermediate nodes have the same behaviour (both are collaborative nodes). The only purpose of such division is to the analytically obtain the time and the overhead required for the subset of destination nodes to detect a selfish node. Thus, the 4D-CTMC states are: $(d_p(t), d_n(t), e_p(t), e_n(t))$, where $e_p(t)$ represents number of intermediate nodes that have a Positive state, $e_n(t)$ the intermediate nodes with a Negative state, $d_p(t)$ the destination nodes with the Positive state and $d_n(t)$ the destination nodes with the Negative state. Note down, in this model, a Negative is a false negative. The states must be verify the following conditions are: $d_p(t) + d_n(t) \leq D$ and $e_p(t)+e_n(t) \leq E$. Our 4D-CTMC model has an initial state (0, 0, 0, 0) (that is, all the nodes have no information). The final states are when $d_p(t) = D$. We define υ as the number absorbing states, that are all the possible permutations of states ({(D, 0, *, *)}) that sum E. It is easy to derive that $\upsilon = p^s(E) = 0.5(E + 1)(E + 2)$. The number of transient states are τ is obtained in similar way:

$\tau = (p^s(D) - 1)\, p^s(E)$. So this model can be expressed by using the following generator matrix is Q:

$$Q = \begin{pmatrix} T & R \\ 0 & 0 \end{pmatrix}, \qquad (2)$$

Where T is a $\tau \times \tau$ matrix with elements q_{ij} denoting the transition rate from transient state s_i to transient state s_j, R is a $\tau \times \upsilon$ matrix with elements q_{ij} denoting the transition rate from the transient state is s_i to the absorbing state s_j, the left 0 is a $\upsilon \times \tau$ zero matrix, and the right 0 is a $\upsilon \times \upsilon$ zero matrix. Now, we have derive the transition rates are q_{ij} . Given the state $s_i = (e_p, e_n, d_p, d_n)$, we have:

$$q_{ij} = \begin{cases} R_p(E - e_p - e_n) & e_p + \\ R_{fn}(E - e_p - e_n) & e_n + \\ R_{fn} e_p & e_p - \\ R_p e_n & e_n - \\ R_p(D - d_p - d_n) & d_p + \\ R_{fn}(D - d_p - d_n) & d_n + \\ R_{fn} d_p & d_p - \\ R_p d_n & d_n - \end{cases} \qquad (3)$$

Where x+ are represents the transition from a state (· · · , x, · · ·) to (· · · , x + 1, · · ·), and x− represents transition from state (· · · , x + 1, · · ·) to (· · · , x, · · ·). Finally, $q_{ij} = -\sum_{i \neq j} q_{ij}$.

The first transition e_p+ is when a intermediate collaborative node changes from NoInfo state to a Positive state $((d_p, d_n, e_p, e_n)$ to $(d_p, d_n, e_p + l, e_n))$. The rate of change depends on the updating of ρ, and on the δ and θ parameters. The reputation value is ρ increments according to the expression 1. This update can be generated by local events and indirect events. First, the local watchdog can be generate a local PosEvt with rate $\lambda p_d(1 - p_{fn})$ so the reputation is incremented by δ. Then, the rate of increment due to the local events is $\lambda \delta p_d (1 - p_{fn})$. Second, updating from an indirect event depends on a number of nodes with Positive and Negative states and the probability of collaboration: $\lambda p_c(c_p - \gamma c_n)$ where $c_p = e_p + d_p$ and $c_n = e_n + d_n$. Malicious nodes affect this updating by generating indirect NegEvt with a rate $\lambda M p_m$. Since we are evaluating the increment, this term must be positive. So, the final rate due to indirect events is $\lambda \max(p_c(c_p - c_n) - M p_m)$. All the previous terms are divided by threshold θ in order to obtain the rate of changing when a node contacts with a collaborative node:

$$R_p = \lambda(\delta p_d(1 - p_{fn}) + \max(p_c(c_p - \gamma c_n) - M p_m, 0))/\theta \qquad (4)$$

Finally, there are $(E - e_p - e_n)$ nodes with the NoInfo state so the final transition rate is $R_p(E - e_p - e_n)$.

The second transition, e_n+, is when a intermediate collaborative node changes from $((d_p, d_n, e_p, e_n)$ to $(d_p, d_n, e_p, e_n + 1)$. This means that a intermediate collaborative node is changes to the Negative state (a false negative). We can derive a similar expression for the rate of change to a (false) Negative state R_{fN}. In this case, when a node contacts with the selfish node, the reputation is decreased with rate $\lambda \delta p_d p_{fn}$, and also by indirect events with rate $\lambda(p_c(\gamma c_n - c_p) + M p_m)$. Finally, we have:

$$R_{fn} = \lambda(\delta d_p p_{fn} + \max(p_c(\gamma c_n - c_p) + M p_m, 0))/\theta \qquad (5)$$

and the transition is $R_{fn}(E - e_p - e_n)$.

The transition e_p- is when the intermediate collaborative node that has a Positive state changes to the NoInfo. So, this event is similar to e_n+ and the transition rate is similar:$R_{fn} e_p$. Note that in this case we can multiply by the number of nodes that have the Positive state instead of a number of pending nodes. In similar way, the transition e_n+ occurs when a intermediate collaborative node that has the Negative state changes to NoInfo. So, transition rate is $R_p e_p$. For transitions regarding destination nodes, the rates is very similar to the previous ones, as seen in the expression 3. Finally, all such transitions retain the exponential distribution of the useful contacts, preserving the Markovian nature of the process.

By using the generator matrix Q we can derive two different expressions: one for a detection time T_d and another for a overall overhead (or cost) O_d. Starting with the detectison time, from the 4D-CTMC we can be obtain how long it will be take for the process to be absorbed. Using the fundamental matrix $N = -T^{-1}$, so ,we can obtain a vector t of the expected time to absorption as $t = N_v$, where v is column vector of ones ($v = [1, 1, \ldots 1]^T$). Each entry t_i of t is represents the expected time to absorption from the state si. Since we only need the expected time from state $s_1 = (0, 0, 0, 0)$ to absorption (that is, the expected time for all the destination nodes to have a Positive state), the detection time T_d, is:

$$T_d = E[T] = v_1 N_v \qquad (6)$$

Where T is a random variable denoting the detection time for all nodes and $v_1 = [1, 0, \ldots, 0]$. Concerning the overhead we have need to obtain the number of transmitted messages for each state is si. First, the duration of each state is si can be obtained using the fundamental matrix N. By definition, the elements of first row of N are to be the expected times in each state starting from state 0. Then, the duration of state s_i is $f_i = N(1, i)$. Now, we calculate the expected number of message sm_i. The number of messages depends on a diffusion model. So the easier exposition, we can start with $\gamma = 0$, that is, only the positive detections are transmitted. From state $s_1 = (0, 0, 0, 0)$ to $s_{E+1} = (0, 0, 0, E)$ no node has a Positive state, so no messages are transmitted and m1 = 0. From states $s_{E+2} = (0, 0, 1, 0)$ to $s_{2E+1} = (0, 0, 1, E - 1)$, one node has a Positive state. In such cases, the Positive can be transmitted to the all nodes (except itself) for the duration of each state i (N(1, i)) with the rate λ and the probability p_c. Then, the expected number of messages can be obtained as $m_i = N(1, i)\lambda(C - 1)p_c$. From states $s_{2E+2} = (0, 0, 2, 0)$ to $s_{3E+1} = (0, 0, 2, E - 2)$, we have two possible senders and $m_i = 2N(1, i)\lambda(C - 1)p_c$. Considering both types of nodes (destination and intermediate), the number of nodes with a Positive for state s_i is $\Phi(s_i) = d_p + e_p$. Summarizing, the overhead of transmission (number of messages) is:

$$O_d = E[Msg] = \lambda(C - 1)\, p_c \sum_{i=1}^{\tau} \Phi(s_i) N(1, i). \qquad (7)$$

Finally, for γ > 0, the ratio of nodes cn that will be transmit a Negative is precisely γ, so $\Phi(s_i) = d_p + e_p + \gamma(d_n + e_n)$.

By using the previous model, we can evaluate the time when the destination nodes D have a “false negative” about the selfish node. In this case absorbing states are {0, D, *, *}, that is, when $d_n = D$. A high rate of the false negatives and the malicious nodes may be cause a false negative state to reached in less time than a true positive detection.

4.4 The Model for False Positives

This model describe, evaluating the effect of false positives. This model evaluates how fast a false positive spreads in the network (the diffusion time). So, in this case, a greater diffusion time stands for a lower impact of the false positives. The diffusion time is similar to detection time of true positives described in the previous subsection, and it can obtained in a similar way. Following process is same that in the previous model for false negatives, we have a 4D-CMTC with a same states (d_p,d_n, e_p, e_n), but in this case $c_p = d_p+e_p$ represents number of nodes with a false positive, and $c_n = d_n + e_n$ the number of nodes with a negative detection. We can derive expressions similar to 4 and 5, for the case of the false positives. So in this case, R_{fp} represents the rate of false positive, and it is derived in a similar way:

$$R_{fp} = \lambda(\delta p_d p_{fp} + \max(p_c(c_p - \gamma c_n) + M p_m, 0))/\theta \qquad (8)$$

and R_n represents the rate of negative detection:

$$Rn = \lambda(\delta p_d\ (1 - p_{fp}) + \max(p_c\ (\gamma c_n - c_p) - M p_m, 0))/\theta \qquad (9)$$

Using these expressions, the transition rates (q_{ij}) of the generator matrix Q are similar to expression 3, substituting R_p and R_{fn} by R_{fp} and R_n, respectively. Finally, by using equations 6 and 7 described in previous model. So, we can obtain the diffusion time and the overhead.

5. CONCLUSION

The CoCoWa as a collaborative contact-based watchdog to reduce the time and improve the effectiveness of detecting a selfish nodes, reducing the harmful effect of false positives, false negatives and a malicious nodes. CoCoWa is a based on diffusion of the known positive and the negative detections. When a contact is occurs between two collaborative nodes, the diffusion module is transmits and processes the positive (and negative) detections.

CoCoWa can be reduce the overall detection time with respect to original detection time when the collaboration scheme is not used, with reduced overhead (message cost). So, this reduction is very significant, ranging from 20 percent for very low degree of collaboration to the 99 percent for higher degrees of collaboration.

The combined effect of the collaboration and reputation of this approach can be reduce the detection time while increasing the global accuracy by using a moderate local precision watchdog.

REFERENCES

[1] S. Abbas, M. Merabti, D. Llewellyn-Jones, and K. Kifayat, “Lightweight sybil attack detection in manets,” IEEE Syst. J., vol. 7, no. 2, pp. 236–248, Jun. 2013.

[2] S. Bansal and M. Baker, “Observation-based cooperation enforcement in ad hoc networks” arXiv:cs.NI/0307012, 2003.

[3] S. Buchegger and J.-Y. Le Boudee, “Self-policing mobile ad hoc networks by reputation systems,” IEEE Commun. Mag., vol. 43, no. 7, pp. 101–107, Jul. 2005.

[4] L. Buttyan and J.-P. Hubaux, “Enforcing service availability in mobile ad-hoc WANs,” in Proc. 1st Annu. Workshop Mobile Ad Hoc Netw. Comput., 2000, pp. 87–96.

[5] L. Buttyan and J.-P. Hubaux, “Stimulating cooperation in selforganizing mobile ad hoc networks,” Mobile Netw. Appl., vol. 8, pp. 579–592, 2003.

[6] Enrique Hern andez-Orallo, Manuel David Serrat Olmos, Juan-Carlos Cano, Carlos T. Calafate, and Pietro Manzoni, “CoCoWa: A Collaborative Contact-Based Watchdog for Detecting Selfish Nodes”, IEEE Transaction on Mobile Computing, Vol. 14, No. 6, pp. 1162- 1175, June 2015.

[7] C. K. N. Shailender Gupta and C. Singla, “Impact of selfish node concentration in MANETs,” Int. J. Wireless Mobile Network., vol. 3, no. 2, pp. 29–37, Apr. 2011.

[8] C. Toh, D. Kim, S. Oh, and H. Yoo, “The controversy of selfish nodes in ad hoc networks,” in Proc. Adv. Commun. Technol., Feb. 2010, vol. 2, pp. 1087–1092.

[9] Y. Yoo, S. Ahn, and D. Agrawal, “A credit-payment scheme for packet forwarding fairness in mobile ad hoc networks,” in Proc. IEEE Int. Conf. Commun., May 2005, vol. 5, pp. 3005–3009.

[10] E. Hernandez-Orallo, M. D. Serrat, J.-C. Cano, C. M. T. Calafate, and P. Manzoni, “Improving selfish node detection in MANETs using a collaborative watchdog”,

IEEE Communication Letters, Vol. 16, No. 5, pp. 642–645, May 2012.

[11] R. Groenevelt, P. Nain, and G. Koole, "The message delay in mobile ad hoc networks", Performance Evaluation, Vol. 62, pp. 210–228, Oct. 2005.

[12] Reshma Lill Mathew, Prof. P. Petchimuthu ,"Detecting Selfish Nodes in MANETs Using Collaborative Watchdogs", International Journal of Advanced Research in Computer Science and Software Engineering ,Vol. 3, No. 3, pp. 37-41, March 2013.

AUTHORS

Miss. Pallavi Shankar Bankar received the Bachelor's of Engineering degree (B.E) in Computer Science and Engineering in 2013 SVERI's COE, Pandharpur. She is now pursuing Master's degree in Computer Science & Engineering at SKNSCOE, Pandharpur.

Prof. Mr. S. S. Ingole is currently working as Assistant Professor in Computer Engineering Department at SKNSCOE, Korti, Pandharpur.

Prof. Mr. R. S. Jamgekar is currently working as Assistant Professor in Computer Engineering Department at SKNSCOE, Korti, Pandharpur.