# Preventing malicious nodes in ad hoc networks using reinforcement learning

Karnkamon Maneenil, Wipawee Usaha

*Abstract*— **This paper proposes an enhancement to an existing reputation method for indicating and avoiding malicious hosts in wireless ad hoc networks. The proposed method combines a simple reputation scheme with a reinforcement learning technique called the on-policy Monte Carlo method where each mobile host distributedly learns a good policy for selecting neighboring nodes in a path search.**

**Simulation results show that the reputation scheme combined with the reinforcement learning can achieve up to 89% and 29% increase in throughput over the reputation only scheme for the static and dynamic topology case, respectively.**

*Keywords*—**Reputation, network security, malicious nodes, mobile ad hoc networks, reinforcement learning, .**

## I. Introduction

In wireless ad hoc networks, each host has a limited transmission range. Successful delivery of packets between hosts outside transmission range of each other therefore relies on cooperation of intermediate nodes. The fundamental assumption for such networks is that the nodes will cooperate and not misbehave. However, hosts join the network on the fly creating a dynamic topology network. The lack of a centralized network management leads ad hoc networks vulnerable to attacks by misbehaving nodes. Consequently, packets are dropped or even misdirected therefore resulting in low network throughput. Figure 1 illustrates an ad hoc network which contains malicious nodes in the shortest paths (that is, via nodes 10 or 5 and 6). With some quantification of node misbehavior, malicious nodes can be identified and the source node is able to send packets along an alternative path (that is, via nodes 1, 2, 3 and 4).

Recently, reputation schemes have been employed to identify and avoid malicious nodes [1]-[6]. The reputation of a node is a function of only the number of data packets that have been previously relayed by the node. Hence, nodes have high reputation when they successfully forward packets they receive. If the nodes have low reputation, it is subsequently weeded out from the ad hoc network. Reputation is an average of recommendations received by a node. Suppose node $A$ receives 100 packets and routed 60 packets but dropped 40 packets. Hence total reputation of node $A$ becomes $(60-40)/100 = 0.2$. Reputation values vary probabilistically depending on the traffic load and behavior of nodes themselves.

The outstanding features of reputation schemes include i) circumvention of malicious nodes, ii) promotion of cooperation among nodes, iii) decentralized collection and storage of reputations and iv) subsequently increase in the average throughput of the ad hoc network [1], [3], [5]. However, many of these works employ threshold reputation which is the minimum acceptable reputation a certain node must possess in order to be selected as the next hop node.

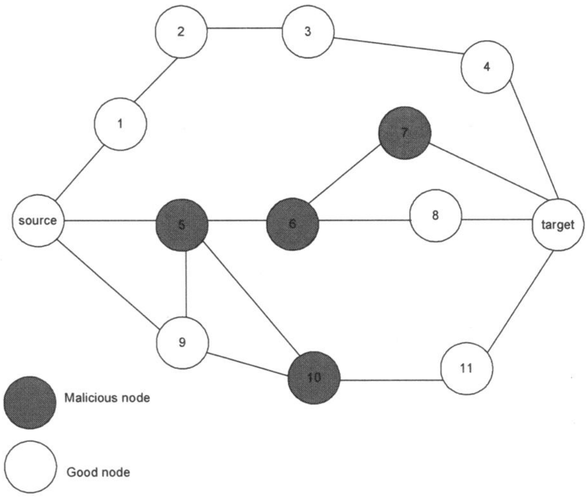

Fig. 1. An ad hoc network with malicious nodes.

The work in [2], [4] propose fixed-threshold reputation schemes for identifying trustworthy routers and relays. The work of [6] also studied reputation schemes with a fixed threshold for peer-to-peer networks.

In [2]-[6], fixed values of reputation thresholds have been selected to discriminate cooperative from noncooperative nodes. However, such static values may not be suitable for every ad hoc environment. In this paper, we integrate a reinforcement learning technique with an existing reputation scheme and determine a good rule to distinguish malicious. The advantage of our approach is that the rule is adaptive to the network dynamics because it is learned by interacting directly with the environment.

## II. Reputation as a reinforcement learning problem

Reinforcement learning is a computational approach for goal-directed learning and decision-making. It uses a formal framework defining the interaction between a learning agent and its environment in terms of states, actions and rewards. Figure 2 shows the agent-environment interaction in reinforcement learning. In this paper, we employ a learning approach based on sample episodes, called the *on-policy Monte Carlo* (ONMC) method [8]. This method uses sample episodes for estimating value functions which

School of Telecommunication Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand 30000 Corresponding author's email: wusaha@ieee.org

0-7803-9206-X/05/$20.00 ©2005 IEEE

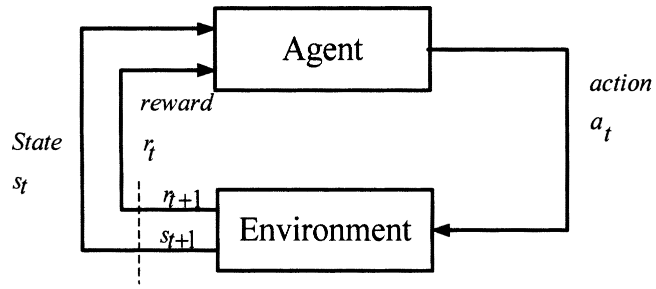

Fig. 2. Diagram of agent-environment interaction in reinforcement learning.

```
Initialize, for all $s \in X_s, a \in A_s$:
    $Q(s,a) \leftarrow$ arbitrary
    $Returns(s,a) \leftarrow$ empty list
    $\pi \leftarrow$ an arbitrary $\in$-soft policy
Repeat forever:
    (1) Generate an episode using $\pi$
    (2) For each pair $s,a$ appearing in the episode:
            $R \leftarrow$ return following the first occurrence of $s,a$
            Append $R$ to returns $(s,a)$
            $Q(s,a) \leftarrow average(Returns(s,a))$
    (3) For each $s$ in the episode:
            $a^* \leftarrow \arg\max_a Q(s,a)$
            For all $a \in A_s$:
                $\pi(s,a) \leftarrow \begin{cases} 1-\varepsilon+\frac{\varepsilon}{|A_s|} & if\ a=a^* \\ \frac{\varepsilon}{|A_s|} & if\ a\neq a^* \end{cases}$
```

Fig. 3. The $\varepsilon$–soft policy ONMC algorithm.

quantify an agent's experience. Value functions are functions of states or state-action pairs that estimate how good it is for the agent to be in a given state or state-action pair. Such functions are computed from average sample returns received from the environment operating under a fixed policy. The experience from the environment is divided into episodes, and it is assumed that each episode terminates eventually irrespective of whatever actions are taken. The ONMC method learns incrementally on an episode-by-episode basis—meaning that the value functions are estimated and policies are improved after each episode. Under certain assumptions, the ONMC method eventually converges to an optimal policy and optimal value functions—given only sample episodes and no other knowledge of the environment's dynamics. Figure 3 shows the pseudo code for the ONMC algorithm.

This method is selected because the episodic nature of route search process in wireless ad hoc networks. An episode starts immediately when a source node initiates a route search to a destination node, and terminates when the target node is found or the maximum number of hop count is reached. As the route search is executed, the intermediate nodes are selected hop-by-hop based on their reputation value. Each time the search is successful, a reward is to every node along all paths found. The goal is to find a rule that selects neighboring nodes based on their reputation values, which optimizes some performance criterion in finite horizon. The finite horizon problem is considered here due to the *episodic* nature of route search process.

## III. Problem formulation

The ONMC method can be applied to an actual ad hoc network or a simulator to obtain a good neighboring node selection policy. Consider a $\mathcal{N}$-node ad hoc network. Each source node $S$, maintains the reputation information to all its neighboring nodes. Suppose that nodes $A$, $B$, $C$, $D$ are neighbors of node $S$. Let $r_A$, $r_B$, $r_C$, $r_D$ be the reputation of nodes $A$, $B$, $C$, $D$ , respectively, where $0 \leq r_A, r_B, r_C, r_D \leq 1$. The reputation scheme is based on [1], [2]. Since the reputation values are real numbers, we quantize the state space at node $S$ as $X_S = \{x_S : [q_A, q_B, q_C, q_D]\}$ where $q_A$, $q_B$, $q_C$, $q_D$ are quantized reputation values of nodes $A$, $B$, $C$, $D$ , respectively. Let the action space at node $S$ be given by $A_S = \{a_S : [\delta_A, \delta_B, \delta_C, \delta_D]\}$ where $\delta_A$ ($\delta_B$, $\delta_C$, $\delta_D$) is the 1 if node $S$ selects node $A$ ($B$, $C$, $D$ ) in the route search and 0, otherwise. The process is repeated at every selected nodes until the destination node is found or the maximum number of hop counts is reached. If the route search is successful, then a reward of +1 is assigned to every node on all successful paths. Otherwise, a reward of 0 is assigned to all nodes involved in the route search. By using the ONMC method in this scenario, we are able to determine a near-optimal neighboring node selection policy based on reputation values.

## IV. Experimental results

We consider a wireless ad hoc network of 23 nodes which includes a number of misbehaving nodes. Two cases of topologies have been considered, i.e., the static and dynamic topology. In the latter case, the topology of the network is generated by a random connectivity model. Reputation values between 0 and 1 at each node reflects how trustworthy of a node is—the higher the reputation values, the more reliable the nodes are. Since reputation values are continuous-valued, the state space is quantized into 5 subintervals, $[0, 0.2)$, $[0.2, 0.4)$, $[0.4, 0.6)$, $[0.4, 0.6)$, $[0.6, 0.8)$ and $[0.8, 1)$ which are represented by integers 1, 2, 3, 4, 5, respectively. Hence, the state space for a give node $s$ has a total of $5^4 = 625$ possible states and for instance, the state $x_S = [1, 3, 2, 1]$ refers to the state when node $s$ has neighbors with reputation values $r_A, r_D \in [0, 0.2)$,$r_B \in [0.4, 0.6)$, $r_C \in [0.2, 0.4)$, etc.

We study three schemes, namely, reputation scheme with a fixed reputation threshold of 0.5 [2], and reputation scheme combined with the ONMC, and a shortest path

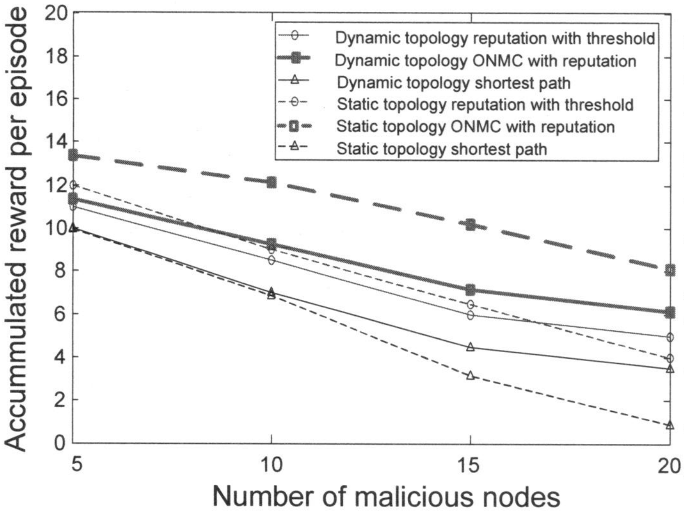

Fig. 4. Accumulated reward per episode as a function of the number of malicious nodes in the test network.

scheme which disregards the reputation values.

Figure 4 shows the accumulated reward per episode obtained in static and dynamic topologies as the number of malicious nodes in the network increases. The maximum number of allowed packets broadcasted in the network is 1000. Under both topologies, the ONMC scheme outperforms the other two schemes as the number of bad nodes increases. The reason is because ONMC can attain good policies for avoiding malicious nodes and is able to find more successful routes when compared to other schemes. Note that when multiple successful paths are found, a reward of +1 is assigned to every node on all successful paths. Therefore, the accumulated reward per episode of ONMC is the highest among the schemes.

Figure 5 shows the number of packets arrived at the destination in static and dynamic topologies as the number of malicious nodes in the network increases. The maximum number of allowed packets broadcasted in the network is 1000. Results show that the reputation with ONMC scheme consistently gives the highest number of packets under both topologies.

Figure 6 shows the relative throughput of the shortest path and reputation with ONMC schemes (i.e., throughput/throughput_reputationonly). Results show that the reputation with ONMC scheme can achieve up to 89% and 29% increase in throughput for static and dynamic topology case, respectively, as the number of malicious nodes increase.

Figure 7 shows the percentage of packets that arrived at the destination as a function of the maximum allowed number of packets. The number of malicious nodes is fixed at 5. Results show that the reputation with ONMC scheme still gives a significantly higher number of packet arrivals compared to the other two schemes—even when we reduce the amount of flooding (i.e., by reducing the allowable packets in the network from 1000 to 10 packets).

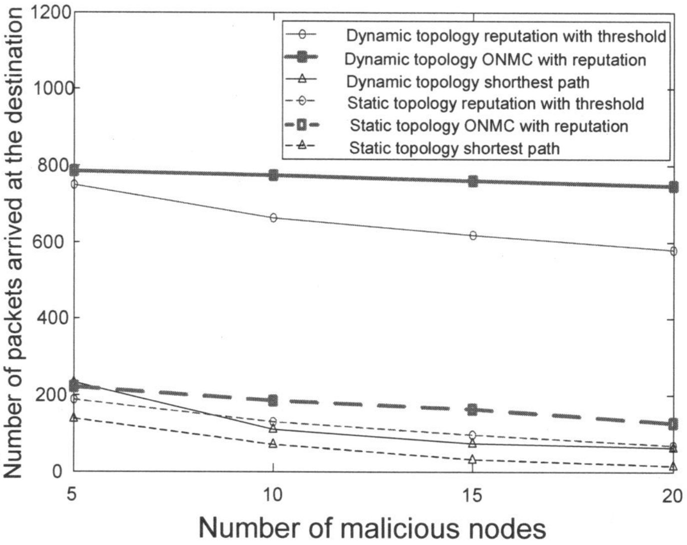

Fig. 5. The number of packets arrived at the destination as a function of the number of malicious nodes in the test network.

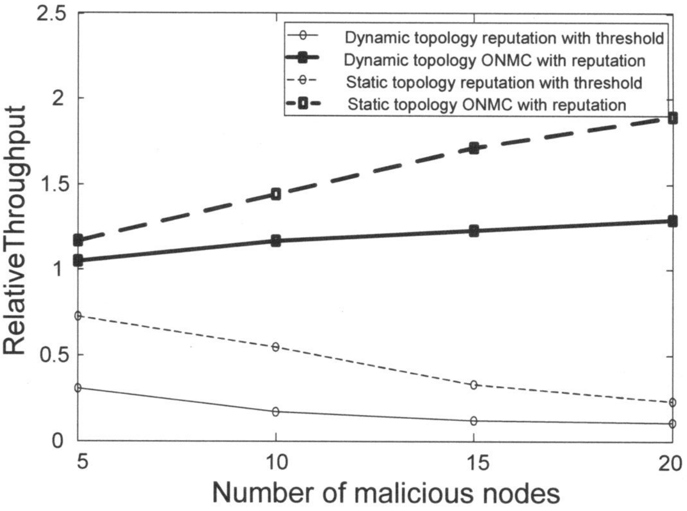

Fig. 6. Relative throughput as a function of the number of malicious nodes in the test network.

## V. Conclusion

The ability to join the network on the fly without centralized infrastructure exposes wireless ad hoc networks to major security vulnerabilities. Secure network functionalities are therefore necessary to defend attacks from malicious nodes. In this paper, we study a reputation scheme combined with ONMC method to learn good rules to identify and therefore select behaving nodes as well as avoid malicious nodes. Numerical studies show that up to 89% of throughput increase can be achieved over the fixed threshold reputation scheme—showing that learning through direct interaction with the network can lead to better reputation decision rules.

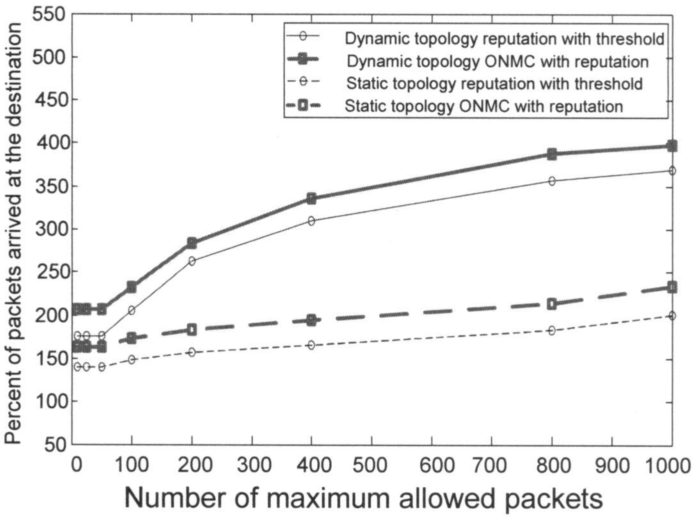

Fig. 7. Percentage of packets that arrived at the destination as a function of the maximum allowed number of packets.

## References

[1] P. Dewan, P. Dasgupta and A. Bhattacharya. On using reputations in ad hoc networks to counter malicious nodes. Parallel and Distributed Systems, 2004. ICPADS 2004. Proceedings. Tenth International Conference on 7-9 July 2004 Page(s):665 - 672.

[2] P. Dewan and P. Dasgupta. Trusting routers and relays in ad hoc networks. Parallel Processing Workshops, 2003. Proceedings of the 2003 International Conference on 6-9 Oct. 2003 Page(s):351 - 358

[3] S. Buchegger, C. Tissieres and J.-Y. Le Boudec. A Test-Bed for Misbehavior Detection in Mobile Ad-hoc Networks - How Much Can Watchdogs Really Do?. Mobile Computing Systems and Applications, 2004. WMCSA 2004. Sixth IEEE Workshop on 02-03 Dec. 2004 Page(s):102 - 111

[4] S. Buchegger and J.-Y. Le Boudec. The effect of rumor spreading in reputation systems for mobile ad hoc networks.In Proc. WiOpt'03 (Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks), 2003.

[5] P.-W. Yau and C.J. Mitchell. Reputation methods for routing security for mobile ad hoc networks. Mobile Future and Symposium on Trends in Communications, 2003. SympoTIC '03. Joint First Workshop on 26-28 Oct. 2003 Page(s):130 - 137

[6] S. Marti and H. Garcia-Molina. Identity crisis: anonymity vs reputation in P2P systems. Peer-to-PeerComputing, 2003. (P2P 2003). Proceedings. Third International Conference on 1-3 Sept. 2003 Page(s):134 - 141

[7] Q. He, W. Dapeng and P. Khosla. SORI: a secure and objective reputation-based incentive scheme for ad-hoc networks. Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE Volume 2, 21-25 March 2004 Page(s):825 - 830 Vol.2

[8] R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction*, Massachusetts: The MIT Press, 1998.