www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 8 August, 2013 Page No. 2371-2376

A Review on “Adaptive Contriving Routing for Multihop Wireless Ad Hoc Networks”

Mubina Begum, C. RamaKrishna, Sasmita Behera

M.tech (CSE) Student in CSE PRRCET Medak (Dist), Andhra Pradesh India 502 300
mubina_mubin@yahoo.com
Asst Professor in CSE PRRCET Medak (Dist), Andhra Pradesh India 502 300
cramakrishna537@gmail.com
Asst Professor in CSE PRRCET Medak (Dist), Andhra Pradesh India 502 300
behera.sasmita2008@gmail.com

ABSTRACT

A distributed adaptive opportunistic routing scheme for multihop wireless ad hoc networks is proposed. The proposed scheme utilizes a reinforcement learning framework to opportunistically route the packets even in the absence of reliable knowledge about channel statistics and network model. This scheme is shown to be optimal with respect to an expected average per-packet reward criterion. The proposed routing scheme jointly addresses the issues of learning and routing in an opportunistic context, where the network structure is characterized by the transmission success probabilities. In particular, this learning framework leads to a stochastic routing scheme that optimally “explores” and “exploits” the opportunities in the network.

Keywords- Opportunistic routing, reward maximization, wireless adhoc networks.

I. INTRODUCTION

Opportunistic routing for multihop wireless ad hoc networks has seen recent research interest to overcome deficiencies of conventional routing [1]–[6] as applied in wireless setting. Motivated by classical routing solutions in the Internet, conventional routing in ad hoc networks attempts to find a fixed path along which the packets are forwarded [7]. Such fixed-path schemes fail to take advantage of broadcast nature and opportunities provided by the wireless medium and result in unnecessary packet retransmissions. The opportunistic routing decisions, in contrast, are made in an online manner by choosing the next relay based on the actual transmission outcomes as well as a rank ordering of neighboring nodes. Opportunistic routing mitigates the impact of poor wireless links by exploiting the broadcast nature of wireless transmissions and the path diversity. The authors in [1] and [6] provided a Markov decision theoretic formulation for opportunistic routing. In particular, it is shown that the optimal routing decision at any epoch is to select the next relay node based

on a distance-vector summarizing the expected-cost-to-forward from the neighbors to the destination. This “distance” is shown to be computable in a distributed manner and with low complexity using the probabilistic description of wireless links. The study in [1] and [6] provided a unifying framework for almost all versions of opportunistic routing such as SDF [2], Geographic Random Forwarding (GeRaF) [3], and ExOR [4], where the variations in [2]–[4] are due to the authors’ choices of cost measures to optimize. For instance, an optimal route in the context of ExOR [4] is computed so as to minimize the expected number of transmissions (ETX), while GeRaF [3] uses the smallest geographical distance from the destination as a criterion for selecting the next-hop.

The opportunistic algorithms proposed in [1]–[6] depend on a precise probabilistic model of wireless connections and local topology of the network. In a practical setting, however, these probabilistic models have to be “learned” and “maintained.” In other words, a comprehensive study and evaluation of any opportunistic routing scheme requires an integrated approach to the issue of probability estimation. Authors in [8] provide a sensitivity analysis for the opportunistic routing algorithm given in [6]. However, by and large, the question of learning/estimating channel statistics in conjunction with opportunistic routing remains unexplored.

In this paper, we first investigate the problem of opportunistically routing packets in a wireless multihop network when zero or erroneous knowledge of transmission success probabilities and network topology is available. Using a reinforcement learning framework, we propose a distributed adaptive opportunistic routing algorithm (d-AdaptOR) that minimizes the expected average per-packet cost for routing a packet from a source node to a destination. This is achieved by both sufficiently exploring the network using data packets and exploiting the best routing opportunities.

Our proposed reinforcement learning framework allows for a low-complexity, low-overhead, distributed asynchronous implementation. The significant characteristics of d-AdaptOR are that it is oblivious to the initial knowledge about the network, it is distributed, and it is asynchronous.

The main contribution of this paper is to provide an opportunistic routing algorithm that: 1) assumes no knowledge about the channel statistics and network, but 2) uses a reinforcement learning framework in order to enable the nodes to adapt their routing strategies, and 3) optimally exploits the statistical opportunities and receiver diversity. In doing so, we build on the Markov decision formulation in [6] and an important theorem in Q-learning proved in [9]. There are many learning-based routing solutions (both heuristic or analytically driven) for conventional routing in wireless or wired networks [10]–[15]. None of these solutions exploits the receiver diversity gain in the context of opportunistic routing. However, for the sake of completeness, we provide a brief overview of the existing approaches. The authors in [10]–[14] focus on heuristic routing algorithms that adaptively identify the least congested path in a wired network. If the network congestion, hence delay, were to be replaced by time-invariant quantities,1 the heuristics in [10]–[14] would become a special case of d-AdaptOR in a network with deterministic channels and with no receiver diversity. In this light, Theorem 1 in Section IV provides analytic guarantees for the heuristics obtained in [10]–[14]. In [15], analytic results for ant routing are obtained in wired networks without opportunism. Ant routing uses ant-like probes to find paths of optimal costs such as expected hop count, expected delay, and packet loss probability.2 This dependence on ant-like probing represents a stark difference with our approach where d-AdaptOR relies solely on data packet for exploration

The rest of the paper is organized as follows:

In Section II, we discuss the system model and formulate the problem.

Section III formally introduces our proposed adaptive routing algorithm, d-AdaptOR.We then state and prove the optimality theorem for d-AdaptOR

In section IV, we discuss optimality of d-AdaptOR.

In Section V, we present the implementation details and practical issues of d-AdaptOR.

In section VI, we discuss about simulation.

Finally, we conclude the paper and discuss future work in Section VII.

II. SYSTEM MODEL

We consider the problem of routing packets from a source node 0 to a destination node in a wireless ad hoc network of d+1 nodes denoted by the set $\Theta = \{0, 1, 2, \ldots, d\}$. The time is slotted and indexed by $n \geq 0$ (this assumption is not technically critical and is only assumed for ease of exposition). A packet indexed by $m \geq 1$ is generated at the source node 0 at time τ_s^m according to an arbitrary distribution with rate $\lambda > 0$. We assume a fixed transmission cost is incurred upon a transmission from node . Transmission cost can be considered to model the amount of energy used for transmission, the expected time to transmit a given packet, or the hop count when the cost is set to unity. We consider an opportunistic routing setting with no duplicate copies of the packets. In other words, at a given time only one node is responsible for routing any given packet. Given a successful packet transmission from node to the set of neighbor nodes , the next (possibly randomized) routing decision includes: 1) retransmission by node ; 2) relaying the packet by a node $j \in S$; or 3) dropping the packet altogether. If node is selected as a relay, then it transmits the packet at the next slot, while other nodes $k \neq j, k \in S$, expunge that packet.

We define the termination event for packet m to be the event that packet m is either received at the destination or is dropped by a relay before reaching the destination. We denote this termination action by . We define termination time τ_T^m to be the stopping time when packet m is terminated. We discriminate among the termination events as follows. We assume that upon the termination of a packet at the destination (successful delivery of a packet to the destination), a fixed and given positive delivery reward R is obtained, while no reward is obtained if the packet is terminated before it reaches the destination. Let r_m denote this random reward obtained at the termination time τ_T^m , i.e., either zero if the packet is dropped prior to reaching the destination node or if the packet is received at the destination.

Let $i_{n,m}$ denote the index of the node which at time transmits packet , and accordingly $c_{i_{n,m}}$ let denote the cost of transmission (equal to zero if at time n packet m is not transmitted).

The routing scheme can be viewed as selecting a (random) sequence of nodes $\{i_{n,m}\}$ for relaying packets m=1,2,…. As such, the expected average per-packet reward associated with routing packets along a sequence of $\{i_{n,m}\}$ up to time N is

$$J_N = \mathbf{E}\left[\frac{1}{M_N}\sum_{m=1}^{M_N}\left\{r_m - \sum_{n=\tau_s^m}^{\tau_T^m - 1} c_{i_{n,m}}\right\}\right] \qquad (1)$$

where M_N denotes the number of packets terminated up to time N and the expectation is taken over the events of transmission decisions, successful packet receptions, and packet generation times.

III. DISTRIBUTED ALGORITHM

Before we proceed with the description of d-AdaptOR, we provide the following notations. Let $\mathcal{N}(i)$ denote the set of neighbors of node including node itself. Let $\mathfrak{S}^i$ denote the set of potential reception outcomes due to a transmission from node $i \in \Theta$, i.e.,$\mathfrak{S}^i = \{S : S \subseteq \mathcal{N}(i), i \in S\}$. We refer to $\mathfrak{S}^i$ as the state space for node 's transmission. Let $A(S) = S \cup \{T\}$ denote the space of all allowable actions available to node upon successful reception at nodes in . Finally, for each node , we define a reward function on states $S \in \mathfrak{S}^i$ and potential decisions $a \in A(S)$ as

$$g(S, a) = \begin{cases} -c_a, & \text{if } a \in S \\ R, & \text{if } a = T \text{ and } d \in S \\ 0, & \text{if } a = T \text{ but } d \notin S. \end{cases}$$

A. Overview of d-AdaptOR

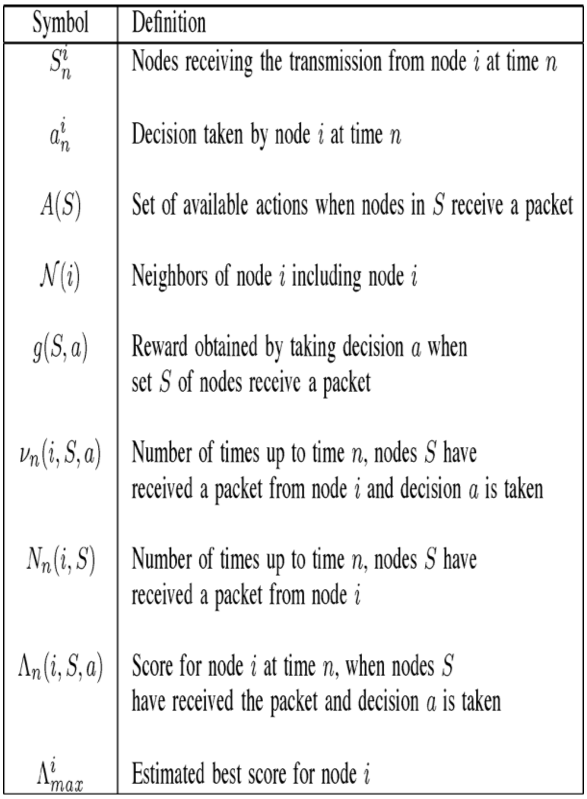
As discussed before, the routing decision at any given time is made based on the reception outcome and involves retransmission, choosing the next relay, or termination. Our proposed scheme makes such decisions in a distributed manner via the following three-way handshake between node and its neighbors $\mathcal{N}(i)$.

1) At time n, node transmits a packet.

2) The set of nodes S_n^i who have successfully received the packet from node , transmit acknowledgment (ACK) packets to node . In addition to the node's identity, the acknowledgment packet of node $k \in S_n^i$ includes a control *message* known as *estimated best score* (EBS) and denoted by $\Lambda^{k}_{\max}$.

3) Node announces node $j \in S_n^i$ as the next transmitter or announces the termination decision in a forwarding (FO) packet.

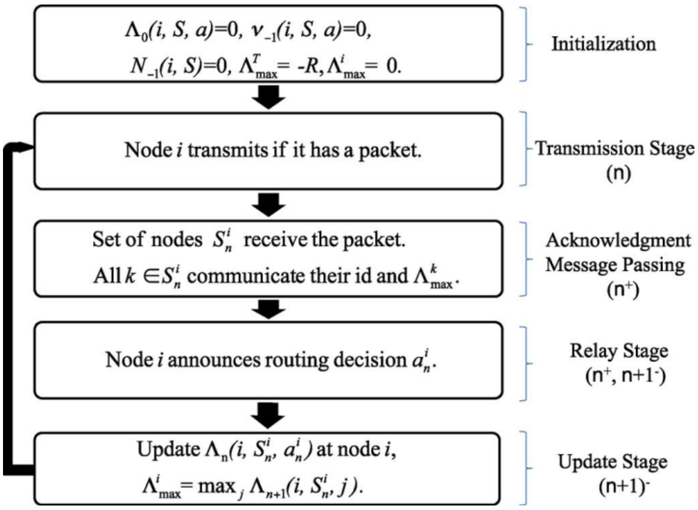
TABLE I
NOTATIONS USED IN THE DESCRIPTION OF THE ALGORITHM



Symbol	Definition
S_n^i	Nodes receiving the transmission from node i at time n
a_n^i	Decision taken by node i at time n
$A(S)$	Set of available actions when nodes in S receive a packet
$\mathcal{N}(i)$	Neighbors of node i including node i
$g(S,a)$	Reward obtained by taking decision a when set S of nodes receive a packet
$\nu_n(i,S,a)$	Number of times up to time n, nodes S have received a packet from node i and decision a is taken
$N_n(i,S)$	Number of times up to time n, nodes S have received a packet from node i
$\Lambda_n(i,S,a)$	Score for node i at time n, when nodes S have received the packet and decision a is taken
Λ^i_{max}	Estimated best score for node i

B. Detailed Description of d-AdaptOR

The operation of d-AdaptOR can be described in terms of initialization and four stages of transmission, reception and acknowledgment, relay, and adaptive computation as shown in Fig. 1. For simplicity of presentation, we assume a sequential Timing for each of the stages. We use n^+ to denote some (small) time after the start of nth slot and $(n+1)^-$ to denote some (small) time before the end of nth slot such that n $<n^+<(n+1)^-$ $<n+1$.



$\Lambda_0(i, S, a)$=0, $\nu_{-1}(i, S, a)$=0,
$N_{-1}(i, S)$=0, $\Lambda^T_{\max}$= -R, $\Lambda^i_{\max}$= 0.
Initialization
Node i transmits if it has a packet.
Transmission Stage (n)
Set of nodes S_n^i receive the packet.
All $k \in S_n^i$ communicate their id and $\Lambda^k_{\max}$.
Acknowledgment Message Passing (n^+)
Node i announces routing decision a_n^i.
Relay Stage (n^+, $n+1^-$)
Update $\Lambda_n(i, S_n^i, a_n^i)$ at node i,
$\Lambda^i_{\max}$= max$_j$ $\Lambda_{n+1}(i, S_n^i, j)$.
Update Stage $(n+1)^-$

Fig. 1. Flow of the algorithm. The algorithm follows a four-stage procedure:
transmission, acknowledgment, relay, and update.

0) **Initialization**:

For all $i \in \Theta, S \in \mathfrak{S}^i, a \in A(S)$, initialize $\Lambda_0(i,S,a) = \nu_{-1}(i,S,a) = N_{-1}(i,S) = \Lambda^i_{\max} = 0$, while $\Lambda^T_{\max} = -R$.

1) **Transmission Stage:**

Transmission stage occurs at time in which node transmits if it has a packet.

2) **Reception and acknowledgment Stage:**

Let S_n^i denote the (random) set of nodes that have received the packet transmitted by node . In the reception and acknowledgment stage, successful reception of the packet transmitted by node is acknowledged to it by all the nodes in S_n^i. We assume that the delay for the acknowledgment stage is small enough (not more than the duration of the time slot) such that node infers S_n^i by time n^+.

For all nodes k $\in S_n^i$, the ACK packet of node k to node i includes the EBS message $\Lambda^k_{\max}$.

Upon reception and acknowledgment, the counting random variable N_n is incremented as follows:

$$N_n(i,S) = \begin{cases} N_{n-1}(i,S)+1, & \text{if } S = S_n^i \\ N_{n-1}(i,S), & \text{if } S \neq S_n^i. \end{cases}$$

3) **Relay Stage:**

Node i selects a routing action $a_n^i \in A(S_n^i)$ according to the following (randomized) rule parameterized $\epsilon_n(i,S) = \frac{1}{N_n(i,S)+1}$.

- With probability $(1 - \epsilon_n(i, S_n^i))$

$$a_n^i \in \arg\max_{j \in A(S_n^i)} \Lambda_n\left(i, S_n^i, j\right)$$

by
is selected.

- With probability $\epsilon_n(i, S_n^i)$

$$a_n^i \in A\left(S_n^i\right)$$

is selected uniformly with probability $\frac{\epsilon_n(i,S_n^i)}{|A(S_n^i)|}$

4) **Adaptive Computation Stage:**
At time $(n+1)^-$, after being done with transmission and relaying, node updates score vector Λ_n(I,.,.) as follows.

- For $S = S_n^i$, $a = a_n^i$

$$\Lambda_{n+1}(i,S,a) = \Lambda_n(i,S,a) + \alpha_{\nu_n(i,S,a)} \times \left(-\Lambda_n(i,S,a) + g(S,a) + \Lambda_{\max}^a\right). \quad (2)$$

- Otherwise

$$\Lambda_{n+1}(i,S,a) = \Lambda_n(i,S,a). \quad (3)$$

Furthermore, node i updates its EBS message $\Lambda_{\max}^i$ for future acknowledgments as

$$\Lambda_{\max}^i = \max_{j \in A(S_n^i)} \Lambda_{n+1}(i, S_n^i, j).$$

C. Computational Issues

The computational complexity and control overhead ofd-AdaptOR is low.

1) Complexity: To execute stochastic recursion (2), thenumber of computations required per packet is order of $O(\max_{i\in\Theta}|\mathcal{N}(i)|)$at each time slot. The space complexity of d-AdaptOR is exponential in the number of neighbors, i.e.,$O(\max_{i\in\Theta} 2^{|\mathcal{N}(i)|})$ for each node. The reduction in storage requirement using approximation techniques in [16] is left as future work.

2) Control Overhead: The number of acknowledgments per packet is order of $O(\max_{i\in\Theta}|\mathcal{N}(i)|)$, independent of network ----size.

3) Exploration Overhead: The adaptation to the optimal performance in the network is guaranteed via a controlled randomized routing strategy that can be viewed as cost of exploration. The cost of exploration is proportional to the total number of packets whose routes deviates from the optimal path. In proof of Theorem 1, we show that this cost increases sublinearly with the number of delivered packets, hence the per-packet exploration cost diminishes as the number of delivered packets grows.Additionally, communication of $\Lambda_{\max}$ adds a very modest overhead to the genie-aided or greedy-based schemes such as ExOR or SR.

IV. ANALYTIC OPTIMALITY OF D-ADAPTOR

We will now state the main result establishing the optimality of the proposed d-AdaptOR algorithm under the assumptions of a time-invariant model of packet reception and reliable control packets. More precisely, we have the following assumptions.
Assumption 1: The probability of successful reception of a packet transmitted by node i at set $S \subseteq \mathcal{N}(i)$ of nodes is $P(S|i)$, independent of time and all other routing decisions. The probabilities $P(\cdot|\cdot)$ in Assumption 1 characterize a packet reception model that we refer to as *local broadcast model*. Note that for all $S \neq S'$, successful reception at S and S' are mutually exclusive and$\sum_{S\subseteq\Theta} P(S|i) = 1$. Furthermore, logically node i is always a recipient of its own transmission, i.e., $P(S|i) = 0 \text{ iff } i \notin S$.

Assumption 2: The successful reception at set S due to transmission from node i is acknowledged perfectly to node i .

V. PROTOCOL DESIGN AND IMPLEMENTATION ISSUES

In this section, we describe an 802.11 compatible implementation for d-AdaptOR.

A. 802.11 Compatible Implementation

The implementation of d-AdaptOR, analogous to any opportunistic routing scheme, involves the selection of a relay node among the candidate set of nodes that have received and acknowledged a packet successfully. One of the major challenges in the implementation of an opportunistic routing algorithm i general, and the d-AdaptOR algorithm in particular, is the design of an 802.11 compatible acknowledgment mechanism at the MAC layer. We propose a practical and simple way to implement acknowledgment architecture.
The transmission at any node is done according to an 802.11 CSMA/CA mechanism. Specially, before any transmission, transmitter i performs channel sensing and starts transmission after the backoff counter is decremented to zero. For each neighbor node $j \in \mathcal{N}(i)$, the transmitter node i then reserves a virtual time slot of duration $T_{ACK}+T_{SIFS}$, where T_{ACK} is the duration of the acknowledgment packet and T_{SIFS} is the duration of Short InterFrame Space (SIFS) [20]. Transmitter i then piggybacks a priority ordering of nodes $\mathcal{N}(i)$ with each data packet transmitted. The priority ordering determines the virtual time slot in which the candidate nodes transmit their acknowledgment. Nodes in the set S^i that have successfully received the packet then transmit acknowledgment packets sequentially in the order determined by the transmitter node. After a waiting time of $T_{\text{wait}} = |\mathcal{N}(i)|(T_{\text{ACK}}+T_{\text{SIFS}})$during which each node in the set S^i has had a chance to send an ACK, Node i transmits a FOrwarding control packet (FO). The FO packets contain the identity of the next forwarder, which may be node i again or any node$j \in S^i$. If T_{wait} expires and no O packet is received (FO packet reception is unsuccessful), then the corresponding candidate nodes drop the received data packet. If the transmitter i does not receive any acknowledgment, node i retransmits the packet. The backoff

window is doubled after every retransmission. Furthermore, the packet is dropped if the retry limit (set to 7) is reached.

In addition to the acknowledgment scheme, d-AdaptOR requires modifications to the 802.11 MAC frame format. Fig. 2 shows the modified MAC frame formats required by d-AdaptOR. The reserved bits in the type/subtype fields of the frame control field of the 802.11 MAC specification are used to indicate whether the rest of the frame is a d-AdaptOR data frame, a d-AdaptOR ACK, or a, FO.The data frame contains

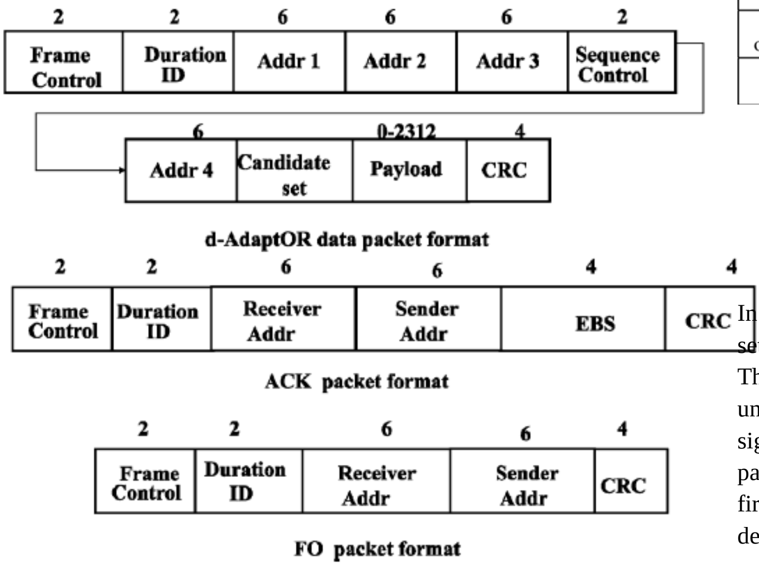


Fig. 2. Frame structure of the data packets, acknowledgment packets, and FO packets.

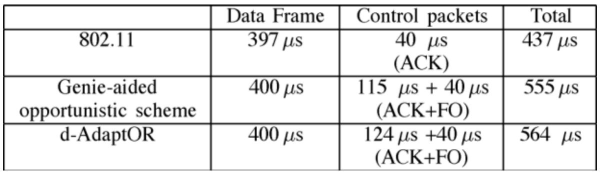
the candidate set in priority order, the payload, and the 802.11 Frame Check Sequence. The acknowledgment frame includes the data frame sender's address and the feedback EBS $\Lambda_{\max}$. The FO packet is exactly the same as a standard 802.11 short control frame that uses different subtype value.

B. d-AdaptOR in a Realistic Setting

1) Loss of ACK and FO Packets: Interference or low signal-to-noise ratio (SNR) can cause loss of ACK and FO packets. Loss of an ACK packet results in an incorrect estimation of nodes that have received the packet, and thus affects the performance of the algorithm. Loss of FO packet negatively impacts the throughput performance of the network. In particular, loss of an FO packet can result in the drop of data packets at all the potential relays, reducing the throughput performance. Hence, in our design, FO packets are transmitted at lower rates to ensure a reliable transmission.

2) Increased Overhead: As it is the case with any opportunistic scheme, d-AdaptOR adds a modest additional overhead to the standard 802.11 due to the added acknowledgment/handshake structure. This overhead increases linearly with the number of neighbors. Assuming a 802.11b physical layer operating at 11 Mb/s with an SIFS time of 10 s, preamble duration of 20 s, Physical Layer Convergence Protocol (PLCP) header duration of 4 s, and 512-B frame payloads, Table II compares the overhead in the data packet due to piggybacking and the control overhead due to ACK and FO packets for unicast 802.11, genie-aided opportunistic scheme, and d-AdaptOR. d-AdaptOR requires communication overhead of 4 extra bytes (for EBS) per ACK packet compared to the genie-aided opportunistic scheme, while unicast 802.11 does not require such overhead.

TABLE II
OVERHEAD COMPARISONS



	Data Frame	Control packets	Total
802.11	397 μs	40 μs (ACK)	437 μs
Genie-aided opportunistic scheme	400 μs	115 μs + 40 μs (ACK+FO)	555 μs
d-AdaptOR	400 μs	124 μs +40 μs (ACK+FO)	564 μs

VI. SIMULATIONS

In this section, we provide simulation studies in realistic wireless settings where the theoretical assumptions of our study do not hold. These simulations not only demonstrate a robust performance gain under d-AdaptOR in a realistic network, but also provide significant insight in the appropriate choice of the design parameters such as damping sequence , delivery reward , etc.We first investigate the performance of d-AdaptOR with respect to the design parameters and network parameters in a grid topology of 16 nodes.We then use a realistic topology of 36 nodes with random placement to demonstrate robustness of d-Adaptor to the violation of the analytic Assumptions 1 and 2.

VII. CONCLUSION AND FUTURE WORK

In this paper, we proposed d-AdaptOR, a distributed, adaptive, and opportunistic routing algorithm whose performance is shown to be optimal with zero knowledge regarding network topology and channel statistics. More precisely, under idealized assumptions, d-AdaptOR is shown to achieve the performance of an optimal routing with perfect and centralized knowledge about network topology, where the performance is measured in terms of the expected per-packet reward. Furthermore, we show that d-AdaptOR allows for a practical distributed and asynchronous 802.11 compatible implementation, whose performance was investigated via a detailed set of QualNet simulations under practical and realistic networks. Simulations show that d-AdaptOR consistently outperforms existing adaptive routing algorithms in practical settings. The long-term average reward criterion investigated in this paper inherently ignores the short-term performance. To capture the performance of various adaptive schemes, however, it is desirable to study the performance of the algorithms over a finite horizon. One popular way to study this is via measuring the incurred "regret" over a finite horizon. Regret is a function of horizon N that quantifies the loss of the performance under a given adaptive algorithm relative to the performance of the topology-aware optimal one. More specifically, our results so far implies that the optimal rate of growth of regret is strictly sublinear in , but fails to provide a conclusive understanding of the short-term behavior of d-AdaptOR. An important area of future work comprises developing adaptive algorithms that ensure optimal growth rate of regret. The design of routing protocols requires a consideration of congestion control along with the throughput performance [26], [27]. Our work, however, does not consider this closely related issue. Incorporating congestion control in opportunistic routing algorithms to minimize expected delay without the topology and the channel statistics knowledge is an area of future research.

ACKNOWLEDGMENT

The authors would like to thank A. Plymoth and P. Johansson for the valuable discussions. They are grateful to the anonymous reviewers who provided thoughtful comments and constructive critique of the paper.

REFERENCES

[1] C. Lott and D. Teneketzis, “Stochastic routing in ad hoc wireless networks,” in *Proc. 39th IEEE Conf. Decision Control*, 2000, vol. 3, pp. 2302–2307, vol. 3.

[2] P. Larsson, “Selection diversity forwarding in a multihop packet radio network with fading channel and capture,” *Mobile Comput. Commun. Rev.*, vol. 2, no. 4, pp. 47–54, Oct. 2001.

[3] M. Zorzi and R. R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks:Multihop performance,” *IEEE Trans. Mobile Comput.*, vol. 2, no. 4, pp. 337–348, Oct.–Dec. 2003.

[4] S. Biswas and R. Morris, “ExOR: Opportunistic multi-hop routing for wireless networks,” *Comput. Commun. Rev.*, vol. 35, pp. 33–44, Oct. 2005.

[5] S. Jain and S. R. Das, “Exploiting path diversity in the link layer in wireless ad hoc networks,” in *Proc. 6th IEEE WoWMoM*, Jun. 2005, pp. 22–30.

[6] C. Lott and D. Teneketzis, “Stochastic routing in ad hoc networks,” *IEEE Trans. Autom. Control*, vol. 51, no. 1, pp. 52–72, Jan. 2006.

[7] E. M. Royer and C. K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *IEEE Pers. Commun.*, vol. 6, no. 2, pp. 46–55, Apr. 1999.

[8] T. Javidi and D. Teneketzis, “Sensitivity analysis for optimal routing in wireless ad hoc networks in presence of error in channel quality estimation,” *IEEE Trans. Autom. Control*, vol. 49, no. 8, pp. 1303–1316, Aug. 2004.

[9] J. N. Tsitsiklis, “Asynchronous stochastic approximation and Q-learning,” in *Proc. 32nd IEEE Conf. Decision Control*, Dec. 1993, vol. 1, pp. 395–400.

[10] J. Boyan and M. Littman, “Packet routing in dynamically changing networks: A reinforcement learning approach,” in *Proc. NIPS*, 1994, pp. 671–678.

[11] J. W. Bates, “Packet routing and reinforcement learning: Estimating shortest paths in dynamic graphs,” 1995, unpublished.

[12] S. Choi and D. Yeung, “Predictive Q-routing: A memory-based reinforcement learning approach to adaptive traffic control,” in *Proc. NIPS*, 1996, pp. 945–951.

[13] S. Kumar and R. Miikkulainen, “Dual reinforcement Q-routing: An on-line adaptive routing algorithm,” in *Proc. Smart Eng. Syst., Neural Netw., Fuzzy Logic, Data Mining, Evol. Program.*, 2000, pp. 231–238.

[14] S. S. Dhillon and P. Van Mieghem, “Performance analysis of the AntNet algorithm,” *Comput. Netw.*, vol. 51, no. 8, pp. 2104–2125, 2007.

[15] P. Purkayastha and J. S. Baras, “Convergence of Ant routing algorithm via stochastic approximation and optimization,” in *Proc. IEEE Conf. Decision Control*, 2007, pp. 340–354.

[16] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific, 1996.

[17] S. Chachulski,M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” in *Proc. ACM SIGCOMM*, 2007, pp. 169–180.

[18] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York: Wiley, 1994.

[19] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, MA: Athena Scientific, 1997.

[20] W. Stallings,*Wireless Communications and Networks*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 2004.

[21] J.Bicket, D. Aguayo, S. Biswas, andR.Morris, “Architecture and evaluation of an unplanned 802.11b mesh network,” in *Proc. ACM MobiCom*, Cologne, Germany, 2005, pp. 31–42.

[22] M. Kurth, A. Zubow, and J. P. Redlich, “Cooperative opportunistic routing using transmit diversity in wireless mesh networks,” in *Proc. IEEE INFOCOM*, Apr. 2008, pp. 1310–1318.

[23] J. Doble, *Introduction to Radio Propagation for Fixed and Mobile Communications*. Boston, MA: Artech House, 1996.

[24] S. Russel and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 2003.

[25] R. Parr and S. Russell, “Reinforcement learning with hierarchies of machines,” in *Proc. NIPS*, 1998, pp. 1043–1049.

[26] P. Gupta and T. Javidi, “Towards throughput and delay optimal routing for wireless ad-hoc networks,” in *Proc. Asilomar Conf.*, Nov. 2007, pp. 249–254.

[27] M. J. Neely, “Optimal backpressure routing for wireless networks with multi-receiver diversity,” in *Proc. CISS*, Mar. 2006, pp. 18–25.

[28] L. Breiman, *Probability*. Philadelphia, PA: SIAM, 1992.

[29] S. Resnick, *A Probability Path*. Boston, MA: Birkhuser, 1998.