Autonomic Reconfiguration Management for Heterogeneous Wireless Networks using Reinforcement Learning

Minsoo Lee, Dan Marconett, Xiaohui Ye and S. J. Ben Yoo

Department of Electrical and Computer Engineering, University of California Davis, Davis, CA 95616, USA

E-mail: {msolee, dmarconett, xye, sbyoo}@ucdavis.edu

Abstract

This paper presents a framework of network systems to address a combination of network control interests by means of an autonomic self-configuration scheme using a cross-layer approach. We propose a network architecture that enables intelligent services to meet QoS requirements. By adding autonomous intelligence, based on reinforcement learning, to the network management agents, the system is better able to reconfigure the network management strategy around areas of interest and adapt to changes. We present OPNET simulation results showing that our autonomous self-configuration approach successfully improves the scalability of the original AODV protocol in a heterogeneous network environment.

Introduction

Due to the fact that current network control systems have limited ability to adapt to dynamic network conditions, providing intelligent network management for globally optimal performance is a challenging problem. There is the limitation of network elements in the user context, network state, interworking scope and application demands. This limitation results in obstacles to intelligent adaptation. In the case of routing in wireless ad hoc networks, battery-powered devices create challenging problems in terms of prolonging the autonomous lifetime of the network. In designing intelligent routing protocols, the various features of sensor networks lead to a set of optimization problems in routing path length, load balancing, consistent link management, and aggregation [1]. Most existing routing techniques are designed to optimize one of these goals. In real scenarios however, these factors are usually in conflict and influence the routing performance in a complex way, leading to the need for a more sophisticated routing scheme that makes optimal trade-offs. Clearly, solving the optimization goals separately does not lead to an optimal solution for the problem here; instead one needs to consider all these optimization goals as a whole in designing routing protocols.

A way to solve these problems might lie in the vision of cognitive networks. The vision of 4G networks can be realized through the advent of cognitive networks. Cognitive wireless networks are capable of reconfiguring their infrastructure, based also on experience, in order to adapt to the continuously changing environment [2]. Cognitive networks [3] are seen as a major facilitator of the 4G network vision. Cognitive networks are capable of continuously adapting to changing environmental conditions and/or user needs. Typically, machine learning like Q-learning [4] helps implement the adaptation methods of self-configuration and self-management in the autonomic computing paradigm.

The self-configuration of network systems will have cross-layer ramifications for the protocol stack, from the physical (PHY), Medium Access Control (MAC), network, and transport layers to the middleware, presentation and application layers. Therefore, cross-layer design [5-8] approaches are critical for the efficient utilization of limited resources with QoS guarantees in future wireless networks. The recent cross-layer architectures showed that a cross-layer approach is advantageous for wireless networks. Better network system performance can be obtained from context exchanges across protocol layers. In this paper we present the concept of autonomous self-configuration by Q-learning in a cross-layer approach, which can overcome the potential scope limitations of network management in heterogeneous wireless networks by allowing networks to observe, act, and learn in order to optimize their performance. Our strategy is to tailor the routing strategy in wireless networks with Q-learning to ensure that the SLA and packet delivery ratio can be executed at desired levels of quality while minimizing additional management overhead. The questions that need to be answered are threefold:

1) What information needs to be gathered and exchanged between layers?
2) What information should be reconfigured by Q-learning? and
3) When should the results of Q-learning be used to determine reconfigurable parameters?

We aim to improve network performance by providing a reasonably accurate picture of the underlying network systems maintained by the proposed Q-learning based autonomic self-configuration.

Specifically, we present a novel model of reconfigurable ad hoc routing management with Q-learning, which enables the nodes to efficiently learn an optimal routing strategy enhancing the packet delivery ratio and QoS support. We also present OPNET simulation results showing that our autonomous self-configuration in a cross-layer approach successfully improves the scalability of the original AODV protocol in a heterogeneous network environment.

The remainder of the paper will be organized as follows: Section 2 presents a belief survey of related work. Section 3 gives an overview of our network architecture with reinforcement learning techniques for autonomic self-configuration. Section 4 describes in detail our Q-learning based intelligent self-configuration scheme. Section 5 explains our OPNET simulation scenario. Section 6 presents our OPNET simulation results. Finally, we discuss future work and conclude in Section 7.

2. Related Work

2.1. Cross-Layer Approaches for Intelligent Network Management in Wireless Networks

The realm of network management covers vast areas. Issues such as IP configuration, security and network monitoring fall
under the management umbrella. While these components are not unique to Mobile Ad hoc Networks (MANETs), they do become more difficult to achieve when nodal mobility, dynamic network membership, and unstable links are introduced to the network [9]. Depending on the speed of the Mobile Nodes (MNs), the mobility can be classified into three categories of increasing speed: static, low mobility, and high mobility. Therefore, the management of a network should be able to take into account any of these three cases and their characteristics. In the case of low mobility, the steady-state performance should be optimized, and incidental updates (e.g., for route discovery) can consume more resources, whereas in the high mobility case, resource consumption and delay due to route maintenance and updating are important factors [10].

However, arriving at a solution to this problem set is certainly non-trivial. Centralized network management architectures have been shown to fail in MANETs. Recently a distributed decision making scheme [11] has been introduced to address these concerns. In this proposed scheme, nodes may only be aware only of their own neighbors and have no sense of the size and extent of the network. Finding a mechanism that can deal with the particular challenges associated with distributed decision making in ad hoc networks is certainly non-trivial. To cope with the demands of cross-layer design [5-8], management solutions are needed for efficient utilization of the limited resources, with QoS guarantees, in future wireless networks. Better network system performance can be obtained from context exchanges across protocol layers, which may not be available in the traditional layering architecture. This paper presents an overview of cross-layer design approaches for routing management in MANET, summarizes our research results, and suggests further research directions. We have deployed the cross-layer mechanism between the application layer and the network layer in a MN for exchanging end-to-end delay with the application layer.

2.2 Challenges in MANETs

Several papers have classified routing protocols in terms of their behavioral characteristics and applicability. We largely adhere to the standard convention of classification, namely Flat, Hybrid, and Geographically oriented protocols. Routing protocols which are not organized in any hierarchical fashion are commonly referred to as flat routing protocols. Flat routing schemes have three main classifications: proactive (table-driven, e.g. Optimized Link State Routing Protocol (OLSR) [13]), reactive (demand-driven, e.g. Dynamic Source Routing protocol (DSR) [14], Ad-Hoc On-Demand Distance Vector protocol (AODV) [15]), and hybrid (e.g. Zone Routing Protocol (ZRP) [16]).

Dynamic Source Routing protocol (DSR) is a reactive protocol which uses source routing as a central mechanism. When a route request (RREQ) is made by a particular node, it uses the destination route stored in its local route cache to send the data packet. Nodes along the path aggressively cache the path from the source node’s cache (which is embedded in the packet itself). However, if the node does not have the required route information cached, the route discovery process is initiated by flooding the network with route request packets. The request packets propagate throughout the network until they reach the destination node, or a node which has a cached path to the destination. This end node then sends a route reply with the newly discovered route source information back to the source node which then caches the path for future source routing. Further, destination nodes respond to all route request packets, thereby increasing the amount of aggressive caching taking place throughout the network.

The Ad-Hoc On-Demand Distance Vector (AODV) routing protocol is another routing protocol for multihop wireless networks, similar in nature to DSR. AODV shares DSR’s on-demand characteristics in that it also discovers routes on an as-needed basis via a similar route discovery process. However, AODV adopts a very different mechanism to maintain routing information. There is only one table entry per destination in a particular node’s routing table. AODV uses sequence numbers to determine the “freshness” of routes in the various routing tables. Without source routing, AODV relies on routing table entries to propagate the route reply (RREP) back to the source and, subsequently, to route data packets to the destination.

An important feature of AODV is the maintenance of timer-based states in each node with parameters (e.g. active route timeout, hello interval, etc) regarding utilization of individual routing table entries. A routing table entry is expired when not used recently. A set of predecessor nodes is maintained for each routing table entry, indicating the set of neighboring nodes which use that entry to route data packets. These nodes are notified by route error (RERR) packets when the next-hop link breaks. Each predecessor node, in turn, forwards the RERR to its own set of predecessors, thus effectively erasing all routes using the broken link.

However effective AODV may be in mobile network conditions, it is less resistant to packet drops than are proactive protocols. AODV does not frequently update the route to the destination. There seems to be a periodicity in the route request generation which could be attributed to poor link failure detection, resulting from large Hello timer values. AODV determines the best effort shortest path, i.e. is essentially the shortest successful path, which need not be the optimal path. In the case of proactive protocols, such as OLSR, there are sufficient exchanges of routing information to result in near-optimal routes. Therefore, OLSR is more resistant to packet drops at the MAC layer. However, one of the drawbacks of OLSR is that it generates routing traffic independent of application traffic. Due to the higher routing overhead in proactive routing protocols we chose the reactive routing protocol, AODV, in our cross-layer approach and tried to enhance the protocol performance with a Q-learning based self-configuration mechanism. This paper focuses on the challenges faced in reconfiguring the critical routing parameters (hello interval and active route timeout) to enhance network performance by dynamic context exchanges in heterogeneous networks.

2.3 Q-learning in MANET Routing

The various features of wireless networks lead to a set of optimization problems in achieving performance objectives. The idea of applying reinforcement learning to routing in networks was first introduced by Boyan and Littman [17], who used Q-learning [4] in a static packet switched network. They have shown that the Q-learning based routing approach can compete
with the shortest path algorithms, without prior knowledge regarding the network topology. Q-learning is also applied to routing in ad-hoc networks [1]. To the best of our knowledge, no existing routing scheme with reinforcement learning takes into consideration optimization goals (routing path length, load balancing, consistent link management, and aggregation) with cross-layer approaches together.

3. A Network Architecture with A Cross-layer Design for Autonomic Self-configuration

The strict layering architecture may not be efficient for wireless networks when heterogeneous traffic is served over a wireless channel with limited network resources. Efficiently utilizing the resources with QoS provisioning requires a cross-layer joint design and optimization approach. As a result better performance can be obtained from information exchanges across protocol layers.

The proposed cross-layer network architecture for our autonomic self-configuration network is illustrated in Fig. 1. One of the main advantages of the cross-layer design is to make protocols aware of the current state of the network from the point of view of the local node. This improves the higher level processes of the middleware, in which our Q-learning mechanism can exploit the knowledge of the network status and potentially improve system performance.

The initial proposals for ways to implement cross-layer interactions are also being made in the literature. These can be put into three categories [8]: Direct communication between layers, a shared database across the layers, and completely new abstractions. Specifically, we present the cross-layer interactions among layers by a shared network status module which supports vertical communications among the layers by acting as a repository for information collected by network protocols. Each module at each layer can access the shared network status module to exchange its data and interact.

Fig. 2 is an illustration of our cross-layer design approach showing how our Q-learning mechanism in the middleware layer can interact with the other reconfigurable modules in the other layers. The following steps describe the detailed procedure of our cognitive network management.

Step 1: The SLA management module in the middleware layer gathers application demands and sets SLA requirements.
Step 2: The Q-learning agent in the middleware layer receives the SLA requirements and sets the level of SLA compliance.
Step 3: At the network layer, the AODV protocol provides the Q-learning agent with the decision variables, including end-to-end delay, RERR and RREP.
Figure 2. Our Cross-layer design for distributed decision making in a mobile node.

Step 4: The Q-learning agent makes a decision about the action to enhance the performance and reconfigures the routing parameter (active route timeout and hello interval).

4. Q-learning based Self-configuration for AODV

4.1 Q-learning

Q-learning, one of the most important reinforcement learning algorithms, was presented by Watkins [4]. Each time an action $a$ is executed, the agent receives an immediate reward $r$ from the environment. It then uses this reward and the expected long-term reward to update the Q-values, which in turn influence future action selection. Its simplest form, one-step Q-learning, is defined as:

$$Q(s,a) = (1 - \alpha)Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a')]$$

where $\alpha$ is the learning rate, which models the rate of updating Q-values. As a model-free RL technique, Q-learning requires no knowledge about the underlying reward or transition mechanism; thus it is applicable to the problem of learning the routing strategy in ad hoc networks.

4.2 Q-learning based self-configuration

In our Q-learning based self-configuration in AODV routing, each node has two Q values, Q1 and Q2. Q1 denotes the Q value for unstable network status, which makes the node take the action of decreasing active route timeout and hello interval. Q2 presents the stability of the network, which will make the node take the action of increasing active route timeout and hello interval.

In our technique, each node makes its self-configuration decision based on the local routing information, represented as the two Q values which estimate the quality of the alternative actions. These values are updated each time the node receives a RREP packet.

If one of the following events occurs, and the reward $r$ is given by:

- **Reward:** $r = n \left(1 / \left(ETE_t / ETE_{max}\right)\right)$,

1) When a ROUTE REPLY packet reaches the source and there is a path from the source to destination.

- **Penalty:** $r = n \left(ETE_t / ETE_{max}\right)$

2a) When a ROUTE ERROR packet is generated as the route is broken.

2b) When a ROUTE REPLY packet reaches the source and there is no path from the source to destination.

where $ETE_t$ is the current end-to-end delay, $ETE_{max}$ is the maximum end-to-end delay that complies with the SLA agreement and $n$ denotes the normalization constant. The reason why end-to-end delay is used to affect the reward and penalty is that SLA is more heavily dependent on end-to-end delay. Fig. 3 shows the distributed decision making and self-configuration by Q-learning at each MN in a heterogeneous network.

5. OPNET Simulation

The performance of our network system has been evaluated with the OPNET simulation tool [18]. As shown in Fig. 4 the simulation network is defined as a flat terrain of $3000 \times 3000$ m with 25 MNs, 2 MANET gateways, 1 mobile wireless video streaming server and 2 Ethernet video streaming servers. The mobile wireless video streaming server and MNs were equipped with a wireless network interface. At the physical and data link layers, the 802.11b standard was used for the analysis. The 802.11b standard was selected, as it is readily available in OPNET and provides the capability of defining different transmission data rates for the mobile nodes. Then it is possible to assign higher-capacity links to those mobile nodes that belong to higher topology levels. It should be noted that the main purpose of the simulation scenarios was to provide a framework to compare the performance of our Q-learning based self-configuration in AODV and AODV protocols.

The traffic model used to gather the simulation results consists of three constant bit rate (CBR) sources. The traffic models were constructed as follows:

- **Traffic #1** MANET-to-MANET: 300 kbps downloading video streaming from Wireless Video Streaming Server to mobile_node_1.

- **Traffic #2** MANET-to-Ethernet: 300 kbps uploading video streaming from mobile_node_2 to Ethernet Video Streaming Server 1.

- **Traffic #3** Ethernet-to-MANET: 300 kbps downloading video streaming from Ethernet Video Streaming Server 2 to mobile_node_3.

In terms of SLA compliance, the maximum end-to-end delay, $ETE_{max}$, was 3ms which complies with the SLA agreement.
In this analysis, node mobility is assumed to be random (i.e., independently selected by each node using a uniform distribution) movement rather than group movement. The mobile nodes are assigned a maximum speed of 15 m/s. In the simulation scenarios, each mobile node changes its location within the network based on the “random waypoint” model; that is, the node randomly selects a destination, moves toward that destination at a speed not exceeding the maximum speed (15 m/s) and then pauses; this interval is known as pause-time. In order to calculate the impact of high mobility on the protocol overhead, pause-time is assigned 0 s. It should be noted that a pause-time of 0 s represents the worst case scenario as the mobile nodes are constantly moving during the simulation. In the training episode, from the beginning of simulation to 150 s, each node randomly chooses actions decreasing or increasing active route timeout and hello interval.

6. Performance Evaluation
The performance of our Q-learning based self-configuration was evaluated in terms of packet delivery ratio, end-to-end delay, SLA compliance, and the overhead load in the network due to control messages generated by the AODV routing mechanism. The performance results were derived under our Q-learning based self-configuration and are compared with the performance
6.1 Control overhead and route errors
The control overhead is measured in terms of the number of control messages generated by the routing algorithm. Fig. 5 illustrates the number of routing control messages generated or relayed in the network. On average, our Q-learning based self-configuration achieves a reduction of control messages equivalent to 55.3 percent. In the Q-learning based AODV protocol simulation, our Q-learning mechanism at each node self-configures the active route timeout and hello interval according to the Q-value. Due to the self-configured parameters, the nodes send RREQs more appropriately to account for failed routes, improving the route freshness and the link failure detection processes. The route error is evaluated as the average number of RERR packets per second. Q-learning based self-configuration achieves a reduction in the number of RERR messages equivalent to 71.9 percent as shown in Fig. 6.

6.2 Protocol Performance
The performance of Q-learning based self-configuration in AODV and of AODV protocol is evaluated in terms of packet delivery ratio, end-to-end delay and SLA compliance metrics. The packet delivery ratio is defined as the percentage of data packets successfully delivered to the intended destination. The end-to-end delay metric is defined as the average elapsed time between the generation and reception of data packets.

Fig. 7 shows the packet received at mobile_node_1 in traffic #1 MANET-to-MANET for the AODV and Q-learning based self-configuration in AODV protocol. From the results in Table 1 and Fig. 7, it is clear that the Q-learning based self-configuration in AODV delivers at least 13 percent more packets than does the standard AODV. This result can be explained in terms of the network overhead due to the control messages generated by the AODV protocol. From the results presented in Table 1, it is clear that the AODV mechanism generates a greater number of control messages than does our AODV protocol with Q-learning based self-configuration. This, in turn, translates into a higher probability of lost control messages in AODV due to collisions in the wireless medium. Consequently, routing paths are less reliable under AODV.
enhancement of end-to-end delay equivalent to 43.7 percent and SLA compliance equivalent to 14.7 percent. From these results, it is clear that under the Q-learning based reconfigurable architecture, it is possible to achieve a lower end-to-end delay metric than under an AODV architecture. Reduced traffic in the wireless medium allows our AODV to realize a shorter queuing delay, resulting in shorter end-to-end delays. It is expected that our optimized AODV will outperform AODV under high network mobility conditions because of its capability of reducing the control overhead and quickly reestablish new routing paths, as well as efficiently using high-capacity links.

7. Conclusions and Future Research
In this paper, we described our vision of a framework for autonomously reconfigured network systems with a cross-layer approach. The Q-learning based self-configuration mechanism has been proposed to improve the performance of AODV. This is applicable to large heterogeneous networks, where the characteristics of the mobile nodes and application demands are different. We also presented the experimental results of our network system using OPNET. The performance results confirm that in comparison to original AODV, our Q-learning based self-configuration mechanism dramatically reduces the protocol overhead (55.3% reduction). It achieves a higher packet delivery ratio (13.1% enhancement) while incurring shorter queuing delays. More specifically, with the Q-learning based self-configuration, it is possible to achieve shorter end-to-end delays (43.7% reduction) while reducing the incidence of lost data packets. Therefore, our autonomous self-configuration mechanism successfully improves the scalability and adaptivity of the original AODV protocol in a heterogeneous network environment.

Future research may include performance evaluation under diverse mobility patterns with other routing protocols (proactive, hierarchical and hybrid). The work in this paper highlights some interesting and potentially important areas for future work. Some of these are enumerated as follows:

<table>
<thead>
<tr>
<th>Items</th>
<th>AODV</th>
<th>Q-learning based Reconfiguration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#1 MANET to MANET</td>
<td>#2: MANET to Ethernet</td>
</tr>
<tr>
<td>Route Discovery Time (s)</td>
<td>0.575856</td>
<td>0.41425</td>
</tr>
<tr>
<td>Route Errors (packets/s)</td>
<td>4.178218</td>
<td>1.178218</td>
</tr>
<tr>
<td>Control Overhead (packets/s)</td>
<td>42.17492</td>
<td>18.80528</td>
</tr>
<tr>
<td>Packet Jitter (sec)</td>
<td>0.247187</td>
<td>0.000873</td>
</tr>
<tr>
<td>ETE delay (s)</td>
<td>1.550288</td>
<td>0.576016</td>
</tr>
<tr>
<td>SLA (%)</td>
<td>16.05308</td>
<td>32.41111</td>
</tr>
<tr>
<td>Packet Delivery Ratio (%)</td>
<td>64.77414</td>
<td>84.22339</td>
</tr>
</tbody>
</table>

Table 1. Summary of our OPNET simulation results
Proactive and reactive network management
There is a fundamental tradeoff between proactive and reactive routing protocols, namely improved delay or reduced control overhead. A proactive routing protocol generates routing traffic independent of application traffic. Due to the higher routing overhead in proactive routing protocols (e.g. OLSR), we have chosen the reactive routing protocol, AODV in our cross-layer approach and tried to enhance the protocol performance with a Q-learning based self-configuration mechanism. However, it is inevitable that certain static networks will have especially high QoS demands which require the use of proactive routing. How to use proactive routing while minimizing the network-layer overhead is of key interest.

Performance evaluations in various network environments
We plan to verify our scheme for other simulation scenarios, i.e. characterized by more heterogeneous networks (e.g. 3G, WiMAX and optical networks), various traffic models and mobility models. It could be useful to provide results as a combination of larger networks and nodes.

Learning rate sensitivity and optimization
We executed the simulations with a relatively low learning rate ($\alpha = 0.001$). In our experiments, we saw that the Q-learning based self-configuration was sensitive to learning rate reassignment. One solution to this problem is to use Bayesian exploration [19], which will be explored in our future work, to tune and optimize learning rate values. There is also a need to investigate optimization accuracy and the process of reward value assignment in the Q-value computation, in addition to the selecting of correct parameters for self-configuration.

Cross-layer design for heterogeneous application traffic with QoS guarantees
To achieve desired QoS guarantees it would be critical to consider the change of user demands in the application layer. For consistent QoS support, the MAC layer could provide the essential context. The MAC layer could provide an indication of the network congestion level and achievable data rates; these could be used to determine whether the lower layer capability can meet the upper layer requirement.

References