MRL-CC: A Novel Cooperative Communication Protocol for QoS Provisioning in Wireless Sensor Networks

Xuedong Liang*
Department of Informatics, University of Oslo[1], Norway
Rikshospitalet University Hospital[2], Norway
E-mail: xuedongl@medisin.uio.no
*Corresponding author

Min Chen
Department of Electrical and Computer Engineering,
University of British Columbia, Canada
E-mail: minchen@ece.ubc.ca

Yang Xiao
Department of Computer Science,
University of Alabama, USA
E-mail: yangxiao@ieee.org

Ilangko Balasingham
Department of Electronics and Telecommunications,
Norwegian University of Science and Technology[1], Norway
Rikshospitalet University Hospital[2], Norway
E-mail: ilangkob@medisin.uio.no

Victor C.M. Leung
Department of Electrical and Computer Engineering,
University of British Columbia, Canada
E-mail: vleung@ece.ubc.ca

Abstract: Cooperative communications have been demonstrated to be effective in combating the multiple fading effects in wireless networks, and improving the network performance in terms of adaptivity, reliability, data throughput and network lifetime. In this paper, we investigate the use of cooperative communications for quality of service (QoS) provisioning in resource-constrained wireless sensor networks, and propose *MRL-CC*, a Multi-agent Reinforcement Learning based multi-hop mesh Cooperative Communication mechanism. In *MRL-CC*, a multi-hop mesh cooperative structure is constructed for reliable data disseminations. The cooperative mechanism that defines cooperative partner assignments, and coding and transmission schemes is implemented using a multi-agent reinforcement learning algorithm. We compare the network performance of *MRL-CC* with *MMCC* (Chen et al., 2009), a Multi-hop Mesh structure based Cooperative Communication scheme, and investigate the impacts of network traffic load, interference and sensor node's mobility on the network performance. Simulation results show that *MRL-CC* performs well in terms of a number of QoS metrics, and fits well in large-scale networks and highly dynamic environments.

Keywords: Cooperative communications; reinforcement learning; quality of service; wireless sensor networks.

Reference to this paper should be made as follows: Xuedong Liang, Min Chen, Yang Xiao, Ilangko Balasingham and Victor C.M. Leung (2004) '*MRL-CC*: A Novel Cooperative Communication Protocol for QoS Provisioning in Wireless Sensor Networks', Int. J. Sensor Networks, Vol. 1, Nos. 1/2/3, pp.64–74.

Biographical notes: Xuedong Liang is a PhD candidate at the Department of Informatics, University of Oslo, Norway. He is also a research fellow at Rikshospitalet University Hospital, Oslo, Norway. His research interests are in the areas of communication protocols, formal modeling and simulation, QoS provisioning in wireless sensor networks.

1 Introduction

Wireless sensor networks (WSNs) have numerous potential applications, e.g., battlefield surveillance, medical care, wildlife monitoring and disaster response. In mission-critical applications, the wireless networks used for communication must ensure that data packets can be delivered to the data processing center reliably and efficiently. In other words, a set of QoS requirements (e.g., end-to-end delay, packet delivery ratio and communication bandwidth) on network performance must be satisfied. However, due to the dynamic topology, time-varying wireless channel, and severe constraints on power supply, computation power and communication bandwidth of sensor nodes, quality of service (QoS) provisioning is challenging in WSNs.

As surveyed by Zhang and Zhang (2008); Hanzo and Tafazolli (2007); Zhang and Mouftah (2005); Al-Karaki and Kamal (2004), a large number of QoS support communication protocols have been proposed for WSNs recently. Most of these protocols are based on network traffic engineering, i.e., sensor nodes maintain network state information (e.g., transmission delay, outage event probability and available communication bandwidth) and use various algorithms to perform QoS routes' computation and maintenance. However, the network state information is inherently imprecise due to the dynamic wireless channel, node mobility and varying duty cycle. Moreover, significant communication overheads are incurred to the network due to the dissemination of network state information throughout the network, especially in large-scale networks. Thus, research on distributed, lightweight and highly adaptive communication protocols with QoS support is still needed.

In recent years, cooperative communications have been proposed to exploit the spatial and time diversity gains in wireless networks (Nosratinia et al., 2004; Hong et al., 2007). Users in cooperative communication systems work cooperatively by relaying data packets for each other, and thus forming multiple transmission paths or virtual multiple-input-multiple-output (MIMO) system to the destination without the need of multiple antennas at each user. By utilizing the broadcast nature of the wireless medium and spatial distribution of sensor nodes, cooperative communications can be used to improve the network performance of WSNs. For instance, users experience severe channel fading can have other users/partners helping them to deliver packets to the destination with satisfied QoS requirements. Cooperative mechanism is the key to the performance of cooperative communication systems, however it is challenging to find the optimal cooperative policies, e.g., when to cooperate, how to cooperate and whom to coopeate with, in dynamic wireless networks (Ibrahim et al., 2008).

In this paper, we investigate the use of cooperative communications for QoS provisioning in resource-constrained WSNs, and propose *MRL-CC*, a Multi-agent Reinforcement Learning based Cooperative Communication protocol. In *MRL-CC*, a multi-hop mesh cooperative structure is constructed for reliable data disseminations. The cooperative mechanism that defines cooperative partner assignments, and coding and transmission schemes is implemented at each node using a multi-agent reinforcement learning algorithm. The cooperative nodes, regarded as multiple agents in the context of reinforcement learning framework, learn the optimal cooperative policy through experiences and rewards without the need of prior knowledge of the wireless network model. Thus, by considering the interactions with both the environment and other agents, multiple agents can cooperatively learn the optimal policy by using locally observed network state information and limited information exchange. Therefore, optimal network performance can be achieved without the need of maintaining precise network state information and centralized control.

The rest of the paper is organized as follows. Section II presents the related work. The background information on reinforcement learning and its applications in WSNs is provided in Section III. Section IV describes the architecture overview, design issues and reinforcement learning algorithm implementations of *MRL-CC*. The performance analysis is presented in Section V. Finally, Section VI concludes the paper and discusses the future research directions.

2 Related Work

A large number of cooperative communication protocols have been proposed recently. Cooperation diversity gains, transmitting, receiving and processing overheads, are investigated by Sadek et al. (2006). Cooperative issues across the different layers of the communication protocol stack, self-interested behaviors and possible misbehaviors are explored in (Conti et al., 2004). Lin et al. (2009) proposed a cooperative relay framework which accommodates the physical, medium access control (MAC) and network layers for wireless ad-hoc networks. In the network layer, diversity gains can be achieved by selecting two cooperative relays based on the average link signal-to-noise ratio (SNR) and the two-hop neighborhood information. A cooperative communication scheme combining relay selection with power control is proposed in (Zhou et al., 2008), where the potential relays compute individually the required transmission power to participate in the cooperative communications. A variety of cooperative diversity protocols are proposed by Laneman et al. (2004), namely, amplify-and-forward, decode-and-forward, selection relaying, and incremental relaying. The performance of the protocols in terms of outage events and associated outage probabilities are evaluated respectively. Coded cooperation (Hunter et al., 2002) integrated cooperation with channel coding and works by sending different parts of each user's code word via two independent fading paths. Sendonaris et al (2003a,b) implemented a coopera-

tion strategy for mobile users in a conventional code division multiple access (CDMA) systems, in which users are active and use different spreading code to avoid interferences. In (Hunter and Nosratinia, 2004), distributed cooperative protocols, including random selection, received SNR selection and fixed priority selection, are proposed for cooperative partner selection. The outage probability of the protocols are analyzed respectively. CoopMAC, a cooperative MAC protocol for IEEE 802.11 wireless networks, is presented by Liu et al. (2006). CoopMAC can achieve performance improvements by exploiting both the broadcast nature of the wireless channel and cooperative diversity. REER, a scalable, energy efficient and error-resilient routing protocol for dense WSNs is proposed by Chen et al. (2008). Based on geographical information, REER's design harnesses the advantages of high node density and relies on the collective efforts of multiple cooperative nodes to deliver data, without depending on any individual ones. *MMCC* is proposed by Chen et al. (2009), which aims to improve the network reliability and prolong the network lifetime. A mesh structure is established for reliable data dissemination, random based and distance based values are used as the forwarding node selection criteria. However, the random timer based criterion cannot achieve optimal performance and incurs extra transmission delay; the distance based value criterion is not always effective in dynamic WSNs.

Most of the previous research is based on the following assumptions:

- orthogonal radios (FDMA, TDMA or CDMA) are available at all nodes,
- the information of fading co-efficients among the source, sink and cooperative partners, are available at each node,
- the sink node always has the updated information (full or partial) of the cooperative partner assignments, handover, and the inter-user channel properties.

However, half-duplex and CSMA/CA based wireless transceivers are employed in most of the sensor node platforms in practice, i.e., sensor nodes cannot transmit and receive signals simultaneously. Besides, due to the distributed nature of WSN applications, the sink node usually does not have the information of the inter-user channel properties in real time, as well as the cooperative scheme and partner assignments among the participants.

Another limitation is that most existing research focus on cooperation between a pair of users. Cooperations among multiple users are investigated in (Sadek et al., 2007; Hunter and Nosratinia, 2004), however the research is limited on one hop communication networks.

3 Reinforcement Learning and its Applications

Reinforcement learning provides a framework in which an agent can learn control policies based on experiences and rewards. In the standard reinforcement learning model, an agent is connected to the environment via perception and action, as shown in Fig. 1. On each step of interaction, the agent receives an input, i, some indication of the current state, s, of the environment; the agent then choose an action, a, to generate as output. The action changes the state of the environment, and the value of the state transition is communicated to the agent through a scalar reinforcement learning signal, r. The agent's behavior, B, should choose actions that tend to increase the long-term sum of values of the reinforcement signal (Kaelbling et al., 1996). The main idea of reinforcement learning is to strengthen the good behaviors of the agent while weaken the bad behaviors through rewards given by the environment (Zhang and Ma, 2007).

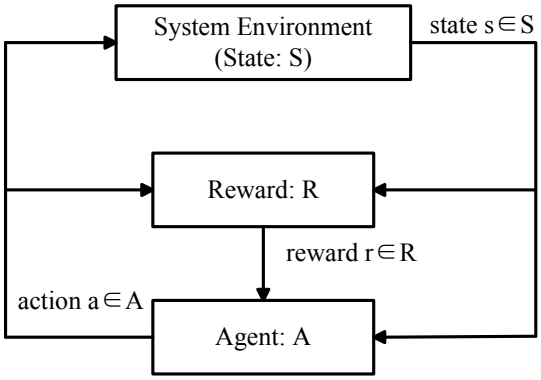

Figure 1: A standard reinforcement learning model

The underlying concept of reinforcement learning is Markov Decision Process (*MDP*). A *MDP* models an agent acting in an environment with a tuple (S,A,P,R), where S is a set of states, A denotes a set of actions. $P(s^{'}|s,a)$ is the transition model that describes the probability of entering state $s' \in S$ after executing action $a \in A$ at state $s \in S$. $R(s,a,s^{'})$ is the reward obtained when the agent executes a at s and enter $s^{'}$. The goal of solving a MDP is to find an optimal policy, $\pi : S \mapsto A$, that maps states to actions such that the cumulative reward is maximized. Detailed information on reinforcement learning can be found in (Kaelbling et al., 1996).

Reinforcement learning has been applied in the design of communication protocols for wireless networks in the last years. (Boyan and Littman, 1993) is the first work which investigated the use of reinforcement learning in packet routing in dynamically changing networks. Yu et al. (2008) presented a novel method for QoS provisioning for adaptive multimedia in wireless networks via average reward reinforcement learning in conjunction with stochastic approximation. Liu and Elhanany (2006) proposed RL-MAC, a reinforcement learning based MAC protocol for WSNs. A node implemented with RL-MAC can adjust the frame active time and duty cycle dynamically to adapt to its own traffic load, as well as its incoming traffic characteristics. A near-optimal transmission strategy that chooses the optimal modulation level and transmission power while adapt-

ing to the incoming traffic rate, buffer and channel conditions, is proposed by Pandana and Liu (2005).

Multi-agent systems (MASs) are systems that multiple agents are connected to the environment and may take actions to change the state of the environment (Stone and Veloso, 2000). In MASs, independent distributed reinforcement learning (IndRL) is a basic learning algorithm, that agent assumes itself is the only one that can change the state of the environment, and does not consider the interactions among itself and other agents. Obviously, this will lead to individual agent's greedy and selfish behaviors, i.e., no cooperation is considered and thus only local optimization can be achieved (Busoniu et al., 2006).

In WSNs, data packets are usually routed to the destination node through multi-hop communications. The QoS performance of the route relies on the overall routing procedures, i.e., each node, which involves in the routing procedure, contributes to the end-to-end QoS performances. It is worth to note that, nodes which are not directly involved in the routing procedure but are within the communication range of the forwarding nodes, may take actions (e.g., packet originating, forwarding) and have impacts on the route's QoS performance as well, due to the shared and contention nature of the wireless channel. Thus, WSNs can be characterized as multi-agent systems, where sensor nodes can be considered as agents, wireless medium and packet flows can be regarded as environment. Intuitively, agents should cooperate with each other to achieve global optimal performance.

A number of distributed reinforcement learning (DRL) algorithms have been proposed for MASs in (Schneider et al., 1999), namely, global reward DRL, local reward DRL, distributed reward DRL, distributed value function DRL (DVF-DRL). In DVF-DRL, by exchanging local state values with the immediate neighboring agents, agents can consider both the rewards of the neighboring and non-neighboring agents, and then choose action to maximize the weighted sum of the rewards of all the nodes in the network. Thus, global cooperation can be achieved and overall network performances can be improved (Tham and Renaud, 2005).

4 Cooperative Mechanism Design and Implementation

In this section, we describe the architecture and design issues of *MRL-CC*. First, an architecture overview of the network organization is presented. Then we describe the three phase operations of *MRL-CC*, namely mesh cooperative structure construction, Q-learning algorithm initialization and data dissemination phases. Finally, the design and implementation of the Q-learning algorithm are illustrated.

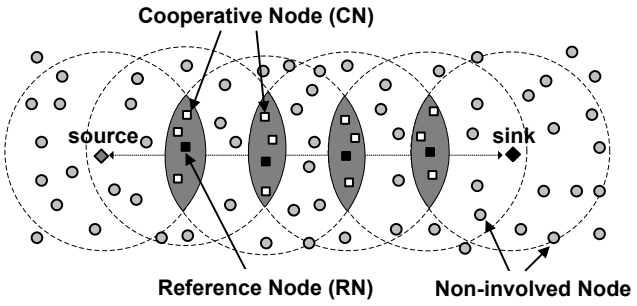

Figure 2: Multi-hop mesh cooperative structure for data dissemination in WSNs

4.1 Architecture Overview

As shown in Fig. 2, *MRL-CC* employs a multi-hop mesh cooperative structure for reliable data disseminations in WSNs, i.e., data packets originated from the source are forwarded to the sink node by groups of cooperative nodes (denoted as CNs) relaying. In each group of CNs, a node will be elected as the forwarding node to forward the data packet to the adjacent group of CNs towards the sink node, and other nodes play as cooperative partners and will help in the packet forwarding in case the forwarding-node-election fails or the packet is corrupted in the transmissions.

The forwarding-node-election in the CNs is based on a multi-agent reinforcement learning algorithm, i.e., each node is implemented with a Q-learning algorithm (Sutton and Barto, 1998), a model-free method which learns the value of a function $Q(s, a)$ to find an optimal decision policy. Each node maintains the Q-values of itself and its cooperative partners, which reflect the qualities (e.g., transmission delay, packet delivery ratio) of the available routes to the sink. When a packet is received by the nodes in a group of CNs, each node will compare its own Q-value with those of other nodes in the CNs; the node which determines that it has the highest Q-value will be elected to forward the data packet to the adjacent group of CNs towards the sink.

Each time a packet is forwarded, all the nodes in the group of CNs will receive immediate rewards from the environment, which represent the quality of packet forwarding in terms of transmission delay and packet loss rate. Nodes then use the rewards to update the Q-values, which will influence their future decisions of forwarding-node-election.

The algorithm will reach convergence after a certain amount of time, depending on the network size, node mobility and density. Then, nodes are able to use the learned policy to take appropriate actions, i.e., node with the highest Q-value will forward the packet to the adjacent group of CNs towards the sink. To adapt to the dynamic nature of WSNs, *MRL-CC* explores the environment with a certain probability of ε, namely ε-greedy method (Sutton and Barto, 1998). That is, with the probability of $1 - \varepsilon$, the node with the the highest Q-value will forward the packet

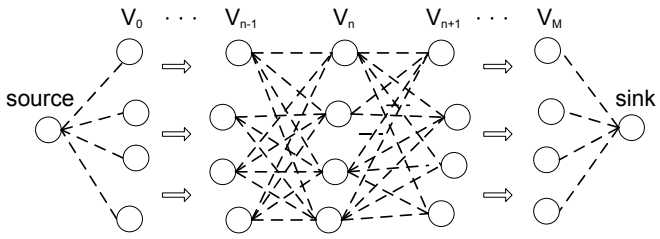

Figure 3: Cooperation between adjacent groups of cooperative nodes

to the adjacent group of CNs; and with the probability of ε, a randomly chosen node will forward the packet to the adjacent group of CNs.

Thus, without using complicated algorithms for the wireless link quality prediction (Mohrehkesh et al., 2006; Shah and Nahrstedt, 2002), or explicitly frequent updating and maintaining of the network state information, nodes can find the optimal cooperative policy through a series of trial-and-error interactions with the dynamic environment.

4.2 Multi-hop Cooperative Structure Construction Phase

To construct a multi-hop mesh cooperative structure, a set of nodes, termed as reference nodes (denoted as RNs) between the source node and the sink node (the source and the sink are also RNs) is first selected. The RNs are determined sequentially starting from the source to the sink, and the distance between two adjacent RNs is an application specific value, which is a trade-off between reliability and energy efficiency. Once the RNs are determined, a set of nodes around each RN will be selected as the cooperative nodes (denoted as CNs), and thus, a multi-hop mesh cooperative structure is constructed in this phase. Data packets originated from the source will be forwarded to the sink by groups of CNs relaying.

A part of the mesh structure is shown in Fig. 3, where the n_{th} cooperative group is denoted by V_n, and its adjacent groups are denoted as V_{n-1} and V_{n+1}, which are one hop farther and closer towards the sink than V_n, respectively. Ideally, each node in V_n is connected with all the nodes in V_{n-1} and V_{n+1}; however, the links are unreliable and the qualities are varying over time and space due to the time-varying wireless channel and dynamic network topology.

The number of cooperative nodes in each CNs, and the number of groups of CNs in the network, depend on the network size, node density and the trade-off between reliability and energy efficiency. Details of the mesh cooperative structure construction and parameters selection can be found in (Chen et al., 2008, 2009).

4.3 Q-learning Algorithm Initialization Phase

In the initialization phase, each node is assigned with an initial Q-value. For node $i \in V_n$, its initial Q-value (denoted as $Q_{n_i}^{ini}$) is calculated based on the relative distance (compared with its cooperative partners in V_n) from node i to the nodes in V_{n+1}, as shown in (1).

$$Q_{n_i}^{ini} = d_{V_n,V_{n+1}}/d_{i,V_{n+1}} \tag{1}$$

where $d_{V_n,\mathrm{V_{n+1}}}$ is the average distance between V_n and V_{n+1}, which can be calculated as in (2).

$$d_{V_n,V_{n+1}} = \frac{1}{N}\sum_{i=0}^{N} d_{i,V_{n+1}} \tag{2}$$

where N is the number of cooperative nodes in V_n (for simplicity, we assume N is identical for each group of CNs). $d_{i,\mathrm{V_{n+1}}}$ is the average distance between node i and the nodes in V_{n+1}, which can be calculated as in (3).

$$d_{i,V_{n+1}} = \frac{1}{N}\sum_{j=0}^{N} d_{i,j} \tag{3}$$

In the initialization phase, node i exchanges its initial Q-value with the nodes in V_{n-1}, V_n and V_{n+1}, by broadcasting the initialization messages.

4.4 Data Dissemination Phase

When a data packet is received by the nodes in V_n, each node will compare its Q-value with those of other cooperative nodes in V_n. The node which determines that it has the highest Q-value will forward the data packet to V_{n+1}, and other nodes in V_n will deduce whether the packet forwarding is successful or not, by monitoring the packet transmission at the next hop, i.e., from V_{n+1} to V_{n+2}.

If the data packet is received by V_{n+1}, the task for the current round of data forwarding for the nodes in V_n is finished. Thus, the nodes in V_n will receive positive rewards and update their Q-values, accordingly. Note in the Q-learning algorithm, all the nodes in V_n, including the elected packet forwarding node and the other cooperative nodes, will receive positive reward. This is because that the other cooperative nodes have made the correct decision of the forwarding-node-election.

If the packet forwarding fails, all the nodes in V_n will receive negative rewards (i.e., get punishment) and their Q-values will be updated. Then, another forwarding-node-election will be conducted by the nodes in V_n for packet re-transmission based on the updated Q-values.

There are two reasons may cause the failure of packet forwarding:

- *forwarding election failure*: in this case, the node elected to forward the data packet is not eligible due to the out-of-date Q-value lists stored at the nodes in V_n,
- *packet transmission failure*: the packet is corrupted or collided during the transmission from V_n to V_{n+1}.

To address the problem of packet forwarding failure, each node maintains a timer T_{rf} for packet re-forwarding. That is, if nodes in V_n do not hear the packet delivering

from V_{n+1} to V_{n+2} before the timer expires, nodes in V_n deduce that the packet is not successfully forwarded from V_n to V_{n+1} and another forwarding procedure will be restarted by the nodes in V_n based on the updated Q-values.

T_{rf} is defined as in (4).

$$T_{rf} = T_{BO_1} + T_l + T_{IFS} + T_{TA} + T_{BO_2} \tag{4}$$

where T_{BO_1} and T_{BO_2} are the maximum allowed backoff time at the nodes in V_n and V_{n+1} respectively, and the values are defined by the employed MAC layer protocols. T_l is the packet transmission time and $T_l = \frac{l_d}{R}$, where l_d is the packet size (including overheads) in bits and R is the data transmission rate of the transceiver in bits per second. T_{IFS} and T_{TA} are the inter frame space (IFS) and the transceiver's transmitting to receiving turnover time respectively, which are specified by the underlying communication protocols.

Once the Q-learning algorithm reaches convergence, nodes are able to use the learned cooperative policy to take appropriate actions, i.e., node with the highest Q-value will be elected to forward the data packet to V_{n+1}, and nodes with lower Q-values monitor the packet forwarding and will help in the packet delivering if the packet forwarding from V_n to V_{n+1} fails.

4.5 Q-learning Algorithm Implementation

In the context of reinforcement learning, for node $i \in V_n$, we define the states, actions and rewards as follows:

State $S_i = \{k\}$, $k \in \{V_{n-1}, V_n, V_{n+1}\}$.

Action

$$A_i = \begin{cases} a_f \\ a_m \end{cases} \tag{5}$$

The execution of a_f represents that node i is elected by the nodes in V_n to forward the packet from V_n to V_{n+1}, and the execution of a_m denotes that node i monitors the packet forwarding.

Reward Function The reward function is defined as in (6).

$$Rwd(i) = \begin{cases} \left(\frac{d_{V_n,sink} - d_{V_{n+1},sink}}{d_{V_n,sink}}\right) \Big/ \left(\frac{T_{V_{n+1}} - T_{V_n}}{T_{rmn}}\right) & \text{(6a)} \\ -\frac{T_{rf}}{T_{rmn}} & \text{(6b)} \end{cases} \tag{6}$$

Eq. (6a) is used to calculate the reward when the packet forwarding is successful. $d_{V_n,sink}$ is the average distance between the nodes in V_n and the *sink*, which can be calculated as in (7).

$$d_{V_n,sink} = \frac{1}{N} \sum_{i=0}^{N} d_{i,sink} \tag{7}$$

$T_{V_{n+1}}$ and T_{V_n} are the packet forwarding time at V_{n+1} and V_n, respectively, observed at node i using the local clock. T_{rmn} is the maximum amount of time that can be elapsed in the remaining path to the sink to meet the QoS requirement on the end-to-end delay. T_{rmn} is updated in each packet forwarding procedure, and the value is encapsulated in the data packet. The positive reward reflects the quality of the packet forwarding, i.e., relative progress towards the sink over time.

Eq. (6b) is used to calculate the reward when the packet forwarding fails. The negative reward reflects the relative delay caused by the unsuccessful packet transmission from V_n to V_{n+1}.

The updating of Q-value iterates at each node in each forwarding procedure, and distributed value function - distributed reinforcement learning algorithm (DVF-DRL) (Schneider et al., 1999) is used in the updating iteration.

For 1-hop packet forwarding, at iteration t, node $i \in V_n$ forwards a packet to V_{n+1}, and then $j \in V_{n+1}$ is elected to continue the packet forwarding towards the sink. Node i updates its Q-value as in (8).

$$\begin{aligned} Q_i^{t+1}(s_i^t, a_i^t) = (1-\alpha) Q_i^t(s_i^t, a_i^t) + \alpha(r_i^{t+1}(s_i^{t+1}) + \\ \gamma w(i,j) \max_{a_j \in A_j} Q_j(s_j^t, a_j^t) + \\ \gamma \sum_{i' \in V_n, i' \neq i} w(i,i') \max_{a_{i'} \in A_{i'}} Q_{i'}(s_{i'}^t, a_{i'}^t)) \end{aligned} \tag{8}$$

where α is the learning rate, which models the updating rate of the Q-value. r denotes the immediate reward of execution of the action. The weight of the future rewards is defined by γ. $w(i,j)$ models how strongly node i weights j's reward in its average. Eq. 8 shows that node i's Q-value is a weighted sum of i's Q-value at the previous state, the action's immediate reward, the maximum Q-value of j which is elected as the forwarding node in V_{n+1} at the next hop, and the Q-values of all of i's cooperative partners in V_n.

Considering the multi-hop forwarding procedure as shown in Fig. 2, node i_0 (the source) originates a packet destinated to the sink node *snk*. In the learning period, a number of nodes (denoted as $i_0, i_1, \ldots i_M$) are elected for packet forwarding sequentially from the source to the sink node. For node i_0, the Q-value is updated as in (9)

$$\begin{aligned} Q_{i_0}(s_{i_0}, a_{i_0}) = \\ (1-\alpha) \sum_{n=0}^{M} (\alpha\gamma)^n \max_{a_{i_n} \in A_{i_n}} Q_{i_n}(s_{i_n}, a_{i_n}) \prod_{j=0}^{n} w(i_j, i_{j+1}) \\ + \sum_{n=0}^{M} \alpha^{n+1} \gamma^n r(s_{i_n}, a_{i_n}) \prod_{j=0}^{n} w(i_j, i_{j+1}) \\ + \sum_{n=0}^{M} (\alpha\gamma)^{n+1} \sum_{i' \in V_n, i' \neq i} \max_{a_{i'} \in A_{i'}} Q_{i'}(s_{i'}, a_{i'}) \prod_{j=0}^{n} w(i_j, i_{j+1}^{'}) \\ + \alpha^{M-1} \gamma^M Q_{snk}(s_{snk}, a_{snk}) \prod_{j=0}^{M} w(i_j, i_{j+1}) \end{aligned} \tag{9}$$

Both of the first and second terms are contributed by the nodes which are directly involved in the forwarding procedure. The first term is the weighted sum of the maximum Q-values of the nodes which are elected as the forwarding nodes sequentially from the source to the sink. The second term is the weighted sum of the immediate rewards achieved by the elected forwarding nodes. The third term defines the weighted sum of the maximum Q-values of the cooperative partners at each group of CNs, contributed by the nodes which are not directly involved in the forwarding procedure. The weighted Q-value of the sink node, which is set as a constant value is modeled in the last term. In the calculation of the expected end-to-end rewards, the future rewards are weighted by the discounting factor $\gamma \in (0, 1)$. The reason is that in dynamic WSNs, it is more appropriate of using the discounted values of the future rewards than using the average values, because the network topology, link qualities, and node's duty cycles are tend to be varying.

Eq. 9 illustrates that although the source node only has locally observed network state information, and only communicates with its cooperative partners within the same group of CNs, it can estimate the end-to-end QoS performance of the routes to the sink, by calculating the weighted sum of its own immediate reward, the rewards that are expected to be achieved by the potentially elected forwarders in the remaining path to the sink, and the rewards of all of the non-directly involved nodes in the network. Therefore, nodes in *MRL-CC* can work in a cooperative manner by choosing actions to maximize the global rewards.

The pseudo code of the Q-learning algorithm is listed at Algorithm 1.

Algorithm 1 The Q-learning algorithm at sensor node i in the n_{th} cooperative group (V_n)

```
begin
initialization
  setup its cooperative partners' Q-value list table
  calculate its initial Q-values using Eq. 1
  exchange the initial Q-value with its partners
loop
  if receive data packet P_m from V_{n-1} then
    compare its Q-value with those of its cooperative partners' in V_n
    if its Q-value is the highest one then
      with the probability of 1 − ε, forward P_m to V_{n+1}
    else
      monitor the packet forwarding
    end if
  end if
  if overhear P_m is delivered from V_{n+1} to V_{n+2} before T_rf expires then
    calculate the reward using Eq. (6a) and update the Q-value
  else
    calculate the reward using Eq. (6b) and update the Q-value
    restart the forwarding-node-election
  end if
end loop
```

Table 1: Simulation Parameters

Parameters	**Value**
Number of sensor nodes	200
Simulation area	400 m × 200 m
Wireless channel model	Log shadowing wireless model
Path loss exponent	2.4
Collision model	Additive interference model
Mobility model	Random waypoint model
Physical and MAC layer	IEEE 802.15.4 standard
Packet length	40 bytes
Communication range	50 m
Data transmission rate	250 kbps
Simulation time	400 s
Number of simulation runs	10
N	4
ε	0.1
α	0.1
γ	0.5
$w(i,j)$	0.5, if j is the forwarding node $\frac{1}{2N}$, if j is the cooperative partners

5 Performance Evaluation

To study the network performance of *MRL-CC*, we compare it with *MMCC*, a multi-hop mesh structure based cooperative communication scheme. A random forwarding-node-election scheme is also implemented and its performance is used as a comparison baseline.

5.1 Simulation Environment

We simulate a WSN where 200 sensor nodes are randomly distributed in a 400m × 200m rectangular area. We assume that nodes are stationary in the simulations, except in the mobile scenario where 50 nodes are randomly chosen as mobile nodes and others keep stationary. The source and the sink nodes are chosen randomly in each simulation run. Constant packet arrival rate with $5p/s$, and varying packet arrival rate (the probability of packet arrival rate of each sensor node follows a Poisson distribution with average $\lambda = 5p/s$), are used in the simulations.

Castalia (Castalia, 2009; Pham et al., 2007) wireless sensor network simulator, which is built based on the OMNeT++ (OMNeT++, 2009; Varga, 2001) discrete event simulation platform, is used as the simulation environment.

Table 1 lists the detailed simulation parameters.

5.2 Comparison with *MMCC*

The average end-to-end delay to the sink node in different wireless channel conditions are shown in Fig. 4 and Fig. 5, respectively.

The simulation results show that when the wireless channel is in a perfect condition, i.e., no error occurs in packet transmissions, *MRL-CC* and *MMCC* (distance based) have the similar performances on the end-to-end delay. However, when the error-prone wireless channel is used in the simulation, *MRL-CC* performs better than *MMCC*.

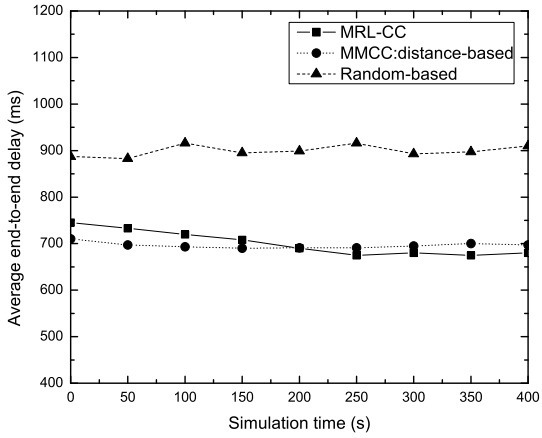

Figure 4: Average end-to-end delay to the sink node (link failure ratio = 0)

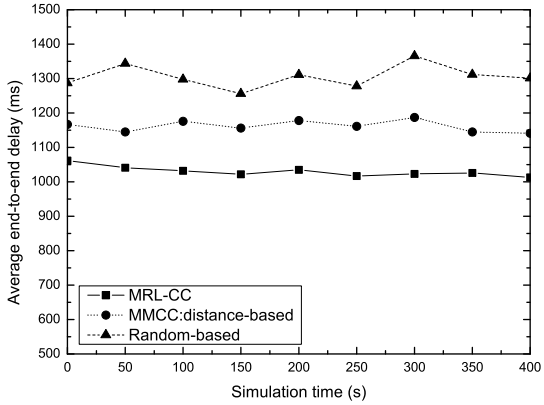

Figure 5: Average end-to-end delay to the sink node (link failure ratio = 0.2)

The reason is that in the perfect wireless channel condition, distance based protocols such as *MMCC*, are always effective in forwarding-node-election, i.e., node which is closest to the sink is often the best forwarding candidate. However, in realistic wireless channel conditions, i.e., time-varying wireless links with packet transmission errors and collisions, it is not always true that nodes closer to the sink always have higher link qualities and should be elected as the forwarding nodes, and thus the use of distance based criterion in forwarding-node-election is not always effective. For *MRL-CC*, by utilizing the policy learned from experiences and rewards, nodes with higher link qualities are more likely to be elected as the forwarding nodes in the groups of CNs, and thus, the forwarding node assignment in *MRL-CC* is more adaptive than that in *MMCC*.

Fig. 6 and Fig. 7 illustrate the average packet delivery ratio from the source node to the sink node with constant and varying packet arrival rates, respectively.

We can observe that with constant packet arrival rate, *MRL-CC* and *MMCC* have similar performances on the packet delivery ratio. However, when the packet arrival rate varies, *MRL-CC* outperforms *MMCC*. The simulation results also verify that the forwarding-node-election

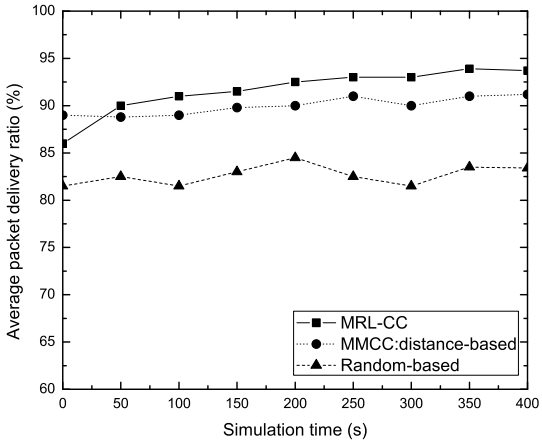

Figure 6: Average packet delivery ratio to the sink node with constant packet arrival rate

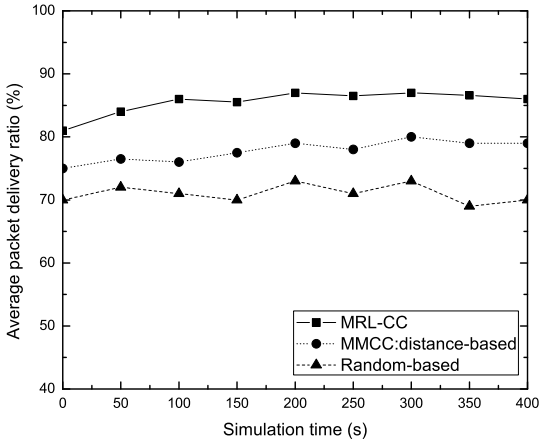

Figure 7: Average packet delivery ratio to the sink node with varying packet arrival rate

scheme in *MRL-CC* is more adaptive and flexible than *MMCC* in dynamic network conditions, by taking nodes' varying processing and queuing delays into account.

The impact of network background traffic load on the average end-to-end delay, and the impact of node mobility on the average packet delivery ratio are shown in Fig. 8 and Fig. 9, respectively.

The simulation results show that *MRL-CC* performs better than *MMCC*, especially when the network background traffic becomes heavy and/or the network mobility level increases. It is because that *MMCC* selects data forwarding nodes either by a random value based criterion or a distance based criterion, and thus lacks of the flexibility of handling network dynamics. In comparison, *MRL-CC* is much more intelligent in data forwarding-node-election since it experiences the dynamic networks and learns the optimal cooperative policy through experiences and rewards. The flexible nature of computer machine learning allows it to adapt to the dynamic environment well, especially in networks with heavy traffic in highly dynamic scenarios.

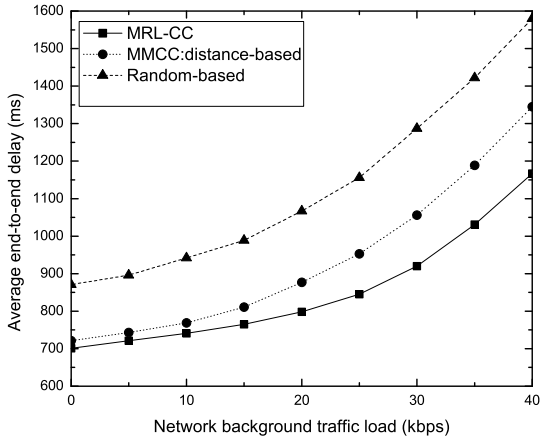

Figure 8: The impact of network background traffic load on average end-to-end delay

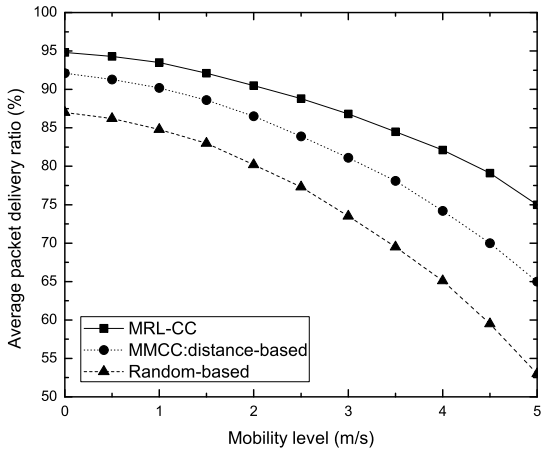

Figure 9: The impact of node mobility on average packet delivery ratio

We have also observed that for all the measured QoS metrics in the simulations, *MRL-CC* performs better after the simulation runs for a certain amount of time (i.e., around 50s). This is mainly because that there is a learning period for any kinds of learning based protocols, in which agents (sensor nodes in this paper) explore all the possible decisions (cooperative policy) and estimate the decision qualities, so that the network performance are improved over time. When the learning procedure is finished, nodes can take the optimal actions according to the network state information.

6 Conclusions and Future Research

In this paper, we have investigated the use of cooperative communications for QoS provisioning in resource-constrained wireless sensor networks, and proposed *MRL-CC*, a multi-agent reinforcement learning based multi-hop mesh cooperative communication mechanism for wireless sensor networks. Simulation results show that *MRL-CC* performs well in terms of a number of QoS metrics and fits well in large scale networks and highly dynamic environments.

In future research, service differentiation and system fairness will be considered in the cooperative mechanism design. Moreover, we will examine the use of adaptive cooperative coding scheme (e.g., channel coding) and employ power allocation scheme to improve the network performance and prolong the network lifetime.

Acknowledgment

This research is in the context of the EU project IST-33826 CREDO: Modeling and analysis of evolutionary structures for distributed services (http://www.cwi.nl/projects/credo/). This work was supported in part by the Canadian Natural Sciences and Engineering Research Council under grant STPGP 322208-05. Professor Yang Xiao's work was partially supported by the US National Science Foundation (NSF) under the Grants No. CNS-0716211 and CCF-0829827.

REFERENCES

Chen, M., Liang, X., Leung, V. C.M. and Balasingham I. (2009) 'Multi-Hop Mesh Cooperative Structure based Data Dissemination for Wireless Sensor Networks', *Proceedings of the 11th International Conference on Advanced Communication Technology (ICACT'09)*, February, pp.102–106.

Zhang, Q. and Zhang, Y. (2008) 'Cross-Layer Design for QoS Support in Multihop Wireless Networks', *Proceedings of the IEEE*, January, Vol. 96, No. 1, pp.64–76.

Hanzo, L. and Tafazolli, R. (2007) 'A Survey of QoS Routing Solution for Mobile AD Hoc Networks', *IEEE Communications Surveys & Tutorials*, July, Vol. 9, No. 2, pp.50–70.

Zhang, B. and Mouftah, H.T. (2005) 'QoS routing for wireless ad hoc networks: problems, algorithms, and protocols', *IEEE Communications Magazine*, October, Vol. 43, No. 10, pp.110–117.

Al-Karaki, J. N. and Kamal, A. E. (2004) 'Routing techniques in wireless sensor networks: a survey', *IEEE Wireless Communications*, December, Vol. 11, No. 6, pp.6–28.

Nosratinia, A., Hunter, T.E. and Hedayat, A. (2004) 'Cooperative communication in wireless networks', *IEEE Communications Magazine*, October, Vol. 42, No. 10, pp.74–80.

Hong, Y.W., Huang, W.J., Chiu, F.H. and Kuo, C.C. J. (2007) 'Cooperative Communications in Resource-Constrained Wireless Networks', *IEEE Signal Processing Magazine*, May, Vol. 24, No. 10, pp.47–57.

Ibrahim, A.S., Sadek, A.K., Su, W. and Liu, K.J. R. (2008) 'Cooperative communications with relay-selection: when to cooperate and whom to cooperate with?', *IEEE Transactions on Wireless Communications*, July, Vol. 7, No. 7, pp.2814–2827.

Sadek, A.K., Wei, Y. and Liu, K.J. R. (2006) 'When Does Cooperation Have Better Performance in Sensor Networks?', *Proceedings of the 3rd IEEE Sensor and Ad Hoc Communications and Networks (SECON'06)*, September, pp.188–197.

Conti, M., Gregori, E. and Maselli, G. (2004) 'Cooperation issues in mobile ad hoc networks', *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops(ICDCSW'04)*, March, pp.803–808.

Lin, Y., Song, J.H. and Wong, V. W.S. (2009) 'Cooperative Protocols Design for Wireless Ad-Hoc Networks with Multi-hop Routing', *Mobile Networks and Applications*, April, Vol. 14, No. 2 pp.143–153.

Zhou, Z., Zhou S., Cui, J.H. and Cui, S. (2008) 'Energy-Efficient Cooperative Communication Based on Power Control and Selective Single-Relay in Wireless Sensor Networks', *IEEE Transactions on Wireless Communications*, August, Vol. 7, No. 8 pp.3066-3078.

Laneman, J. N., Tse, D. N.C. and Wornell G.W. (2004) 'Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behavior', *IEEE Transactions on Information Theory*, December, Vol. 50, No. 12, pp.3062–3080.

Hunter, T.E. and Nosratinia, A. (2002) 'Cooperation Diversity through Coding', *Proceedings of the IEEE International Symposium on Information Theory (ISIT'02)*, June, pp.220.

Sendonaris, A., Erkip, E. and Aazhang, B. (2003) 'User Cooperation Diversity-Part I: System Description', *IEEE Transactions on Communications*, November, Vol. 51, No. 11, pp.1927–1938.

Sendonaris, A., Erkip, E. and Aazhang, B. (2003) 'User Cooperation Diversity-Part II: Implementation Aspects and Performance Analysis', *IEEE Transactions on Communications*, November, Vol. 51, No. 11, pp.1939–1948.

Hunter, T.E. and Nosratinia, A. (2004) 'Distributed Protocols for User Cooperation in Multi-User Wireless Networks', *Proceedings of the 47th IEEE annual Global Telecommunications Conference (GLOBECOM'04)*, November, pp.3788–3792.

Liu, P., Tao, Z., Lin, Z., Erkip, E. and Panwar, S. (2006) 'Cooperative Wireless Communications: a Cross-Layer Approach', *IEEE Wireless Communications*, August, Vol. 13, No. 4, pp.84–92.

Chen, M., Kwon, T., Mao, S., Yuan, Y. and Leung, V. C.M. (2008) 'Reliable and Energy-Efficient Routing Protocol in Dense Wireless Sensor Networks', *International Journal on Sensor Networks*, August, Vol. 4, No. 1/2, pp.104–117.

Sadek, A. K., Su, W., and Liu, K.J. R. (2007) 'Multinode Cooperative Communications in Wireless Networks', *IEEE Tansactions on Signal Processing*, January, Vol. 55, No. 1, pp.341–355.

Kaelbling P.L., Littman, L.M. and Moore W.A. (1996) 'Reinforcement Learning: A Survey', *Journal of Artificial Intelligence Research*, May, Vol. 4, No. 12, pp.237–285.

Zhang, D. and Ma, H. (2007) 'A Q-Learning-based Decision Making Scheme for Application Reconfiguration in Sensor Networks', *Proceedings of the 11th International Conference on Computer Supported Cooperative Work in Design (CSCWD'07)*, April, pp.1122–1127.

Boyan, A. J. and Littman, M. L. (1993) 'Packet Routing in Dynamically Changing Networks: A Reinforcement Learning Approach', *Proceedings of the 7th Neural Information Processing Systems (NIPS'93)*, December, pp.671–678.

Yu, F.R., Wong, V. W.S. and Leung, V. C.M. (2008) 'A New QoS Provisioning Method for Adaptive Multimedia in Wireless Networks', *IEEE Transactions on Vehicular Technology*, May, Vol. 57, No. 3, pp.1899–1909.

Liu, Z. and Elhanany, I. (2006) 'RL-MAC: A QoS-Aware Reinforcement Learning based MAC Protocol for Wireless Sensor Networks', *Proceedings of the IEEE 2006 International Conference on Networking, Sensing and Control (ICNSC'06)*, April, pp.768–773.

Pandana, C. and Liu, K.J. R. (2005) 'Near-optimal reinforcement learning framework for energy-aware sensor communications', *IEEE Journal on Selected Areas in Communications*, April, Vol. 23, No. 4, pp.788–797.

Stone, P. and Veloso M. (2000) 'Multiagent Systems: A Survey from a Machine Learning Perspective', *Autonomous Robots*, June, Vol. 8, No. 3, pp.345–383.

Busoniu, L., Babuska, R. and De Schutter, B. (2006) 'Multi-Agent Reinforcement Learning: A Survey', *Proceedings of The 9th International Conference on Control, Automation, Robotics and Vision (ICARCV'06)*, December, pp.1–6.

Schneider, J., Wong, W., Moore, A. and Riedmiller, M. (1999) 'Distributed Value Functions', *Proceedings of the 16th International Conference on Machine Learning*, June, pp.371–378.

Tham, C.K. and Renaud, J.C. (2005) 'Multi-Agent Systems on Sensor Networks: A Distributed Reinforcement Learning Approach', *Proceedings of the 2005 International Conference on Intelligent Sensors, Sensor*

Networks and Information Processing Conference (ISSNIP'05), December, pp.423–429.

Sutton, R. S. and Barto, A. G. (1998) 'Reinforcement Learning: An Introduction', *MIT Press*.

Mohrehkesh, S., Fathy, M. and Yousefi, S. (2006) 'Prediction Based QoS Routing in MANETs', *Proceedings of the 8th International Conference on Distributed Computing and Networking (ICDCN'06)*, December, pp.46–51.

Shah, S.H. and Nahrstedt, K. (2002) 'Predictive location-based QoS routing in mobile ad hoc networks', *Proceedings of the 2002 IEEE International Conference on Communications (ICC'02)*, April, pp.1022–1027.

Castalia Wireless Sensor Network Simulator (2009) *http://castalia.npc.nicta.com.au*

Pham, H.N., Pediaditakis, D. and Boulis, A. (2007) 'From Simulation to Real Deployments in WSN and Back', *Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'07)*, June, pp.1–6.

OMNeT++ Discrete Event Network Simulator (2009), *http://www.omnetpp.org*

Varga, A. (2001) 'The OMNeT++ Discrete Event Simulation System', *Proceedings of the 15th European Simulation Multiconference (ESM'01)*, June, pp.319–324.