Engineering Applications of Artificial Intelligence 46 (2015) 23–32

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Application of reinforcement learning to medium access control for wireless sensor networks

Yi Chu *, Selahattin Kosunalp, Paul D. Mitchell, David Grace, Tim Clarke

Communications and Signal Processing Research Group, Department of Electronics, University of York, United Kingdom

ARTICLE INFO

Article history:
Received 6 February 2015
Received in revised form
26 June 2015
Accepted 7 August 2015
Available online 15 September 2015

Keywords:
Reinforcement learning
Wireless sensor networks
Medium access control

ABSTRACT

This paper presents a novel approach to medium access control for single-hop wireless sensor networks. The ALOHA-Q protocol applies Q-Learning to frame based ALOHA as an intelligent slot selection strategy capable of migrating from random access to perfect scheduling. Results show that ALOHA-Q significantly outperforms Slotted ALOHA in terms of energy-efficiency, delay and throughput. It achieves comparable performance to S-MAC and Z-MAC with much lower complexity and overheads. A Markov model is developed to estimate the convergence time of its simple learning process and to validate the simulation results.

1. Introduction

Wireless sensor networks (WSNs) represent a rapidly emerging technology with a wide range of industrial and military applications for environmental monitoring (Akyildiz et al., 2002). A WSN typically comprises a large number of inexpensive nodes, which are capable of sensing, processing and communicating data in a collaborative fashion over multi-hop links such that they are robust to topological changes and failure of a small proportion of the nodes in a network should not affect its operation. Sensor nodes are often battery-powered and when deployed in a distributed fashion in areas which are difficult to access, recharging or replacing the batteries is very difficult. Therefore, energy efficiency is critically important in extending the operational life of such networks and is usually a priority in the design of communication protocols.

In a WSN, nodes share the same medium with their neighbours within a certain range. The Medium access control (MAC) protocol plays an important role in maximising throughput and energy efficiency. A well designed protocol should ensure the successful delivery of the data whilst minimising unnecessary energy consumption arising from collisions and associated retransmissions, control packet overheads, idle listening and overhearing (Demirkol et al., 2006). Many MAC protocols for WSNs have been proposed that significantly improve energy efficiency and throughput performance. However, these improvements incur higher overheads and exhibit ever increasing complexity. Performance evaluation through simulation is common but the practicality of recently proposed schemes is questionable. Given the longer term vision of huge numbers of devices embedded into machines everywhere, much simpler protocols are needed that can nonetheless provide energy-efficient communication, good throughput and acceptable delay.

ZigBee (The ZigBee Alliance, 2015) is a commercial specification for small and low data rate personal area networks, and it is designed based on IEEE 802.15.4 standard (IEEE Standard for Local and Metropolitan area networks, 2012). The MAC layer channel access in IEEE 802.15.4 mainly uses Carrier Sense Multiple Access (CSMA) and ALOHA for random access, and slot scheduling for contention-free access. CSMA based schemes avoid packet collisions by using channel sensing and hand shake procedures (send requests and responses before actual data transmission), however these increase system overheads as well as energy consumption, and still cannot completely eliminate the hidden terminal problem. Slot scheduling ensures success of transmissions but it brings more complexity and computation to the system, while having more overheads than random access schemes.

ALOHA based schemes do benefit from simplicity, low computation and overheads, but they suffer from poor throughput and low energy efficiency through collisions which arise from their blind transmission strategy. Intelligent selection of transmission times offers the potential to significantly improve channel performance and energy efficiency. In this paper, reinforcement learning is applied to ALOHA for this purpose. It enables nodes to develop

* Corresponding author at: Communications and Signal Processing Research Group, Department of Electronics, University of York, United Kingdom.
E-mail addresses: yi.chu@york.ac.uk (Y. Chu), sk772@york.ac.uk (S. Kosunalp), paul.mitchell@york.ac.uk (P.D. Mitchell), david.race@york.ac.uk (D. Grace), tim.clarke@york.ac.uk (T. Clarke).

http://dx.doi.org/10.1016/j.engappai.2015.08.004
0952-1976/

a more effective transmission strategy based on their prior experience. Compared with CSMA based schemes, channel sensing and hand shake procedures are not required, and the only overheads are Acknowledgement (ACK) packets which are commonly implemented in MAC protocols. Nodes can sleep when they have no data to transmit to extend the network lifetime. Compared with contention-free schemes, the learning schemes can achieve the same collision-free channel access without any scheduling information exchange. The only cost is the time for learning algorithm to converge to an optimal steady state, which this paper provides a detailed investigation in Sections 4–6.

Reinforcement learning enables entities to learn effective strategies through trial-and-error interactions in a dynamic environment, with future actions determined by prior experience (Kaelbling et al., 1996). This established artificial intelligence strategy has recently been applied to communication problems and MAC layer protocols. In (Li et al., 2010) and (Tang et al., 2011), several reinforcement learning based MAC protocols are proposed. Although they are not designed for WSNs, a similar strategy can be brought to the design of MAC protocols for WSNs. In this paper, one reinforcement learning method, Q-Learning, is applied to an adaptation of slotted ALOHA as an intelligent slot selection strategy to avoid collisions and retransmissions in single-hop networks. A similar approach has also been evaluated for multi-hop WSNs in (Chu et al., 2012a,2012b). The primary purpose of this paper, here, is to provide a thorough analysis of the application of reinforcement learning to the medium access control problem. A new novel protocol is introduced for single hop networks (ALOHA-Q), which achieves perfect scheduling in steady states with the only overheads being ACK packets. A Markov model of the ALOHA-Q learning scheme is developed to analyse its convergence behaviour and to validate a simulation model. The steady-state performance of ALOHA-Q is then evaluated and compared with S-MAC and Z-MAC through simulation.

The rest of the paper is organised as follows. Section 2 presents related work on MAC protocols for WSNs. Section 3 introduces the principles behind combining Slotted ALOHA and Q-Learning and introduces the ALOHA-Q scheme. Section 4 discusses the tradeoff and fairness issues of the ALOHA-Q scheme. Section 5 provides a Markov model to estimate the convergence time of the learning process. Section 6 presents simulation results, and Section 7 concludes the paper.

2. Related work

A large number of protocols have been proposed for WSNs in an effort to improve energy-efficiency and provide good quality of service in terms of throughput and delay. The majority are contention-based and derived from CSMA since this is a natural approach for distributed multi-hop networks and is employed in the IEEE 802.15.4 standard. Variants of CSMA are also well established for single-hop networks due to standardisation of IEEE 802.11. Some of the more pertinent scheme are summarised in this section.

To improve energy efficiency and the lifetime of a sensor network, S-MAC (Wei et al., 2004) is a well known contention-based scheme which periodically switches nodes between sleep and listening modes. During the listening period, nodes exchange synchronisation and scheduling information and during the sleep period, nodes can either initiate data transmission or remain in sleep mode. The data transmission process is the same as that employed in IEEE 802.11 with Request to Send (RTS) and Clear to Send (CTS) packets exchanged prior to data packet transmission. S-MAC effectively increases network lifetime through the duty cycle, but this serves to increase delay and reduce throughput in densely deployed networks.

Based on S-MAC, DS-MAC (Lin et al., 2004) applies a dynamic duty cycle to achieve a better balance between latency and energy consumption at different traffic levels. The packets in DS-MAC have a field to record their one-hop latency which is used to estimate the current traffic level. Besides the average latency, nodes also keep track of energy consumption on a per packet basis and use this as an indicator of energy efficiency. If the latency of a node rises above a certain level, it will change its duty cycle so that it remains awake for the same duration but reduces its sleep time. It offers better channel performance than S-MAC at high traffic levels, but similar to S-MAC, latency is still a problem.

Demand Wakeup MAC (DW-MAC) (Sun et al., 2008) is another extension of S-MAC which introduces a new low-overhead scheduling algorithm to wake up nodes when they need to transmit or receive in order to increase the effective channel capacity and adapt the network to a variety of traffic loads.

Instead of RTS and CTS packets, DW-MAC transmits scheduling frames at different time points to indicate the length of the data transmissions. It integrates scheduling and access control to achieve low latency and high power efficiency. However, this feature also makes DW-MAC very sensitive to synchronisation errors.

Two intelligent CSMA based schemes are proposed in (Barcelo et al., 2011, Fang et al., 2013). By applying slots and repeating frames to CSMA, the scheme described in (Barcelo et al., 2011) starts with random access, and users continually select slots with successful transmissions until two consecutive collisions. Similarly, the scheme presented in (Fang et al., 2013) is initialised to random access, and the user keeps using the slot with correct packet reception until a collision, then this slots have a decreasing probability of reselection (other slots have equal probability of selection). The embedded intelligence further avoids collisions and improves the channel performance.

By combining the advantages of TDMA and CSMA, Z-MAC (Rhee et al., 2008) achieves better channel utilisation and lower latency. Nodes broadcast ping packets for neighbour discovery after initialisation and select unique time slots according to DRAND (Rhee et al., 2006) to avoid the hidden terminal problem. Nodes have two modes, low contention level (LCL) and high contention level (HCL). At the LCL, any node can contend to transmit in any slot but at the HCL, only the owners of the slot and their one-hop neighbours are allowed to contend for the channel. The owner of the slot always has the highest priority but if it does not have any data to send, other nodes can steal the slot. The priority of the slot owner is applied by using a certain number of contention slots (which are much shorter than data slots) while transmitting packets. The slots owners can always transmit before the non-owners, so that the non-owners can overhear the transmission and avoid collisions. The selection of the length of each contention slot is based on the precision of the synchronisation. Z-MAC performs well at different traffic levels and has robustness to synchronisation errors with the cost of overheads.

Quorum-based MAC (Q-MAC) (Chao and Lee, 2010) adapts sleep schedules to improve energy efficiency and delay performance. The quorum-based wake up scheme determines the wake-up frequency of a node by considering current traffic load. Nodes with a light load switch to a sleep mode more frequently. To reduce the delay caused by long sleep periods, Q-MAC applies a list of next-hop nodes to increase the transmission opportunities for relaying packets.

Spatial Correlation-based Collaborative MAC (CC-MAC) (Vuran and Akyildiz, 2006) improves energy efficiency by making the data transmission process more selective according to the event-based characteristic of WSNs. When an event occurs, nodes within a

certain spatial range may detect the same event and generate data with high similarity. Instead of transmitting all the data, CC-MAC allows a small number of nodes to transmit representative data to avoid redundant data from the neighbour nodes.

In (Chen et al., 2013, 2014) two hybrid cross-layer schemes based on Low-Energy Adaptive Clustering Hierarchy (LEACH) (Schurgers and Srivastava, 2001) are proposed. Authors of (Chen et al., 2013) have presented a Clustering Algorithm based on Social Insect Colonies (CASIC) with Particle Swarm Optimisation (PSO) scheme (CASIC-PSO) to form a WSN. CASIC-PSO uses PSO algorithm to select cluster heads by considering the node's remaining energy and the distance to base station (thereby improve network life time), the cluster heads then form their own cluster member nodes and schedule sensing data transmissions. To extend the life time of LEACH, the scheme proposed in (Chen et al., 2014) has combined intersection-based coverage algorithm (IBCA) (Wang et al., 2005) and power-efficient gathering in sensor information system (PEGASIS) (Lindsey and Raghavendra, 2002) and reduced overall energy consumption. In the proposed scheme, IBCA is applied to locate redundant nodes and switch them to sleep state to conserve energy, while ensuring sufficient number of active nodes to complete data transmission in the network.

Authors of Shafiullah et al. (2013) have proposed another cross-layer scheme E-BMA for WSNs, designed for railway monitoring applications. Several sensors are deployed in each railway wagon to collect the health data and send it to the base station located in the middle of the train. LEACH is applied as its network layer architecture, benefiting from the specific node deployment: nodes in each wagon form a cluster and the cluster head is selected from them. The MAC layer of E-BMA is designed based on the bit-map-assist (Li, 2004) protocol (BMA). Similar as SMAC, BMA divides time into contention and transmission phases. During contention phases nodes exchange scheduling information, and during transmission phases nodes have collision-free data transmission. E-BMA further improves energy efficiency by piggybacking information of future data packets instead of scheduling packets exchange during contention phase, specifically for the high traffic load feature while the train is operating.

Schedule-based schemes represent an alternative approach, although schedules are difficult to arrange in a distributed way. A good example is TRAMA (Rajendran et al., 2006) which incorporates mechanisms for neighbour discovery, schedule exchange and determination of node priority in successive slots. The resulting schedules alleviate the traditional energy waste mechanisms (idle listening, overhearing, collisions), but the resulting overheads and complexity are not amenable to basic sensor nodes. It is clear that the most promising techniques will find application in certain types of wireless sensor network, but much simpler solutions need to be developed to support more basic devices.

Many MAC protocols for WSNs employ certain CSMA features (e.g. S-MAC, Z-MAC), notably channel sensing before transmission. Although sensing is able to avoid the majority of collisions arising from transmitters within a one hop distance, transmissions from hidden nodes cannot be detected through channel sensing. Additional techniques are common in order to avoid collisions with hidden nodes such as the exchange of two-hop scheduling information in Z-MAC which requires additional overheads and increases energy consumption. ALOHA-Q does not require channel sensing and does not require idle listening (for transmitter nodes, in single-hop networks) but achieves perfect scheduling after convergence, giving it a distinct advantage over CSMA based schemes. Slot selections of the nodes are determined by their historical transmissions (collision or success), and the network approaches the steady state by exploration and exploitation. The applicability of ALOHA-Q to multi-hop networks has been evaluated in (Chu et al., 2012a, 2012b). In this paper a detailed study and evaluation of the convergence of ALOHA-Q is presented for single-hop networks.

3. ALOHA-Q protocol design

Frame based ALOHA is considered for the application of reinforcement learning and stateless Q-Learning (Suttong and Barto, 1998) is used to capture the learning experience. A frame comprises a fixed number of slots as a system wide parameter. Each node has individual Q values for every slot in the frame which are updated by transmission outcomes (success or failure). The largest value determines which slot is selected for the net transmission. Nodes only wake up when they need to transmit in the slots with the highest Q values and to receive the associated acknowledgements (ACKs). Idle listening is not used in ALOHA-Q (except the sink node). Time references for synchronisation are embedded in the ACK packets sent from the sink node, so that the transmitting nodes are able to maintain synchronisation with the sink node as long as they transmit data packets to the sink and receive ACK packets.

Nodes in the network all start with random access (all Q values are 0), learn through transmission, and finally reach their optimal transmission strategy in which nodes have found unique slots for contention free transmission. Q values are denoted$Q(i,k)$, indicating the preference of node i to transmit a packet in slot k. The previous Q values and current reward contribute to the Q value update according to (1) after every data packet transmission

$$Q_{t+1}(i,k) = Q_t(i,k) + \alpha(r - Q_t(i,k)) \tag{1}$$

where α is the learning rate and r is the current reward. One of the Q values of a node is updated after each data packet transmission.

If a transmission succeeds, a reward of $+1$ is returned otherwise the reward is -1. Slots with higher Q values are preferred but if multiple slots have the same higher Q value, one (or more) will be randomly selected from the set. Fig. 1 shows an example of the frame structure and Q-Learning algorithm. It shows successful/collided packet transmissions and Q value updates of one node in a WSN. In this example each frame contains 3 slots and $\alpha = 0.1$.

4. Discussions of tradeoff and fairness

4.1. Learning process and steady state

The learning process results in a node having different Q values for each slot. According to (1), a negative reward will have a greater impact on the current Q value, when the Q value is positive and vice versa. A slot which regularly receives a negative reward is therefore unlikely to be the preferred slot. This results in the node seeking a slot which will continually return a positive reward. Through this learning process the network tends towards to an optimal steady state condition where all nodes have unique slots. It behaves like a schedule-based network but without the need for scheduling information exchange or determination of node priorities in each slot, which is critical in WSNs without centralised control.

To reach the optimal steady state, the parameters need to be appropriately set. The learning rate $\alpha \in [0,1]$ controls the speed at which a Q value converges to the current reward. The higher α, the

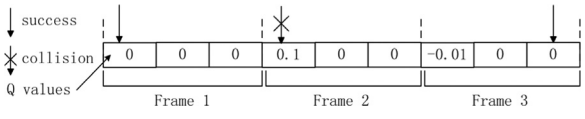

Fig. 1. Example of Q values and repeating frames.

faster the Q value converges to reward r. α is usually set to a small value so that the steady state will exhibit some robustness to small changes in channel conditions (e.g. infrequent collisions). Another important parameter is the frame size N. Nodes will be unable to find unique slots if there are insufficient slots in a frame. N should be just large enough to ensure this. Overestimating N will introduce additional latency and reduce the maximum achievable throughput. In a single hop network, the number of nodes deployed will often be known. In a multi-hop network, an appropriate value for N can be determined by estimating the number of nodes within communication range, based on the deployment density.

The learning algorithm is able to adapt to changes in the network topology should nodes die or additional nodes be deployed. When a node dies, its preferred slots automatically become available for others to use. New nodes will learn from scratch, but reach an optimal steady state much more quickly than if the whole network is initialised, because they are learning from a steady environment and can more easily find unique slots. The learning schemes can achieve perfect scheduling in the steady state following a period of convergence. The time taken to converge is important to the network and it varies due to the random slot selection process when multiple slots have the same maximum Q value. In Section 5 an analytical model is developed to estimate the convergence time of a simple learning process.

4.2. Fairness on quality of service

When the network converges to the steady state, nodes achieve perfect scheduling and experience an equal Quality of Service (QoS). However, during the learning process, nodes usually find their preferred slots sequentially, which means that some nodes obtain better QoS earlier than others, resulting in an imbalance in fairness between the nodes. Such fairness issues exist in the majority of the learning based MAC schemes. For example in (Barcelo et al., 2011, Fang et al., 2013), nodes obtain their slots after different periods of time, and nodes have different probabilities of success before the network converges, which makes the network convergence time an important measurement.

Compared with the learning schemes proposed in (Barcelo et al., 2011, Fang et al., 2013) which have an equal probability of selecting other slots when a collision occurs in one slot, Q-Learning has certain advantages and benefits from recording the transmission history. When only a minority of nodes has not converged, they have experience with regard to slot choice based on their transmission history, and the probability of them finding their preferred slots increases as time passes. Consider, for example, a network with 100 nodes where 99 nodes have found their preferred slots and one node is searching for its own unique slot. Assuming the worst case that it has no prior transmission experience (all Q values are zero), it needs a maximum of 100 transmissions (99 collisions and one success) to find its preferred slot and the probabilities of locating its preferred slot during each transmission are 1/100, 1/99, 1/98 etc. until the 100th transmission is reached and the probability becomes 1. In later sections we present a Markov model to estimate the convergence time and present simulations results of convergence time with different learning rates. Note that the convergence time is that measured when all nodes in the network find their preferred slots (not the average convergence across all nodes).

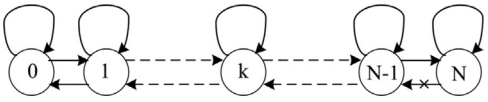

Fig. 2. Markov model.

5. Convergence time of ALOHA-Q

5.1. Markov model

We consider a single-hop network with N nodes and saturated traffic conditions (nodes always have packets to transmit). The learning rate (α) is set to 1 and the Q values are all initialised to -1, so that they can either be $+1$ or -1. The frame size is set equal to the number of nodes N and each node is allowed to transmit one packet per frame. Fig. 2 depicts the Markov process which describes this scheme. State k represents the current number of steady-state nodes (nodes which have a Q value of $+1$ for a particular slot) in the network. State transitions take place in every slot, and the process can only move forwards or backwards one state, or stay in the same state after each slot. A node which has reached steady-state is referred to as a *steady* node whereas a node still learning is referred to as a *hopping* node. When the process reaches state N, all nodes have found their unique slots and the system is deemed to have converged.

Let $p_{i,j}$denote the state transition probability from state i, $i=0, 1, 2\cdots N$ to state j, $j=0, 1, 2\cdots N$. For the model in Fig. 2, the relevant transition probabilities are: $p_{k,k}$, $p_{k,k-1}$ and $p_{k,k+1}$, $k=0, 1, 2\cdots N$, which arise from the following situations:

- $p_{k,k}$: success of a *steady* node in an *occupied* slot; collision of two or more *hopping* nodes in an *unoccupied* slot; this slot is empty.
- $p_{k,k-1}$: collision of a *steady* node in an *occupied* slot.
- $p_{k,k+1}$: success of a *hopping* node in an *unoccupied* slot.

Where an *occupied* slot represents a slot for which only one node has a $+1$ Q value. An *unoccupied* slot is a slot for which all nodes have a -1 Q value. Based on the previous definition, a *hopping* node is a node which has a -1 Q value for all slots.

More specifically, to stay in the same state ($p_{k,k}$) either:

- The current slot is *occupied* and no *hopping* nodes select the current slot.
- The current slot is *unoccupied* and two or more *hopping* nodes transmit packets in it.
- The current slot is *unoccupied* and there are no transmissions in it.

To move down one state ($p_{k,k-1}$):

- The current slot is *occupied* and one or more *hopping* nodes transmit packets in it.

To move up one state ($p_{k,k+1}$):

- The current slot is unoccupied and only one hopping node transmits a packet in it.

For a given value of k and knowing N, the state transition probabilities can be obtained as

$$p_{k,k}=\frac{k}{N}\left(\frac{N-1}{N}\right)^{N-k}+\frac{N-k}{N}\left(1-\frac{N-k}{N}\left(\frac{N-1}{N}\right)^{N-K-1}\right) \tag{2}$$

$$p_{k,k-1}=\frac{k}{N}\left(1-\left(\frac{N-1}{N}\right)^{N-k}\right) \tag{3}$$

$$p_{k,k+1}=\left(\frac{N-k}{N}\right)^{2}\left(\frac{N-1}{N}\right)^{N-k-1} \tag{4}$$

Convergence time estimation

The convergence time for this scheme is the accumulated time the system spends in all states before reaching state N, because once the process reaches state N it will never move back to the previous states. Consider $\boldsymbol{P}$ as the state transition probability matrix which has the elements:$p_{i,j}$, $i,\ j=0,\ 1,\ 2\cdots N$. In this model, $\boldsymbol{P}$ is a sparse matrix. Defining $\boldsymbol{P}^2=\boldsymbol{P}\boldsymbol{P}^{\boldsymbol{T}}$, the matrix element

$$p_{i,j}^2=\sum_{m=0}^{N}(p_{i,m}p_{m,j}) \tag{5}$$

represents the probability that the process visits state j via state m (where m represents any state) starting in state i.

The elements in $\boldsymbol{P}^3$ are

$$p_{i,j}^3=\sum_{m=0}^{N}(p_{i,m}^2p_{m,j}) \tag{6}$$

denote the probabilities that the process visits state j via all possible states after two transitions from state $i\ \forall\ i,j$, via any two transition states.

So, from (5) and (6), $\boldsymbol{P}^n$ is the matrix of visit probabilities between any state and all others via n arbitrary transitions. The element $p_{i,j}^n$ is the probability that the process visits state j after n transitions, starting in state i. Alternatively, $p_{i,j}^n$ can be described as the expected number of visits to state j at the nth state transition starting from state i, or the expected number of slots the process stays in state j at the nth state transition starting at state i. As with $\boldsymbol{P}$, each row sum of $\boldsymbol{P}^n$ will always equal unity.

Convergence of the scheme (unity probability of reaching state N) can be proved if

$$\lim_{n\to\infty}p_{i,N}^n=1,\ i=0,\ 1,\ 2\cdots,N \tag{7}$$

Proof. We base our proof on the relationship between $\boldsymbol{P}$ and its Jordan normal form (Finkbeiner 1978) $\boldsymbol{J}$, and invoking the associated similarity transform property

$$\boldsymbol{B}^{-1}\boldsymbol{P}\boldsymbol{B}=\boldsymbol{J}\Leftrightarrow\boldsymbol{P}^{\boldsymbol{n}}=\boldsymbol{B}\boldsymbol{J}^{\boldsymbol{n}}\boldsymbol{B}^{-1} \tag{8}$$

to obtain the Jordan normal form $\boldsymbol{J}$, we need to calculate the eigenvalues of $\boldsymbol{P}$ by solving its characteristic equation

$$det(\lambda\boldsymbol{I}-\boldsymbol{P})=0 \tag{9}$$

where $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{P}$ is an $(N+1)$ square matrix which has only one non-zero element in the last row.

Using conventional matrix indexing

$$\boldsymbol{P}(N+1,N+1)=p_{N,N}=1 \tag{10}$$

we expand (9) by using the last row according to the Laplace Expansion (Poole, 2005)

$$det(\lambda\boldsymbol{I}-\boldsymbol{P})=(-1)^{N+1+N+1}(\lambda-1)M_{N+1,N+1} \tag{11}$$

where $M_{N+1,N+1}$ is the determinant of the $N+1,N+1$ minor matrix of $\boldsymbol{P}$, which is the $N\times N$ matrix resulting from removing the last column and row of $\boldsymbol{P}$.

Clearly $\lambda=1$ is one eigenvalue of $\boldsymbol{P}$. To determine the range of the remaining eigenvalues, we employ the Gershgorin Circle Theorem (Varga, 2004). Begin by calculating absolute row sums excluding non-diagonal elements over $\boldsymbol{P}$

$$R_i=\sum_{j\neq i}|p_{i,j}|,\ i=0\cdots N \tag{12}$$

Then a set of Gershgorin discs can be drawn, centred on p_{ii} with radius R_i in the complex domain as $D(p_{i,i},R_i)$. The Gershgorin Circle Theorem states that the eigenvalues of $\boldsymbol{P}$ lie within the relevant Gershgorin disc. Except for $p_{N,N}$, the eigenvalues associated with all other diagonal elements have an absolute value less than unity, because each row sum of $\boldsymbol{P}$ equals unity.

Suppose $\boldsymbol{J}$ has m Jordan blocks and $\boldsymbol{J}$ can be represented by J_i, $i=1,2\cdots m$ where

$$J_i=\begin{bmatrix}\lambda_i & 1 & 0 & \cdots & & 0\\ & \lambda_i & 1 & 0 & \cdots & 0\\ & & \lambda_i & \ddots & \ddots & \vdots\\ & & & \ddots & & 0\\ & & & & \lambda_i & 1\\ 0 & & & & & \lambda_i\end{bmatrix}_{r_i\times r_i} \tag{13}$$

where r_i is the multiplicity of eigenvalue λ_i.

We know that $J_m=1$. The matrix $\boldsymbol{J}^n$ can be calculated by diagonally aggregating the individual Jordan Blocks J_i^n, $i=1,2\ldots m$, each of which can be obtained from

$$J_i^n=\begin{bmatrix}\lambda_i^n & \binom{n}{1}\lambda_i^{n-1} & \cdots & \binom{n}{r_i-1}\lambda_i^{n-r_i+1}\\ & \lambda_i^n & \ddots & \vdots\\ & & & \binom{n}{1}\lambda_i^{n-1}\\ 0 & & & \lambda_i^n\end{bmatrix}_{r_i\times r_i} \tag{14}$$

From the calculated bound on the eigenvalues above, with $n\to\infty$, we can see that $\boldsymbol{J}^n$ is an $(N+1)$ square matrix with the $(N+1,\ N+1)$ indexed element equal to unity and all others equal to zero.

We calculate the matrix $\boldsymbol{B}$ from $\boldsymbol{B}^{-1}\boldsymbol{P}\boldsymbol{B}=\boldsymbol{J}\to\boldsymbol{P}\boldsymbol{B}=\boldsymbol{Q}=\boldsymbol{B}\boldsymbol{J}$. Suppose the last column of $\boldsymbol{B}$ is $b_{i,N+1}$, $i=1\cdots N+1$, then calculate the last column of $\boldsymbol{Q}$ as $q_{j,\ N+1}$, $j=1\cdots N+1$. From $\boldsymbol{Q}=\boldsymbol{P}\boldsymbol{B}$ we get

$$q_{j,\ N+1}=\sum_{i=1}^{N+1}p_{j,i-1}b_{i,N+1} \tag{15}$$

From $\boldsymbol{Q}=\boldsymbol{B}\boldsymbol{J}$ we can get

$$q_{j,N+1}=b_{j,N+1} \tag{16}$$

Moreover, we have $\sum_{i-1}^{N+1}p_{j,i-1}=1,j=1\cdots N+1$. By substituting (16) into (15) the last column of $\boldsymbol{B}$ can be obtained as all 1 s

$$b_{i,N+1}=1,\ i=1\cdots N+1 \tag{17}$$

The last row of $\boldsymbol{B}$ are $b_{N+1,i}$, $i=1\cdots N+1$. Calculating the last row of $\boldsymbol{Q}$ as $q_{N+1,j},j=1\ldots N+1$. From $\boldsymbol{Q}=\boldsymbol{P}\boldsymbol{B}$ we can get

$$q_{N+1,j}=b_{N+1,j},\ j=1\cdots N+1 \tag{18}$$

From $\boldsymbol{Q}=\boldsymbol{B}\boldsymbol{J}$ we can get

$$q_{N+1,j}=\sum_{i=1}^{N+1}b_{N+1,i}J_{i,j},\ j=1\cdots N+1 \tag{19}$$

Substituting (18) into (19) we have $b_{N+1,1}=b_{N+1,1}J_{1,1}$, where$J_{1,1}$ is non-zero (see the proof in Appendix A), so $b_{N+1,1}=0$. Then we have $b_{N+1,2}=b_{N+1,2}J_{2,2}$, so $b_{n+1,2}=0\cdots$ and we can calculate the rest: $b_{N+1,i}=0$, $i=1\cdots N$. The matrix $\boldsymbol{B}$ has its last column all 1 and last row all 0 except for $b_{N+1,N+1}=1$

$$\boldsymbol{B}=\begin{bmatrix}\boldsymbol{B}_{N\times N} & & 1\\ & & \vdots\\ 0 & \cdots & 1\end{bmatrix}_{(N+1)\times(N+1)} \tag{20}$$

Then we calculate $\mathbf{B}^{-1}$ by using

$$\boldsymbol{B}^{-1}=\frac{\boldsymbol{B}^*}{det(\boldsymbol{B})} \tag{21}$$

Where $\boldsymbol{B}^*$ is the adjoint matrix of $\boldsymbol{B}$. By expanding the last row of $\boldsymbol{B}$, we can get $\det(\boldsymbol{B})=det(\boldsymbol{B}_{N\times N})$. $\boldsymbol{B}^*$ can be represented by the transpose of a matrix which has its element equal to $(-1)^{i+j}M_{i,j}$, $i,\ j=1,2,\cdots N+1$, the cofactors of $\boldsymbol{B}$. $M_{i,j}$ is the determinant of the i,j minor matrix of $\boldsymbol{B}$. The last row of $\boldsymbol{B}^*$ are:

Table 1
Simulation Parameters.

Parameters	Values
Channel bit rate	250 kbits/s
Data packet length (simulation)	1044 bits
Data packet length (practical)	935 bits
ACK packet length (simulation)	20 bits
ACK packet length (practical)	144 bits
Slot length	1100 bits

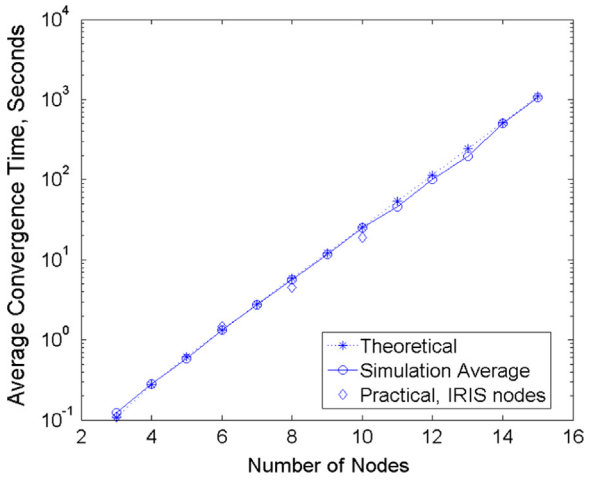

Fig. 3. Average simulation convergence time.

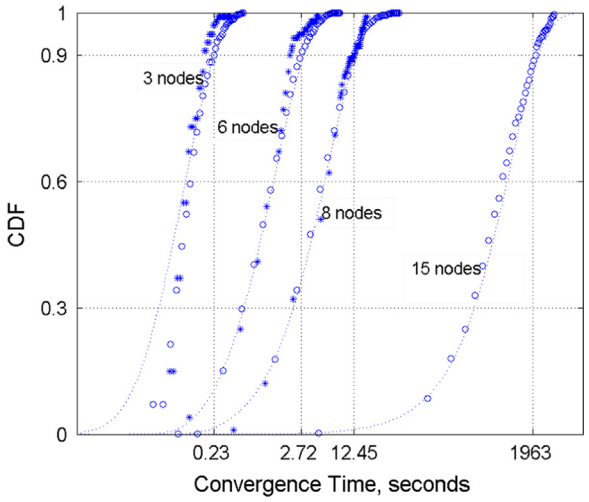

Fig. 4. CDF of convergence time measured through analysis and simulation.

$(-1)^{i+N+1}M_{i,N+1}$, $i=1\cdots N+1$, and we also know that $M_{i,N+1}=0$, $i=1\cdots N$, $M_{N+1,N+1}=\det(\boldsymbol{B}_{N\times N})=\det(\boldsymbol{B})$. So the last row of $\boldsymbol{B}^{-1}$ has zero entries except for the final one which is unity. Finally we calculate $\lim_{n\to\infty}\boldsymbol{P}^{\boldsymbol{n}}=\boldsymbol{B}\boldsymbol{J}^{\boldsymbol{n}}\boldsymbol{B}^{-1}$ using the previous results as

$$\lim_{n\to\infty}\boldsymbol{P}^{\boldsymbol{n}}=\begin{bmatrix}0 & \vdots & 1\\ & & 1\end{bmatrix}_{(N+1)\times(N+1)} \tag{22}$$

This proves the convergence of the learning scheme.

To calculate the time before convergence we need the expected time that this process stays in all states except state N, which is equal to the expected number of visits to these states across all nth transitions where $n=0,\ 1,\ 2\cdots,\infty$

$$E\{\text{convergence time}\}=\sum_{n=1}^{\infty}\sum_{j=0}^{N-1}{p_{ij}}^{n} \tag{23}$$

Which is the expected convergence time starting with state i.

The expected convergence time from the initialisation of the process can be obtained by calculating

$$E\{\text{convergence time}\}=\sum_{n=1}^{\infty}\sum_{j=0}^{N-1}{p_{0j}}^{n} \tag{24}$$

6. Performance evaluation

The convergence time obtained from the Markov model is compared to simulations from OPNET and practical experiments for the purpose of validation. The steady-state performance characteristics of ALOHA-Q are then evaluated in OPNET and compared with slotted ALOHA, S-MAC and Z-MAC to show the improvements achieved by applying learning and the overall capability of the scheme compared to well established schemes for WSNs. The throughput performance is also evaluated through practical experiments. For the performance evaluation, a single-hop network is considered with 200 nodes that all generate data packets and send them directly to the sink. The practical experiments use 20 nodes due to the limited number of devices. All nodes have the same mean packet inter-arrival time and the inter-arrival time is exponentially distributed. The IRIS (http://www.memsic.com/userfiles/files/datasheets/wsn/iris_datasheet.pdf, 2015) nodes are used in all practical experiments. The IRIS nodes are IEEE 802-15.4-comliant devices embedded with TinyOS (TinyOS: An operating system for wireless sensor networks, 2005), equipped with anAtmega1281 low-power microcontroller and an AT86RF230 radio transceiver (http://www.memsic.com/userfiles/files/datasheets/wsn/iris_datasheet.pdf, 2015). All nodes are synchronized with the sink. A list of simulation and practical experiments parameters can be found in Table 1. Due to the IRIS device limitations, the 20-bit ACK packet size cannot be achieved and a larger ACK packet size is applied in practical experiments. However the convergence results will not be affected by this difference because the slot length of both simulations and practical experiments are the same.

6.1. Convergence results

Experiments were undertaken which matched all the assumptions made in the analysis in Section 5. The other simulation parameters can be found in Table 1. Fig. 3 shows the average convergence time with different numbers of nodes in the network and each marker represents the average of 200 simulations. The convergence time has been determined for a maximum of 15 nodes in Fig. 3 since this is sufficient to demonstrate a close match between the analytical model and simulation, and the theoretical computation becomes prohibitive for larger numbers of nodes. Theoretically, the expected convergence time requires a large number of computations, as (24) shows with $n\to\infty$. When the convergence time is calculated for 15 nodes, $n=10^9$ so that $\boldsymbol{P}^{n}$ converges. This requires multiplication of two 16×16 matrices 10^9 times and it takes about 6 h to obtain the results. Moreover, the time required to calculate the convergence time increases exponentially with each additional node, so results cannot be easily provided for larger number of nodes and the trend of convergence time can be clearly observed in Fig. 3. The 95% confidence interval for the simulation results associated with 3 nodes is about $\pm3\%$, and the confidence interval increases with

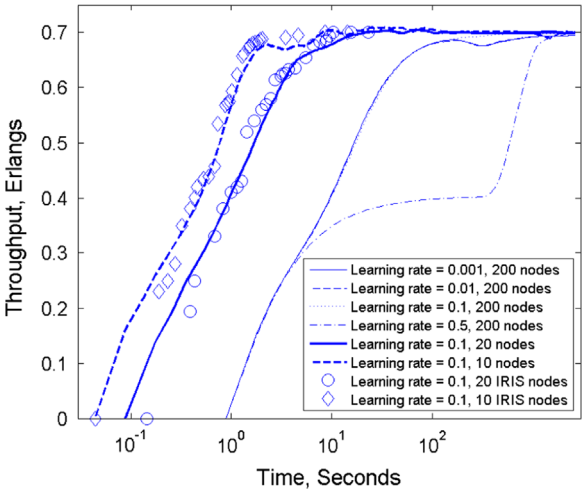

Fig. 5. Real-time throughput.

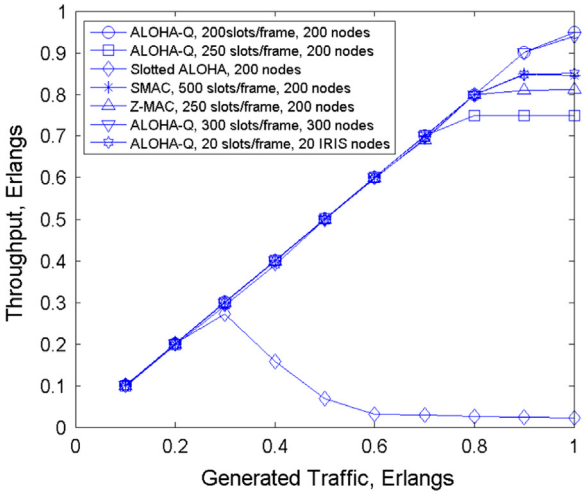

Fig. 6. Normalised throughput.

the number of nodes. The confidence interval for 15 nodes is approximately $\pm$12%. The confidence intervals are not shown on the figure as they are almost invisible given the logarithmic scale of convergence time on the y axis. Practical experiments are implemented for 3, 6, 8, and 10 nodes (due to the larger time costs) to evaluate simulation and theoretical results. Each marker represents the average of 100 trials and a closed match to both simulation and theoretical results can be observed in Fig. 3. Fig. 4 shows the Cumulative Density Function (CDF) of the convergence time of the networks with different number of nodes in the network. Each curve shows the results of 1000 simulations, 100 practical trials are implemented due to larger time cost. The dotted lines are produced by the Markov model, the circle markers are samples of the simulation results and the star markers are samples of the practical results. The results show that the distribution of the convergence time (both simulated and practical) matches the analysis. The discrepancy in the curve for 3 nodes is caused by the particularly small convergence times which are therefore sensitive to minor fluctuations. The simulation model and practical trials determine whether convergence has been achieved at the beginning of each frame, so the reported convergence time is restricted to multiples of the frame length and usually larger than the estimated convergence time obtained from the Markov model. This discrepancy does not exist for larger number of nodes.

It can be seen that the convergence times increase rapidly with an increasing number of nodes in the network, but the observed times need placing in context. In the majority of sensor network applications, the intention is to deploy a set of nodes for long periods of times (potentially years). In this respect, an initial convergence time of minutes (or even hours) is not a significant problem. It is also important to note that the protocol operates in a perfectly adequate fashion and offers performance benefits prior to convergence. To demonstrate this, Fig. 5 shows the real-time running throughput achieved from initialisation to each time step with different learning rates and network sizes. Note that the Markov model proposed in Section 5 uses a learning rate of 1.0. Different learning rates are applied in these simulations to examine the impact of this parameter on performance, and practical trials are implemented for evaluation. Each simulation curve represents the average of 100 simulations and each marker represents the average of 50 practical trials. Networks with 10, 20 and 200 nodes are simulated, the frame size is optimum and the generated traffic load is 0.7 Erlangs. Practical trials with 10 and 20 nodes are implemented to evaluate the simulation results. In simulations the running throughput is calculated at the beginning of each frame. It can be seen that a learning rate of 0.5 (200 nodes simulation) performs worse than the smaller learning rates, with suboptimal throughput for about 400 s before rising and reaching 0.7 Erlangs after about 1500 s. Learning rates equal to or less than 0.1 provide much more rapid convergence and almost identical performance characteristics. Throughput increases with time and reaches 0.7 Erlangs about 100 s after initialisation. The experienced throughput exceeds that obtainable with conventional slotted ALOHA after just 10 s. It is worth noting that Z-MAC requires a similar set up phase where nodes are unable to transmit data packets before it is completed (Rhee et al., 2008). According to the analysis in (Rhee et al., 2006), DRAND requires approximately 600 s to determine the owner of each slot for a network with 200 nodes. ALOHA is able to provide substantial data throughput, approaching its maximum of 0.7 Erlangs after 200 s. For smaller networks the ALOHA-Q converges much faster. For the network of 10 nodes, both simulated and practical results show that ALOHA-Q has converged at about 2 s, and for the network of 20 nodes, ALOHA-Q has converged within 10 s.

6.2. Steady state results

ALOHA-Q, Slotted ALOHA with exponential back-off (Kwak et al., 2005), S-MAC and Z-MAC have been simulated in OPNET to evaluate and compare their performance in terms of throughput, end-to-end delay and energy cost per bit throughput. All schemes are simulated for a network of 200 nodes, and ALOHA-Q (with optimal frame size) has one extra simulation for 300 nodes to show its performance with a network of larger node density. Trials of 20 IRIS nodes are implemented to demonstrate the throughput performance under practical network environments. A 10% duty cycle is used for S-MAC as commonly employed (Wei et al., 2004). Each contention slot of Z-MAC has a length of 0.5 bits, and we use the same contention window size employed in (Rhee et al., 2008) (8 contention slots for slot owners and extra 32 contention slots for non-owners) for the simulations of Z-MAC in this paper. We turn off the LCL, HCL and Explicit Contention Notification (ECN) because they make no difference in a single-hop network, in accordance with (Rhee et al., 2008), which assumes that all nodes in the network can transmit packets to and receive packets from each other, so that there are no hidden node problems and the channel sensing provides correct information. Note that this assumption (all nodes are within one-hop range) is not required for ALOHA-Q, and hidden nodes may exist even in a practical single-hop network, which will affect the channel performance. According to the experiments in

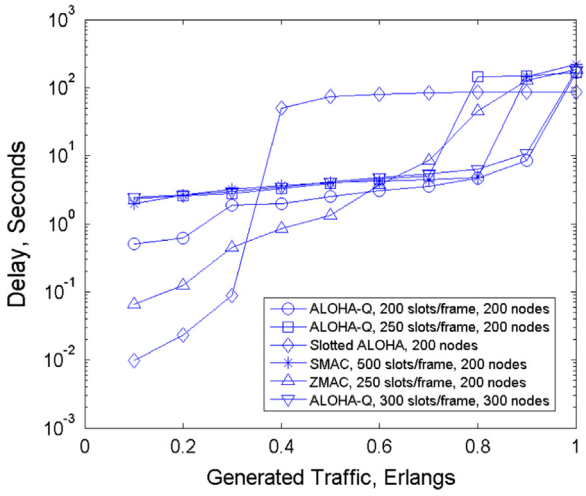

Fig. 7. End-to-end delay.

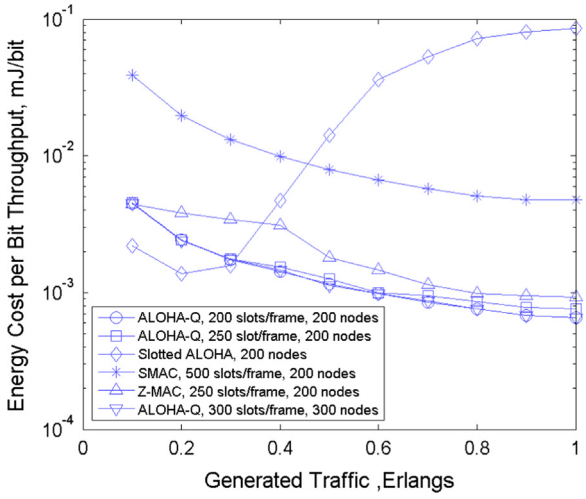

Fig. 8. Energy cost per bit throughput.

(Rhee et al., 2008), a 250 slots frame length is used for Z-MAC, which is typically larger than the number of nodes in the network.

Fig. 6 shows the normalised throughput performance. It can be seen that the throughput of ALOHA-Q with 200 slots per frame increases linearly with the generated traffic load and reaches a maximum close to 0.95 Erlangs, which is the theoretical maximum for the simulation parameters in Table 1 (1044 data bits/1100 bits slot length ≈ 0.95). ALOHA-Q with 300 slots per frame (300 nodes network) also achieves a similar throughput, indicating that its performance is not affected by larger number of nodes. Maximum throughput (935 data bits/1100 bits slot length ≈ 0.85) is also achieved under the 20 IRIS nodes deployment, note that the maximum throughput is lower than simulations because a larger ACK packet size is used. S-MAC has a slightly lower maximum throughput capability because of the fixed overheads present in the listening period (10% in the simulations). The performance of ALOHA-Q with 250 slots per frame is shown to demonstrate the impact of overestimating the number of nodes in the network. The throughput exhibits a similar linear increase but to a lower maximum, because the frame is oversized and some slots remain unused. Slotted ALOHA achieves a maximum throughput of 0.27 Erlangs, one third of that achievable with ALOHA-Q given the marginal increase in complexity, demonstrating the effectiveness of intelligent slot selection through learning. Z-MAC achieves a throughput of about 80% of the channel capacity, which is similar to the results observed in (Rhee et al., 2008). Its performance is limited by the contention windows (8 slots for owners and additional 32 slots for non-owners) used for channel sensing. When the network comprises a large number of nodes, the probability of two or more nodes applying channel sensing in the same slot is high under high traffic conditions, which causes more collisions and limits the throughput performance. Z-MAC has better performance than S-MAC for multi-hop networks, but under this scenario it is not optimum.

The average end-to-end delay experienced by packets is shown in Fig. 7 as a function of generated traffic load. When the traffic load is low, Slotted ALOHA offers the lowest delay because nodes are able to access the channel almost immediately after packets are generated and relatively few experience a collision. Z-MAC has good delay performance under low traffic conditions, because it is similar to Slotted ALOHA in allowing a node to transmit a packet straight after it has been generated when the channel is clear. All ALOHA-Q schemes offer less than 3 s delay before reaching the maximum traffic level. ALOHA-Q with 300 nodes has slightly higher delay than ALOHA-Q with 200 nodes (at low and medium traffic loads) because of the larger frame size. However it has lower delay than ALOHA-Q with 250 slots per frame at high traffic loads because the optimal frame size can achieve higher throughput. The slightly higher delay of S-MAC is caused by the long frame structure and sleep period.

Fig. 8 presents the energy efficiency results. The total energy cost is the sum of energy consumed by transmitting/receiving packets, idle listening and overhearing. According to the IRIS node datasheet (Kwak et al., 2005), 51 mW is considered as the transmitting power and 48mW is considered as the receiving/idle listening power. The sleep mode power consumption is negligible to other RF transceiver power consumption. The energy cost per bit throughput is calculated by dividing the total energy cost (mJ) by the amount of data received at the sink (in bits). Slotted ALOHA has the best energy efficiency at low traffic loads because of the low probability of collision, but energy costs rises rapidly with increasing traffic load due to the overheads of retransmission. The two ALOHA-Q schemes have similar performance characteristics as a function of traffic load and better performance at higher traffic levels. Similar performance can be observed for ALOHA-Q with 300 nodes, which indicates that its energy efficiency does not degrade due to increasing network size. S-MAC exhibits the same performance trend but generally higher energy costs resulting from the additional overheads in the frame and idle listening. Z-MAC has slightly higher energy cost than ALOHA-Q because of channel sensing (a node applies channel sensing in every Z-MAC slot when it has packets to send).

7. Conclusions and future works

In this paper, ALOHA-Q has been proposed as a low complexity MAC protocol capable of providing high energy-efficiency performance combined with high throughput and adequate delay. The main contribution of this protocol is its simplicity and low overheads compared with the majority of current schemes. Q-Learning is employed as an intelligent slot selection strategy in a frame based Slotted ALOHA scheme which results in migration from random access to perfect scheduling in steady state conditions. A detailed study of the convergence properties of the approach has been evaluated through a Markov model of the learning process. Results for a single hop network demonstrate that ALOHA-Q is able to provide rapid convergence to steady-state conditions and improved energy-efficiency, comparable throughput and delay to both S-MAC and Z-MAC despite being much simpler.

In ALOHA-Q and other learning based ALOHA schemes for multi-hop WSNs proposed in (Chu et al., 2012b, 2012a), fixed and predefined frame size is applied. Section 6 has demonstrated the impact of different frame size to system performance. Overestimating frame size generates unused slots and underestimating frame size introduces packet collisions, so they both affect channel performance. To improve system adaptability and channel performance under non-optimal frame size conditions, a learning algorithm of intelligently selecting optimal frame size is under development. By integrating learning based slot and frame size selection strategies, the network would start from random access with non-optimal frame size and reach perfect scheduling with optimal frame size after both learning algorithms have converged. A further key requirement is the ability for the network to adapt to different densities of node deployment, without requiring a fixed and pre-estimated frame size configuration.

Appendix A

Proof in chapter V section B

To prove the diagonal elements of the Jordan form of $\boldsymbol{P}$ are all positive (or the eigenvalues of $\boldsymbol{P}$ are all positive), we need to prove that the diagonal elements of $\boldsymbol{P}$ are all larger than 0.5, so that according to the Gershgorin Circle Theorem the eigenvalues will lie within the disc centred between 0.5 and 1 with the radius of less than 0.5, which means the eigenvalues are positive. We need to prove

$$p_{k,k}=\frac{k}{N}\left(\frac{N-1}{N}\right)^{N-k}+\frac{N-k}{N}\left(1-\frac{N-k}{N}\left(\frac{N-1}{N}\right)^{N-k-1}\right)>0.5 \tag{25}$$

where N is and integer larger than 2, and k is an integer between 0 and $N-1$. Move 0.5 to the left

$$\frac{2k}{N}\left(\frac{N-1}{N}\right)^{N-k}+\frac{N-2k}{N}-2\left(\frac{N-k}{N}\right)^{2}\left(\frac{N-1}{N}\right)^{N-k-1}>0 \tag{26}$$

Multiply $-N^{N-k+1}$on both sides

$$2(N-k)^2(N-1)^{N-k-1}-2k(N-1)^{N-k}-(N-2k)N^{N-k}<0 \tag{27}$$

We enlarge the first part of (26) by multiplying $N-1/N-k$ assuming $N>k\geq 1$, then we have

$$LHS(26)\leq 2(N-k)(N-1)^{N-k}-2k(N-1)^{N-k}-(N-2k)N^{N-k}<0 \tag{28}$$

Transform (27) to

$$\begin{aligned}&(N-2k)(N-1)^{N-k}-(2k-N)(N-1)^{N-k}+(2k-N)N^{N-k}<0\\&(N-2k)(N-1)^{N-k}+(2k-N)(N^{N-k}-(N-1)^{N-k})<0\\&(2k-N)(N^{N-k}-(N-1)^{N-k})<(2k-N)(N-1)^{N-k}\end{aligned} \tag{29}$$

For the situation that $2k-N<0$, (28) becomes

$$(N-1)^{N-k}-N^{N-k}<(N-1)^{N-k} \tag{30}$$

Which is obviously true. For the situation that $2k-N>0$, (28) becomes

$$N^{N-k}<2(N-1)^{N-k} \tag{31}$$

By using $x=e^{\ln x}$, (30) can be rewritten as

$$e^{(N-k)\ln\frac{N}{N-1}-\ln 2}<1 \tag{32}$$

To prove (31) we need to use the inequality that

$$\frac{x}{1+x}<\ln(1+x)<x \tag{33}$$

where $x>-1$ and $x\neq 0$. (32) can be proved as follow:

Proof of the right side

$$f(x)=\ln(1+x)-x\rightarrow f'(x)=\frac{1}{1+x}-1 \tag{34}$$

When $x>0$, $f'(x)<0$, and $f(x)<0$ because $f(0)=0$. When $x<0$, $f'(x)>0$ and $f(x)<0$ because $f(0)=0$.

Proof of the left side

$$f(x)=\frac{x}{1+x}-\ln(1+x)\rightarrow f'(x)=\frac{-x}{(1+x)^2} \tag{35}$$

When $x>0$, $f'(x)<0$, and $f(x)<0$ because $f(0)=0$. When $x<0$, $f'(x)>0$ and $f(x)<0$ because $f(0)=0$.

Go back to the proof of (31), we assum $x=(1/N-1)$, then we have

$$\frac{1}{N}<\ln\frac{N}{N-1}<\frac{1}{N-1} \tag{36}$$

And (31) becomes

$$LHS(31)<e^{\frac{N-k}{N-1}-\ln 2}<1 \tag{37}$$

By using the minimum value of $k=(N+1/2)$, (36) becomes

$$e^{\frac{1}{2}-\ln 2}<1 \tag{38}$$

And $\ln 2\approx 0.69$, (37) is proved. So we can say that (24) is true when $k\neq 0$ *or* $N/2$ (when N is even).

For the situation that $k=0$, (24) becomes

$$1-2\left(\frac{N-1}{N}\right)^{N-1}>0 \tag{39}$$

$$2\left(\frac{N-1}{N}\right)^{N-1}=e^{\ln 2+(N-1)\ln\frac{N-1}{N}}<e^{\ln 2+(N-1)\frac{-1}{N}}<1 \tag{40}$$

For the situation that $k=N/2$, (24) becomes

$$\frac{1}{2}\left(\frac{N-1}{N}\right)^{\frac{N}{2}}-\frac{1}{4}\left(\frac{N-1}{N}\right)^{\frac{N}{2}-1}>0 \tag{41}$$

Multiply both sides of (40) by $4((N-1)/N)^{1-\frac{N}{2}}$

$$2\left(\frac{N-1}{N}\right)-1>0 \tag{42}$$

which is obviously true because N is larger than 2.

Terms and explanations

(a) ALOHA (Abramson, 1970): a simple MAC protocol for packet based wireless communication systems, developed in 1970s. For the simplest ALOHA, a node sends a packet (to the destination) immediately after the packet is ready. A more detailed study of ALOHA schemes can be found in (Yi, 2013) chapter 5.

(b) Carrier Sense Multiple Access (CSMA) (Kleinrock and Tobagi, 1975): widely used MAC of many wireless systems. A commonly used scheme is that the sender sends an RTS first, the receiver replies a CTS if it is not busy, the sender then sends data and the receiver replies ACK (or no ACK depends on service requirements) when data transmission completes. Before all transmissions the sender first senses the channel, and if the channel is free then the sender starts to transmit

(c) Time Division Multiple Access (TDMA): to allow multiple nodes to send packets on shared radio resource (channel), nodes can access the channel during different time periods, because collision will happen if multiple nodes send packets

on the same channel at the same time. More details can be found in (Yi, 2013) chapter 3.

(d) Node: the basic unit of a WSN. A node usually has four components: a sensing module (sensor) to generate data, a processing module (micro processor) for data processing, a power module (batteries) and a communication module (radio).

(e) Slot: the basic transmission unit in a TDMA based system. A slot represents a period of time on a channel. In theoretical analysis a slot is usually assumed to be the time required to send a data packet (length of a data packet on a channel). In WSNs research, a slot usually contains the length of a data packet, the length of an ACK packet plus some guard band in case of synchronisation offset (clock drift).

(f) Frame: a frame is composed of a certain number of slots. By dividing time into repeating frames, the slots of each frame can be numbered to apply learning algorithms. In case of this paper, each slot in the repeating frames is given a Q value.

References

Akyildiz, F., Sankarasubramaniam, Y., Cayirci, E., Su, W., 2002. Wireless sensor networks: a survey. Comput. Netw. 40, 393–422.

Abramson N., 1970. The aloha system: another alternative for computer communications, In: Proceedings of the AFIPS '70 (Fall), pp. 281–285.

Barcelo, J., Bellalta, B., Cano, C., Sfairopoulou, A., Oliver, M., Verma, K., 2011. Towards a collision-free WLAN: dynamic parameter adjustment in CSMA/E2CA. EURASIP J. Wirel. Commun. Netw.

Chu Y., Mitchell P.D., Grace D., 2012a. Reinforcement learning based ALOHA for multi-hop wireless sensor networks with informed receiving, In: Proceedings of the IET Conference on Wireless Sensor Systems (WSS), pp. 1–6.

Chu Y., Mitchell P.D., Grace D., 2012b. ALOHA and Q-Learning based medium access control for wireless sensor networks, In: Proceedings of the International Symposium on Wireless Communication Systems (ISWCS), pp. 511–515.

Chao, C.-M., Lee, Y.-W., 2010. A quorum-based energy-saving MAC protocol design for wireless sensor networks. IEEE Trans. Veh. Technol. 59 (2), 813–822.

Chen, Y.L., Wang, N.C., Chen, M.Y., Huang, Y.F., Shih, Y.N., 2013. A concentric clustering architecture with PSO algorithm in a wireless sensor network. Sens. Mater.: Int. J. Sens. Technol. 26 (5), 325–332.

Chen, Y.L., Chen, M.Y., Cheung, F.K., Chang, Y.C., 2014. A new hybrid architecture with an intersection-based coverage algorithm in wireless sensor networks. Comput. Sci. Inf. Syst. J. 11 (3), 1017–1035.

Demirkol, I., Ersoy, C., Alagoz, F., 2006. MAC protocols for wireless sensor networks: a survey. Commun. Mag. 44, 115–121.

Fang, M., Malone, D., Duffy, K., Leith, D., 2013. Decentralised learning MACs for collision-free access in WLANs. Wirel. Netw. 19 (1), 83–98.

Finkbeiner II, D.T., 1978. Introduction to Matrices and Linear Transformations. Freeman, San Francisco.

IEEE Standard for Local and Metropolitan area networks, 2012. Part 15.4: Low-Rate Wireless Personal Area Networks.

Kaelbling, L.P., Littman, M.L., Moore, A.W., 1996. Reinforcement learning: a survey. Artif. Intell. Res. 4, 237–285.

Kwak, B.-J., Song, N.-O., Miller, L.E., 2005. Performance analysis of exponential backoff. IEEE/ACM Trans. Netw. 13, 343–355.

Kleinrock, L., Tobagi, F.A., 1975. Packet switching in radio channels: Part 1-carrier sense multiple-access modes and their throughput-delay characteristics. IEEE Trans. Commun. 23, 1400–1416.

H. Li, D. Grace, P.D. Mitchell, 2010. Cognitive radio multiple access control for unlicensed and open spectrum with reduced spectrum sensing requirements, In: Proceedings of the International Symposium on Wireless Communication Systems (ISWCS), pp. 1046–1050.

P. Lin, C. Qiao, X. Wang, 2004. Medium access control with a dynamic duty cycle for sensor networks, In: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), pp. 1534–1539.

Lindsey S., Raghavendra, C.S., 2002. PEGASIS: power-efficient gathering in sensor information systems, In: Proceedings of the IEEE Aerospace Conference, Big Sky, Montana, pp. 1124–1130.

Li, J., 2004. A Bit-Map Assisted Energy-Efficient MAC Scheme for Wireless Sensor Networks (MS. thesis). Electrical Engineering, Mississippi State University, Starkville, MS.

Poole, D., 2005. Linear Algebra. A Modern Introduction. Cengage Learning, Boston.

Rhee, I., Warrier, A., Aia, M., Min, J., Sichitiu, M.L., 2008. Z-MAC: A hybrid mac for wireless sensor networks. IEEE/ACM Trans. Netw. 16, 511–524.

Rhee I., Warrier A., J. Min, Xu L., 2006. Drand: distributed randomized TDMA scheduling for wireless ad hoc networks, In:Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), pp. 190–201.

Rajendran, V., Obraczka, K., Garcia-Luna-Aceves, J.J., 2006. Energy-efficient, collision-free medium access control for wireless sensor networks. Wirel. Netw. 12.

Y. Sun, S. Du, O. Gurewitz, D. Johnson, 2008. DW-MAC: A low latency, energy efficient demand-wakeup MAC protocol for wireless sensor networks, In: Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), pp. 53-62.

C. Schurgers, M.B. Srivastava, 2001. Energy efficient routing in wireless sensor networks, In:Proceedings of the MILCOM Commununications for Network-Centric Operations: Creating the Information Force, Mclean.

Shafiullah, G.M., Azad, Salahuddin A., Shawkat Ali, A.B.M., 2013. Energy-efficient wireless MAC protocols for railway monitoring applications. IEEE Trans. Intell. Transp. Syst. 14 (2), 649–659.

Suttong, R.S., Barto, A.G., 1998. Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA.

The ZigBee Alliance, 2015. ⟨www.zigbee.org⟩ (retrieved 06.02.15).

Tang Y., Grace D. Clarke T., Wei J., Multichannel non-persistent CSMA MAC schemes with reinforcement learning for cognitive radio networks, In: Proceedings of the International Symposium on Communications and Information Technologies (ISCIT), pp. 502–506.

TinyOS: An operating system for wireless sensor networks, 2005. Ambient Intelligence, Springer-Verlag, Berlin, 115–148.

Vuran, M.C., Akyildiz, I.F., 2006. Spatial correlation-based collaborative medium access control in wireless sensor networks. IEEE/ACM Trans. Netw. 14 (2), 316–329.

Varga, R.S., 2004. Geršgorin and His Circles. Springer-Verlag, Berlin.

Wei, Y., Heidemann, J., Estrin, D., 2004. Medium access control with coordinated adaptive sleeping for wireless sensor networks. IEEE/ACM Trans. Netw. 12 (3), 493–506.

Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R., Gill, C., 2005. Integrated coverage and connectivity configuration for energy conservation in sensor networks. ACM Trans. Sens. Netw. 1 (1), 36–72.

Yi, Chu, 2013. Application of Reinforcement Learning on Medium Access Control for Wireless Sensor Networks (Ph.D. thesis). University of York, UK.

⟨http://www.memsic.com/userfiles/files/datasheets/wsn/iris_datasheet.pdf⟩ (retrieved 06.02.15).