CORE
Metadata, citation and similar papers at core.ac.uk
Provided by UCL Discovery

Reinforcement learning based routing for energy sensitive wireless mesh IoT networks

Y. Liu, K.-F. Tong and K.-K. Wong

With the huge growth of the Internet of Things (IoT) in manufacturing, agricultural and numerous other applications, connectivity solutions have become increasingly important especially for those covering wide remote area in the scale of kilometre squares. Although many low-power wide-area network (LPWAN) technologies such as Long Range (LoRa) are supposed to support long-range low-power wireless communication, the underneath star topology limits the scalability of the networks due to the need of a central hub. To provide connectivity to a wider area, the authors propose to build the mesh topology upon these LPWAN technologies. One of the challenges of meshing these networks is the routing mechanism originally designed for star networks is not energy sensitive. In this letter, the authors address this issue by proposing a distributed as well as energy-efficient reinforcement learning (RL) based routing algorithm for the wide area wireless mesh IoT networks. They evaluate the failure rate, spectrum and power efficiencies of the proposed algorithm by simulations, which resemble the long-range IoT networks, by comparing it to that of a random routing with loop-detection algorithm and a centralised pre-programmed routing algorithm which represents the ideal-scenario. They also present a progressive study to demonstrate how the learning in the algorithm reduces the power consumption of the entire network.

Introduction: Internet of Things (IoT) are revolutionising the world by enabling automatic data collection to data utilisation through machine-to-machine (M2M) communications. For those applications deployed in remote areas, the devices are typically left unattended for a long period of time and expected to be powered by batteries and not to be connected to the power grids. There have been a number of network protocols to support these IoT/M2M applications, such as narrowband IoT (NB-IoT) over cellular networks, LoRa and Sigfox. Despite these technologies being widely commercialised, there remain some key challenges.

Power efficiency is an issue. In order to maximise the lifespan of these unattended battery-powered remote nodes, the network itself ought to be power efficient and simple to manage. Also, data needs to be accessed at any time for analysis, so high availability is necessary. In addition, as the coverage grows, the network must be able to handle the addition or reduction of nodes and changes of the network geometry as it might happen at any time in these networks. The throughput and delay of the network are usually less important, as most of these networks do not require real-time access so requirement on throughput and delay is more flexible. We will focus on the three issues listed above in this letter.

Wireless mesh networks (WMN) are often considered as non-scalable as they likely suffer from interferences [1], [2]. However, this may not be the case for applications such as soil quality monitoring networks in the farms and air quality monitoring networks in the urban areas, where the remote sensing and monitoring nodes are mostly static, and the only change may happen to the network topology is the addition of nodes from the extra deployment or reduction due to battery power failures. This is the scenario this letter aims to address, where routing needs to be energy aware and adaptive to the battery life of the IoT devices.

Routing for traditional star networks is mostly done by the shortest path first (SPF) algorithm, such as Dijkstra's Algorithm. However, this algorithm is not suitable for the mesh IoT network because in SPF, when there is a change in the network, the routing table of every node, including those who have not transmitted or received in this transmission, must be re-generated to update the optimal path. Due to the dynamic power status of nodes in the mesh IoT/M2M networks considered in this letter, this is too power demanding and inefficient to be implemented.

In this regard, using deep learning for routing in WMN might seem an attractive solution [3]. However, this may not always be possible because a large dataset which represents all the possible scenarios of routing is required to train the neural network, and the intelligence to go beyond what appears in the dataset is very limited, or the method will fail if the network experiences changes that do not exist in the training dataset. In the energy aware IoT network considered in this letter, the size of the dataset will grow exponentially with the number of the IoT devices, which makes deep learning a less practical solution.

To maximise energy efficiency and keep up with the changes of the status of the nodes, we opt for the reinforcement learning (RL) approach to update and manage the routing table in a distributed fashion and make routing decision based on the experience the nodes gather over time as the network operates. Unlike supervised learning which requires a good dataset beforehand, RL is fully adaptive to changes and can also maintain a good balance between exploration and exploitation of the network [4]. In order to optimise the usability of IoT networks in remote areas, we propose an RL-based routing algorithm for energy sensitive wireless mesh IoT networks. By focusing on the energy awareness of each node and the entire network, the algorithm improves usability of the network in terms of failure rate as well as energy and spectral efficiency. In the proposed algorithm, the routing table is constantly populated and updated with a model-free RL method called temporal difference (TD) learning to find the estimated best route in a continuing basis. We also propose a new cost function that considers the battery life of the nodes.

The Network Model: This letter considers an IoT network with a finite number of nodes randomly distributed in a given area. Each node can communicate with adjacent nodes within range. Transmission from a node to an adjacent node takes place in synchronised time slots and each link transmission costs a unit of bandwidth. Simultaneous transmissions take place over non-overlapping channels so that no interference will occur. Each node has a finite battery which is recharged periodically. Fig. 1 illustrates some examples of randomly generated networks.

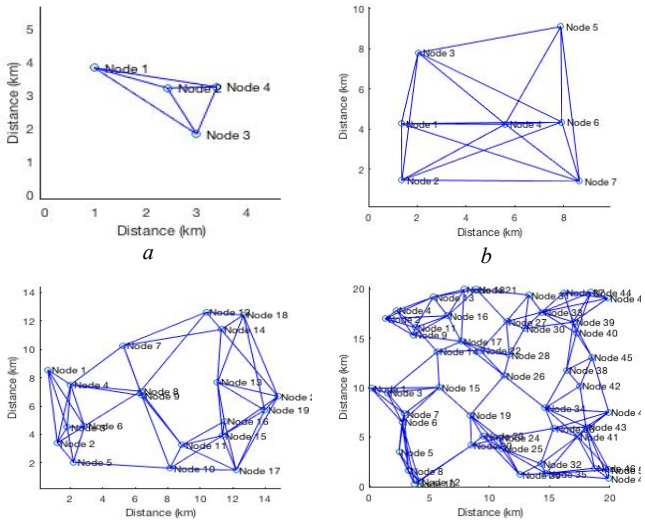

Fig. 1 *Examples of the randomly generated networks.*
a sample of a 4-node network.
b A sample of a 7-node network.
c A sample of a 20-node network.
d A sample of a 50-node network.

RL Based Routing Algorithm: A route (in contrast to link) transmission is an action of transferring one data packet from a source node (SN) to a destination node (DN) through several intermediate nodes (INs) along a route in the mesh network. Every node in the network can be equally assigned as an SN, DN, or IN at any time according to the need of the transmission. Each node keeps a routing table that is used to decide the next hop in a transmission. Inside that routing table, there are entries of possible next nodes (NNs) which reach all possible DNs in the network. It is kept on a list of active adjacent nodes. Any two nodes with sufficient energy for direct communication will consider each other as active adjacent nodes. Each node will check and update its active adjacent nodes list when the transmission is completed. Additionally, a routing metric (RM) that indicates the probability of selecting a particular NN from the active adjacent nodes list is also stored. The probability of a node being selected is calculated by using a Boltzmann exploration process as

$$p(NN_n) = \frac{e^{RM(NN_n)/\tau}}{\sum_{i=1}^{n} e^{RM(NN_i)/\tau}}. \quad (1)$$

In (1), $p(NN_n)$ represents the probability of the n^{th} possible NN (NN_n) being selected by the SN in the route. The possible NNs are fetched from the routing table of SN before a packet is dispatched from the SN. We

denote the hyperparameter as τ to control the spread of the SoftMax distribution of all the possible routes. If the routing table of the SN does not have an entry of the DN, the SN will initialise a new entry for that DN and assign an averaged RM value between all the active adjacent nodes, i.e., RM = 1 / number of active adjacent nodes, as the initial RM value. By using the Boltzmann process generated probability instead of directly using the metric, *RM(NN$_n$)*, we are able to balance exploration and exploitation, which gives RL the ability to adapt to changes and evaluate the optimality of the current best solution [[5]].

After the SN picks the NN, the NN will be assigned as IN$_1$. When the packet has been dispatched, the SN will wait for the feedback of the transmission from the DN when the transmission is finished. Then the RMs of all the visited nodes will be updated from the feedback of the path quality (*PQ*) value using RL. Specifically, to compute the PQ value, the cost of the current leg of transmission (C) will be used, given by

$$C = \mathrm{W}_1 P_t - \mathrm{W}_2 \log(P_{TN}) - \mathrm{W}_3 \log(P_{RN}), \quad (2)$$

where W$_1$, W$_2$, and W$_3$ are some prescribed positive weights, and P_t represents the transmission power used in the corresponding leg of transmission. The value of P_t is calculated by

$$P_t = \frac{(2^{\frac{R}{\mathrm{BW}}}-1)*(\mathrm{I}+\mathrm{N})*\mathrm{d}^{\alpha}}{|\mathrm{h}|^2}, \quad (3)$$

where R stands for the transmission rate, BW stands for the transmission bandwidth, I and N are the interference and noise levels, d is the distance between the transmitting node and receiving node of the current leg (e.g. SN and IN$_1$ in the first leg), α is the pathloss exponent, and h is the channel gain, which is assumed known in this letter.

In (2), P_{TN} and P_{RN} represent the remaining power (in percentage) at the transmit and receive nodes, respectively. We use log operation to have a smaller impact on the cost value when the power available at the node is high, while the impact will be high when the power available at is low. Considering the remaining power levels in both nodes in the cost function will empower the network to be energy aware for routing.

We attach a variable in the header of the packet to keep track of the nodes that the packet has visited to avoid looping. The visited nodes will be removed from the active adjacent nodes list in the current node. The same loop-checking process will be used in every leg of the transmission. When moving on to the next leg of the transmission, we consider IN$_1$ as the new SN and IN$_2$ to be the new NN and we add IN$_1$ to the header and continue the same routing process until the packet successfully reaches the DN. If a node has no possible NN due to no active adjacent node or all active adjacent nodes have been removed because they have been visited, then this leg will be deemed unsuccessful. We will then roll back the transmission to the previous node and retry the same node selection process with a different possible NN. If the total number of retries (NR) is greater than a preset maximum number (NR$_{max}$) or all the possible NNs have been tried, then the transmission from SN to DN is unsuccessful. Once the transmission is completed, we work out the *PQ* value by

$$PQ = SB - \sum_{over\ path} C \quad (4a)$$

$$SB = \begin{cases} \mathrm{SBV}, & if\ transmission\ is\ successful \\ 0, & if\ transmission\ is\ unsuccessful \end{cases}. \quad (4b)$$

The PQ equation consists of two parts; the sum of the cost values of all the legs in the transmission from the current node to the DN and a success bonus (*SB*). To calculate the PQ value, the positive *SB* value (SBV) will be added to a successful transmission to compensate the cost values. Zero *SB* will be given to the unsuccessful transmissions, and in this case, *PQ* will be negative. We calculate *PQ* of each node (SN and INs) in a route and use the *PQ* values along the route to the DN to update the RM of each node accordingly by using the TD-learning based equation given by

$$RM^*(NN) = RM(NN) + \beta(PQ + \gamma RM(NN') - RM(NN)), \quad (5)$$

where *RM*(NN)* denotes the updated value, *RM(NN)* represents the start RM value at the node for selecting the route toward DN via node NN, *PQ* is the path quality value obtained from the feedback of the current transmission, *RM(NN')* represents the expected RM value of the selected route which is calculated by averaging all the possible routes in this leg of transmission, γ is the discount rate to gradually stabilise the value over time. We also use a learning rate β to control the speed of learning.

Finally, we also consider the scenario when the battery of each node can be recharged over time. To do so, we reset the available energy in the nodes to the maximum power (P$_{max}$) at every charging cycle (CC). The nodes will also refresh their active adjacent nodes in the routing tables.

With transmissions taking place over time, the proposed RL algorithm repeatedly updates the RM values in the routing tables of the nodes from the PQ feedback when transmission is completed. This newly learnt RM will direct onwards transmissions to a more optimised route.

Simulation Results: For each episodic simulation, transmissions were generated using a Poisson Process to randomly appoint the SN and DN with a fixed transmission rate (R). Around 20,000 transmissions were generated in a time span of 20,000s. We used networks of 4 different sizes to represent different scales and coverage of the sensor networks. We began with the 4-node networks in an area of 5 km by 5 km to represent the simple mini IoT networks. The second set of networks had 7 nodes distributed in a 10 km by 10 km area while the third series of networks had an area of 15 km by 15 km and consisted of 20 nodes, representing the medium size IoT/M2M networks. Finally, the random-generated 50-node networks covering a 20 km by 20 km area represented the large-scale mesh IoT networks. We have chosen these configurations of the network to represent different scenarios that the remote IoT networks may be deployed. As the proposed network aims to provide remote monitoring in remote areas which may not be covered by any wireless system, a network of 50 nodes covering 20 km by 20 km represents our general use case. When more nodes or a larger area needs to be covered, it can work with other systems, such as mobile network, to form a joint network. It will be more efficient as it is more suitable for covering a much larger number of mobile nodes in a smaller area, but this is outside the scope of this letter. We generated 10 different networks of each size in the simulations to average the results. Each node was initialised with a maximum power (P$_{max}$) of 15 Wh and was periodically recharged in a 1000s CC. The available links between nodes were also randomly picked. In the medium- and large-scale networks, each node was connected with up to 5 closest nodes as the active adjacent nodes to simplify the simulations. Each transmission used the fixed BW of 125 kHz, which is the same as the minimum transmission BW of LoRa [6]. The channel and transmission related parameters used in the simulation are shown in Table 1 and Table 2.

Table 1 Channel parameters used in the simulation

BW (kHz)	N (dBm)	α	h (fixed)	I
125	-130	2.8	2	0

Table 2 Transmission related parameters used in the simulation

R (kb/s)	PZ (kb)	P$_{max}$ (Wh)	CC (s)	NR$_{max}$	τ	β	γ
1	1	15	1000	5	0.5	0.8	0.8

The results of a centralised SPF algorithm are provided to serve as the performance upper bound. In the centralised SPF simulations, the energy available in each node is always assumed to be infinite and the routing information is generated and stored in a centralised database that oversees the entire network topology. We also provide the results of a random routing method as the lower bound benchmark. We considered random routing without learning because the traditional ad hoc routing is done in a very similar way by choosing any possible available route.

We investigated the average failure rate (%), the spectrum efficiency (bit/Hz) and energy efficiency (bit/kJ) of each set of networks for each algorithm. The spectral efficiency $\eta_{spectral}$ and energy efficiency η_{energy} are, respectively, defined as

$$\eta_{spectum} = \frac{D_{transmitted}}{BW*NC_{used}}, \quad (6)$$

$$\eta_{energy} = \frac{D_{transmitted}}{P_{total}}, \quad (7)$$

where $D_{transmitted}$ is the total amount of data successfully transmitted from SNs to DNs of all transmissions during one simulation, NC_{used} represents

the total number of used carriers, and P_{total} is the total power consumed in the simulations for calculating the energy efficiency η_{energy}.

Each simulation covered a period of 20,000s with randomly generated approximately 20,000 SN-to-DN transmissions. We ran simulations of each scale of network for 10 times with the same set of SN-to-DN transmissions and average the results of each scenario.

Fig. 2 shows the failure rate results and as we can see, the proposed RL routing algorithm significantly reduces the failure rate over random routing in the 7-, 20- and 50-node networks. The small difference in the 4-node networks is due to the simplicity of the network and the lack of routing choice, and hence the RL algorithm is hardly making a significant difference. In addition, unsurprisingly, the centralised SPF method has shown no failure because of its global understanding of the network.

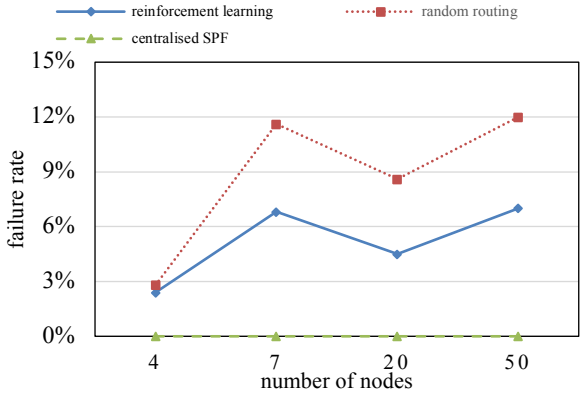

Fig. 2 *Average failure rate.*

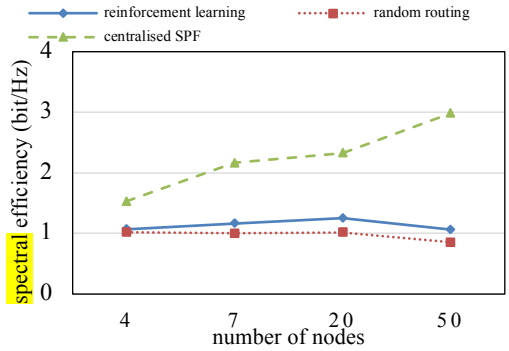

Fig. 3 *Spectral efficiency.*

In Fig. 3, the results demonstrate that the RL algorithm also has gain in terms of spectral efficiency over random routing, but the advantage appears to be much less noticeable. This is because the proposed RL algorithm considers the energy consumption pattern in the cost function when learning the network, to avoid nodes with limited residual battery. Therefore, it is remarkable that the proposed method can still improve spectral efficiency while at the same time reducing the failure rate. Nonetheless, the centralised method shows significant superiority in the spectral efficiency as it will always use the most effective route, given also an infinite energy supply for all the nodes.

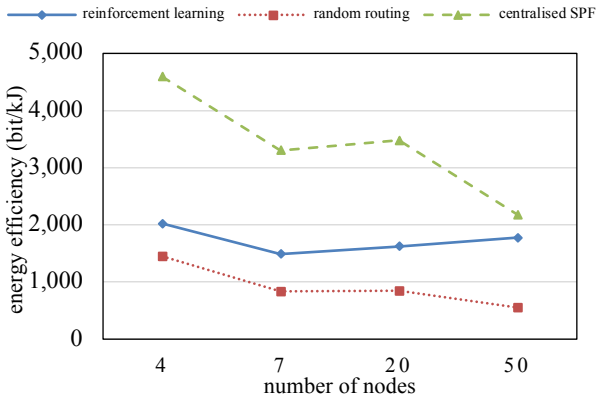

Fig. 4 *Energy efficiency.*

In terms of the energy efficiency as provided by the results in Fig. 4, similar observation can be made. With the increasing of the number of nodes, the RL learning algorithm demonstrates apparent superiority over the random routing scheme. In the case of 50-node networks, it even shows a comparable result to the idealised centralised method.

Finally, we plot the progressive timeline of the series of 10 simulations of 50-point largest network conducted in Fig. 5. It can be observed that the failure rate of the learning algorithm gradually reduces overtime, demonstrating an increased network stability over time. Besides, the energy efficiency is also increased as learning progresses.

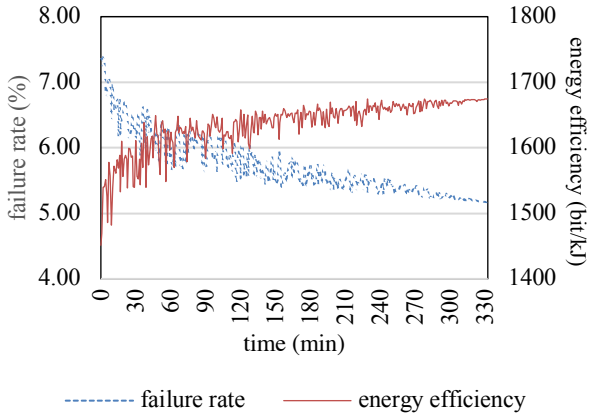

Fig. 5 *Time series of average failure rate and energy efficiency for the 50-node mesh IoT networks.*

Conclusion: By using RL for routing in the IoT/M2M energy sensitive mesh networks, considerable improvements in power efficiency, failure rate and spectrum efficiency have been demonstrated. With the increase of the scale of network, the benefit over the routing algorithm without learning capabilities become more significant. Moreover, the compelling improvement of the performance contrast to the modest addition of the complexity clearly demonstrates the potential of the method.

Y. Liu, K.-F. Tong and K.-K. Wong, (*Department of Electronic and Electrical Engineering, University College London, London, United Kingdom*)

E-mail: yu.liu@ucl.ac.uk

References

[1] Srivathsan S., Balakrishnan N., Iyengar S.S,: 'Scalability in Wireless Mesh Networks:' in *Guide to Wireless Mesh Networks.*, Misra S.; Misra, S. C.; Woungang, I., London: Springer-Verlag, 2009, pp 325-347, ISBN 978- 1-84800-908-0

[2] Sampaio S., Souto P., Vasques F.,:'A review of scalability and topological stability issues in IEEE 802.11s wireless mesh networks deployments' , *International Journal of Communication Systems*, 2016,29, (4), pp 671-693, doi: 10.1002/dac.2929

[3] Tang, F., Mao, B., Fadlullah Z.M., Kato N., Akashi, O., Inoue, T., Mizutani, K.: 'On Removing Routing Protocol from Future Wireless Networks: A Real-time Deep Learning Approach for Intelligent Traffic Control', *IEEE Wireless Communications*., 2018, 25, (1), pp.154-160, doi: 10.1109/MWC.2017.1700244

[4] Cesa-Bianchi, N., Gentile,C., Lugosi,G., Neu, G.: 'Boltzmann Exploration Done Right', 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, December 2017.

[5] Kaelbling, L.P., Littman, M. L., Moore, A. W. 'Reinforcement Learning: A Survey', *Journal of Artificial Intelligence Research, 1996, (4), pp. 237-285, doi: 10.1613/jair.1.11396*

[6] LoRa Alliance, 'LoRaWANTM 1.1 Specification', https://lora-alliance.org/sites/default/files/2018-04/lorawantm_specification_-v1.1.pdf, accessed March 2019.