Deep Reinforcement Learning-Based Multi-Hop State-Aware Routing Strategy for Wireless Sensor Networks

Aiqi Zhang, Meiyi Sun, Jiaqi Wang, Zhiyi Li, Yanbo Cheng and Cheng Wang *

School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China; aiqiz@bupt.edu.cn (A.Z.); meiyi123@bupt.edu.cn (M.S.); jia_qi@bupt.edu.cn (J.W.); hexing@bupt.edu.cn (Z.L.); Cheng_Yanbo@bupt.edu.cn (Y.C.)
* Correspondence: wangcheng@bupt.edu.cn

Abstract: With the development of wireless sensor network technology, the routing strategy has important significance in the Internet of Things. An efficient routing strategy is one of the fundamental technologies to ensure the correct and fast transmission of wireless sensor networks. In this paper, we study how to combine deep learning technology with routing technology to propose an efficient routing strategy to cope with network topology changes. First, we use the recurrent neural network combined with the deep deterministic policy gradient method to predict the network traffic distribution. Second, the multi-hop node state is considered as the input of a double deep Q network. Therefore, the nodes can make routing decisions according to the current state of the network. Multi-hop state-aware routing strategy based on traffic flow forecasting (MHSA-TFF) is proposed. Simulation results show that the MHSA-TFF can improve transmission delay, average routing length, and energy efficiency.

Keywords: multi-hop state-aware; traffic flow prediction; deep reinforcement learning; wireless sensor networks

1. Introduction

The Internet of Things (IoT) connects a range of physical devices through a variety of transmission technologies, e.g., WiFi, 5G, wireless sensing networks, etc. IoT has become an important network infrastructure in the world of the future [1–3]. The resilience, reliability, and efficiency of agriculture and smart city services can be effectively enhanced by IoT technologies [4]. Among these communication technologies applied in the IoT networks, wireless sensor networks (WSNs) play an important role [5]. The wireless sensor network is generally composed of sensor nodes equipped with wireless communications equipment. These nodes are autonomous and spatially distributed. These nodes in operation can realize real-time monitoring of the physical environment, provide massive data for back-end services to analyze, and build an intelligent model of the sensing layer. However, the typical features of these nodes generally are limited storage, power, and computing resources. Since the purpose of WSNs is to perceive and transmit information efficiently, how to design the appropriate routing strategy will directly affect the performance of the network. Therefore, the core goal of the WSN routing strategy is to reduce energy consumption, enhance network damage resistance, and prolong network life.

The routing strategy of WSNs has attracted the attention of many researchers and institutions. The Ad hoc On-Demand Distance Vector Routing (AODV) was designed for mobile ad-hoc networks [6]. The behavior of the AODV routing protocol in situations of link failures was investigated. The Destination Sequence Distance Vector Routing (DSDV) was designed to attempt to cope with the geographic mobility of network nodes [7]. The Dynamic Source Routing (DSR) was defined by [8]. The data sender can determine precisely the needed sequence of the nodes to propagate the packet. However, DSR has the
disadvantage of high signaling overheads. Besides, it has been found that these routing protocols have poor performance under changing network topologies, which is very common in WSNs due to factors such as node sleeping [9]. The flooding routing strategy [10] is designed primarily to deal with network scalability and energy efficiency. The time tag is used to combine signal and flooding routing, which can reduce the signaling overheads.

The above research results on routing strategies need to be further considered for the change of network topology. Since the scale of IoT applications and services is increasing day by day, the network topology is becoming more complex. The local topology is more complex and easy to change. As the wireless sensor nodes have limited computing and communications capabilities, the network will use an appropriate sleeping strategy to save node energy, which entails changing the network topology. In addition, both the sensor nodes and the entire sensor network are vulnerable to attacks. Therefore, resilient routing strategies are needed to handle the routing problem under changing network topology, whether caused by the sleeping scheme of the network or by an external attack. Inspired by this view, a failure-resilient routing strategy via IoT networks was proposed in [11]. The routing strategy in [11] can automatically explore and establish optimal local routes in the event that unpredictable failures cause the previous route to fail. In addition, the temporary rerouting will be abandoned when the failures have been repaired. A dynamic Attack-Resilient Routing (ARR) strategy was designed in [12] in software-defined networks. The ARR dynamically routes the data fragments through a set of reliable and lightly loaded paths to achieve multi-path diversity, thus improving the data availability of the destination in the presence of attacks. According to the above research results, multi-path routing in wireless sensor networks is one of the more effective methods to deal with network topology changes. However, the multi-path routing strategy generally requires the computing capacity of the node, and it will increase the signaling overhead, which needs further study.

Recently, deep learning has been garnering research attention in the network routing area [13]. The deep-learning-based intelligent routing strategies are discussed in [14–16]. It has been proved in these papers that deep learning methods can be used to design the routing strategy in WSNs. A Deep Reinforcement Learning (DRL)-based Quality-of-Services (QoS) routing strategy was proposed in [17] to handle the secure routing problem in a software-defined network. A deep-learning-based link prediction model was proposed in [18]. A resilient routing algorithm was presented to ensure the integrality of transmitted data and shorten the length of the transmission path selectively. A value matrix-based deep learning approach was proposed in [19]. In [19], the deep Convolutional Neural Network (CNN) and the Deep Belief Network (DBN) were used together to deal with the large-scale network problem. Through the above research, we find that the traffic between nodes has some regularity, which can help to improve the performance of the routing strategy.

In this paper, the traffic flow prediction strategy based on the Recurrent Neural Network (RNN) combined with the Deep Deterministic Policy Gradient (DDPG) method is proposed to predict the network traffic distribution. Furthermore, the node state of the WSNs is analyzed, which considers various situations such as free, busy, and congested. While the multi-hop node state is given, a Double-Deep Q Network (DDQN) is trained to achieve the optimal next hop. Therefore, a Multi-Hop State-Aware routing strategy based on Traffic Flow Forecasting (MHSA-TFF) is proposed. In the MHSA-TFF, the routing strategy is divided into two parts, the traffic flow prediction part and the next-hop selection part. Simulation results show that the MHSA-TFF can improve transmission delay, average routing length, and energy efficiency.

The remainder of this paper is organized as follows. In Section 2, the system model of WSNs is presented. The topology of the network and the multi-hop node state-aware model are described. In Section 3, the traffic flow prediction part based on the RNN combined with the DDPG method is presented. In addition, the optimal next hop decision routing strategy based on the Double Deep Q Network is discussed in detail. In Section 4,
the simulation results are discussed, including the packet delivery ratio and the end-to-end delay under the topological state changes. Finally, Section 5 draws the conclusion.

2. System Model

Due to the development of heterogeneous node types in MSNs, a wireless sensor network that consists of a large number of ordinary nodes and a few enhanced nodes is considered in this paper. The transmitting power, energy, and computing capacity of ordinary nodes are limited, while the enhanced nodes have sufficient energy and computing capacity. Therefore, these enhanced nodes can take responsibility for processing deep learning models and controlling the routing strategy of the whole network. These enhanced nodes will collect the historical traffic flow information of ordinary nodes, and based on this information, complete the prediction of future traffic. Next, these enhanced nodes are responsible for updating the node-wide multi-hop topology information and intelligently performing local routing optimization. Furthermore, these enhanced nodes will regularly exchange network topology information so that the entire network topology information will be periodically updated. These enhanced nodes send the routing results to the ordinary nodes so that the ordinary nodes can complete the next-hop node selection process of the forwarding packets. Indeed, the signaling interactions caused by the enhanced nodes will consume network resources and incur additional overhead. However, such overhead can be reduced by controlling the interaction frequency. The scale of these overheads will not be significant because the network topology does not change very quickly in a real scenario, such as the industrial IoTs, intelligent agriculture, etc. The node connectivity architecture of WSN is shown in Figure 1.

As it is shown in Figure 1, the enhanced node is represented by the square icon, and the ordinary node is represented by the circular icon. Ordinary nodes have at least one routing path, and some of the ordinary nodes connect directly with enhanced nodes. Enhanced nodes have connections to multiple ordinary nodes. Some enhanced nodes have communication links with each other, while others need to be routed through ordinary nodes. From this topology, it can be seen that each enhanced node is responsible for the ordinary node within multi-hop. The node connectivity architecture is modeled as a direct graph $G = (V, E)$. $V$ is the set of all enhanced nodes and ordinary nodes, which is represented as $\{v_1, v_2, ..., v_n\}$. $E$ represents the set of communication links between each node. Since there are two kinds of nodes in the wireless sensor system, we define the $V^E$ and $V^O$ as the set of all enhanced nodes and ordinary nodes, respectively. In addition, we use
the $E^E$, $E^O$, $E^{OO}$ as the communication links between two enhanced nodes, enhanced node and ordinary node, and two ordinary nodes, respectively. Therefore, through interactive link information among nodes, enhanced nodes can obtain the whole network topology information and calculate the hop number between any two nodes. Therefore, the $\Gamma_i(v)$ is defined as the set of $i$-hop neighbors of node $v$, where $i = \{1, 2, ..., k\}$. In this paper, the enhanced node only requires the collection of the neighbor node state within a two-hop range. The definition of the symbols used later is summarized in Table 1.

Table 1. Definition of the symbols.

| Symbol | Definition |
|--------|------------|
| $G$    | The direct graph of the WSN |
| $V$    | The set of nodes |
| $E$    | The set of links |
| $V^E$  | The set of enhanced nodes |
| $V^O$  | The set of ordinary nodes |
| $E^E$  | The communication link between two enhanced nodes |
| $E^{EO}$ | The communication link between enhanced node and ordinary node |
| $E^{OO}$ | The communication link between two ordinary nodes |
| $\Gamma_i(v)$ | The set of $i$-hop neighbors of node $v$ |

3. Multi-Hop State-Aware Routing Strategy Based on Traffic Flow Forecasting

Based on the system model proposed above, two problems are studied in this paper. The first problem is how to establish a traffic prediction model between two nodes and make routing decisions based on the results of the traffic prediction model. The second problem is how to make the enhanced nodes respond intelligently to the changing of network topology.

To the first problem, the traffic flow prediction strategy based on the RNN network combined with the Deep Deterministic Policy Gradient (DDPG) method is proposed to predict the network traffic distribution. To the second problem, the node state of the WSNs is analyzed, and a Double-Deep Q Network is trained to achieve the optimal next hop under the condition of topological changes.

3.1. Traffic Flow Prediction Strategy Based on RNN Combined with the DDPG Method

According to the research results in [20,21], it can be assumed that the size of traffic between two nodes is related to the geographical location between the two nodes. In this section, the traffic model between two nodes is proposed. It is assumed that each node can obtain its own average traffic after running for a period of time, which is defined as $j_i$ and belongs to node $v_i$. Therefore, the traffic between node $v_i$ and $v_j$ can be calculated as (1).

$$j_{ij} = j_i \times \frac{j_j / l_{ij}}{\sum_{k \in I_n} j_k / l_{ik}}$$

where $l_{ij}$ is the spatial distance between node $v_i$ and $v_j$. In addition, the network traffic of each node is composed of two components: the periodical flow and the random flow [22]. Therefore, traffic merits using neural networks in order to discover the periodic feature.

After the network has been running for some time, the enhancement node can obtain the topology of the network. Then, each enhanced node takes the ordinary nodes closest to them as a node-set and collects the position information and traffic information of the ordinary node. After the collection of enhanced nodes is completed, the spatiotemporal distribution of traffic can be calculated according to Equation (1). In this section, the traffic
The traffic flow prediction strategy based on RNN combined with the DDPG method is shown (Figure 2.)

![Diagram](image)

**Figure 2.** The traffic flow prediction strategy based on RNN combined with the DDPG method.

The interaction process between the environment and the agent is considered as a Markov Decision Process (MDP) by the reinforcement model. The element in MDP is defined as $M = (S, A, R, P, \lambda)$, where the space of state, action, reward is $S$, $A$, $R$, respectively. The transition probability function is $P$, and the discount factor is $\lambda$ ($\lambda \in [0,1]$). According to the policy, the agent will decide an action under a different state [22]. The Q value is used in this model to evaluate the performance of the policy. Therefore, the Q value under $(s_t, a_t)$ is

$$Q(s_t, a_t) = E[\sum_{n=0}^{\infty} \lambda^n R(s_{t+n}, a_{t+n})]$$  \hspace{1cm} (2)

In online network usage, the Q value is more commonly expressed as

$$Q(s_t, a_t) = E[R_t + \lambda Q(s_{t+1}, a_{t+1})]$$  \hspace{1cm} (3)

In the proposed model, the loss function in DDPG is used to process the back propagation, which can be expressed as

$$L(\phi) = E[(Q(s_t, a_t | \phi) - \psi_t)^2]$$  \hspace{1cm} (4)

where the $Q(s_t, a_t | \phi)$ represents the Q value expressed by a set of parameters $\phi$ in the neural network. $\psi_t$ is

$$\psi_t = R(s_t, a_t) + \lambda Q(s_{t+1}, \mu(s_{t+1}) | \phi)$$  \hspace{1cm} (5)

The policy gradient of DDPG can be computed as

$$\nabla_\phi J = E[\nabla_\phi Q(s, \mu(s | \phi^0) | \phi^0)]$$

$$= E[\nabla_\phi Q(s, a | \phi^0) \nabla_\phi \mu(s | \phi^0)]$$  \hspace{1cm} (6)

Algorithm 1 shows the detailed DDPG method used in this section.
Algorithm 1 DDPG algorithm

Randomly initialize $\phi^\theta$ and $\phi^\phi$
Create target network $Q'$ and $\mu'$ with $\phi'^\theta \leftarrow \phi^\theta$ and $\phi'^\phi \leftarrow \phi^\phi$
Initialize replay buffer $R$

for episode = 1 to $M$:
Initialize a random process $N$ for action exploration
Receive the initial state $s_i$

for iteration $t = 1$ to $T$
do
Execute action $a_t$ and observe $r_t$ and $s_{t+1}$
Store $(s_t, a_t, r_t, s_{t+1})$ in $R$
sample a minibatch of $(s_t, a_t, r_t, s_{t+1})$ from $R$
set $\psi_t = r_t + \lambda Q'(s_{t+1}, \mu'(s_{t+1} | \phi'^\phi))$
update $\phi^\theta$ with
$$L = \frac{1}{N} \sum_i (y_i - Q(s_i, a_i | \phi^\theta))^2$$
update $\phi^\phi$ with
$$\nabla_{\phi^\phi} J = \frac{1}{N} \sum_i [\nabla_s Q(s_i, a_i | \phi^\phi) \nabla_{\phi^\phi} \mu(s_i | \phi^\phi)]$$
update target network with
$$\phi'^\theta \leftarrow \lambda \phi'^\theta + (1-\lambda) \phi^\theta$$
$$\phi'^\phi \leftarrow \lambda \phi'^\phi + (1-\lambda) \phi^\phi$$
end for
end for

As the traffic between nodes shows the periodic feature, the RNN is used as the input neural network. Figure 3 shows the structure of RNN. The input data is the traffic snapshot of the traffic distribution. Therefore, the output data can be treated as one of the input data in the next time step. The traffic condition in the next time step is predicted according to the information of past traffic patterns. The hyperparameters of the RNN are similar to [23]. So far, the traffic prediction between nodes is finished, and the enhanced node will use this information to make further routing decisions.

![Figure 3. The structure of RNN.](image)

3.2. Setup and Update of Node State

In this section, we proposed a node state awareness mechanism, processed at the enhanced nodes. Each enhanced node takes the ordinary nodes closest to them as a node-
set and collects node states. In the proposed MHSA-TFF, each enhanced node keeps a Node State Table (NST). The node states are stored as the style in Table 2.

Table 2. Link state table.

| Node | State  | Energy  | Traffic Load | Timestamp |
|------|--------|---------|--------------|----------|
| v_i  | Online | Medium  | Free         | t        |
| v_j  | Online | High    | Congested    | t        |
| v_k  | Offline| Low     | Busy         | t        |

It can be seen from Table 2 that the energy and traffic load of ordinary nodes are collected. There are three levels of energy and traffic load, low/medium/high, free/busy/congested, respectively. The online/offline represents whether the ordinary node returns the ACK packet. The commonly used hello packet strategy is adopted to achieve the link state table [24]. The enhanced node sends hello packets to nodes belonging to its ordinary node-set and receives the reported packet. When the enhanced node receives the message from its ordinary node-set, it updates NST. Algorithm 2 shows the process of the multi-hop state-aware scheme.

Algorithm 2 Multi-hop State-Aware Scheme

State Updating:
while true do
    broadcast hello packets to its ordinary node set
    wait $t_{\text{hello}}$
end while
if receive ACK within $t_{\text{ack}}$ then
    if ($t$ is up to date) then
        NST$(v_i).\text{state} \leftarrow \text{Online}$
    else
        drop the ACK message
    end if
else
    NST$(v_i).\text{state} \leftarrow \text{Offline}$
end if

Energy and Traffic Load Updating:
Receive reported packet
if ($t$ is up to date) then
    continue
else
    drop the reported packet
end if
if (NST$(v_i).\text{energy is not equal to energy}) then
    NST$(v_i).\text{energy} \leftarrow \text{energy}$
else if (NST$(v_i).\text{traffic load is not equal to traffic load}) then
    NST$(v_i).\text{traffic load} \leftarrow \text{traffic load}$
else
    drop the packet
end if

Based on the above information, the enhanced node has the traffic flow and the state of the ordinary node within the multi-hop. This information will be used to make the routing decision.
3.3. Routing Strategy Based on DDQN

It is mentioned in Section 3.1 that an agent is modeled as \( \{S, A, P, R\} \) in deep reinforcement learning. As the WSN routing strategy is combined with the deep reinforcement learning, the model \( \{S, A, P, R\} \) is constructed as the Table 3.

| Parameters | Definition | Action | Possibility | Reward |
|------------|------------|--------|-------------|--------|
| S          | State([s, v, NST]) | \( v_{next} \) | \( P_{next} \) | \( r \) |

where the source node is \( s \), the destination node is \( v_d \), and the current NST at the enhanced node. \( v_{next} \) represents the next-hop decided by the enhanced node. For example, if a packet needs to be routed from \( s \) to \( v_d \), the reward \( r \) is expressed as

\[
r = \begin{cases} 
  r_{success} & \text{if } v_{next} = v_d \\
  -(\alpha \cdot r_{offline} + \beta \cdot r_{low} + \gamma \cdot r_{congested}) & \text{if } v_{next} = v_d \\
  -\text{diff}(v_{next}, v_d) & \text{if } Other 
\end{cases}
\]

where \( \text{diff}(v_{next}, v_d) = l_{next,d}^2 \) represents the distance between the next hop node and the destination node, \( r_{success} \) represents success routing and \( r_{offline}, r_{low}, r_{congested} \) are the punishments for next hop offline/energy low/traffic congested events, respectively. \( \alpha, \beta, \gamma \) are the weight coefficients of the three events. The training process of DDQN is referred to [21].

Since the state, energy, and traffic load are considered in DDQN as the input parameters, the proposed routing strategy can respond to the network topology changes. The workflow of MHSA-TFF is shown in Algorithm 3.

Algorithm 3 The Workflow of the MHSA-TFFDDPG algorithm

1: For each enhanced node, do the prediction of the traffic between nodes.
2: Evaluate node states and update the NST
3: if the NST changes then
   Interacts with other enhanced nodes to update the network topology
end if
4: Load DDQN at the enhanced node
5: If an ordinary node has the routing requirement, input the \([s, v, NST]\) to the corresponding enhanced node.
6: The enhanced node outputs the next-hop \( v_{next} \) and sends to the ordinary node
7: The ordinary node transmits the packet to \( v_{next} \)

MHSA-TFF takes the multi-hop node state information as the input to the routing decision process, which is partially updated. The extensive wireless sensor network is divided into several small pieces by the setting of enhanced nodes. Each enhancement node is responsible for gathering information within the multi-hop region and making routing decisions within this multi-hop region.

4. Simulation Results

The input part of MHSA-TFF was completed by Python, and the deep learning model was built by TensorFlow with the following experimental environment: AMD Ryzen R5 @ 3.5 GHz × 4, 16 G RAM, gtx1060 GPU, 6 G VRAM.

Table 4 presents the detailed simulation parameters and experimental environment of the wireless sensor network. A total of 120 ordinary nodes and 8 enhanced nodes are randomly deployed in the 150\( \text{m} \times 150\( \text{m}\) simulation area. The direct graph \( G = (V, E) \) is
generated based on the geographic location information. The communication radius of ordinary nodes and enhanced nodes are limited to 20 m and 50 m, respectively. When the scene is initialized, a probability value \( p \in [0,1] \) will be generated for the nodes within the communication range. If \( p \) is greater than 0.4, a two-way communication link will be established, and each node will be guaranteed to have at least one two-way communication link.

**Table 4. Simulation parameters and environment.**

| Parameters                        | Value       |
|----------------------------------|-------------|
| Number of Ordinary Nodes         | 120         |
| Number of Enhanced Nodes         | 8           |
| Simulation Area                  | 150 m × 150 m |
| Energy cost per TX               | 5 nJ/bit    |
| Energy cost per RX               | 5 nJ/bit    |
| Initial Energy of each Node      | 1 J         |
| Simulation Time Steps            | 1000        |

The AODV in [6] and the Flooding in [10] are chosen as the baseline methods. We compared the performance of the MHSA-TFF with that of AODV and flooding in four indicators, which are the transmission delay, the average routing length, the residual energy state, and the number of dead nodes. The average value of 150 independent replicates for each indicator was taken in our simulation to obtain a statistical average value. In the simulation, we assume that the network topology is basically unchanged under the first 400 step sizes, which means that there are no emergency events such as node damage or shutdown. In the following 600-step size, we assume that 5% of ordinary nodes are damaged at random every 200 steps (first happens at the 400 step mark), but we still guarantee that there is at least one communication link for the undamaged ordinary nodes. The above settings were used to simulate algorithm performance under network topology changes.

Figure 4 shows the comparison on the transmission delay. Due to the passive routing strategy of AODV, the broadcasting scheme is used in AODV to establish the routing path for data transmission. Therefore, this leads to an increasing in transmission delay when the network topology changes. The MHSA-TFF can automatically adapt to the changes in network topology. Thus, the transmission delay of MHSA-TFF increases with the occurrence of ordinary node corruption events but is better than that of AODV and flooding.
Figure 4. The comparison on the transmission delay.

Figure 5 shows the comparison on the average routing length. Before the change of network topology, it can be seen that the routing length values of the two strategies are similar. After the change of network topology, the average routing length of MHSA-TFF is smaller than that of AODV. This proves that the MHSA-TFF can resist the change of network topology to some extent through the application of global topology information and neural networks. Moreover, the results of the flooding routing strategy are not recorded and displayed. The reason is that if the flood routing strategy is adopted, the route length is always equal to the number of edges at each step in the network.

Figure 5. The comparison on the average routing length.

Figures 6 and 7 show the comparison on the total residual energy and the number of dead nodes, respectively. These two figures prove that the MHSA-TFF strategy has good energy utilization efficiency. This is because our routing strategy runs on enhanced nodes, while ordinary nodes only need to receive signaling and forward packets without making
routing decisions. In addition, the use of deep learning reduces the frequency of recalculated routes. This scenario can be used in the industrial internet, industrial IoT, and intelligent agriculture.

![Figure 6. The comparison on the total residual energy.](image)

![Figure 7. The comparison on the number of dead nodes.](image)

5. Conclusions

In this paper, a deep reinforcement learning-based multi-hop state aware routing strategy was proposed for wireless sensor networks. We used the DDPG strategy combined with the RNN network to estimate inter-node business traffic. Furthermore, we proposed a multi-hop node state awareness mechanism based on enhanced nodes. By reporting and collecting energy and traffic load information for ordinary nodes, enhanced nodes can train the DDQN network based on this information and make routing decisions based on the current multi-hop node state. The advantage of the proposed strategy is that it can cope with changes in the network topology state. The simulation results prove the feasibility of the strategy.
Author Contributions: Conceptualization, C.W. and A.Z.; software, A.Z. and M.S.; validation, Z.L. and Y.C.; data curation, A.Z., M.S., and Y.C.; writing—original draft preparation, A.Z. and J.W.; writing—review and editing, M.S., C.W.; project administration and funding acquisition, C.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (NSFC) under the Grant No. 61801033.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. Future Gener. Comput. Syst. 2013, 29, 1645–1660, doi:10.1016/j.future.2013.01.010.
2. Chen, L.; Thombre, S.; Jarvinen, K.; Lohan, E.S.; Alen-Savikko, A.K.; Leppakoski, H.; Bhuiyan, M.Z.H.; Bu-Pasha, S.; Ferrara, G.N.; Honkala, S.; et al. Robustness, Security and Privacy in Location-Based Services for Future IoT: A Survey. IEEE Access 2017, 5, 8956–8977, doi:10.1109/access.2017.2695525.
3. Da Xu, L.; He, W.; Li, S. Internet of Things in Industries: A Survey. IEEE Trans. Ind. Inf. 2014, 10, 2233–2243, doi:10.1109/tii.2014.2300753.
4. Brewster, C.; Roussaki, I.; Kalatzis, N.; Doolin, K.; Ellis, K. IoT in Agriculture: Designing a Europe-Wide Large-Scale Pilot. IEEE Commun. Mag. 2017, 55, 26–33, doi:10.1109/mcom.2017.1600528.
5. Agrawal, D.P. Embedded Sensor Systems. In Embedded Sensor Systems; Springer: Singapore, 2017; pp. 35–63.
6. Pereira, N.C.V.N.; De Moraes, R.M. A comparative analysis of AODV route recovery mechanisms in wireless Ad Hoc networks. IEEE Latin Amer. Trans. 2009, 8, 1–6, doi:10.1109/latincom.2009.5304742.
7. Manjunath, M.; Manjaiyah, D. Spatial DSDV (S-DSDV) routing algorithm for mobile ad hoc network. In Proceedings of the 2014 International Conference on Contemporary Computing and Informatics (IC3I), Mysore, India, 27–29 November, 2014; pp. 625–629.
8. Al-Mekhlafi, Z.G.; Hassan, R. Evaluation study on routing information protocol and dynamic source routing in Ad-Hoc network. In Proceedings of the 2011 7th International Conference on Information Technology in Asia, Sarawak, Malaysia, 12-14 July, 2011; pp. 1–4.
9. Usop, N.S.M.; Abdullah, A.; Abidin, A.F.A. Performance evaluation of AODV, DSDV & DSR routing protocol in grid environment. IJCSNS Int. J. Comput. Sci. Netw. Secur. 2009, 9, 261–268.
10. Subitha, A. Combined Single and Flooding Routing using Time-tag in Intermittently Connected Mobile Networks. In Proceedings of the 2019 International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC); Institute of Electrical and Electronics Engineers: New York, NY, USA, 2019.
11. Fan, K.; Lu, J.; Sun, D.; Jin, Y.; Shen, R.; Sheng, B. Failure Resilient Routing via IoT Networks. In Proceedings of the 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing ( CPSCom) and IEEE Smart Data (SmartData), Exeter, UK, 21–23 June 2017; pp. 845–850.
12. Mohan, P.M.; Gurusamy, M.; Lim, T.J. Dynamic Attack-Resilient Routing in Software Defined Networks. IEEE Trans. Netw. Serv. Manag. 2018, 15, 1146–1160, doi:10.1109/tmsm.2018.2846294.
13. Kato, N.; Fadlullah, Z.M.; Mao, B.; Tang, F.; Akashi, O.; Inoue, T.; Mizutani, K. The Deep Learning Vision for Heterogeneous Network Traffic Control: Proposal, Challenges, and Future Perspective. IEEE Wirel. Commun. 2017, 24, 146–153, doi:10.1109/mwc.2016.160317wcc.
14. Fadlullah, Z.M.; Tang, F.; Mao, B.; Kato, N.; Akashi, O.; Inoue, T.; Mizutani, K. State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow’s Intelligent Network Traffic Control Systems. IEEE Commun. Surv. Tutorials 2017, 19, 2432–2455, doi:10.1109/comst.2017.2707140.
15. 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 Wirel. Commun. 2018, 25, 154–160, doi:10.1109/mwc.2017.1700244.
16. Mao, B.; Fadlullah, Z.M.; Tang, F.; Kato, N.; Akashi, O.; Inoue, T.; Mizutani, K. Routing or Computing? The Paradigm Shift Towards Intelligent Computer Network Packet Transmission Based on Deep Learning. IEEE Trans. Comput. 2017, 66, 1946–1960, doi:10.1109/tc.2017.2709742.
17. Guo, X.; Lin, H.; Li, Z.; Peng, M. Deep-Reinforcement-Learning-Based QoS-Aware Secure Routing for SDN-IoT. IEEE Internet Things J. 2019, 7, 6242–6251, doi:10.1109/jiot.2019.2960033.
18. Huang, R.; Ma, L.; Zhai, G.; He, J.; Chu, X.; Yan, H. Resilient Routing Mechanism for Wireless Sensor Networks With Deep Learning Link Reliability Prediction. *IEEE Access* 2020, 8, 64857–64872, doi:10.1109/access.2020.2984593.

19. Fadlullah, Z.M.; Tang, F.; Mao, B.; Liu, J.; Kato, N.; Akashi, O.; Inoue, T.; Mizutani, K. On Intelligent Traffic Control for Large-Scale Heterogeneous Networks: A Value Matrix-Based Deep Learning Approach. *IEEE Commun. Lett.* 2018, 22, 2479–2482, doi:10.1109/lcomm.2018.2875431.

20. Lakhina, A.; Papagiannaki, K.; Crovella, M.; Diot, C.; Kolaczyk, E.D.; Taft, N. Structural analysis of network traffic flows. In *SIGMETRICS 2004 / PERFORMANCE 2004: Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems*; Association for Computing Machinery: New York, NY, USA, 2004; pp. 61–72.

21. Wang, W.D.; Wang, C.; Wang, H.W.; Xu, P. Dynamic cache allocation routing strategy of Internet of things satellite node based on traffic prediction. *J. Commun.* 2020, 41, 25–35, doi:10.11959/j.issn.1000-436x.2020038.

22. Silver, D.; Lever, G.; Heess, N.; Degris, T.; Wierstra, D.; Riedmiller, M. Deterministic policy gradient algorithms. In Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 21–26 June 2014; pp. 387–395.

23. Ramakrishnan, N.; Soni, T. Network Traffic Prediction Using Recurrent Neural Networks. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17-20 December, 2018; pp. 187–193.

24. Fortz, B.; Thorup, M. Internet traffic engineering by optimizing OSPF weights. In *Proceedings of the IEEE INFOCOM 2000*, Tel Aviv, Israel, 26-30 March, 2002; pp. 519–528.