

# A Multi-Agent System for Environmental Monitoring Using Boolean Networks and Reinforcement Learning

Hanzhong Zheng<sup>1</sup> and Dejie Shi<sup>2,\*</sup>

<sup>1</sup>Department of Computer Science, The University of Pittsburgh, Pittsburgh, PA 15213, USA

<sup>2</sup>School of Computer and Information Engineering, Hunan University of Technology and Business, Changsha, 410205, China

\*Corresponding Author: Dejie Shi. Email: zpy\_s7@163.com

Received: 10 February 2020; Accepted: 28 June 2020

**Abstract:** Distributed wireless sensor networks have been shown to be effective for environmental monitoring tasks, in which multiple sensors are deployed in a wide range of the environments to collect information or monitor a particular event. Wireless sensor networks, consisting of a large number of interacting sensors, have been successful in a variety of applications where they are able to share information using different transmission protocols through the communication network. However, the irregular and dynamic environment requires traditional wireless sensor networks to have frequent communications to exchange the most recent information, which can easily generate high communication cost through the collaborative data collection and data transmission. High frequency communication also has high probability of failure because of long distance data transmission. In this paper, we developed a novel approach to multi-sensor environment monitoring network using the idea of distributed system. Its communication network can overcome the difficulties of high communication cost and Single Point of Failure (SPOF) through the decentralized approach, which performs in-network computation. Our approach makes use of Boolean networks that allows for a non-complex method of corroboration and retains meaningful information regarding the dynamics of the communication network. Our approach also reduces the complexity of data aggregation process and employ a reinforcement learning algorithm to predict future event inside the environment through the pattern recognition.

**Keywords:** Multi-agent system; reinforcement learning; environment monitoring

## 1 Introduction

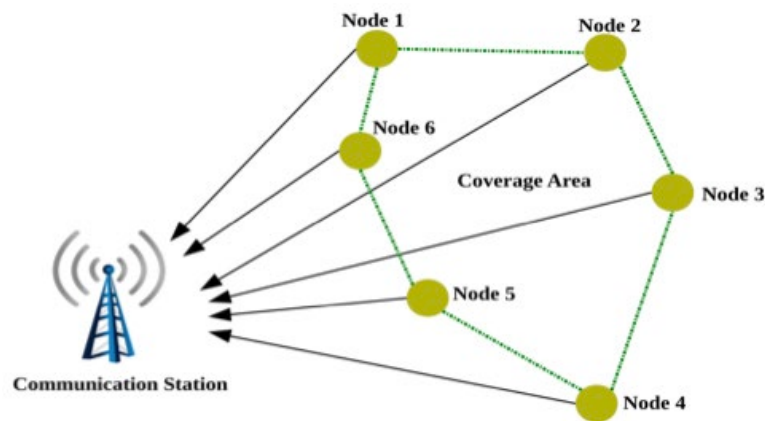
Using wireless multi-sensor networks to collect environmental information has been the site of much attention among computer science researchers in recent years. Environmental monitoring refers to the processes that are required to monitor the conditions of the environment in terms of the quality of air and water, soil, temperature and other factors. The purposes of using wireless multi-sensor networks can be classified into two categories: environmental data collection or particular event monitoring. The existing and potential applications in the field of environmental data collection include endangered species protection [1], zoology study [2], pollution detection [3], marine monitoring [4] and many more [5]. Researchers usually integrate different types of sensors to obtain diverse in the environmental data collection. There are also many existing applications that monitor particular event using different sensors in the network such as Hallway monitoring [6], intruder detection [7], forest fire detection [8] and flood prediction [9]. In event monitoring, researchers focus on the data aggregation that analyzes the environmental information using one type of sensor. In their wireless sensor networks, each sensor is usually represented as a node and each node is able to communicate with a base station, which stores and



processes all the information.

Due to the extreme inherent complexities of these tasks, the success of such applications can be primarily attributed to the communication ability of sensor network to monitor the large-scale environment. Monitoring the complex and dynamic large scale environment requires substantial number of sensors. Wireless sensor network can easily generate massive and multi-dimensional data from these sensors through continuous environmental monitoring task. Unstructured environmental data is very difficult to analyze and recognize data patterns. In addition, the sensor network also faces the data transmission failure problem.

The current research on environmental monitoring mostly uses a centralized approach, where nodes collect the data from separate locations and send the data across the whole network to a centralized server. The server has a high computational power and big database to store, manipulate, and aggregate the data from each node in the network. Other researchers concentrate on the integration of different types of sensors and camera into the network to provide diverse environmental information [4] or leave the interactions and dynamics of the greater network completely unexplored. Therefore, the current wireless sensor networks cannot utilize all the resources and there are many drawbacks and disadvantages in their structure.



**Figure 1:** The traditional model of centralized wireless sensor network

Fig. 1 demonstrates the traditional model of wireless sensor network using 6 sensors deployed in the environment. The base station in the network is responsible for all computational processes. The solid edges in the graph represents the communication between each node and the base station. Each node only has sensor components, focuses on sensing the environment, and transiting the data to the base station. However, node 2 and node 3 are the furthest nodes from the base station in the network. The data transmission between them and the base station needs to travel across the whole network. The larger area that the network wants to cover, the longer distance the data has to travel and the higher cost it can generate.

The failure of the base station directly can cause the failure of environmental monitoring task and the rest of the system is no longer valuable. When the number of sensor nodes in the network dramatically increases, the number of communication in the network will correspondingly increases. In this paper, using the idea from the mathematical field of dynamic systems, we propose to develop a wireless multi-sensor network for environmental monitoring where each node has the ability to aggregate and store the information. We employ a Boolean network that is a particular kind of dynamical system, to model information exchange and aggregation during a monitoring task using a simple Boolean rule. Particularly, we concentrate on modelling the interactions between the sensors, and study the dynamics produced from those interactions during an environmental monitoring task.

We conduct experiment in a simulated environment, in order to study the scale ability of area coverage and robustness to the network failure. Our experiment results indicate that our system can solve these constrains, especially using the reinforcement learning methods on each node. We conduct

experiments in the simulation to show: the sensors are able to reach a consensus regarding an event. Our model is desirable when operating error-prone sensors which are common and inexpensive, and are often attractive for large-scale multi-sensor systems.

## 2 Related Work

The common usage of the wireless sensor network is the surveillance and event monitoring. Hammoudeh [10] proposes an idea of designing a wireless sensor network to deal with the challenges of increasing illegal immigrants. Kishino [1] establishes a centralized wireless sensor network for monitoring endangered fish using wireless sensors that are spreading in the water by measuring dissolved oxygen, water temperature, air temperature, humidity, and illuminance. Kridi et al. [2] build a wireless sensors network system for monitoring the pre-swarmer colony behavior and develop a predictive algorithm based on pattern recognition using clustering data mining techniques on bees' cyclical daily behaviors. The high temperature, lack of food, pressure and humidity change can cause the bees' swarming behavior that commonly brings economic loss to farmers. Their system also faces the problems of data transmission costs. The costs are exponentially increasing when trying to cover larger areas.

The wireless sensor network also has been applied in monitoring human living environment. Baumgartner et al. [6] builds a sensor network to monitor the activities in the hallway with 180 load sensors connected to 30 wireless sensor nodes. They deploy the load sensors with floor tiles and sensors are able to share the information between each other. Each sensor node is connected with the PC that is responsible for collecting and aggregating the environmental data from the sensor nodes, then, the aggregated data is visualized and can reflect the current situation of the hallway. In [11], Nasipuri et al. build a centralized wireless sensor network that is deployed in a substation for monitoring the potential problems such as circuit breakers, transformers and transformer bushings. Nodes in the system are able to communicate using a multi-hop mesh network, which uses a dynamic link-quality based on routing protocol. The sensor network transmits all sensor data to a base station that performs data analysis and visualization. The development of the wireless sensor network has also been used for monitoring noise pollution monitoring. Silvia Santini et al. [12] built a system using wireless sensor network that is based on the Tmote invent prototyping platform. They also developed a system based on tiny LAB, which is a Matlab-based tool that enables real-time acquisition, processing and visualizing the data collected by the wireless sensor network. In their system, each sensor is able to provide enough actual data on noise sources and the system is able to assess the noise pollution levels using their predefined noise indicator that is a mathematical formula. However, the server needs to have computational power. The entire network has the high probability of Single Point of Failure. In Peckens et al. [13] study, a wireless sensing unit was developed that possesses the same functionality. The wireless sensing unit utilizes on-board data processing techniques to monitor noise by computing equivalent continuous sound levels, which effectively minimizes data transmission and increases the overall longevity of the node.

Machine learning recently has been applied to the field of environmental monitoring for dealing with large amount of data. In the application of flood protection [14], they developed an Artificial Intelligence (AI) component that integrated into an Early Warning System (EWS) platform of the Urban Flood project for the flood protection. The machine learning is able to detect the abnormal behavior of the object and to provide early indicators for the decision support system. Moreover, machine learning recently also has been adopted to help agents and robots to make better decisions. Dike conditions are monitored by the sensors in the network and environmental conditions are the input data, which are analyzed by the classification Neural Clouds (NC). Neural Clouds combines Advanced K-Mean clustering algorithm (AKM) and extended Radial Basis Functions (RBF). AKM modifies k-means algorithm with an adaptive calculation of optimal number of clusters. In this paper, we use a reinforcement learning approach on a wireless sensor network. Each sensor in the network is an agent that is able to interact with surrounding environment and use the "feedback" to gradually improve its behavior. However, in a system that requires sensors to make decision in a highly dynamic environment. The reinforcement learning has its own advantages in the long-term planning.

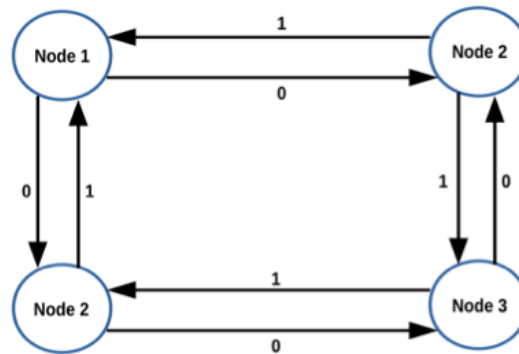
### 3 Method

#### 3.1 Boolean Network and In-Grid Computation

Boolean networks have been used for modeling networks in which the node activity can be described by states, 1 and 0. Each node is updated based on its logical relationships with other nodes. The notion of Boolean network was first explored as a model in computational genomics, in an attempt to capture the mechanics of gene regulation in a formal way. Boolean network is a discrete system that are capable of exhibiting complex patterns and behaviors in spite of the simplicity of their fundamental structure. Moreover, Boolean network is easy to understand and manipulate. In this work, each node also possesses a rule by which its state is updated. We note that in the context of our research, this rule is equivalent for all nodes. The network uses the mathematical model to calculate each sensors observation.

The design of the wireless sensor network is to build sensor grids that is able to monitor certain events occurring in a wild area and reduce the cost of implementing the task of environmental monitoring. Each sensor grid contains four sensors that are capable of communicating with other sensors in the same grid. Each sensor aggregates the information from other sensors in the grid. Then, each grid eventually outputs a decision on the occurrence of the event based on the information coming from the four sensors in the grid. Within the wireless sensor network, we establish a Boolean network that enables the environmental information to be represented by Boolean values without losing any valuable information. Using a Boolean network modeling approach similar to that of previous research [15].

Fig. 2 describes an example of a Boolean network. Each sensor grid in the network is constructed by such a Boolean network. This particular Boolean network consists of 4 sensor nodes, each with states  $r_{ri}$  with  $i = \{1, \dots, 4\}$  and the edges containing the value of their state that is sent to its neighboring robot. We concentrate on a particular environmental task, where the goal of each sensor is to accurately detect whether the event has occurred in its environment. This model can be applied to fire detection, animal tracking, or sudden reactions (chemical or other).

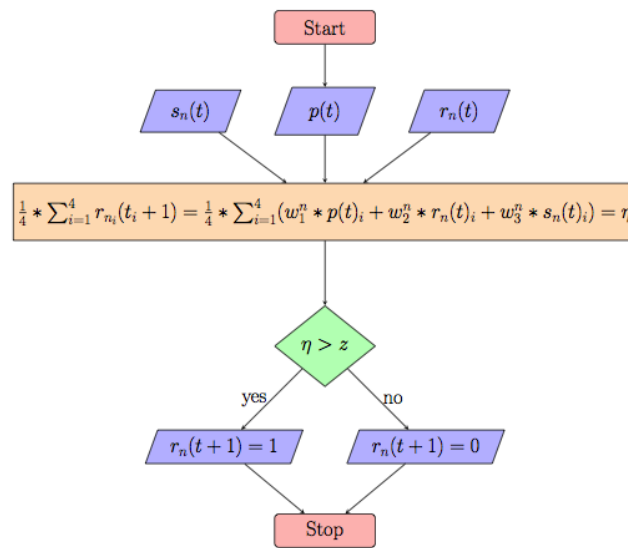


**Figure 2:** the Boolean network in one grid of the wireless sensor network at one tick count. The nodes represent the sensor agent, while the edges represent the communication flow between two sensor agents

Each sensor has a specific state. A state is assumed to take on only two values, 1 (implying that the event happened) and 0 (event did not happen). These states are therefore conveniently applied to Boolean network. The states of the sensor are influenced by the current sensor reading of the agent, the last aggregated value of the state in the grid, and the agents own state history. Each sensor is responsible for aggregating the states received from the other sensor agents, together with its own past state. Then the whole grid eventually outputs a value that is used to determine whether the grid believes the event is occurring. More precisely, the grid's state set to 1 at the next "Tick Count" if the final aggregated value has surpassed a given threshold  $z$ . Each sensor is independent to other sensors. It is responsible for calculating its own weighted value about the event through its current sensor reading (received by its own sensor components), its own past-states, and the aggregated state other sensors. Until each sensor finishes updating its latest information about current event, then the grid outputs a final aggregated value for the

occurrence of the event and each sensor will update to the final value. This update procedure in each agent has been mathematically expressed in Eq. (1), where  $p(t)$  represents the current sensor reading values of the sensor in the network,  $r_n(t)$  represents the average aggregated value of the event in the grid, and  $s_n(t)$  represents the previous determination of the occurrence of the event calculated by the grid. For the threshold value  $z$ , we note:  $z \in [0, 1]$ , such that  $\omega_1^n + \omega_2^n + \omega_3^n = 1$ . These weight values are therefore used to express preference between the three inputs to the function, allowing for a wide range of dynamics to be given by an Eq. (1).

$$r_n(t+1) = \begin{cases} 1 & \omega_1^n \times p(t) + \omega_2^n \times r_n(t) + \omega_3^n \times s_n(t) > z \\ & \Sigma \omega_i^n = 1 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$



**Figure3:** Flow of the decision making process in each grid of wireless sensor network

The underlying process for each sensor agent’s decision making regarding the event occurrence is given by Algorithm 1 and illustrated in Fig. 3. The sensor generates its initial state, calculate an aggregated state, and obtain updated sensor readings. In the example of the wireless sensor network, the  $\eta$  in the Fig. 3 represents the average of the sum of the all aggregated values of each sensor agents calculated using Eq. (1). Comparing  $z$  with a pre-defined threshold value to determine the final aggregated value of the whole grid and transmit the value to each sensor agent. When  $\eta$  is greater than threshold  $z$ , the final aggregated result about the event is 1, otherwise is 0. We assume that the sensor use a well-defined procedure based on their sensor components to determine whether or not any event for which they are monitoring has taken place, which outputs a binary decision (e.g., a yes or no, or 1 or 0 answer). Finally, each sub-grid makes the final decision on the occurrence of the event and each sensor receives the decisions.

```

Data: Initial state  $s(0)$ 
Result:  $p(t)$ 
while not finished with the monitoring task do
  for each sensor agent in the Boolean network do
    monitor the event in the network;
    if the event is within gridi then
      for each sensor agent in gridi do
        update its  $p(t)_i$ ;
        send  $p(t)_i$  to other sensor agents in the gridi;
        calculate  $r_n(t + 1)$  using Equation 3.1 ;
        send  $r_n(t + 1)$  to other sensor agents in the gridi;
        calculate the grid final aggregated value using the
          procedure illustrated in Figure 4;
        update  $r_n(t + 1)$  using the grid final aggregated value;
      end
    end
  end
   $t = t + 1$ ;
end

```

**Algorithm 1:** Algorithm used by sensor agent  $n$  in wireless sensor network

The procedure of each sensor is illustrated in Algorithm 1. To be clear, 1 or 0 values are the sensors final aggregated value, and 1 represents that the system believes that the event occurs, while 0 represents that the system does not believe the event occurs. Furthermore, we also can see the advantages of the network from its algorithm. The network firstly determines which grid the object is at so that only the sensor agents that in that grid need to communicate with other agents that are in the same grid instead of that sensor agents in the whole network needs to communicate with others. Using this approach, the network can greatly reduce communication cost through decreasing communication times between sensor agents and Single Point of Failure.

### 3.2 Reinforcement Learning

Learning ability in applications has been becoming more and more popular, especially for a complex task. The increasing the complexity for an agent based system, utilizing the learning behaviors becomes very necessary for the system [16,17]. Each sensor is able to improve their performance on detecting the occurrence of the event through the machine learning. In our proposed method, the accurate weight values in the mathematical equation are very critical as they directly influence the accuracy of the detection of the occurrence of the event. Although we can find the best weights value in the mathematical equation through conducting many experiments and tests, we have to face the high costs brought by those experiments and tests. In order to find the most appropriate weights values, we introduce the reinforcement learning in the decision making procedure of each sensor agent.

Reinforcement learning can maximize some notions of cumulative reward of software gents through manipulating their actions in an environment. In many other definitions, reinforcement learning is also referring to approximate dynamic programming. It aims to improve agent behaviors by helping them find optimal solutions. In our system, the performance of the wireless sensor network is determined by the decision making procedure in each grid. We initially assume that each input plays the same importance in determining the aggregated value of the event so that the default weight value for each of them is 1/3. However, the aggregation method has a very low performance on detecting the event using the default

weight values. Hence, we applied reinforcement learning in the decision making procedure to find out the weights value of the equations that can maintain a high performance on event detection.

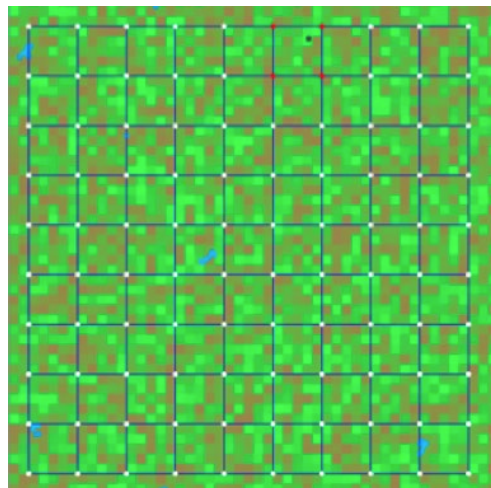
When the grid computes the average aggregated value, we use it to compare with the real event value. If there is a big difference between the real event value and the average aggregated value generated by the grid, the network has a very low performance on monitoring the event and it receives a punishment, then we need to modify the decision making components of the system. On the contrary, if there is a small or no difference between the real event value and the average aggregated value generated by the grid, the network successfully detects the event within the network and it receives a reward. Then, the network does not need to modify its behaviors in order to efficiently detect the event in the network. The amount of punishment and reward that the network receives is dependent on the reward function. In the simulation, we can see the dynamic changes on the weights.

## 4 Experiment

### 4.1 Test Environment

The wireless sensor network was simulated using the Repast Symphony 2.4.0, which is an agent-based modeling and simulation platforms. Repast Symphony is a plug-in tool on the Eclipse. It enables to develop extremely flexible models of interacting agents on work stations and computing clusters. Eclipse is an integrated development environment (IDE) used in computer program, and is the most widely used Java IDE.

Fig. 4 shows the simulation of the wireless sensor network. There are four types of agents: Grass, River, Sensor, and Object interacting with each other. The Grass and River constitute the simulated environment, which greatly models the prairie environment. The task of Grass and River agents are to interfere the performance of the network in order to test its robustness. The object agent is the simulated object that is used to test whether the wireless sensor network can successfully detect the occurrence of the event. As a result, the position of the object agent is constantly changing within the environment. In the simulation, the object agent is represented using black point on the two-dimensional plane. The black point is randomly moving around within the network. Sensor collects environmental information and process the current information and store the historical information. The Object agent is the simulated object that moves around within the network. This simulation enables to model the real natural environment.



**Figure 4:** The simulated environment of the wireless sensor network

Each sensor agent is able to make its own decision on the occurrence of the event based on its collected environmental information from sensor components. Then, each sensor will send its decision



with other agents, which are in the same sensor grid. Each sensor in the grid aggregates the information from others and itself using the method. Eventually, the grid will output an aggregated decision about the occurrence of the event. In the simulated wireless sensor network, each grid is consisted by four sensors. The sensor agent is represented using white points on a two-dimensional plane.

## 5 Experiment Results

We conducted simulation experiments to test our Boolean network model with a large number of simulated sensors and to provide a more in-depth analysis of our model. Our simulation experiment involved the development of a simulation of a multi-sensor team applied to an abstract environmental monitoring task. The purpose of designing such a simulation is to prove the correctness of our model, which would be difficult and costly to conduct conventional physical implementations.

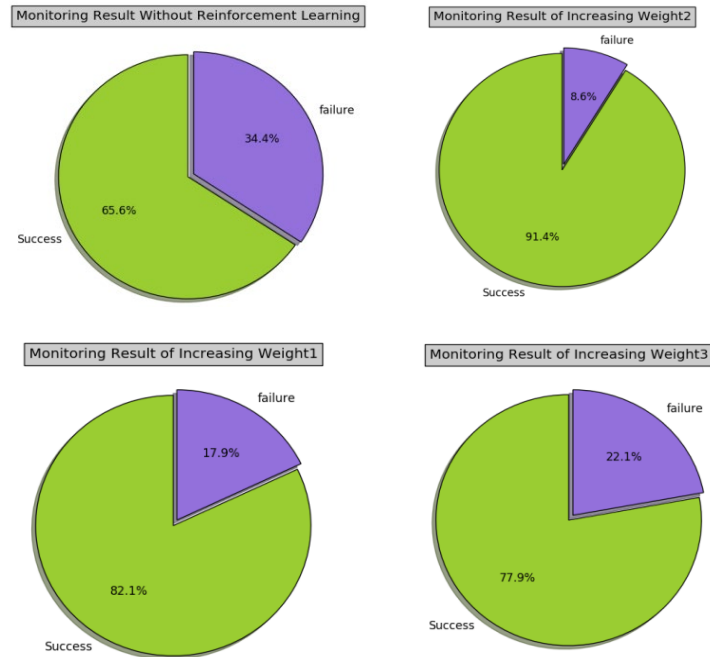
In our simulation, we also implement that the inaccurate readings of each sensor, so there is a small probability that each sensor will provide a false reading. For simplicity and for the purpose of illustration of our method, we assume that the information received by the sensor agents is represented by a Bernoulli random variable. We use 100 sensor agents. Through many previous experiments, the most appropriate threshold  $z$  value is 0.7, which allows the system to have the best performance on detecting the object. The initial values of  $p(t)$  and  $r(t)$  can be updated by current sensor reading and calculation. For the purpose of simplicity, we initial assigns(t) with value of 1 without influencing next result.

Eq. (2) shows the accuracy of the aggregated state values for the four tested scenarios. The complexity of our experiments originates from the object being occasionally discarded because of the failed sensor detection or the object being out of the sensor detection range. When the sensor fails to detect the object that is actually within the range of the network, a sensor must depend on the observations of the others, as well as their own histories, for their sensory data. For all four scenarios, the object randomly moves within the network for 1000 ticks count. Under the first scenario, we did not introduce the reinforcement learning to the system. We design two scenarios to test our system performance: (1) we study the effect of increasing one weight, while keeping others the same. (2) the change of system performance with and without the reinforcement learning. The amount of increasing and decreasing on weights depends on the amount of reward or punishment the agent receives at each ticks count.

$$Q[i] = \left( \frac{\text{Optimal Reward}}{\beta} - |real_i - r_i(t+1) \times \gamma| \right) \quad (2)$$

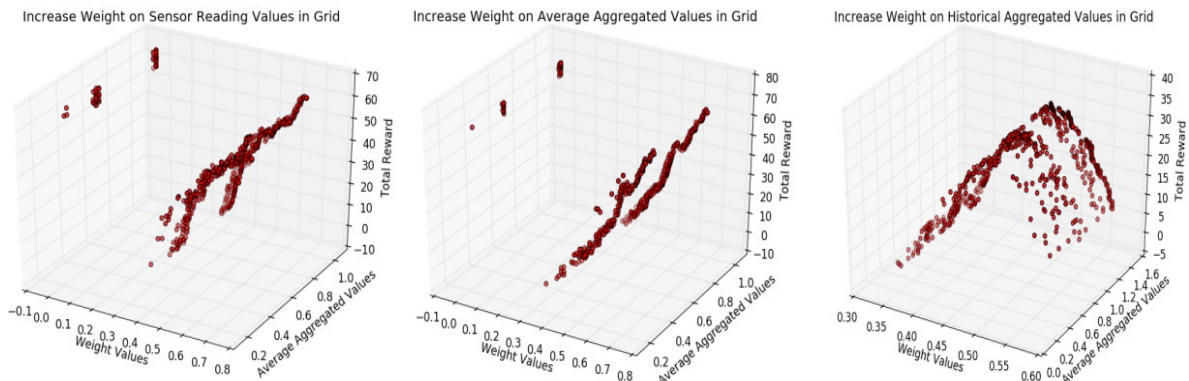
We compared the performance of those two scenarios by calculating the accuracy percentage of detecting the event within the range of the network. For the method with introducing the reinforcement learning, we also compared the performance of the system by determine which weight that we increased its value has the best performance using the same way of calculating the accuracy percentage (showed in Fig. 5). The reward function determines the amount of reward or punishment each agent should receive.  $Q[i]$  is the amount of the reward the agent receives at ticks count  $i$ . The optimal Reward is the maximum amount of reward that the agent can receive is real event value at ticks count  $i$  and  $r_i(t+1)$  is the grid aggregated value at ticks count  $i$ .  $\beta$  is the number of parameters needed to calculate  $r_i(t+1)$ . In our experiments,  $\beta$  is 3 because there are three parameters in the calculation.  $\gamma$  is a threshold value that is used to control the amount of the reward receives at each ticks count. When the agent receives higher previous reward or punishment,  $\gamma$  correspondingly has a higher value in the calculation of the current reward or punishment.



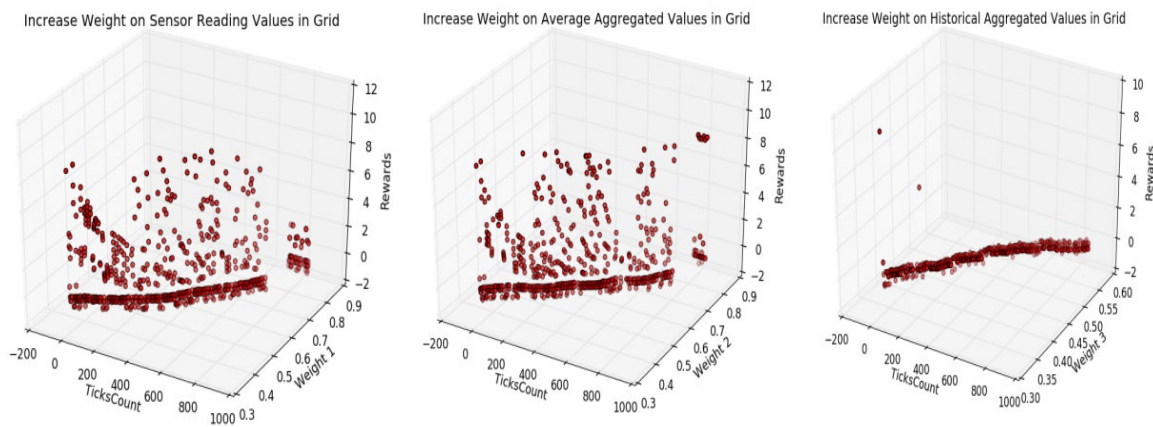


**Figure 5:** The average accuracy of the wireless sensor network with reinforcement learning and without reinforcement learning, and alternating different weight values

In addition, in the test scenario of increasing the weight values in the equation of calculating the aggregated value of the grid, we still need additional information to determine the best model for the network to have the best performance on detecting the event within the network. We visualized the results of increasing the weight value in Fig. 6 and Fig. 7. Figs. 6 and 7 show the change the total reward values of the network receives during the 800 ticks count. Fig. 6 is a three-dimensional scatter plot that allows us to see the relationship between the ticks count, weight values, and individual rewards that the network receives at each ticks count. The x, y, z-axes represent the ticks count, weight values, and individual rewards. Although the majority of the points are gathered at the bottom part of the graphs, meaning that they received low reward or punishment at current ticks count, there are certain number of points are at the higher part of the graphs of increasing the weight on current sensor reading and grid aggregated values, meaning that they received a high reward at that ticks count. The network with increasing the weight on grid aggregated values receive more number of times of high reward. Fig. 7 represents the relationship between the weight values, the grid aggregated values, and the total reward values of the network. The x, y, z-axes represent the weight values, the grid aggregated values, and the total reward values of the network. From Fig. 7, increasing weights on current sensor reading and grid aggregated value demonstrate an increasing pattern of total reward values, and increasing weight value on grid historical aggregated value demonstrates a decreasing pattern after increasing for a short time. Furthermore, although increasing weights on current sensor reading and grid aggregated values both have the increasing pattern of total reward values, the later model has a higher rate in the increasing of its total reward values. In addition, it also has the higher number in its total amount of reward values and a higher accuracy of detecting the moving object. The network with increasing weight on grid aggregated value has a better performance than the network with increasing weight on current sensor reading. It is reasonable because detecting the moving object only depends on a single sensor agent has the higher probability of failing detect the object because of the detecting range. Therefore, we should increase the weight on grid aggregated values to allow the system efficiently monitor the environment, since it demonstrates the best behaviours from the experiment results.



**Figure 6:** The total reward values of the network receives in the scenarios of increasing weights on  $p(t)$ ,  $r_n(t)$ , and  $s_n(t)$



**Figure 7:** The weight values of the network in the scenarios of increasing  $p(t)$ ,  $r_n(t)$ , and  $s_n(t)$

## 6 Conclusion

In this paper, we described a Boolean network based multi-sensor environmental monitoring system. The task of environmental monitoring can easily generate complex data that requires a machine with high computational power and efficient algorithm to process. In our system, we incorporated Boolean network that only has two states: yes or no, 0 or 1. Boolean network dramatically reduces the complexity of the environmental data while it still keeps the useful information. We incorporate Boolean network to reduce the complexity of environmental data with useful information. Besides, in-grid computation utilizes distributed computing fashion that greatly saves system's resources. Our system also uses the reinforcement learning that is able to dynamically improve the behaviors of the system through giving a reward or punishment based on the average aggregated values.

When the system demonstrates a poor behavior on detecting the event, it receives a penalty and the system immediately modifies the weight values in the decision making component. When the system accurately detects the event, it receives a reward and add the reward into the total reward the system receives. As a result, the system initially demonstrates a very poor performance on detecting the event but it is still able to gradually improve its behaviors and eventually can accurately monitor the event within the network. From the simulation results, we demonstrate that our network combines Boolean network and reinforcement learning to give the accurate results. The network is able to accurately detect the moving object when we increase the weight on grid aggregated value while decreasing the weights on

current sensor reading and grid historical grid aggregated value. Therefore, the results shows that increasing the importance of grid average aggregated value is the best model for the system in two of tested scenarios. In the future, we plan to implement the real experiments in prairie or urban areas for various environmental monitoring task.

**Funding Statement:** This research is supported by Natural Science Foundation of Hunan Province (No. 2019JJ40145), Scientific Research Key Project of Hunan Education Department (No. 19A273), and open Fund of Key Laboratory of Hunan Province (2017TP1026).

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] Y. Kishino, Y. Yanagisawa, Y. Shirai, S. Mizutani, F. Naya *et al.*, “A habitat-monitoring system for an endangered fish using a sensor network,” in *Proc. UbiComp/ISWC'15*, Osaka, Japan, pp. 133–136, 2015.
- [2] C. G. N. D. Carvalho, D. S. Kridi and D. G. Gomes, “A predictive algorithm for mitigate swarming bees through proactive monitoring via wireless sensor networks,” in *Proc. PE-WASUN*, New York, NY, USA, pp. 41–47, 2014.
- [3] A. Boubrima, W. Bechkit, R. Hervé and L. Soulhac, “Toward a better monitoring of air pollution using mobile wireless sensor networks,” in *Proc. MOBICOM*, Snowbird, UT, USA, pp. 534–536, 2017.
- [4] J. B. Sharma, “Wireless sensor networks in marine environment monitoring,” *International Journal of Advance Research and Innovative Ideas in Education*, vol. 2, no. 3, pp. 4414–4425, 2016.
- [5] Y. Yu, C. Qian and X. Li, “Distributed and collaborative traffic monitoring in software defined networks,” in *Proc. ACM the Third Workshop on Hot Topics in Software Defined Networking*, New York, NY, USA, pp. 85–90, 2014.
- [6] T. Baumgartner, S. P. Fekete and A. Krölller, “Hallway monitoring with sensor networks,” in *Proc. SENSYS*, Berkeley, CA, USA, pp. 331–332, 2009.
- [7] Q. Yu, Z. Luo and P. Min, “Intrusion detection in wireless sensor networks for destructive intruders,” in *Proc. 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Hong Kong, HK, China, pp. 68–75, 2015.
- [8] P. Dasari, G. K. J. Reddy and A. Gudipalli, “Forest fire detection using wireless sensor networks,” *International Journal on Smart Sensing and Intelligent Systems*, vol. 13, no. 1, pp. 1–8, 2020.
- [9] K. H. Chan, C. S. Cheang and W. W. Choi, “Zigbee wireless sensor network for surface drainage monitoring and flood prediction,” in *Proc. 2014 International Symposium on Antennas and Propagation Conference Proceedings*, Kaoshiung, Taiwan, pp. 391–392, 2015.
- [10] M. Hammoudeh, “Putting the lab on the map: A wireless sensor network system for border security and surveillance,” in *Proc. ICIOTCC*, New York, NY, USA, pp. 621–632, 2016.
- [11] A. Nasipuri, R. Cox, H. Alasti, L. V. D. Zel, B. Rodriguez *et al.* “Wireless sensor network for substation monitoring: Design and deployment,” in *Proc. SENSYS*, New York, NY, USA, pp. 365–366. 2008.
- [12] S. Santini, B. Ostermaier and A. Vitaletti, “First experiences using wireless sensor networks for noise pollution monitoring,” in *Proc. ACM Workshop on Real-world Wireless Sensor Networks*, Glasgow, Scotland, UK, pp. 61–65, 2008.
- [13] C. Peckens, Porter, Cédric and T. Rink, “Wireless sensor networks for long-term monitoring of urban noise,” *Sensors*, vol. 18, no. 9, pp. 3161–3177, 2018.
- [14] G. Zhao, B. Pang, Z. Xu, D. Peng and L. Xu, “Assessment of urban flood susceptibility using semi-supervised machine learning model,” *Science of The Total Environmen*, vol. 6591, pp. 940–949, 2019.
- [15] M. Hopfensitz, C. Mussel, C. Wawra, M. Maucher, M. Kuhl *et al.*, “Multiscale binarization of gene expression data for reconstructing boolean networks,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 2, pp. 487–498, 2012.

- [16] M. Tomar, A. Sathuluri and B. Ravindran, “MaMiC: Macro and micro curriculum for robotic reinforcement learning,” in *Proc. AAMAS*, Montreal, QC, Canada, pp. 2226–2228, 2019.
- [17] S. Kraus, “Computer agents that interact proficiently with people,” in *Proc. the 2nd IKDD Conference on Data Sciences*, Bangalore, India, pp. 1, 2015.