Using Boolean Networks for Consensus in Multi-Robot Environmental Monitoring Tasks

Hanzhong Zheng and Janyl Jumadinova

Abstract— **Robotic systems have been shown to be effective for environmental monitoring tasks, in which one or more robots survey an environment for a particular event. Multi-robot systems, consisting of several interacting robots, have been successful in a variety of applications where their ability to accomplish tasks as a team surpasses the abilities and capacities of a single robot. However, multi-robot systems can generate a large degree of complexity due to the required coordination of movement, communication, and the tolerance for incorrect sensor readings. In this paper, we present a novel approach to multi-robot environmental monitoring based on dynamical systems, in which a robot team overcomes data misinterpretation and aggregation difficulties through an effort of collaboration between all members of the team. Our approach makes use of Boolean networks, which allow for a non-complex method of corroboration, while still retaining meaningful information regarding the dynamics of the robotic system. Using our Boolean network model we apply mathematical tools from dynamical systems and chaos theory to analyze the overall behavior of the robotic dynamic over time. Here we observe how different parameters affect the behaviors of the system. We also empirically and experimentally show that, despite the simplification of the robots' states into Boolean states, our Boolean network model produces accurate results when compared to real events of the environment.**

I. Introduction

The study and design of systems of interacting robots, known as multi-robot systems, have been the site of much attention among computer science researchers in recent years. The existing and potential applications of this field are highly diverse; multi-robot systems have been developed for manufacturing [1], surveillance [2], landmine detection [3], search-and-rescue operations [4], and many more [5]. Due to their extreme inherent complexities of these tasks the success of such applications can be primarily attributed to the ability of robot teams to accomplish tasks that would be difficult or impossible for single robots to attempt. However, with the combined capabilities from their individualized specializations, multi-robot systems contribute a large degree of complexity in the design of their software systems. A specific protocol has to be followed for the transfer of information between the members of robot teams, as well as the aggregation of information from these members.

The information exchange and aggregation in multi-robot systems are particularly important for a class of tasks known as *environmental monitoring*. 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, and other factors. Used as fundamental data-gathering tools, robotic systems today find applications in monitoring environmental changes in the oceans, tracking the spread of the pollution, and in studying volcanoes [6]. However, the current research that uses robotic systems for environmental monitoring concentrates on tracking and searching issues or utilizes robots for data collection from the sensors [7]. Others applications concentrate on the design and development of one robot [8] or they leave the interactions between the robots and the dynamics of the greater robotic systems completely unexplored [9].

In this paper, using ideas from the mathematical field of dynamical systems [10], we develop a multi-robot system for environmental monitoring. In our approach, 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 modeling the interactions between the robots, and study the dynamics produced from those interactions during an environmental monitoring task. In general, Boolean networks have been used for modeling networks in which the node activity can be described by two states, 1 and 0. Each node is updated based on its logical relationships with other nodes. The notion of a 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 [11]. This application made use of a major advantage of Boolean networks: they are discrete systems that are capable of exhibiting complex patterns and behaviors in spite of the simplicity of their fundamental structure. Moreover, Boolean networks are easy to understand and manipulate. Using tools from dynamical systems and chaos, we can analyze an evolution of the aggregated information under various scenarios. For instance, the technique known as *mean-field approximation* allows the behavior of a dynamical system to be expressed using probabilistic estimates, which is useful when the exact state of the dynamical system is not known. This technique has been applied to robotic swarms in [12], where the authors showed that with the large number of objects, their model for interacting objects can be approximated by the mean field approach. In this work, the state of a Boolean network is given by a collection of nodes, each of which possesses a Boolean state. Each node in this network 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. A node represents a robot in our system, with a boolean network of nodes that model a team of interacting robots that are monitoring an environment for a particular event. In this paper, we use a mean-field approach

Allegheny College, Department of Computer Science, 520 N. Main St., Meadville, PA {zhengh,jjumadinova}@allegheny.edu

to generate a recursive mathematical model to determine the probability of finding a node in state 1, at any given instant. Then using the mathematical model, we analyze the dynamics of the nodes in the network and the parameters that influence their behavior. We conduct experiments, both in simulation and using actual robots, to show that, using our model, the robots are able to reach a consensus regarding an event. Our model is desirable when operating error-prone sensors which are common in inexpensive robots, and are often attractive for large-scale multi-robot systems.

II. Related Work

Over the last two decades wireless sensor networks have been popular in applications for environmental monitoring tasks. Sensor networks have been used to monitor habitats [13], the moisture level of the soil [14], conditions in and around building [15]. However, wireless sensor networks are usually stationary, and each sensor generally monitors its local environmental conditions from a two-dimensional setting. In [16] the authors study the mobile sensor network from the dynamic aspect that depends on the mobility of sensors (continuous movement of sensors). In their work, sensors are deployed in the environment by using a random initial deployment strategy and the sensor mobility model, which depends on the density of the sensors at different locations and the stationary two-dimensional Poisson process. They show that the mobility of the sensors can be exploited to compensate for the shortage of sensors and to improve network coverage.

Robotic systems can be viewed as mobile wireless sensors that can be used in cooperation with the stationary wireless sensors. Tekdas *et al.* [17] develop a system that integrates mobile robots with wireless sensors. The authors utilize autonomous robots as data mules that visit static sensors within their communication range, get the data from the sensors, and return to a remote base station to offload the collected data. One of the benefits of this approach is that a sensor-based system saves on energy consumption which prolonges the lifetime of the sensor network.

Robotic systems can also be used in applications where mobility is important and thus the usage of the stationary wireless sensors would not be possible. In [18] a simple, yet novel, approach based on a biased random walk is proposed to locate and track gradient sources such as temperature, light, PH and salinity. The proposed approach is validated through experiments involving one robot in a phototaxis experiment. Jadaliha and Choi [19] develop scheme for the problem of monitoring an environmental process in a large region by a small number of robotic sensors. They test their approach by monitoring the temperature of an outdoor swimming pool, as sampled by an autonomous aquatic surface robot. In many environmental monitoring applications, robots need to make decisions about how to interpret their sensing, what kinds of information to communicate to other robots, and how information obtained from other robots should be aggregated. Although, robotic systems have been applied to environmental monitoring problems, they have been generally used as data collecting and measuring agents. In this paper, we develop a multi-robot system based on a Boolean network where each robot has the intelligence to process its own data and aggregate the information received from the other robots of their teams.

The Boolean network approach has already been used to model a variety of real or artificial networks including: genetic regulatory networks (e.g. [20]), biology (e.g. [21]), neural networks (e.g [22]), artificial life (e.g. [23]), and prediction markets [24]. Boolean networks have also been proposed for robotic behavioral control in [25]. Here, Boolean networks are used to design a robotic system, where sensors and actuators are mapped to the network's inputs and outputs. Their experimental results show that a Boolean network based robot is able to alternate phototaxis and antiphototaxis behavior depending on a sound signal. In our work, we also use a Boolean network based model, however we design a Boolean network to study the dynamics of the robots' interactions, whereas in [25], Boolean networks were used to model the inputs and outputs of single robots.

III. Boolean Network based Robotic System

Using a Boolean network modeling approach similar to that of previous research [22], [24], we have generated a mathematical model for the environmental monitoring system in which a node represents a robot that aggregates the information from the other nodes that are within its neighborhood at each time point. The neighborhood indicates the robots that are within a communication range of each other. The nodes in the Boolean network, representing robots that are able to communicate with each other, have an edge between them. Figure 1 describes an example of such a Boolean network. This particular Boolean network consists of six robot nodes, each with state r_{r_i} with $i = \{1, .., 3\}$, 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 robot is to accurately detect whether a specific event has occurred in its environment. The examples of potential applications include the fire detection, the collapse of buildings or trees, or sudden reactions (chemical or other), that is conducted by a number of the robots in a large space.

In our system, each robot has a specific state. A state is assumed to take on only two values, 1 (implying that the event has happened) and 0 (event did not happen). These states are therefore conveniently applied to Boolean networks. The states of the robots are influenced by the external information sensed by the robot's sensors, the last aggregated state of the robots, and the robot's own state history. Each robot is responsible for aggregating the states received from the other robots, together with its own past state. More precisely, a robot node is set to 1 at the next time point if the weighted average of its external information (received by its own sensors), its own past-state, and the aggregated state of the robots to which it is connected, has surpassed a given threshold, z. This update procedure has been mathematically expressed in Equation 1, where

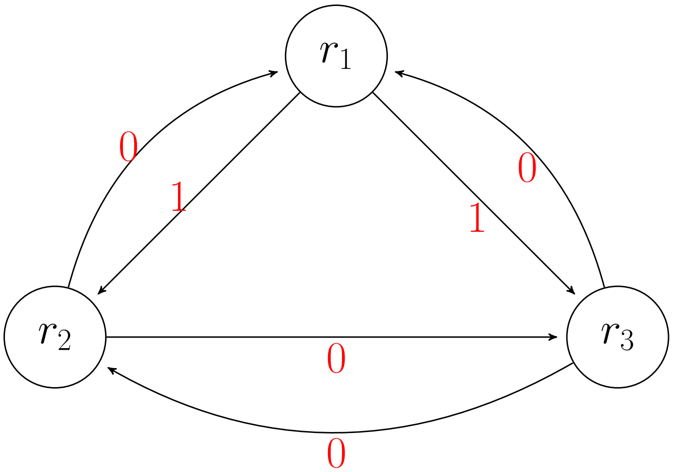


Fig. 1. An example of the Boolean network based robotic system at one time period. The nodes represent the robots, while the edges represent the communication flow between two robots.

$p(t)$ represents the ratio of nodes in the network with a 1 state to the total number of nodes, $r_n(t)$ represents the previous state of the node n, and $s_n(t)$ represents the current determination of the occurrence of the event based on sensor input of the node n. For the threshold value z, we note: $z \in [0, 1]$, and each weight value w_i^n is also $\in [0, 1]$, such that $w_1^n + w_2^n + w_3^n = 1$. 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 a Equation 1.

$$r_n(t+1) = \begin{cases} 1, & \text{if } w_1^n p(t) + w_2^n r_n(t) + w_3^n s_n(t) > z, \\ & \sum w_i^n = 1; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Data: Initial state $s(0)$
Result: p(t)
while *not finished with the monitoring task* **do**
  get the states of the neighboring robots;
  calculate an aggregated state $p(t)$ using Equation 2;
  get $s_n(t)$, new information from its sensor data;
  update the state, $r_n(t+1)$, using Equation 1;
  for *all robots in the Boolean network* **do**
    if *robot k is within its communication range at time t* **then**
      send $r_n(t+1)$ to k;
    end
  end
  $t = t + 1$;
end

Algorithm 1: Algorithm used by the robot n.

The underlying process for each robot's decision making regarding the event occurrence is given by Algorithm 1 and illustrated in Figure 2. The robots initially navigate the environment using a random wandering algorithm and then generate their initial states. Each robot must then calculate an aggregated state and obtain updated sensor readings. We assume that the robots use a well-defined procedure based on their sensors 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, once each robot has updated its state based on this new information and the aggregated state of the neighbors, it transmits its updated state to its neighboring robots.

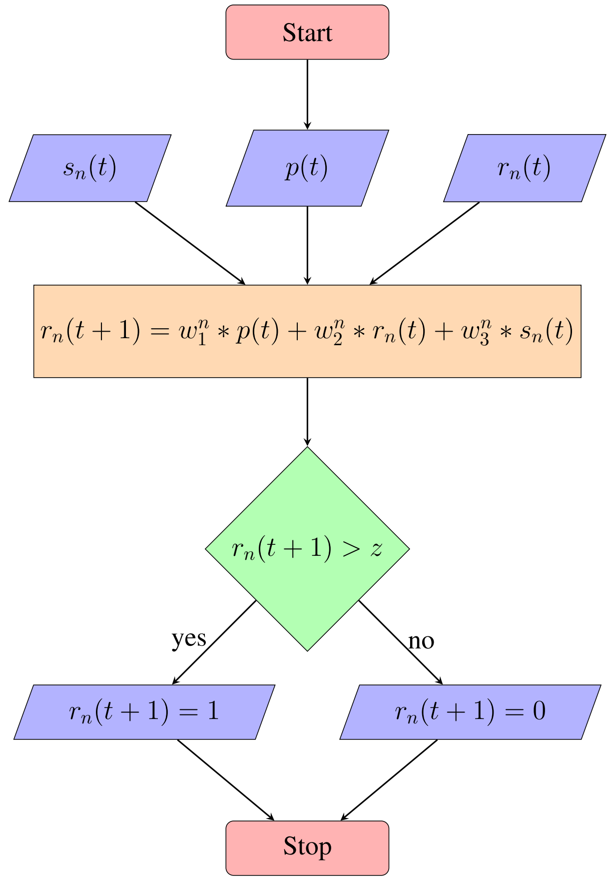


Fig. 2. Flow of the decision making process

For our Boolean network environmental monitoring system, its form is restricted to the structure described in Algorithm 1. The iteration of the procedures in the algorithm allows the system to continuously monitor the environment. The system outputs a binary decision to determine whether or not any event occurs by using the 1 or 0 value.

Mathematical Model for State Aggregation

The aggregated state of robots that believe that the event will occur is given by the fraction of robots that are in state 1 at time t. In our algorithm, this aggregated value is represented through the fraction of nodes that have state 1. However, a robot may not have access to the aggregated value of the whole Boolean network if it has a limited neighborhood. Therefore, to estimate the aggregated state, each robot uses a mean-field approach specific to statistical physics, for the generation of a recursive mathematical model to calculate the probability of finding a node in state 1, for any given time point t.

Let $prob(t)$ be the probability that a robot's state is 1 at time period t, and $1-prob(t)$ the probability that the robot is in state 0 at time period t. We calculate $prob(t+1)$ in terms of $prob(t)$, using a probabilistic approach based on the law of total probability and the assumption of independence of inputs of the rules governing the dynamics of the prediction market. Following the rule of total probability, $prob(t+1) = P(r_n(t+1)=1) = P(r_n(t+1)=1|r_n(t)=1)prob(t) + P(r_n(t+1)=1|r_n(t)=0)(1-prob(t))$, where $P(r_n(t+1)=1)$ is used to denote the probability of the robot's state being 1 at time $t+1$. This gives us Equation 2.

$$\begin{aligned} prob(t+1) &= P\left(w_1 prob(t) + w_2 + w_3 s_n(t) > z\right) prob(t) \\ &\quad + P\left(w_1 prob(t) + w_3 s_n(t) > z\right)(1-prob(t)) \\ &= P\left(s_n(t) > \frac{z - w_1 prob(t) - w_2}{w_3}\right) prob(t) \\ &\quad + P\left(s_n(t) > \frac{z - w_1 prob(t)}{w_3}\right)(1-prob(t)). \end{aligned} \tag{2}$$

Equation 2 can be used with both discrete and continuous distributions for the external information. However, in the numerical and simulated investigations we will focus on the Bernoulli random variable, with $s_n(t)$ being the value of the Bernoulli random variable with probability q_n that $s_n(t)=1$ and probability $1-q_n(t)$ of $s_n(t)$ being 0.

IV. Results

A. Experimental Results

First, we conducted small experiments repeatedly to provide a proof of concept for our Boolean Network system, using four Turtlebot II robots. The precise event that the robots are monitoring is the movement of an object in the environment. This simple experiment allows us to test our Boolean network model in a controlled environment. Our experimental environment is a laboratory room (6 x 10 meters). The sensors on the Turtlebot can detect the moving status of the objects in front of them by measuring the distance between themselves and the object. When the distance between the Turtlebot and the object has changed over a short period time, the status of the object also changes to reflect this movement. Turtlebots are initially spread out in the area where they continuously detect the object in their environment. The detection is performed by employing a random wandering algorithm to explore the area. Turtlebots follow the procedure described in Algorithm 1. When the Turtlebot detects that the status of the object in the area has not changed (the object is not moving), it sets its state to 0. However, if it determines that the object has changed its status (a movement), then Turtlebot sets its state to 1. These values are then immediately shared with the other robots and each then calculates the value of the aggregated state. Since we used a relatively small controlled environment and the Turtlebots communicated with each other through a local wifi, all robots were always within communication range of each other.

We ran experiments for various weight values and threshold values used by our Boolean network under three different scenarios: 1) the object remains still, 2) the object only alternates its status once, 3) the object is constantly alternating its status. Each experiment was repeated ten times and the average values were reported. Our experimental results for the parameter values were aligned with what was expected from our numerical calculations, omitted in this paper. We also recorded the actual movement of the object and calculated the accuracy as a fraction of the correct object status detections by the robotic team over the total number of time steps.

TABLE I

ACCURACY OF THE AGGREGATED STATES FROM THE PHYSICAL EXPERIMENTS.

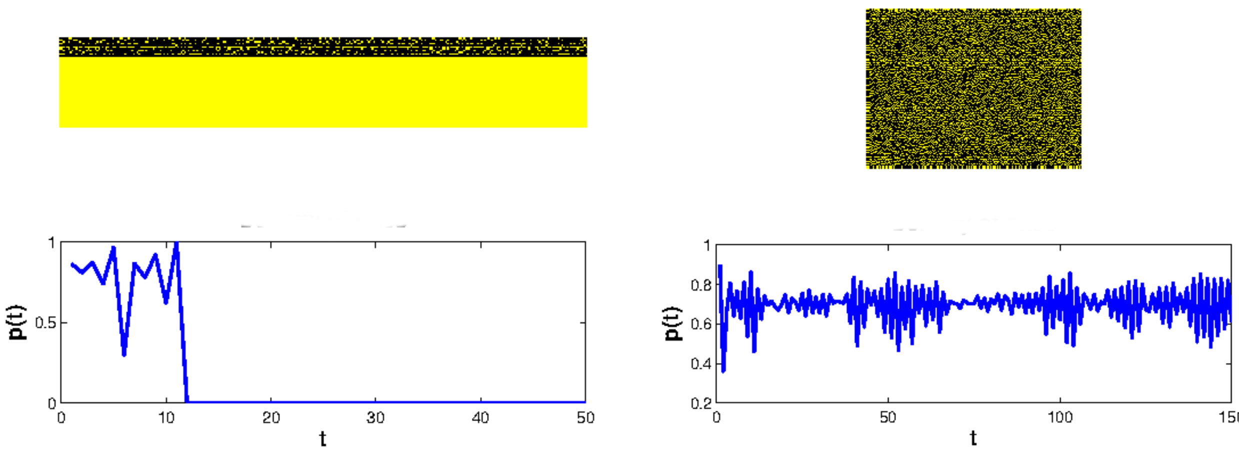
Testing Case	Accuracy
Scenario 1	98%
Scenario 2	91%
Scenario 3	85%

Table I shows the accuracy of the aggregated state values for the three tested scenarios. The complexity of our experiments originate from the object being occasionally being hidden from view from a robot. When out of view, a robot must depend on the observations of the others, as well as their own histories, for their sensory data. Under the first scenario, the aggregated state that is generated is always 0, which satisfies the status of the stationary object. During the second scenario, a state changes from $0 \rightarrow 1$, and then back again. This occurs at the same when the object starts to move at time period 7. Under the third scenario, the object begins with a stationary status, then it moves around in the area and stops again, and repeats this process until the end of the experiment. The aggregated state generated under this scenario reflects the alternating values from $0 \rightarrow 1$ and then back to 0 as the object changes its status continuously. Under the three different scenarios, the Boolean network based robotic system performs accurately as it detects the changing status of the object over time.

B. Simulation Results

We conducted simulation experiments to test our Boolean network model with a large number of simulated robots and to provide a more in-depth analysis of our model. Our simulation experiment involved the development of a simulation of a multi-robot team applied to an abstract environmental monitoring task. This simulation was developed in the Python programming language, and uses random number generation to simulate the reception of true and false sensor readings for each of the robot agents in the system. The purpose of designing such a simulation is to provide us with a large amount of data about the performance of our algorithm, which would be difficult and time-consuming to gather using conventional physical implementations.

The simulated robots are represented using points on a two-dimensional plane. Each robot uses an abstract "sensor"



(a) (b)

Fig. 3. Pattern formation plot and the aggregated state in the robot team, starting with a random initial condition and for parameters (a) $q = 0.2, z = 0.6, w = (0.2, 0.3, 0.5)$ and (b) $q = 0.6, z = 0.6, w = (0.2, 0.3, 0.5)$

to receive readings about its environment. At any given stage in the simulation, there is a location on the two-dimensional plane that represents an "event" taking place. Within a specific radius of this location, the readings received by a robot's sensor will increase in value. However, there is a small probability that each robot's sensor will provide a false reading, even outside of this radius. For simplicity and for the purpose of illustration of our method, we assume that the information received by the sensors is represented by a Bernoulli random variable. The behavior of the simulated robots follows the algorithm described in Section III. In our simulated experiments we use 100 robot agents and we report the average results after running each experiment 20 times.

Figure 3 shows pattern formation plots at the top and the aggregated state graphs at the bottom. Here, q is the probability that the Bernoulli random variable is 1. Pattern formation plots are generated by arranging the nodes, representing the robots, in a one-dimensional array from left to right along the x-axis. The y-axis represents time periods that increase downwards, with $t = 1$ starting at the top. The black and neutral dots represent states 1 and 0, respectively. The fraction of the robots having state 1 over time can be noted in the aggregated state graph. Figure 3(a) shows a scenario of one event occurring at the beginning of the simulation and then no events occurring during the remainder of the time, with $q = 0.2$. We observe that the robot team is able to detect the occurrence of the event and, over time, the robot team is able to verify the discontinuation event. Figure 3(b) shows a pattern formation plot and the aggregated state graph for $q = 0.6$, where the event occurrence changes over time. In this case the fraction of ones oscillates as the robot agents detect several events over the period of 150 time steps.

C. Aggregated State using the Mean-Field Approximation

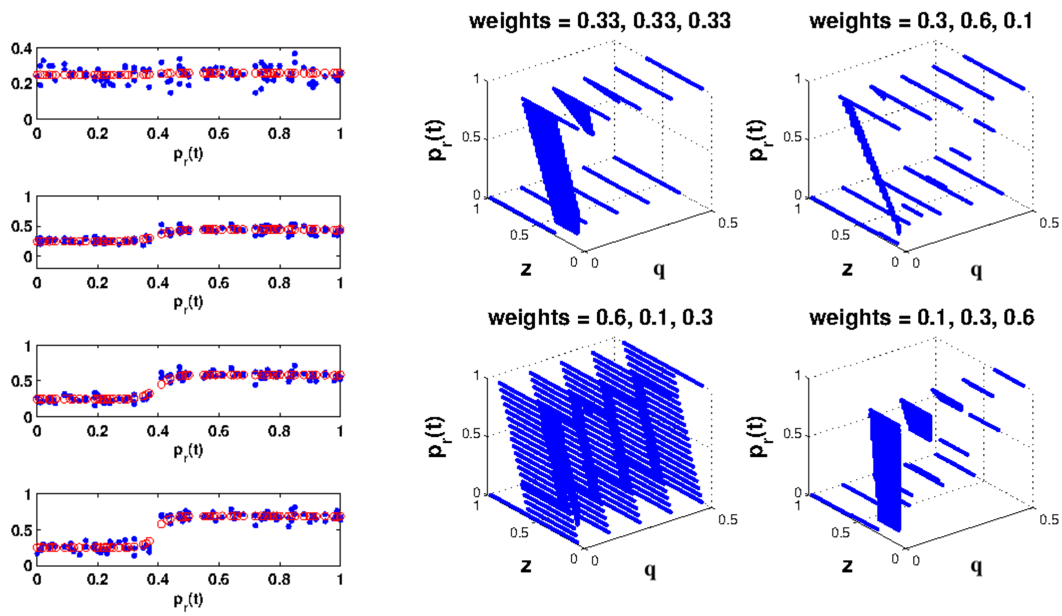
We also compared the results obtained with the mean-field formula by the robot agents to the actual fraction of nodes in state 1 for a simulated network in Figure 4(a). We observed that the match accurately indicates the aggregated state, approximated using the mean-field calculations, since it closely resembles an actual fraction of robots that have state 1.

We generated bifurcation diagrams using the mathematical model (specific to chaos theory) to further analyze the behavior of the aggregated state and identify the effect of various parameters of the model. The diagrams show values visited or approached asymptotically as functions of the varied parameters. Figure 4(b) shows three-dimensional bifurcation diagrams along the parameters q and z for different weight values. We observe that when the robot agents have equal weights for their inputs (e.g., aggregated state, previous state and the external information), then more complex behavior occurs for smaller values of q (top left). When robot agents use higher weights for the aggregated state, we can identify symmetric complex behavior (bottom left). On the other hand, if robot agents put higher weights on their own previous states, then the behavior is stable (top right). Finally, when the robots trust the information signals the most (bottom right), we observe more complex behavior for small values of q and z, meaning there is more accountability for negative information, indicating the likelihood of the event not happening.

V. CONCLUSION

In this paper we have described a Boolean network based multi-robot environmental monitoring system. Our system is able to calculate the aggregated state of the robots within the network and analyze the behavior of the robot population in response to various model parameters, such as information received from the sensors or the robots' previous states. We demonstrate that our Boolean network approach produces accurate results. We also use a mean-field based mathematical modeling to approximate the aggregated state of all the robots in the network by the robots.

In the future we plan to conduct more experiments with the robots for more complex and continuous monitoring tasks. We also plan to do statistical comparison of our Boolean net-



(a) (b)

Fig. 4. (a) Fitting of four iterations of the aggregated state and the mean-field generated aggregated state (from top to bottom: $t = 1, t = 10, t = 30, t = 50$). (b) Bifurcation diagrams along parameters q and z for different weights.

work results with other commonly used aggregation methods, such as Kalman filter.

References

[1] G. S. Tewolde, C. Wu, Y. Wang, and W. Sheng, "Distributed multi-robot work load partition in manufacturing automation," in *Automation Science and Engineering. CASE 2008.* IEEE, 2008, pp. 504–509.

[2] A. Kolling and S. Carpin, "Multi-robot surveillance: an improved algorithm for the graph-clear problem," in *ICRA 2008.* IEEE, 2008, pp. 2360–2365.

[3] P. Dasgupta, J. Baca, K. Guruprasad, A. Muñoz-Meléndez, and J. Jumadinova, "The comrade system for multirobot autonomous landmine detection in postconflict regions," *Journal of Robotics*, vol. 2015, 2015.

[4] J. L. Baxter, E. Burke, J. M. Garibaldi, and M. Norman, "Multi-robot search and rescue: A potential field based approach," in *Autonomous robots and agents.* Springer, 2007, pp. 9–16.

[5] L. E. Parker, "Multiple mobile robot systems," pp. 921–941, 2008.

[6] M. Dunbabin and L. Marques, "Robots for environmental monitoring: Significant advancements and applications," *Robotics & Automation Magazine, IEEE*, vol. 19, no. 1, pp. 24–39, 2012.

[7] V. Isler, "Robotic sensor networks for environmental monitoring." in *AAAI Spring Symposium: AI, The Fundamental Social Aggregation Challenge*, 2012.

[8] Z. Chen, S. Shatara, and X. Tan, "Modeling of biomimetic robotic fish propelled by an ionic polymer–metal composite caudal fin," *IEEE/ASME Tran. on Mechatronics*, vol. 15, no. 3, pp. 448–459, 2010.

[9] R. HeÃŸ, F. Kempf, J. Tautz, and K. Schilling, "Remote biological and robotic sensor networks for environmental monitoring," in *Telematic Applications*, vol. 3, no. 1, 2013, pp. 138–143.

[10] K. Alligood, T. Sauer, and J. Yorke, *Chaos: An Introduction to Dynamical Systems*, ser. Chaos: An Introduction to Dynamical Systems. New York, NY, 1997.

[11] S. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, no. 3, pp. 437 – 467, 1969.

[12] J.-Y. Le Boudec, D. McDonald, and J. Mundinger, "A generic mean field convergence result for systems of interacting objects," in *Quantitative Evaluation of Systems, 2007. QEST 2007.*, 2007, pp. 3–18.

[13] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, 2002, pp. 88–97.

[14] R. Cardell-Oliver, K. Smettem, M. Kranz, and K. Mayer, "Field testing a wireless sensor network for reactive environmental monitoring [soil moisture measurement]," in *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004.* IEEE, 2004, pp. 7–12.

[15] W.-S. Jang, W. M. Healy, and M. J. Skibniewski, "Wireless sensor networks as part of a web-based building environmental monitoring system," *Automation in Construction*, vol. 17, no. 6, pp. 729–736, 2008.

[16] B. Liu, P. Brass, O. Dousse, P. Nain, and D. Towsley, "Mobility improves coverage of sensor networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2005, pp. 300–308.

[17] O. Tekdas, V. Isler, J. H. Lim, and A. Terzis, "Using mobile robots to harvest data from sensor fields," *IEEE Wireless Communications*, vol. 16, no. 1, p. 22, 2009.

[18] A. Dhariwal, G. S. Sukhatme, and A. A. Requicha, "Bacterium-inspired robots for environmental monitoring," in *IEEE International Conference on Robotics and Automation, 2004. ICRA'04.*, vol. 2. IEEE, 2004, pp. 1436–1443.

[19] M. Jadaliha and J. Choi, "Environmental monitoring using autonomous aquatic robots: Sampling algorithms and experiments," *IEEE Tran. on Control Systems Technology*, vol. 21, no. 3, pp. 899–905, 2013.

[20] I. Shmulevich, E. R. Dougherty, and W. Zhang, "From boolean to probabilistic boolean networks as models of genetic regulatory networks," *Proceedings of the IEEE*, vol. 90, no. 11, pp. 1778–1792, 2002.

[21] L. Raeymaekers, "Dynamics of boolean networks controlled by biologically meaningful functions," *Journal of Theoretical Biology*, vol. 218, no. 3, pp. 331–341, 2002.

[22] M. T. Matache and J. Heidel, "Random boolean network model exhibiting deterministic chaos," *Physical review E*, vol. 69, no. 5, p. 056214, 2004.

[23] S. Wolfram, *A new kind of science*. Wolfram media Champaign, 2002, vol. 5.

[24] J. Jumadinova, M. T. Matache, and P. Dasgupta, "A multi-agent prediction market based on boolean network evolution," in *Proceedings of the 2011 WI-IAT*, 2011, pp. 171–179.

[25] A. Roli, M. Manfroni, C. Pinciroli, and M. Birattari, "On the design of boolean network robots," *Applications of Evolutionary Computation*, pp. 43–52, 2011.