

2011

A Multi-Agent Prediction Market Based on Boolean Network Evolution

Janyl Jumadinova

University of Nebraska at Omaha

Mihaela Teodora Matache

University of Nebraska at Omaha, dvelcsov@unomaha.edu

Prithviraj Dasgupta

University of Nebraska at Omaha, pdasgupta@unomaha.edu

Follow this and additional works at: <https://digitalcommons.unomaha.edu/mathfacproc>

 Part of the [Mathematics Commons](#)

Please take our feedback survey at: https://unomaha.az1.qualtrics.com/jfe/form/SV_8cchtFmpDyGfBLE

Recommended Citation

Jumadinova, Janyl; Matache, Mihaela Teodora; and Dasgupta, Prithviraj, "A Multi-Agent Prediction Market Based on Boolean Network Evolution" (2011). *Mathematics Faculty Proceedings & Presentations*. 3.
<https://digitalcommons.unomaha.edu/mathfacproc/3>

This Conference Proceeding is brought to you for free and open access by the Department of Mathematics at DigitalCommons@UNO. It has been accepted for inclusion in Mathematics Faculty Proceedings & Presentations by an authorized administrator of DigitalCommons@UNO. For more information, please contact unodigitalcommons@unomaha.edu.

A Multi-Agent Prediction Market Based on Boolean Network Evolution

Janyl Jumadinova*, Mihaela T. Matache[†] and Prithviraj Dasgupta*
*Department of Computer Science, University of Nebraska at Omaha,
Email: {jjumadinova, pdasgupta}@unomaha.edu
[†]Department of Mathematics, University of Nebraska at Omaha,
Email: dmatache@unomaha.edu

Abstract—Prediction markets have been shown to be a useful tool in forecasting the outcome of future events by aggregating public opinion about the events’ outcome. Previous research on prediction markets has mostly analyzed the prediction markets by building complex analytical models. In this paper, we posit that simpler yet powerful Boolean rules can be used to adequately describe the operations of a prediction market. We have used a multi-agent based prediction market where Boolean network based rules are used to capture the evolution of the beliefs of the market’s participants, as well as to aggregate the prices in the market. We show that despite the simplification of the traders’ beliefs in the prediction market into Boolean states, the aggregated market price calculated using our BN model is strongly correlated with the price calculated by a commonly used aggregation strategy in existing prediction markets called the Logarithmic Market Scoring Rule (LMSR). We also empirically show that our Boolean network-based prediction market can stabilize market prices under the presence of untruthful belief revelation by the traders.

Keywords-Prediction markets, Boolean networks, distributed information aggregation, complex systems modeling.

I. INTRODUCTION

A prediction market is a market-based aggregation mechanism that is used to combine the opinions on the outcome of a future, real-world event from different people and forecast the event’s possible outcome based on their aggregated opinion. Recently, [5], [6], [15], [19] have used multi-agent systems to analyze the operation of prediction markets, where the behaviors of the market’s human participants are implemented using software agents. Most of the existing agent-based models of prediction markets use game theoretic [4], [6], [7], or decision theoretic [8], [17], [23] techniques to analyze the interactions and behavior of the agents. In this paper, we propose a form of a dynamical system, called a Boolean Network (BN) that uses simple Boolean rules to model the operation of a prediction market. In a BN, each node is represented by a binary state while the network edges represent rules that update the state of the node that the edges are incident on. Although inherently simple, BNs can be used to analyze essential aspects of complex networks such as values of parameters that effect a specific behavior and the time required to reach that behavior. It also makes sense to use Boolean networks in the context of a prediction market because there is a direct correspondence

between Boolean values output by the Boolean network’s rules and the binary outcomes of events predicted by a prediction market. The main contributions of our paper are to develop simple Boolean rules for updating the beliefs for each of the market’s participants and for aggregating the participants’ belief information into a single market price. We show that despite the simplification of the traders’ beliefs in the prediction market into Boolean states, the aggregated market price calculated using our BN model is strongly correlated with the price calculated by a commonly used aggregation strategy in existing prediction markets called the Logarithmic Market Scoring Rule (LMSR). Our experimental results show that our BN model also eliminates the problem of frequently fluctuating prices that are known to be a drawback of the LMSR. We also use our BN model to analyze the dynamics of the prediction market with respect to different market parameters and determine the conditions under which the market price converges. Finally, we also model the untruthful belief revelation by the market participants, a commonly encountered problem in prediction markets, using the presence of noise in the Boolean rules of our prediction market and obtain similar results as the conventional (non-Boolean) prediction markets. And finally, we show that the market price tends to stabilize better as the number of trading agents increases.

II. RELATED WORK

Prediction Markets. Prediction markets have been used in various scenarios such as predicting the outcome of geopolitical events such as U.S. presidential elections, determining the outcome of sporting events, predicting the box office performance of Hollywood movies, etc. Companies such as Google, Microsoft, Yahoo and Best Buy have used prediction markets internally to collate the private information from their workers to make predictions about future product and business trends. The seminal work on prediction market analysis [8], [23] has shown that the mean belief values of individual traders about the outcome of a future event corresponds to the event’s market price. Since then researchers have studied prediction markets from different perspectives. Some researchers have studied traders’ behavior by modeling their interactions within a game theoretic framework such as a Shapley-Shubik game

[7] or a Bayesian game [6]. Other researchers have focused on designing rules that a market maker can use to combine the opinions (beliefs) from different traders such as the logarithmic market scoring rule (LMSR) [4], [19] and an information based market maker [5], [9]. In contrast to these approaches, our paper proposes to use simple Boolean rule to model the operation of a prediction market.

Boolean Networks. Boolean network models [16] have been used for modeling networks in which the node activity can be described by two states, 1 and 0. The edges of the network affect the rules that determine the state transitions of the nodes. BN modeling allows exploring the dynamics of relevant nodes and predicting their future states, as well as exploring the overall dynamics of the network. This is especially useful for large networks like prediction markets where analyzing the global behavior of the system and tracking the individual nodes is computationally intensive. The BN approach has already been used to model a variety of real or artificial networks including among others, genetic regulatory networks [21], strongly disordered systems that are common in physics [16], biology [1], neural networks [18], scale-free networks [11], and artificial life [24]. To the best of our knowledge, this is the first time a BN has been applied to model prediction markets.

III. BOOLEAN NETWORK-BASED PREDICTION MARKET

A. Prediction Market Preliminaries

The major participants in an agent-based prediction market are the set of *trading agents* and a central entity called the *market maker agent*. The outcome of an event is binary (will happen/won't happen) and the trading agents place monetary bets related to this outcome. A prediction market consists of T *trading periods* and the trading agents place a bet at each trading period t . The bets are in the form of financial instruments called *securities* related to the event which can be traded (bought/sold/held) in discrete quantities by the trading agents. The market maker agent aggregates the prices at which securities of the event are traded by the trading agents and comes up with an aggregated and normalized *market price* which expresses the probability of the outcome of the event. When the actual decision on the event is made in the real world - the event happens or does not - each trading agent gets paid for each security, the difference between the price at which it bought the security and \$1 if the event happens, or loses the money spent in buying securities if the event does not happen. Because of the binary nature of the event outcomes, it makes sense to use Boolean functions to represent the beliefs of the traders in prediction markets [3].

B. BN-based Prediction Market

Our BN-based prediction market consists of three major entities: trading agents, a market maker agent, and information sources that are external to the market but provide

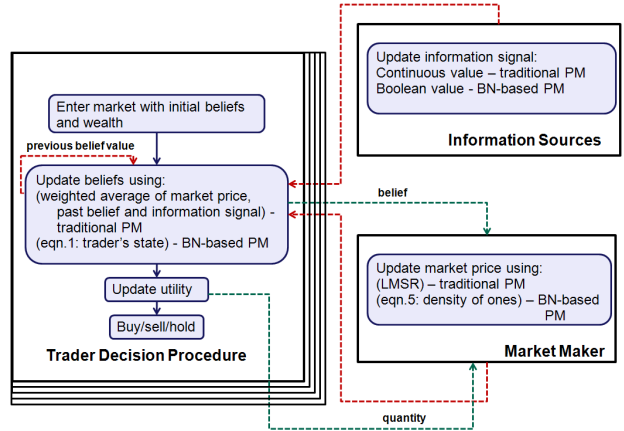


Figure 1. The sequence of operations done by the trading agents and the market maker agent in one trading period in a prediction market.

information to the market's agents. The basic operations of our BN-based prediction market are based on the traditional prediction market's operations, however the trading agents' beliefs are updated using a Boolean function and a novel technique using the Boolean beliefs of the trading agents is used to calculate the market price. Figure 1 shows the operation of both the conventional and the BN-based prediction market proposed in this paper¹. In our prediction market human traders are represented by software trading agents that buy and sell securities on behalf of the human traders. To do this, each trading agent maintains a belief about the outcome of the security corresponding to the event and updates this belief at certain intervals using the aggregated market price, past belief values and external information. In our BN-based prediction market each trading agent uses a variable called a *state* to represent this belief. Each state can take one of two values: 1 or ON, meaning that the trading agent believes that the event will happen, or 0 or OFF, meaning that the trading agent believes the event will not happen. Following the belief update rule in a conventional prediction market, trading agents update the value of their state at each trading period t based on the current aggregated market price, their past state, and the information signal they receive about the event. The state update procedure is represented as a Boolean function which is described in the next section. After the trading agents update their state, they calculate their expected utility using their past state and the current market price and use this utility to determine the optimal quantity of each security to buy or sell. The optimal quantity to buy or sell is given by the quantity that maximizes the expected utility of the trading agent. The trading agents send the quantity of securities they want to buy or sell and their current belief/state to the market maker

¹For the simplicity of our discussion and without the loss of generality, we assume there is one event in our prediction market with two possible outcomes - event happens/does not happen.

Operation	Conventional PM	BN-based PM
Belief update	1. Trading agents calculate their beliefs as a weighted average of the market price, their past belief and the information signal [23] with all the parameters $\in [0, 1]$. 2. Trading agents submit their beliefs as a discrete value $\in [0, 1]$.	1. Trading agents' beliefs are represented through their Boolean states which are updated as a threshold function of the weighted average of the market price $\in [0, 1]$, their past Boolean state $\in \{0, 1\}$ and the Bernoulli information signal $\in \{0, 1\}$. 2. Trading agents send their belief (i.e. their state) as a Boolean value $\in \{0, 1\}$.
Aggregation rule	3. The market maker uses some rule such as LMSR to aggregate the beliefs of the traders and set the market price [4].	3. The market maker uses the fraction of traders that are ON to calculate the market price.
External information service	4. Most prediction markets use a continuous probability distribution to model the external information signal.	4. Following [14], we use a Boolean value for the signal.

Table 1

DIFFERENCES BETWEEN CONVENTIONAL LMSR-BASED PREDICTION MARKET AND OUR BN-BASED PREDICTION MARKET.

agent. The market maker agent updates the market price after aggregating the beliefs received from the trading agents. The market maker agent also calculates the cost of each trading agent's transaction and sends it back to each trading agent.

In the next section we describe the Boolean function formulation of the operations by the trading agents and the market maker agent in a prediction market. A summary comparison between the operations of our BN-based prediction market and a conventional (LMSR-based) prediction market is given in Table I.

C. Trading Agents' Boolean Belief Update

Let N be the set of trading agents in the prediction market. The state of a trading agent $n \in N$ is determined by the three variables defined below:

- 1) $p_r(t)$ - the aggregated market price at trading period t . In our BN-based prediction market we call the current aggregated market price the *density of ones* which is the fraction of trading agents that are in state 1 at a given trading period t .
- 2) $r_n(t)$ - state of the n -th trading agent at trading period t representing its belief about the outcome of the event.
- 3) $\overline{w}^n = (w_1^n, w_2^n, w_3^n)$ - a vector of weights representing the trust that the n -th trading agent holds for the accuracy of the posted market price, its own past belief, and the new information signal it obtains, respectively, following [10]. These trusts are represented as weights $w_i^n \in [0, 1]$, such that $\sum_{i=1}^3 w_i^n = 1$, for $i = 1, 2, 3$.

Let $B_n(t)$ be the information signal received by the n -th trading agent at trading period t . For simplicity and for

the purpose of illustration of this method, we assume that $B_n(t)$ is the value of a Bernoulli random variable with probability q_n of obtaining a 1, that is positive information, and probability $1 - q_n$ of obtaining a 0, that is negative information. The rule that generates the new state of the n -th trading agent can be written as follows and is shown as a diagram in Figure 2:

$$r_n(t+1) = \begin{cases} 1 & , \text{if } w_1^n \cdot p_r(t) + w_2^n \cdot r_n(t) + w_3^n \cdot B_n(t) > z, \\ \sum_{i=1}^3 w_i^n = 1, w_i^n \in [0, 1], \text{ for } i = 1, 2, 3; \\ 0 & , \text{otherwise.} \end{cases} \quad (1)$$

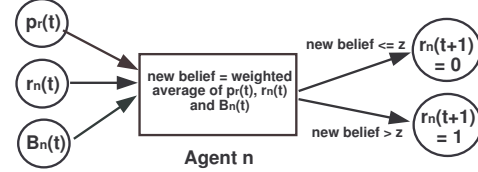


Figure 2. The Boolean belief update used by agent n at trading period t .

Here $z \in [0, 1]$ represents a threshold parameter used to convert the quantity $w_1^n \cdot p_r(t) + w_2^n \cdot r_n(t) + w_3^n \cdot B_n(t)$ into a Boolean value. The rule basically says that the trading agent r_n is turned ON at trading period $t+1$ if the weighted sum of the market price, its own past state, and the external information signal is greater than some threshold value z at trading period t . Thus the trading agent rule is a linear threshold function. The value of z indicates the boundary between what is considered negative or positive overall impact of the aggregated information on each trading agent's belief. For simplicity, we will assume that z is fixed for all trading agents. Although in real prediction markets different agents may have different ways of evaluating information and reflecting on their past experiences, for simplicity we assume that the trust weights \overline{w}^n and the Bernoulli distribution B are the same for all trading agents. Also, in real prediction markets different trading agents may have different thresholds or predispositions for believing that an event will take place; therefore future work will allow for generalizations with varying thresholds. In Section IV-A we show how the weights, w_1^n, w_2^n, w_3^n , can be learned using a neural network.

D. Mean-field Analysis for Calculating the Aggregated Market Price by Market Maker agents

The fraction of trading agents in state 1 at trading period t give the aggregated belief of trading agents that believe the event will happen at trading period t . In our model this value is represented through the density of ones which is calculated by the market maker agent. The market maker agent uses a mean-field approach specific to statistical physics to generate a recursive mathematical model for the density of ones. The mean-field approach assumes a sufficiently large number of nodes so that potential local correlations can be

ignored. This makes the computations more manageable. Moreover, as we will point out in the numerical simulations,

Let $p_r(t)$ be the probability that a (generic) trading agent is ON at trading period t , and $1 - p_r(t)$ the probability that the trading agent is OFF at trading period t . We find $p_r(t+1)$ in terms of $p_r(t)$, using a probabilistic approach typical for derivations of mean-field formulae, based on the law of total probability and the assumption of independence of inputs of the rules governing the dynamics of the prediction market. Since the trust weights $\overline{w^n}$ and Bernoulli distribution $B_n(t)$ is assumed to be the same for all agents, in the derivation below we will drop the trading agent index n .

Observe that by the rule of total probability, $p_r(t+1) = P(r(t+1) = 1) = P(r(t+1) = 1|r(t) = 0)(1 - p_r(t)) + P(r(t+1) = 1|r(t) = 1)p_r(t)$, where $P(A)$ is used to denote the probability of an event A . We note that

$$P(r(t+1) = 1|r(t) = 0) = P(w_1 p_r(t) + w_3 B > z) = \quad (2)$$

$$= P\left(B > \frac{z - w_1 p_r(t)}{w_3}\right) \text{ and similarly}$$

$$P(r(t+1) = 1|r(t) = 1) = P(w_1 p_r(t) + w_2 + w_3 B > z) = \quad (3)$$

$$= P\left(B > \frac{z - w_1 p_r(t) - w_2}{w_3}\right).$$

Putting equations (2) and (3), we get:

$$p_r(t+1) = P\left(B > \frac{z - w_1 p_r(t)}{w_3}\right) (1 - p_r(t)) + \quad (4)$$

$$+ P\left(B > \frac{z - w_1 p_r(t) - w_2}{w_3}\right) p_r(t).$$

To simplify the notation, denote $F_B(b) = P(B > b)$, the complementary cumulative distribution function associated to the random variable B . Then the formula for $p_r(t+1)$ becomes

$$p_r(t+1) = F_B\left(\frac{z - w_1 p_r(t)}{w_3}\right) (1 - p_r(t)) + \quad (5)$$

$$+ F_B\left(\frac{z - w_1 p_r(t) - w_2}{w_3}\right) p_r(t).$$

Observe that this formula can be used with both discrete and continuous distributions for the external information. However, in the numerical investigations we will focus on the Bernoulli random variable. For the Bernoulli case, we can actually compute the values of F_B according to the relative positions of $\frac{z - w_1 p_r(t) - w_2}{w_3} < \frac{z - w_1 p_r(t)}{w_3}$ with respect to the two possible values of B , namely 0 and 1. Recall that q is the probability that B is 1. By a straightforward computation we obtain:

$$p_r(t+1) = \begin{cases} 1, & \text{if } p_r(t) > \frac{z}{w_1} \\ q(1 - p_r(t)) + p_r(t), & \text{if } \max\left\{\frac{z - w_2}{w_1}, \frac{z - w_3}{w_1}\right\} < p_r(t) \leq \frac{z}{w_1} \\ p_r(t), & \text{if } \frac{z - w_2}{w_1} < p_r(t) \leq \frac{z - w_3}{w_1} \\ q, & \text{if } \frac{z - w_3}{w_1} < p_r(t) \leq \frac{z - w_2}{w_1} \\ qp_r(t), & \text{if } \frac{z - w_2 - w_3}{w_1} < p_r(t) \leq \min\left\{\frac{z - w_2}{w_1}, \frac{z - w_3}{w_1}\right\} \\ 0, & \text{if } p_r(t) \leq \frac{z - w_2 - w_3}{w_1}. \end{cases} \quad (6)$$

The mathematical model for the density of ones not only represents the aggregated market price but can also be used to analyze the dynamics of the prediction market. Observe

that the function (6) represents a *map* (that is a function whose domain and codomain are the same) on $[0, 1]$ whose *fixed points* can be computed. Let us denote it by $f(p)$. A point $p \in S$ is a fixed point of the map f if $f(p) = p$. It is known from chaos theory that the fixed points of a map drive the dynamics of the map. More precisely, if say p is a fixed point of f , then if $|f'(p)| < 1$, the fixed point p attracts all other points close enough to p . More precisely, if x is a point close to p , then $f^n(x) \rightarrow p$ as $n \rightarrow \infty$, where $f^n(x)$ is the n -th iterate of f at the point x . The set $\{x, f(x), f^2(x), \dots, f^n(x), \dots\}$ is called the orbit of x . On the other hand, if $|f'(p)| > 1$, the fixed point p repels all the orbits starting at points x in a neighborhood of p .

We find the fixed points for the map given in (6) in our BN-based prediction market by setting $p_r(t+1) = p_r(t)$. The analysis of the stability of the fixed points of the map (6) reveals that the fixed points 0 and 1 are always stable. On the other hand, if $w_2 < w_3$ then there is a third stable point q . The orbits will be attracted to one of these three fixed points, depending on the parameters. If $w_2 > w_3$, we may also end up with the case where all points in $[0, 1]$ are fixed points, which means that all states are frozen from the very beginning, so the system is unstable. This can happen if $\frac{z - w_2}{w_1} < 0$ and $\frac{z - w_3}{w_1} > 1$ which means $w_1 + w_3 < z < w_2$. Higher order iterations of the map (6) do not reveal more complexity. Thus, in case the external information is modeled by a Bernoulli random variable, the behavior of the model is non-complex and can be easily predicted. In future research we will consider more sophisticated random processes to account for the external information.

IV. EXPERIMENTAL RESULTS

A. Learning the trust values by trading agents

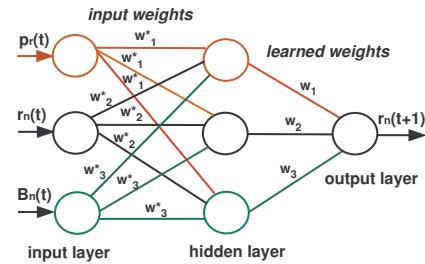


Figure 3. One hidden layer neural network used for learn trust weights.

To find the correct combination of weight parameters, w_1^n, w_2^n, w_3^n used by the n -th trading agent's belief update rule given in Equation 1, we use a neural network representation. We construct a neural network with one hidden layer, where the market price at trading period t , $p_r(t)$, the state of the trading agent at trading period t , $r_n(t)$, and the Bernoulli variable representing the information parameter, $B_n(t)$, are the inputs to the network. The new state of the

trading agent at trading period $t+1$, $r_n(t+1)$, is the output of the neural network. The representation of the neural network used is shown in Figure 3. The initial input weights are set randomly, while the learned (output) weights are learned for different values of the parameters in our BN model, namely z - the threshold parameter, and q - the probability that the Bernoulli random variable is 1. We use the backpropagation technique to learn the weights in our neural network [22]. The training set used for the neural network was obtained by simulating the prediction market for over 200 different combinations of values of z and q parameters in our BN. For the data generated for the training set $p_r(t)$ was calculated as the fraction of the trading agent nodes that are equal to 1 at time t , $r_n(t)$ was set to 1 if the belief that maximizes the expected utility of the trading agent [8] was above z and 0 otherwise, and $B_n(t)$ was set to 0 or 1 based on the value of q . A set of learned weights was generated for each combination of z and q values. The learned values are used in the numerical investigations given below.

B. Patterns and validation of the mean-field based price aggregation mechanism

Having too few trading agents may lead to discrepancies between the mathematical mean-field model and the actual simulation of prediction market due to the fact that for a mean-field approximation the prediction market has to be large enough to ignore local correlations. In our experiments we found that a prediction market with 100 trading agents is sufficiently large for a good match in the fraction of trading agents that are *ON* between the mathematical model and the actual network. Therefore, in all of our experimental results except those presented in Section IV-E we use 100 trading agents. Similar results were obtained with a larger number of trading agents. We start our experimental analysis by presenting *pattern formation plots* generated with a Boolean network governed by the rules presented in Section III-C. More precisely, pattern formation plots are obtained by arranging the nodes, representing the trading agents, in a one-dimensional array and numbering them from left to right. Then we choose an initial state of the prediction market and iterate it a number of trading periods with time evolving downwards. We plot a black dot when the state of the trading agent is 1 and a neutral dot when it is 0. Figure 4 shows the pattern formation plots and the corresponding aggregated market price using BN at each trading period of the prediction market's evolution. This is done for four distinct parameter combinations. We can see in Figure 4(a), that for a low value of $q = 0.3$ and a medium value of $z = 0.6$ (which means that the most weight is given to the information value), the aggregated market price oscillates in a narrow range of values around 0.3. The corresponding pattern formation plot in Figure 4(b) looks random but with more nodes in state 0 (more neutral dots). Figures 4(c) and (d) show similar result for $q = 0.7$

and $z = 0.7$. Here the aggregated market price does not reach stability, but it oscillates within a narrow range of values around 0.7 and therefore its corresponding pattern has more nodes in state 1 (more black dots). The overall higher values for the aggregated market price are due to the fact that the probability of information signal being 1 is high ($q = 0.7$) and the weight corresponding to the information signal is also high ($w_3 = 0.8$). In Figures 4(e) and (f) the parameters are $q = 0.2$ and $z = 0.8$. We can see that the aggregated market price is stable around 0.7 and thus the pattern is stationary with neutral and black vertical stripes showing that trading agents are either in state 0 or in state 1 throughout the prediction market's duration. Finally, for the parameters $q = 0.2, z = 0.2$ in Figure 4(g) and (h) it takes less than 25 trading periods for the aggregated market price to reach stability. This can be seen more clearly from the pattern formation plot where the top of the plot shows clear randomness, while the rest of the plot is black meaning all of the trading agents are in state 1. The aggregated market price is able to converge here mostly because of the low value of the threshold parameter z . Thus from these results we can see that the aggregated market price can be used as a predictor for future market dynamics. It can also estimate the trading period needed to reach a certain type of long-term behavior, e.g. convergence.

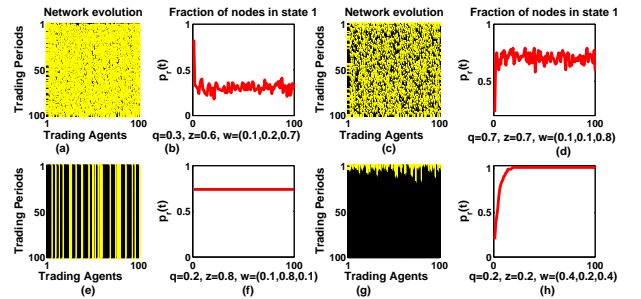


Figure 4. (a),(c),(e),(g): Pattern Formation plots for a prediction market starting with a random initial condition and the parameters specified in the associated right plots. (b),(d),(f),(h): The corresponding aggregated market price. The parameters are set as specified in the graphs.

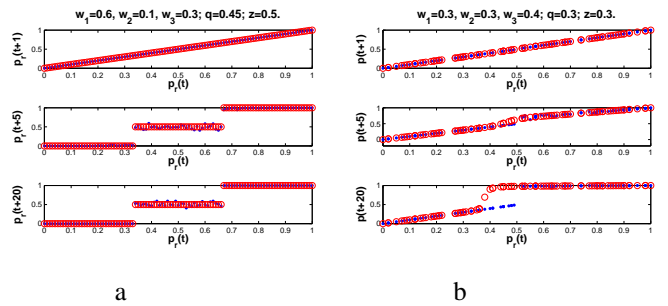


Figure 5. The system (blue dots) versus the mathematical model (red circles) for the 1st, 5th, and 20th trading periods. Note the apparent match between them.

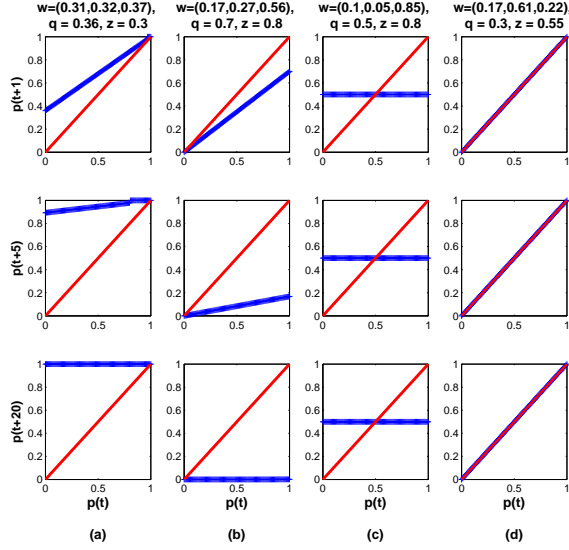


Figure 6. Graphical illustration of the iterations of the aggregated market price (density of ones map given in Equation 6) (blue ‘+’) versus the main diagonal (red line). The intersection between them yields the fixed points of the map.

We now check that the mean-field model for the aggregated market price, $p_r(t)$, derived in Equation (5) is a good approximation for the fraction of nodes in state 1 obtained by evolving the actual BN. We do this by graphing on the same plot both $p_r(t)$ and the actual fraction of trading agents in state 1 for the 1st, 5th, and the 20th trading periods as shown in Figure 5. On the x -axis we plot the initial conditions for the fraction of trading agents in state 1 ($\{0, \frac{1}{N}, \frac{2}{N}, \dots, \frac{(N-1)}{N}\}$), representing how many traders are initially in state 1, i.e. believe that the event will happen. We first apply the mathematical model to each of these initial conditions, iterate them, and plot the results with a red straight line. We then apply the prediction market evolution for a network state corresponding to each initial fraction of the trading agents in state 1, evolve the prediction market, and plot it with a blue ‘+’. For each given initial fraction of ones we randomly select trading agents that are in state 1. Figure 5 shows the comparison results for two different combinations of the parameters w, q , and z . We performed exhaustive simulations for the possible ranges of q, z , and their corresponding weights learned via neural network, and obtained similar results to those in Figure 5. We can see from Figure 5 that the first iteration matches perfectly. Then, as the prediction market and the mathematical model evolve during their transient phase, the match becomes a little less perfect due to the actual correlations that are building up in a prediction market. These correlations are ignored in the mean-field approach. Despite the assumption of no correlations, in the long run the mathematical model for $p_r(t)$ is a very good approximation of the evolution of the

aggregated market price for the actual network.

We also illustrate the behavior of our mean-field based model for the aggregated market price using BN by generating multiple iterations of the mathematical model for $p_r(t+1)$ (blue line marked with ‘+’) for various values of q and z in Figure 6. We also plot the line representing $p_r(t+1) = p_r(t)$ (red straight line). Note that the intersection of each iteration with the first diagonal generates the fixed points. As we discussed in Section III-D our system has fixed points (when $p_r(t+1) = p_r(t)$) at 0, 1 and q . We find that our prediction market mainly conforms to one of four behaviors shown in Figure 6. Figure 6(a) shows the case when the system converges to 1, while Figure 6(b) illustrates the case when the prediction market converges to 0, and Figure 6(c) shows the existence of the fixed point at q . Figure 6(d) shows the last case when $p_r(t+1) = p_r(t)$, so the two lines overlap, meaning chaos.

C. Comparison to Conventional Prediction Markets

In this section we compare the aggregated market price obtained using the BN-based prediction market model to the aggregated market price obtained with a Logarithmic Market Scoring Rule (LMSR) aggregation mechanism [4] while using the same underlying parameters. To illustrate

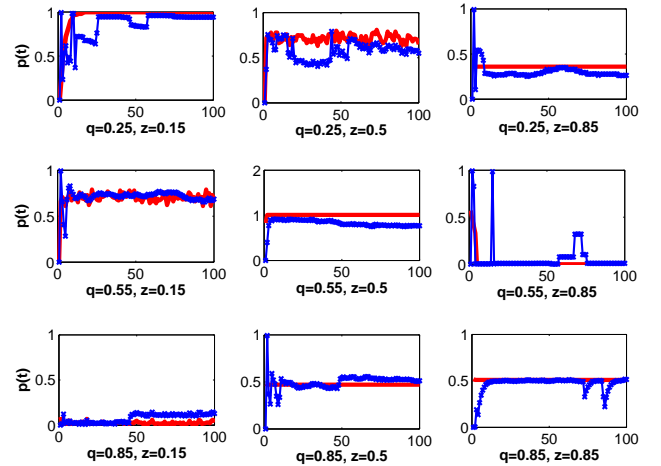


Figure 7. Comparison of the market prices set by LMSR (marked by ‘x’ - blue) and the aggregated market price using BN (marked by ‘◇’ - red) by trading periods. Note the similarities between the two models, as well as the robustness of the BN model as opposed to the increased variation of the LMSR model.

the comparison we graph both market prices on the same plot for different values of q and z . The x -axis represents the number of trading periods, while the y -axis is the market price. We can see from Figure 7 that in the long run, both the LMSR and BN models yield approximately the same results. It is also revealed that the aggregated market price

does not fluctuate as much as the LMSR price, which is a known and reported problem of the LMSR pricing [19]. We also note in Table II that the correlation coefficients between the data from the LMSR and BN models are fairly close to 1, revealing a strong correlation between them. Thus, the BN-based model is a realistic model of prediction markets.

q	z	Correlation
0.25	0.15	0.7925
0.25	0.5	0.7421
0.25	0.85	0.8556
0.55	0.15	0.8827
0.55	0.5	0.7912
0.55	0.85	0.7591
0.85	0.15	0.8404
0.85	0.5	0.8295
0.85	0.85	0.8661

Table II
CORRELATIONS BETWEEN THE LMSR MARKET PRICE AND THE AGGREGATED MARKET PRICE USING BN FROM FIGURE 7. OBSERVE THAT THE NUMBERS ARE FAIRLY CLOSE TO 1 WHICH INDICATES A SIGNIFICANT CORRELATION BETWEEN THE LMSR AND BN MODELS.

D. Robustness to noise

It is known that real networks (biological/genetic, physical, neural, chemical, social, financial etc.) are always subject to disturbances and have the ability to reach functional diversity and aim to maintain the same state under environmental noise. Prediction markets can also be affected by some disturbances in the form of manipulation by the trading agents that reveal their beliefs untruthfully. For example, in the Tradesports 2004 presidential markets there was an apparent manipulation effort. An anonymous trader sold many securities corresponding to the event “George W. Bush will win the 2004 Presidential elections” at a very low value. This caused the market price to be driven to zero, implying a zero percent chance of the event happening. However, this manipulation effort failed, as the market price of the security related to this event rebounded rapidly to its previous level [20]. As prediction markets get more attention and become more widely known among the public, it is likely that some individuals or groups will be motivated to manipulate them. Inducing disturbance in the system by changing the value of certain nodes in the network (according to a deterministic or stochastic rule) is a good model for an environmental or intrinsic type of perturbation. A similar procedure has been used for example by Bilke and Sjunnesson [2] where one randomly chosen variable is inverted after the system has reached a limit cycle in the Kauffman model, or by Goodrich and Matache [12] who show that the introduction of noise can stabilize a certain type of BN for a wide range of parameters. We will analyze the response to disturbances of the prediction market in this paper under a simple noise process to assess the robustness of the BN-based model to potential non-truthful trading agents.

We employ the following noise procedure, called the “flip” rule: at each trading period t we randomly select j trading agents and flip their state before applying the Boolean rule. This procedure has been used in [12]. Since the number of zeros and ones changes due to the perturbation, the value of $p_r(t)$ is modified prior to the application of the model (5). Now, if j nodes are chosen at random, then $j \cdot p_r(t)$ of them are in state 1 and $j \cdot (1 - p_r(t))$ are in state 0. By the flip rule, the total number of trading agents in state 1 is decreased by $j \cdot p_r(t)$ since they are changed to 0. On the other hand, the number is increased by $j \cdot (1 - p_r(t))$ since the zeros become ones. Thus, the proportion of trading agents in state 1, that is $p_r(t)$, becomes $p_r(t) \frac{j \cdot p_r(t)}{N} + \frac{j \cdot (1 - p_r(t))}{N} = p_r(t) + \frac{j}{N} (1 - 2p_r(t))$. Clearly this number is in $[0, 1]$. Then the formula (5) can be written as follows:

$$p_r(t+1) = \quad (7)$$

$$F_B \left(\frac{z - w_1(p_r(t) + \frac{j}{N}(1 - 2p_r(t)))}{w_3} \right) (1 - p_r(t) - \frac{j}{N}(1 - 2p_r(t))) + F_B \left(\frac{z - w_1(p_r(t) + \frac{j}{N}(1 - 2p_r(t))) - w_2}{w_3} \right) (p_r(t) + \frac{j}{N}(1 - 2p_r(t))).$$

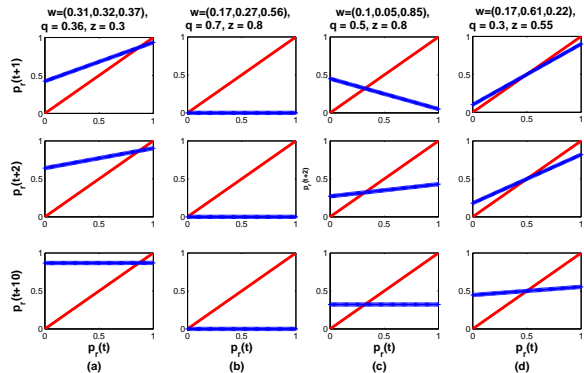


Figure 8. Iterations of the noisy aggregated market price (blue ‘+’) with $j = 10$ versus the main diagonal (red line). We used the same parameters as in Figure 6.

Figure 8 illustrates iteration plots analog to those in Figure 6 (for the same parameter values), but with induced perturbations. These results show that the noise generated by the “flip rule” can stabilize the prediction market as seen from the Figure 8(d). In that case the prediction market was chaotic without noise, and now it stabilizes around 0.5. This result supports the result obtained by Hanson [13], where he showed that the manipulator in the prediction market can aid its accuracy. For other parameter combinations, noise may change the value of the fixed points, maintaining stability, as can be seen in the other plots of Figure 8. The fixed points changed from 1 to 0.9 (Figure 8(a)), and from 0.5 to 0.4 (Figure 8(c)). From Table III we can see that the Euclidian distance between the aggregated market price without noise from Figure 6 and with noise from Figure 8 is greater for

Parameters	Euclidian distance
$w = (0.31, 0.32, 0.37), q = 0.36, z = 0.3, time = t + 1$	2.0661
$w = (0.17, 0.27, 0.56), q = 0.7, z = 0.8, time = t + 1$	3.3728
$w = (0.1, 0.05, 0.85), q = 0.5, z = 0.8, time = t + 1$	4.2224
$w = (0.17, 0.61, 0.22), q = 0.3, z = 0.55, time = t + 1$	4.7510
$w = (0.31, 0.32, 0.37), q = 0.36, z = 0.3, time = t + 2$	5.0965
$w = (0.17, 0.27, 0.56), q = 0.7, z = 0.8, time = t + 2$	5.3222
$w = (0.1, 0.05, 0.85), q = 0.5, z = 0.8, time = t + 2$	5.4979
$w = (0.17, 0.61, 0.22), q = 0.3, z = 0.55, time = t + 2$	5.6092
$w = (0.31, 0.32, 0.37), q = 0.36, z = 0.3, time = t + 10$	5.7047
$w = (0.17, 0.27, 0.56), q = 0.7, z = 0.8, time = t + 10$	5.7094
$w = (0.1, 0.05, 0.85), q = 0.5, z = 0.8, time = t + 10$	5.7178
$w = (0.17, 0.61, 0.22), q = 0.3, z = 0.55, time = t + 10$	5.7290

Table III

EUCLIDIAN DISTANCE BETWEEN THE AGGREGATED MARKET PRICE WITHOUT NOISE FROM FIGURE 6 AND THE AGGREGATED MARKET PRICE WITH NOISE FROM FIGURE 8.

larger time periods. Also, the distance is greater when there is a fixed point at q (Figure 6 (c)) and when $p_r(t+1) = p_r(t)$ (Figure 6 (d)). In Figure 9 we show similar iteration plots but for parameter combinations that yield piecewise functions. We note that there may be multiple fixed points this time. However, all of them are stable since the derivative at those points is always less than 1. Therefore, the stability of the prediction market is either preserved or induced by the introduction of noise.

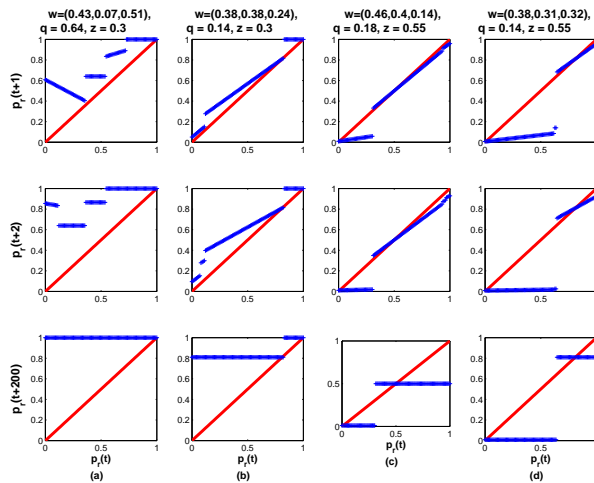


Figure 9. Iterations of the noisy aggregated market price (blue '+') with $j = 5$ versus the main diagonal (red line). Note that these parameter values yield piecewise functions with the possibility of multiple fixed points that are stable.

E. Scalability

In this section we test the scalability of our prediction market and analyze how the changes in the number of trading agents affect the dynamics of the prediction market. Figure 10 shows that our mean-field based model's accuracy for $p_r(t)$, the aggregated market price, improves as the

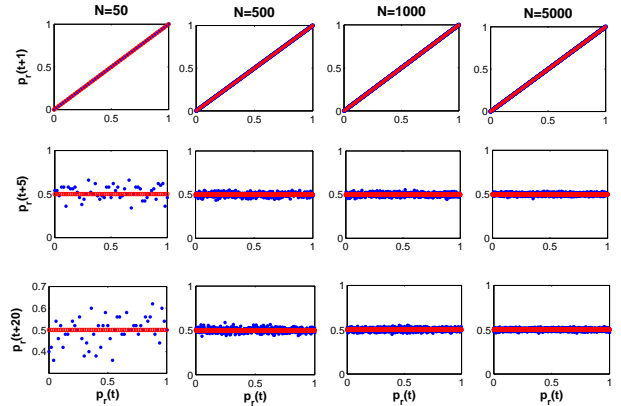


Figure 10. The system (blue dots) versus the mathematical model (red circles) under "flip" noise procedure for 50, 500, 1000, 5000 trading agents for the 1st, 5th, and 20th trading periods. The parameters are fixed as follows: $w_1 = 0.1, w_2 = 0.05, w_3 = 0.85, z = 0.8, q = 0.5$.

number of trading agents increases. This is expected, since a mean-field formula is valid in the limit $N \rightarrow \infty$. Figure

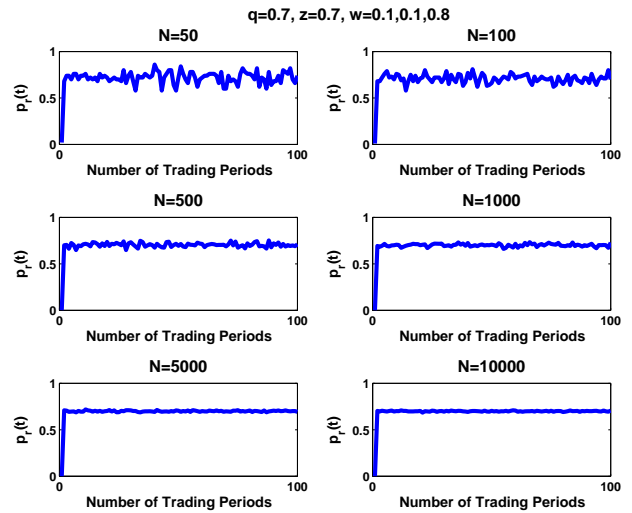


Figure 11. The aggregated market price by time steps, for $N = 50, 100, 500, 1000, 5000, 10000$.

11 shows the aggregated market price using a BN for 50, 100, 500, 1000, 5000, and 10000 trading agents for $q = 0.7, z = 0.7, w_1, w_2, w_3 = 0.1, 0.1, 0.8$. This combination of z and q parameters yields a more dynamic behavior of the prediction market as seen in Figure 4 (c,d), however here we can see that as the number of trading agents increase the aggregated market price becomes less dynamic. However, there is not much difference in the market price dynamics when $N = 5000$ and when $N = 10000$, leading us to believe that in this case 5000 trading agents are enough to lead to an accurate prediction market.

V. CONCLUSION

In this paper, we have described a Boolean network based prediction market and used it to calculate the aggregated market price and analyze the behavior of the trading agent population in response to various market parameters such as information flow or past beliefs. We show that the BN approach gives results similar to the LMSR model with less fluctuations of the market price. In addition to proposing a new method to calculate the aggregated market price using BN and mean-field based mathematical modeling, we also show how it can be used to analyze and predict the dynamics of the prediction market.

In the future we plan on acquiring some reliable real prediction market data and generating a statistical comparison of the market prices generated by our model and the real data. It is possible that some assumptions may need to be adjusted to account for any type of extra information we may be able to derive from real data. However, the main purpose of this paper was to show that a Boolean network approach is reasonable and provides some advantages despite being a simpler model than other conventional models. Also, in order to make the model more realistic we plan to allow for variation of weight and threshold parameters and also account for asynchronous information transmission by the trading agents and the market maker agent. The asynchrony can be applied in a deterministic or stochastic way, and we will analyze the importance of the type of asynchrony on the dynamics of the network. We also plan on making use of the existing, game theory-based trading strategies and translating them into a set of Boolean rules that will govern the dynamics of the network. We may refine the Boolean approach to more than two possible states. Finally, we are interested in exploring truthful revelation mechanisms that can be used to limit untruthful bidding in prediction markets.

Acknowledgments. This research has been partially supported as a part of the COMRADES project funded by the U.S. DoD Office of Naval Research, grant number N000140911174.

REFERENCES

- [1] G.L. Beck and M.T. Matache, "Dynamical behavior and influence of stochastic noise on certain generalized Boolean networks," *Physica A*, Vol. 387(19-20), 2008, pp. 4947-4958.
- [2] S. Bilke and F. Sjunnesson, "Stability of the Kauffman Model," *Physical Review*, Vol. 65(1), 2001.
- [3] Y. Chen, T. Mullen, and C-H. Chu, "Theoretical Investigation of Prediction Markets with Aggregate Uncertainty," *Proc. of the 7th International Conference on Electronic Commerce Research (ICECR)*, 2004, pp. 81-90.
- [4] Y. Chen and D. Pennock, "Utility Framework for Bounded-Loss Market Maker," *Proc. of the 23rd Conference on Uncertainty in Artificial Intelligence (UAI)*, 2007, pp. 49-56.
- [5] S. Das, "The Effects of Market-making on Price Dynamics," *Proc. of the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2008, pp. 887-894.
- [6] S. Dimitrov and R. Sami, "Nonmyopic Strategies in Prediction Markets," *Proc. of ACM Conference on Electronic Commerce (EC)*, 2008, pp. 200-209.
- [7] J. Feigenbaum, L. Fortnow, D. Pennock, and R. Sami, "Computation in a Distributed Information Market," *Theor. Comp. Science*, Vol. 343(1-2), 2005, pp. 114-132.
- [8] S. Gjerstad, "Risks, Aversions and Beliefs in Predictions Markets," mimeo, U. of Arizona, 2004.
- [9] L.R. Glosten and P.R. Milgrom, "Bid, Ask and Transaction Prices in a Specialist Market with Heterogeneously Informed Traders," *Journal of Financial Economics*, Vol. 14, 1985, pp. 71-100.
- [10] B. Golub and M. Jackson, "Naive Learning in Social Networks and the Wisdom of Crowds," *American Economic Journal*, Vol. 2(1), 2010, pp. 112-149.
- [11] R. Goodman and M.T. Matache, "Dynamics of Directed Boolean Networks under Generalized Elementary Cellular Automata Rules, with Power-Law Distributions and Popularity Assignment of Parent Nodes," *Complex Systems*, Vol. 17(4), 2008, pp. 357-379.
- [12] C.S. Goodrich and M.T. Matache, "The stabilizing Effect of Noise on the Dynamics of a Boolean Network," *Physica A*, Vol. 379, 2007, pp. 334-356.
- [13] R. Hanson, "A Manipulator Can Aid Prediction Market Accuracy," *Economica*, Vol. 76, 2009, pp. 304-314.
- [14] L. Jian and R. Sami, "Aggregation and Manipulation in Prediction Markets: Effects of Trading Mechanism and Information Distribution," *Proc. of ACM Conference on Electronic Commerce (EC)*, 2010, pp. 207.
- [15] J. Jumadinova and P. Dasgupta, "A Multi-Agent System for Analyzing the Effect of Information on Prediction Markets," *International Journal of Intelligent Systems*, Vol. 26(5), 2011, pp. 383-409.
- [16] S.A. Kauffman, *The Origins of Order*, Oxford University Press, 1993.
- [17] C. Manski, "Interpreting the Predictions of Prediction Markets," *Economic Letters*, Vol. 91(3), 2006, pp. 425-429.
- [18] M.T. Matache and J. Heidel, "Random Boolean Network Model Exhibiting Deterministic Chaos," *Phys. Rev. E*, Vol. 69, 2004.
- [19] A. Othman, D. Pennock, D. Reeves, and T. Sandholm, "A Practical Liquidity-Sensitive Automated Market Maker," *Proc. of ACM Conference on Electronic Commerce (EC)*, 2010, pp. 377-386.
- [20] P. Rhode and S. Koleman, "Manipulation in Political Stock Markets," mimeograph, University of North Carolina, 2005.
- [21] I. Shmulevich, E.R. Dougherty, and W. Zhang, "From Boolean to Probabilistic Boolean Networks as Models for Genetic Regulatory Networks," *Proceedings of the IEEE*, Vol. 90(11), 2002, pp. 1778-1792.
- [22] S. Russel and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2002.
- [23] J. Wolfers and E. Zitzewitz, "Prediction Markets," *Journal of Economic Perspectives*, Vol. 18(2), 2004, pp. 107-126.
- [24] S. Wolfram, *A new Kind of Science*, Wolfram Media, Champaign, 2002.