Probabilistic Modeling of Information Dynamics in Networked Cyber-Physical-Social Systems

Yan Wang

***Abstract*—Cyber-physical-social systems (CPSS) are physical devices with highly integrated functions of sensing, computing, communication and control, and are seamlessly embedded in human society. The levels of intelligence and functions that CPSS can perform rely on their extensive collaboration and information sharing through networks. In this paper, information diffusion within CPSS networks is studied. Information dynamics models are proposed to characterize the evolution of information processing and decision making capabilities of individual CPSS nodes. The data-driven statistical models are based on a mesoscale probabilistic graph model, where the individual nodes' sensing and computing functions are represented as the probabilities of correct predictions, whereas the communication functions are represented as the probabilities of mutual influences between nodes. A copula dynamics model is proposed to explicitly capture the information dependency among individuals with joint prediction probabilities and estimated from extremal probabilities. A topology-informed vector autoregression model is proposed to represent the mutual influence of prediction capabilities. A spatial-temporal hybrid Gaussian process regression model is also proposed to simultaneously capture correlations between nodes and temporal correlation in the time series.**

***Index Terms*—Cyber-Physical Systems, Information Diffusion, Graph Theory, Copula, Vector Autoregression, Gaussian Process Regression.**

I. Introduction

CYBER-physical systems (CPS) are physical devices with highly integrated functions of sensing, computing, communication, and actuation. They share information, work collaboratively, and form networks, also known as Internet of Things (IoT). Such devices can have different sizes and physical forms at micro- or macro-scales. They are the essential elements in smart home and smart office, intelligent manufacturing, personalized medicine, autonomous and safe transportation, omnipresent energy supplies, and many other applications. Given the intensive interactions between CPS and human society as well as human's heavy reliance on CPS, the terms cyber-physical-social systems (CPSS) and social Internet of things are also used by researchers to describe the highly integrated systems with the additional social dimension. In CPSS, information collection, processing, and decision making are done in a decentralized fashion. The intelligence level of CPSS is enhanced by intensive information sharing. The level of dependencies among CPSS devices for their computation and decision making is unprecedented.

There are unique engineering challenges in designing CPSS. First, given the evolution nature of cyber and physical technologies, adaptability that enables the capabilities of self-learning, self-organization, and context awareness is important to design open systems that can evolve along technology advancement [1], [2]. Adaptability goes beyond traditional closed-loop control. Intelligent agents need to consider local behaviors, soft constraints, and uncertainty for their reasoning [3]. Second, the complexity of the CPSS has significantly increased from traditional products and devices. The CPSS products are connected through IoT and heavily rely on data exchange from each other to realize their functions. Communication between devices plays a major role. Therefore, how to design systems of CPSS which have dependable communication is important. Traditional security measurements along do not guarantee dependability of large IoT systems with heterogeneous hardware and software protocols [4]. Reliable large-scale networked systems that do not fail are impossible to achieve. Resilient systems that can recover automatically from partial failures are more likely to be realizable [5], [6]. In addition, the coordination for the complex distributed systems with aggregated information is challenging [7]. Third, the high-dimensional design space of CPSS includes not only the cyber and physical subspaces, but also the social subspace. Examples of the emerging research issues are how to design the modalities for human-system interaction [8], how to enable context awareness and personalized communication between CPSS and humans [9], how to provide incentives to improve reputations [10], how to protect user privacy by sharing partial data in physical domain without affecting social status [11], and how to quantify trustworthy strategic relationships for information sharing [12], [13].

Compared to traditional products, the design of CPSS requires engineers to have better understanding of the systems level behaviors. One such behavior is the dynamics of information flow in the CPSS networks. In a dynamically evolving CPSS network, the effects of information generation and sharing need to be quantified and analyzed so that the long-term behaviors of such networks can be predicted. The understanding of the deep dependency and mutual influences between CPSS nodes in decision making is essential to design such systems. Examples of the design questions are how and where sensors or information source nodes need to be placed to ensure certain information about the surrounding environment is fully aware of in the network, how soon the network reaches the consensus or the equilibrium state of

This work was supported by National Science Foundation under grant CMMI-1663227. The author is with Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332, USA (e-mail: yan-wang@gatech.edu)

information propagation, and what if a disruption occurs. Thus modeling the information dynamics is useful to test systems engineering strategies towards the system design of networks for adaptability and scalability.

The information flow in computer networks and social networks has been studied extensively. The propagation of information can be modeled in different ways such as the epidemic model [14], influence model [15], and the event-driven approaches [16], [17]. However, there is still a lack of studies on information diffusion in CPSS networks. CPSS networks have different behaviors from traditional computer networks and social networks, because CPSS nodes possess additional functionalities of sensing and actuation. Not just simply forwarding the received information, CPSS nodes continuously generate a large amount of new data or information based on both what they receive and what they collect themselves. Decisions of collecting and sharing information are done locally by individual agents in the distributed systems. Furthermore, the actuation or control function of a CPSS node is intended to change its external environment thus affecting the neighboring nodes at a much larger scale. The CPSS and the external environment become tightly coupled. The behavior of a CPSS node can be affected by and simultaneously affect many other nodes in the network. Therefore the behaviors of CPSS nodes, and thus the information being shared, are highly correlated.

In this paper, the evolution of CPSS behaviors in a network as a result of information dynamics is studied. A data-driven statistical modeling approach is taken to model how CPSS nodes have influences on each other when information is exchanged and how the behaviors of nodes evolve dynamically, given that uncertainties are associated with sensing and communication. The proposed information dynamics models are based on a generic probabilistic graph model of CPSS networks [6], [18], where information exchange and processing at nodes are modeled at the mesoscale. The mesoscale model quantifies the overall functionalities of sensing, computing, and communication of each node with probabilities and simulates directly with those probability values, instead of at the fine-grained scale of packets or individual messages as in most network simulators. In the probabilistic graph model, the sensing and computing capability of each node is characterized by a prediction probability, whereas the communication capabilities between nodes are captured by pairwise reliance probabilities. The prediction probability associated with a node measures how well the node can utilize the available information, make sound decision, and accurately predict the true state of the world. The reliance probability associated with a pair of nodes measures the extent of influence that the information source node can make to the receiving node.

Instead of explicitly modeling the propagation of information elements in existing information dynamics models, the proposed information dynamics models predict the evolution of node behaviors as a result of information exchange. The behavior of each node is characterized by its capability of sensing and computing and quantified with the prediction probability. Here it is also referred to as prediction capability. The prediction accuracy of nodes is influenced by each other, given that the decision of each node is made based on information gathered from itself as well as its neighboring nodes. In addition, information dependency is explicitly captured here with the proposed models, given that the information sent by a node is based on its prediction capability and the capability is highly influenced by the information it receives from others.

Two types of models are proposed. The first one is called copula dynamics modeling, where correlations of prediction capabilities between nodes are represented by joint probabilities or copulas and approximated by extremal probabilities. The dynamics is modeled with the time series of copulas. The second type is called functional interdependency modeling, where the correlations are captured by linear or nonlinear functional relationships. We demonstrate a new topology-informed vector autoregression model where the topology of networks is applied to improve the efficiency and interpretability of regression. We also present a new Gaussian process regression model for time series where the spatial-temporal correlations are simultaneously captured as a result of information sharing between nodes. The novelties of the proposed models include the new concepts of copula dynamics modeling, topology-constrained vector autoregression, and hybrid discrete-continuous kernels in Gaussian process regression for time series.

In the remainder of the paper, Section II provides the background of this work, including an overview of information dynamics modeling in traditional computer networks and social networks in Section II-A and the probabilistic graph model in Section II-B. In Section III, the copula dynamics modeling approach is introduced where extremal probabilities are used to estimate the joint probabilities and their dynamics are modeled as time series. In Section IV, the proposed functional interdependency modeling approach is introduced where new topology-constrained vector autoregression and Gaussian process regression models are described. In Section V, the three models are demonstrated with CPSS network simulation data.

II. Background

Here, the relevant work of modeling information diffusion in networks is reviewed. The probabilistic graph model that the proposed models are based on is also introduced.

A. Information Dynamics Modeling for Networks

Several approaches have been taken to model the information diffusion in computer networks. In the epidemic modeling approach, information propagation is treated in the same way as disease spread. The transmission probability, which is the probability that a piece of information is shared to a neighboring node, is the main parameter to study the diffusion process. Probabilistic models of edge and node percolation have been developed to study the speed of propagation within networks [14]. Alternatively, the susceptible-infected-recovered (SIR) epidemic model was also used to model information diffusion in populations without explicitly considering network topology [19], [20] The SIR model has also been extended with the considerations of additional factors such as contacted state

[21], exposed state [22], immunization [23], and uncertain state [24].

In the event-driven modeling approaches, the adoptions of new information by nodes are characterized by discrete Poisson processes [16], [25], non-Poisson [17], non-homogeneous Poisson process with memory effect [26], branching process [27], and continuous hazard function [28].

The dynamics of influence and popularity have been studied particularly for social networks [29], [30]. A node's influence is characterized by the frequencies of its generated information being forwarded. Influence has been modeled as time-dependent accumulative functions [15]. The spatial-temporal influence propagation can be modeled with partial differential equations such as diffusive logistic model [31] and hydrodynamic model [32]. In the reinforced Poisson process model [33], the transmission rate is dependent on other factors such as popularity and aging effect of information. To incorporate social behaviors in information forwarding in social networks, game theory [34], [35], incentive mechanisms for ego networks [36], as well as government intervention [37] were studied for their effects on information propagation.

These models have been widely applied to study the propagation of keywords and phrases among blogs [38] and within social networks [39]. From the collected data and observations, the parameters of models [40] or the network structure [41] can also be inferred and recovered. Other design problems such as finding a subset of nodes to maximize the speed of diffusion [42] and searching for the influential spreaders [43] have also been studied.

The above modeling approaches are from the perspective of how specific pieces of information such as keywords and ideas being propagated within the network after they are created. One unique aspect of CPSS is that new information is continuously being generated with each node's sensing and reasoning capability. Furthermore, the information that a CPSS device receives can directly affect how additional new information is generated locally for the next moment. The actuation function of CPSS also results in strong coupling between the systems and the surrounding environment. Therefore, the information dynamics modeling for CPSS networks needs to incorporate the deep interdependency between nodes.

There is only limited work in information propagation in CPS or CPSS. Yagan et al. [44] investigated the information transmission within a conjoint social and physical network in the context of epidemic models. Lu et al. [45] developed an algorithm to maximize the information diffusion between nodes which are connected probabilistically. Wang et al. [46] studied the diffusion with the epidemic model when nodes make local decisions of whether forwarding information to others based on the game theory. Yi et al. [47] extended the dimension of states for each node into both physical and social spaces for the epidemic model so that the influence of social behavior on information propagation is modeled. Here a different approach is taken to model the information dynamics in CPSS networks. The effect of information diffusion is modeled instead, based on a mesoscale probabilistic graph model of CPSS networks.

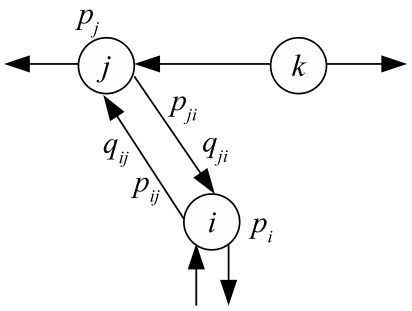


Fig. 1: The probabilistic graph model

B. Probabilistic Graph Model

The information dynamics models presented here are based on a mesoscale description of CPSS networks. The probabilistic graph model was proposed to provide an abstraction of information collection, processing, and exchange between CPSS in an IoT environment [5], [6], [18]. The model is introduced as follows.

A probabilistic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{P}, \mathcal{R})$, where $\mathcal{V} = \{v_k\}$ is a set of nodes, $\mathcal{E} = \{(v_i, v_j)\}$ is a set of directed edges, as shown in Fig. 1. Each node v_k is associated with a *prediction probability* $p_k \in \mathcal{P}$, whereas each edge (v_i, v_j) is associated with some *reliance probabilities* $p_{ij} \in \mathcal{R}$. The prediction probability p_k is the probability that node k detects the true state of world θ and is defined as

$$P(x_k = \theta) = p_k, \tag{1}$$

where x_k is the state variable of node k. With loss of generality, here only binary-valued state variables are considered. The information dependency between node j and node i is described by P-reliance probability

$$P(x_j = \theta | x_i = \theta) = p_{ij}, \tag{2}$$

and Q-reliance probability

$$P(x_j = \theta | x_i \neq \theta) = q_{ij}. \tag{3}$$

P-reliance probability indicates the positive effect of information exchange between nodes, whereas Q-reliance probability captures the negative influence. It is also possible to have $P(x_k = \theta | x_k = \theta)$ and $P(x_k = \theta | x_k \neq \theta)$ indicating how much a node's prediction relies on its own observation.

Each CPSS node collects information by its own sensor and from its neighbors. With the new information, the prediction probabilities are updated. Different *information fusion rules* can be adopted by nodes to update their prediction probabilities. Example rules include *best-case*, *worst-case*, and *Bayesian* rules. They are listed as follows. To simplify the notation, we use $P(x_k)$ to denote $P(x_k = \theta)$, $P(x_k^c)$ to denote $P(x_k \neq \theta)$, and $P(x_j|x_i)$ to denote $P(x_j = \theta | x_i = \theta)$.

The *best-case* or *optimistic* fusion rule is

$$P'(x_k) = 1 - \Pi_{i=1}^{M}(1 - P(x_k|x_i)), \tag{4}$$

where node k has a positive prediction with updated probability P' if any of the M nodes as its information sources

provides a positive cue. Some variations of this rule can also be used, such as

$$P'(x_k) = 1 - \Pi_{i=1}^{M_1}(1 - P(x_k|x_i))\Pi_{j=1}^{M_2}(1 - P(x_k|x_j^c)), \quad (5)$$

where both positive cues from M_1 nodes and negative cues from M_2 nodes ($M_1 + M_2 = M$) are considered. Another version could be

$$P'(x_k) = 1 - \Pi_{i=1,i\neq k}^{M}(1 - P(x_k|x_i)), \quad (6)$$

where the node's own prior observation is not included in the update.

The *worst-case* or *pessimistic* fusion rule is

$$P'(x_k) = \Pi_{i=1}^{M} P(x_k|x_i), \quad (7)$$

where the prediction of a node is positive only if all cues it receives from other nodes are positive. Similarly, some variations of the rule exist, such as

$$P'(x_k) = \Pi_{i=1}^{M_1} P(x_k|x_i)\Pi_{j=1}^{M_2} P(x_k|x_j^c), \quad (8)$$

and

$$P'(x_k) = \Pi_{i=1,i\neq k}^{M} P(x_k|x_i). \quad (9)$$

The *Bayesian* fusion rule is

$$P'(x_k) \propto P(x_k)\left[(P(x_k))^r\,(1 - P(x_k))^{M-r}\right], \quad (10)$$

where the prediction of node k is updated from prior prediction probability $P(x_k)$ given that r out of a total of M cues provided by others are positive.

Further generalization of the above binary-valued state variables to multi-valued discrete state variables is straightforward. Suppose there are a finite set of discrete values $\{\theta_1, \ldots, \theta_N\}$ that the state variable x_k can take. The multi-valued prediction probability $P(x_k = \theta_n)$ ($n \in \{1, \ldots, N\}$) can be obtained similar to binary values. Similarly, reliance probabilities $P(x_j = \theta_n|x_i = \theta_m)$ ($m, n \in \{1, \ldots, N\}$) can be obtained enumeratively. Continuous state variables can be discretized in digital implementation.

The above information fusion rules can be similarly extended to multi-valued state variables. For instance, the optimistic fusion rule in Eq.(5) becomes

$$\begin{aligned} P'(x_k) = 1 - &\Pi_{i=1}^{M_1}(1 - P(x_k|x_i = \theta_1)) \cdots \\ &\Pi_{j=1}^{M_N}(1 - P(x_k|x_j = \theta_N)), \end{aligned} \quad (11)$$

whereas the pessimistic fusion rule in Eq.(8) becomes

$$P'(x_k) = \Pi_{i=1}^{M_1} P(x_k|x_i = \theta_1) \cdots \Pi_{j=1}^{M_N} P(x_k|x_j = \theta_N), \quad (12)$$

where $M_1 + \cdots + M_N = M$. Obviously, if only one of the N values is true or of concerned, the problem setting can be simplified and converted back to the binary case.

Based on the probabilistic graph model, we propose two types of information dynamics models to capture the evolution of CPSS networks where nodes produce information about the state of the world. New information is produced by sensing and computing units of nodes. When a node receives some information from others, the received information is combined and digested, which is then used to update the prediction of the node. The prediction probabilities of nodes are dynamically updated with the mutual influences among each other. Thus the influences and interdependency between information producers and consumers need to be modeled. The strong correlation between nodes is explicitly modeled in the proposed information dynamics models. In the copula dynamics modeling, the dynamic changes of joint prediction probabilities among nodes along time are used to capture the correlation. In the functional interdependency modeling, the correlation is represented as either linear or nonlinear functional relationships between prediction probabilities, and the dynamics of prediction probabilities is modeled directly. In the proposed two-dimensional time-series models, while structural dependency in the graph is modeled as the correlations between the state variables, which are prediction probabilities indexed with nodes, time dependency between the variables is captured with the discrete time series. Because these models capture the correlations directly, the global effect of local changes can be predicted as a result of strong structural and temporal dependencies between variables.

III. Copula Dynamics Modeling

The copula $C : [0,1]^n \longrightarrow [0,1]$ of a random vector $(X_1, \ldots, X_n)$ is a function that maps the marginal cumulative distribution functions $F(X_1), \ldots, F(X_n)$ to the joint cumulative distribution $F(X_1, \ldots, X_n) = C(F(X_1), \ldots, F(X_n))$. In the copula dynamics modeling approach, the joint probabilities of predictions among nodes are used to model the information dynamics. The joint probabilities with a large number of variables however are not easy to be calculated directly, especially given that the number of nodes in a network can be very large. Therefore copulas are proposed here to estimate the joint probabilities of prediction. The precise joint probabilities or copulas are however unknown. They can be estimated from the ones in some extreme scenarios, such as perfectly positive or negative correlation, or completely independence. The corresponding joint probabilities are known as extremal probabilities. The extremal probabilities of perfect correlations or independence form the bounds of the actual but unknown copulas and can be regarded as the vertices that form a convex hull of unknown joint probabilities in the space of distributions. If the copulas for extremal probabilities can be obtained, then the actual joint probabilities can be estimated. From the joint probabilities, the marginal probabilities of predictions can be easily calculated. In the information dynamics modeling, the evolutions of extremal probabilities are explicitly modeled. Time series models of the copulas for extremal probabilities can be trained from experimental or simulation data. Then the future behavior of the system can be predicted. Because the number of copulas grows exponentially as the number of nodes increases, this approach however is prohibitively expensive for large-scale networks.

Given the joint probability $P[x_1 = \theta, \cdots, x_n = \theta]$ for the case that all predictions are positive and simply denoted as $P(x_1, \cdots, x_n)$, the corresponding copula is defined as

$$\begin{aligned} C(p_1, \cdots, p_n) &= P[P(x_1) \leq p_1, \cdots, P(x_n) \leq p_n] \\ &= P(x_1, \cdots, x_n) \end{aligned} \quad (13)$$

This is due to the simplicity of $P(x_i \leq \theta) = P(x_i = \theta)$ in the case of binary values $x_i = \theta$ or $x_i \neq \theta$. The copulas are typically difficult to calculate. But their bounds, known as extremal probabilities, are much easier to obtain.

A. Extremal Joint Probabilities

The well-known Fréchet bounds of the copula $C(P(x_1), \ldots, P(x_n))$ are given as

$$\begin{aligned} &\max\{0, P(x_1) + \cdots + P(x_n) + 1 - n\} \leq \\ &C(P(x_1), \ldots, P(x_n)) \leq \min\{P(x_1), \ldots, P(x_n)\} \end{aligned} \tag{14}$$

where the lower bound corresponds to the perfectly negative correlation, whereas the upper bound corresponds to the perfectly positive correlation. For two random variables, the bounds are

$$\begin{aligned} &\max\{0, P(x_1) + P(x_2) - 1\} \leq \\ &C(P(x_1), P(x_2)) \leq \min\{P(x_1), P(x_2)\} \end{aligned} \tag{15}$$

The bounds for perfectly positive and negative correlations are regarded as the extremal probabilities. The perfectly positive correlation case is also called comonotonic, whereas the perfectly negative correlation case is called countercomonotonic. There are different ways to quantify correlation. In the sense of linear correlation, defined as $\rho(x, y) = Cov(x, y)/\sqrt{\sigma^2(x)\sigma^2(y)}$, the linear correlation coefficient takes the maximum value $\rho = +1$ for the perfectly positive linear dependency and the minimum value $\rho = -1$ for the perfectly negative linear dependency. Other correlation definitions include Spearman's rank correlation $\rho_s(x, y) = \rho(F_x(x), F_y(y))$ defined by the linear correlation of random variables' distribution functions, Kendall's rank correlation $\rho_\tau(x, y) = P[(x_1 - x_2)(y_1 - y_2) > 0] - P[(x_1 - x_2)(y_1 - y_2) < 0]$ defined by the probability of monotonicity trend in the random values.

The extremal probabilities can be extended to multiple variables or nodes. The nodes in $\mathcal{V} = \{\mathcal{V}_1, \mathcal{V}_2\}$ are categorized into two subsets. Within the first subset $\mathcal{V}_1 = \{x_1, \ldots, x_m\}$, all nodes are perfectly positively correlated. Within the second subset $\mathcal{V}_2 = \{x_{m+1}, \ldots, x_n\}$, all nodes are also perfectly positively correlated. However, between any node in $\mathcal{V}_1$ and another one in $\mathcal{V}_2$, they are negatively correlated. That is, the opposite opinions are formed between the two homogenized groups. The predictions between the two groups are contradictory. Given the partition $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2$, when the correlation within either group is perfectly positive but perfectly negative between the two groups, the extremal joint probability is

$$\begin{aligned} &C_{\{\mathcal{V}_1, \mathcal{V}_2\}}(P(x_1), \ldots, P(x_n)) = P(x_1, \ldots, x_n | \{\mathcal{V}_1, \mathcal{V}_2\}) \\ &\quad = \max\{0, \min_{i \in \mathcal{V}_1}\{P(x_i)\} + \min_{j \in \mathcal{V}_2}\{P(x_j)\} - 1\} \end{aligned} \tag{16}$$

When all nodes have perfectly positive correlation without partition, the extremal joint probability is

$$\begin{aligned} &C_{\{\mathcal{V}^+\}}(P(x_1), \ldots, P(x_n)) \\ &= P(x_1, \ldots, x_n | \{\mathcal{V}^+\}) = \min_{i \in \mathcal{V}}\{P(x_i)\} \end{aligned} \tag{17}$$

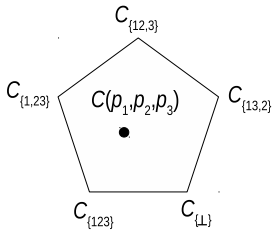


Fig. 2: The unknown copula is bounded by the convex hull formed by five extremal joint probabilities in three-node networks

Another extremal joint probability is

$$\begin{aligned} &C_{\{\mathcal{V}^\perp\}}(P(x_1), \ldots, P(x_n)) \\ &= P(x_1, \ldots, x_n | \{\mathcal{V}^\perp\}) = \prod_{i \in \mathcal{V}}\{P(x_i)\} \end{aligned} \tag{18}$$

when all nodes are independent from each other.

Although the precise form of the copula $C(P(x_1), \ldots, P(x_n))$ is unknown, it can be approximated by the combination of extremal distributions, based on the decomposition principle as

$$\begin{aligned} &C(P(x_1), \ldots, P(x_n)) \\ &= \alpha C_{\{\mathcal{V}^+\}} + \sum_{j=1}^{N} \beta_j C_{\{\mathcal{V}_1, \mathcal{V}_2\}} + (1 - \alpha - \sum_{j=1}^{N} \beta_j) C_{\{\mathcal{V}^\perp\}} \end{aligned} \tag{19}$$

where $N = 2^{n-1} - 1$ indicates all possible partitions of n nodes into two subsets, weight coefficients α and β_j's can be regarded as the chances that the copula takes the respective forms of extremal distributions. All coefficients sum up to one. For instance, the case that $\alpha = 1$ and $\beta_j = 0$ $(\forall j = 1, \ldots, N)$ corresponds to the perfect positive correlation among all nodes, whereas $\alpha = \beta_j = 0$ $(\forall j = 1, \ldots, N)$ corresponds to the complete independence among all nodes. The copula $C(P(x_1), \ldots, P(x_n))$ thus is located in the convex set formed by the extremal joint probabilities, if all extremal probabilities can be calculated. This is illustrated in Fig. 2, where the copula of a three-node network is bounded by the convex hull formed by five possible partitions of nodes. The dynamics of the copula is estimated from the linear combinations of the dynamics of those five extremal probabilities. The challenge of estimating copulas however is to find out the weight coefficients α and β_j's.

The decomposition principle in Eq. (19) can be further generalized. If the nodes are self-organized into M different independent groups and nodes between groups are uncorrelated, then

$$C(P(x_1), \ldots, P(x_n)) = \prod_{m=1}^{M} C_m \tag{20}$$

where copulas C_m's are estimated according to Eq. (19) from the extremal distributions for partitions and independence within each group. Furthermore, if nodes are conditionally independent within each partition, the extremal probabilities

can also be estimated more accurately. In other words, additional dependency information among nodes helps identify the extremal probabilities more precisely. With these independent assumptions, copulas of large networks can be decomposed into smaller groups, and the complexity of information dynamics based on copulas can still be managed.

B. Copula Dynamics

The evolution of extremal probabilities can be generally modeled as

$$\frac{dC_{\{\mathcal{V}_1,\mathcal{V}_2\}}(t)}{dt} = f(C_{\{\mathcal{V}_1,\mathcal{V}_2\}}(t)) + \epsilon \tag{21}$$

where $\{\mathcal{V}_1, \mathcal{V}_2\}$ corresponds to any partition of nodes as in Eq. (16), nonlinear function f captures the dependencies in time evolution, and ϵ is the random noise term. Each copula of extremes in Eq. (19) has a respective dynamics model. A simple numerical approximation of Eq. (21) as time series autoregressive (AR) model is

$$C_{\{\mathcal{V}_1,\mathcal{V}_2\}}(k) = \gamma_0 + \sum_{l=1}^{L} \gamma_l C_{\{\mathcal{V}_1,\mathcal{V}_2\}}(k-l) + \epsilon \tag{22}$$

where the k-th time step value depends on the values of previous L steps. L is the order or lag of AR models. γ_0 is the intercept, γ_l's are the model coefficients, and $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$ follows a normal distribution. The AR model in Eq. (22) captures the time correlation of the expected values of copulas.

The data-driven approaches are necessary to calibrate the dynamics models. Based on the probabilistic graph model in Section II-B, Monte Carlo sampling can be used to simulate the evolutions of the prediction probabilities. The information dynamics model can be trained through regular data fitting or Bayesian approaches. For the copula dynamics modeling, two training procedures are needed. First, the weight coefficients α and β_j's in Eq. (19) need to be trained and calibrated so that the actual joint probabilities of state variables can be estimated from the extremal probabilities. The constraint that the weight coefficients sum up to one needs to be enforced during the training. Therefore, constrained optimization needs to be applied instead of regular least-square fitting. Second, the parameters γ_0 and γ_l's of the dynamics models of copulas in Eq. (22) also need to be calibrated. After parameter calibration, the models can be applied to predict the future values.

IV. Functional Interdependency Modeling

In the second approach, the marginal prediction probabilities are used in the dynamics model, where the interdependency and coupling between them are captured implicitly as functional relationships. The prediction probability of one node is a function of the probabilities from its neighbors.

The second approach to model the interdependency between predictions is to use analytical functions. The dynamics of prediction probability for node i can be modeled by

$$\frac{dP(x_i,t)}{dt} = g_i(P(x_1,t),\ldots,P(x_n,t)) + \epsilon_i \tag{23}$$

where $g_i(\cdot)$'s can be linear or nonlinear functions to capture the interdependency between the trajectories of prediction probabilities $P(x_k)$'s, and $n = |\mathcal{V}|$ is the total number of nodes in the network. If the prediction probabilities of all nodes are considered as a vector $\boldsymbol{P}(t) = [P(x_1,t),\ldots,P(x_n,t)]^T$, the model is written as

$$\frac{d\boldsymbol{P}(t)}{dt} = \boldsymbol{G}(\boldsymbol{P}(t)) + \boldsymbol{\epsilon} \tag{24}$$

In the proposed approach for time series analyses, the model is simplified to

$$\boldsymbol{P}(s) = \boldsymbol{F}(\boldsymbol{P}(s-1),\ldots,\boldsymbol{P}(s-L)) + \boldsymbol{\epsilon} \tag{25}$$

where $\boldsymbol{F}(\cdot)$ is a linear or nonlinear function, s is the discretized time step, and the prediction probabilities are the functions of previous L time steps. That is, the prediction probability of one node at a time step is a function of the prediction probabilities for all other nodes at the previous several time steps, given that nodes share information and have influence on each other. The memory of nodes is limited to L time steps. The function can be linear as the vector autoregression model proposed here, where variables are linearly dependent. More complex nonlinear functions can also be applied. Here, a Gaussian process regression model, which is a highly flexible nonlinear regression model, is proposed.

A. Topology-Constrained Vector Autoregression

A linearized vector autoregression (VAR) model is

$$\boldsymbol{P}(s) = \boldsymbol{A}_0 + \sum_{l=1}^{L} \mathbf{A}_l \boldsymbol{P}(s-l) + \boldsymbol{\epsilon} \tag{26}$$

where the vector value at the s-th time step is related to the values at the previous L steps, $\boldsymbol{A}_0$ is the vector of intercepts, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ is the multi-variant normal random variables, and the $n \times n$ coefficient matrices $\mathbf{A}_l$'s capture the interdependency between prediction probabilities. The VAR model in Eq. (26) captures the time and location dependencies of nodes simultaneously as the linear relationships.

The linear VAR model in Eq. (26) is a generic regression model that can be applied to different dynamics models. To model the network information dynamics, additional physical information of the networks can be applied as the additional constraints. Since the network topology is given, the paths of information flow along edges are known. That is, each node updates its prediction probabilities based on its current prediction and the ones from its source nodes. Thus, this dependency information can help reduce the number of parameters to train in Eq. (26).

Specifically, the topology-constrained VAR model is

$$\boldsymbol{P}(s) = \boldsymbol{A}_0 + \sum_{l=1}^{L} (\mathbf{D}^T \circ \mathbf{A}_l) \boldsymbol{P}(s-l) + \boldsymbol{\epsilon} \tag{27}$$

where $\mathbf{D}$ is the adjacency matrix of the graph, T is the matrix transpose operator, and $\circ$ is the element-wise Hadamard product between two matrices. In the adjacency matrix $\mathbf{D} = (d_{ij})_{n\times n}$, its element is defined as $d_{ij} = 1$ if there is a directed edge from node i to node j; otherwise, $d_{ij} = 0$. Particularly, $d_{ii} = 1$ for all i's. That is, only the coefficients in $\mathbf{A}_l$'s that correspond to direct connections between nodes contribute to

the model. For those without direct connections, there is no immediate information dependency and the coefficients in $\mathbf{A}_l$'s are set to be zeros. With the adjacency information included, the non-zero coefficients in $\mathbf{A}_l$'s now can be interpreted. They can be interpreted as the reliance probabilities. The coefficients in $\boldsymbol{A}_0$ are associated with the nodes' own prediction capability.

In original VAR model, the parameters to be calibrated are vector $\boldsymbol{A}_0$ and matrices $\mathbf{A}_l$'s. The total number of parameters is $n + n \times n \times L$. In the constrained VAR model, the number of parameters to be trained is reduced to $n + e \times L$ where $e = |\mathcal{E}|$ is the number of edges in the directed graph. The training of Eq.(27) can be generalized to minimizing the loss function subject to constraint

$$\sum_{l=1}^{L} ||(\mathbb{I} - \mathbf{D}^T) \circ \mathbf{A}_l|| = 0 \tag{28}$$

where $\mathbb{I}$ is a matrix with the same size of $\mathbf{D}$ and all elements as 1's, and $||\cdot||$ indicates the matrix norm. In addition, to ensure the interpretability of coefficients, all non-zero coefficients in $\mathbf{A}_0$ and $\mathbf{A}_l$'s should be between 0 and 1 so that they can be interpreted as probabilities.

B. Spatial-Temporal Gaussian Process Regression

Gaussian process regression (GPR) is a widely used non-linear model. The unique advantage of GPR is its support of sequential sampling. A GPR model is specified by a mean function $\mu(\boldsymbol{x}) = \mathbb{E}[Y(\boldsymbol{x})]$ and a covariance function $Cov(Y(\boldsymbol{x}), Y(\boldsymbol{x}')) = k(\boldsymbol{x}, \boldsymbol{x}')$. Based on a finite set of basis functions $\{\phi_j(\boldsymbol{x}), j = 1, \ldots, m\}$, the GPR model is

$$Y(\boldsymbol{x}) = \sum_{j=1}^{m} w_j \phi_j(\boldsymbol{x}) + \epsilon = \boldsymbol{\phi}(\boldsymbol{x})^T \boldsymbol{w} + \epsilon \tag{29}$$

where $\boldsymbol{w} = [w_1, \ldots, w_m]^T$ is the weight vector, $\boldsymbol{\phi}(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \ldots, \phi_m(\boldsymbol{x})]^T$ is the vector of m basis functions, and Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma_0^2)$ is assumed to be associated with observations. Let the prior distribution of the weight vector $\boldsymbol{w}$ be Gaussian $N(\mathbf{0}, \boldsymbol{\Sigma}_w)$. The posterior mean of weights $\hat{\boldsymbol{w}}$ is estimated by minimizing the negative logarithmic of the posterior probability, as

$$\hat{\boldsymbol{w}} = \sigma_0^{-2} \mathbf{A}^{-1} \boldsymbol{\Phi}^T \boldsymbol{y} \tag{30}$$

where $\boldsymbol{y} = [y_1, \ldots, y_n]^T$ is the vector of n observations, $\mathbf{A} = \boldsymbol{\Sigma}_w^{-1} + \sigma_0^{-2} \boldsymbol{\Phi}^T \boldsymbol{\Phi}$, and $\boldsymbol{\Phi}$ is the $n \times m$ design matrix with elements $\phi_j(\boldsymbol{x}_i)$'s.

For a new input $\boldsymbol{x}^*$, the predicted mean value is

$$\mathbb{E}[Y(\boldsymbol{x}^*)] = \boldsymbol{\phi}(\boldsymbol{x}^*)^T \hat{\boldsymbol{w}} = \sigma_0^{-2} \boldsymbol{\phi}(\boldsymbol{x}^*)^T \mathbf{A}^{-1} \boldsymbol{\Phi}^T \boldsymbol{y} \tag{31}$$

and the predicted variance is

$$\mathbb{V}[Y(\boldsymbol{x}^*)] = \sigma_0^2 + \boldsymbol{\phi}(\boldsymbol{x}^*)^T \mathbf{A}^{-1} \boldsymbol{\phi}(\boldsymbol{x}^*) \tag{32}$$

Here, the GPR model is used as a time series model to predict the future prediction probabilities of CPSS nodes. To simplify, we choose $m = 1$ in this paper. Different from traditional time-series problems, the prediction probabilities as the responses are strongly correlated because of information sharing between nodes. Therefore, the GPR model proposed here has two-dimensional inputs and multi-dimensional output predictions. The inputs are time and node label. Time is continuous, whereas node label has discrete values. The two inputs allow us to capture both spatial and temporal correlations that exist in the variables. The number of outputs is the same as the number of nodes.

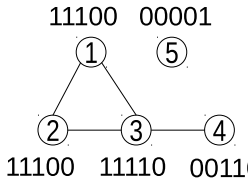


Fig. 3: An example of binary node labels for a 5-node network

To construct the spatial-temporal GPR model, choosing the kernel function is critical. The proposed kernel function

$$k((s, b), (s', b')) = k_1(s, s') k_2(b, b') \tag{33}$$

is a product of temporal kernel

$$k_1(s, s') = \exp(-2 \sin^2(\pi |s - s'|/p)/l^2) \tag{34}$$

and spatial kernel

$$k_2(b, b') = \exp(-0.5 d_H(b, b')/z^2) \tag{35}$$

where s and s' are two different time as continuous variables, whereas b and b' are the labels of two different nodes. The hyperparameters of temporal kernel includes period p and length scale l. The hyperparameter of spatial kernel is scale z. Hamming distance $d_H(b, b')$ between node labels is applied to quantify the differences between nodes. The sinusoidal kernel for temporal correlation is selected because of the fluctuation nature of prediction probabilities ranging between 0 and 1.

The proposed node labels are based on the adjacency of nodes. When two nodes are directly connected, they have stronger interdependency because of information sharing. For a network with n nodes, an n-bit binary string is used to label each node, each bit corresponding to a node. When a node is directly connected to another node, in the label of the first node, the bit corresponding to the second node is "1". Otherwise, the bit is "0". The Hamming distance between two labels indicates how strong the correlation between two nodes. The rationale is that strong correlation exists between two nodes when they have similar information sources, as a result of similar neighborhood connections. When the neighborhoods are similar, the Hamming distance between the two nodes is small. Fig. 3 shows an example of node labels for a network, where a binary string that indicates the pairwise connectivity is assigned to each node. Node 1 has the smallest Hamming distance to Node 2 and they share the same information source. In contrast, Node 1 has the largest distance to Node 5 and they are independent from each other.

V. DEMONSTRATIONS

In this section, several examples are used to demonstrate the proposed information dynamics models. The copula dynamics model will be demonstrated with a simple three-node network.

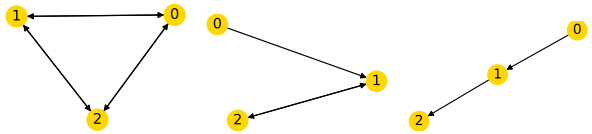


Fig. 4: Three examples of 3-node-6-edge, 3-node-3-edge, and 3-node-2-edge networks to demonstrate copula dynamics

The VAR and GPR models will be demonstrated with larger networks. The implementation can be found at github [48].

A. Demonstrations of The Copula Dynamics Model

For demonstrations, three three-node random networks are created where the nodes are connected at different probabilities. Three example networks in Fig. 4 are tested, which contain six, three, and two edges respectively. The values of the initial prediction probabilities as well as the P- and Q-reliance probabilities are randomly generated. Monte Carlo sampling is applied to simulate the process of prediction probability updates and generate the training data.

1) Data generation: The simulation algorithm is listed in Algorithm 1. In each time step, random samples of observations are generated for each node based on its current prediction probability. Then the observations are shared to the neighboring nodes, and the shared information is sampled based on the reliance probabilities. When a node receives the information from its source nodes, a fusion rule (e.g. worst-case, best-case, Bayesian) is applied to update its prediction. The predictions are compared with the randomly generated ground truth state value and the correct instances are recorded. The above sampling procedure repeats many times, and the probability of correct prediction for each node is obtained and updated for this time step. The joint probabilities for all nodes for all possible combinations of correct and incorrect predictions are also obtained. The simulation clock advances and the next iteration of update is done in the same way.

2) Training: After the simulation data are obtained, the extremal probabilities also need to be calculated based on Eqs. (16)-(18). To calibrate the weight coefficients α and β_j's in Eq. (19), these copulas of extremal probabilities will be used as the inputs for model training, whereas the outputs will be the joint probabilities. For a network of three nodes, the number of extremal probabilities according to the number of node partitions is 5. The number of joint probabilities as the number of binary-valued combinations is $2^3 = 8$. Therefore, for the three-node network, a total of $5 \times 8 = 40$ different extremal probabilities are used as the inputs during the training of the copula model in Eq. (16). The outputs are the 8 joint probabilities. The training can be done by solving the least-squared error optimization problem under the constraints that the weight coefficients are nonnegative and sum up to one. After the training, the relation between the extremal probabilities and joint probabilities is obtained to predict future joint probabilities.

The AR model in Eq. (22) is also built for each of the 40 extremal probabilities. The purpose is to predict the future copulas with extremal probabilities from the existing data. The calibration of coefficients γ_0 and γ_l's can be similarly done with regressions.

Algorithm 1 The Monte Carlo sampling algorithm to generate sequences of prediction and joint probabilities for nodes along time

```
while ( maximum time step is not reached ) do ▷ main iterations for time
    while ( number of samples is not enough ) do
        Randomly generate a ground truth;
        for each node i in the graph do
            Randomly generate a sample based on P(x_i);
            for each destination node k w.r.t. node i do
                if node i predicts correctly then
                    Generate a sample based on P(x_k|x_i);
                else
                    Obtain a sample based on P(x_k|x_i^C);
                end if
            end for
            Update prediction based on fusion rule;
            Record in the tally;
        end for
    end while
    Update P(x_k) from the tally;
end while
```

After the two training procedures, the 40 AR models are used to predict the future values of the 40 respective extremal probabilities from existing simulation data. From the forecast of extremal probabilities as the inputs of Eq. (19), the 8 joint probabilities for a future time step can be estimated. From the 8 joint probabilities, 3 marginal prediction probabilities can be obtained.

3) Results: For the first example, all three nodes are fully connected with 6 directional edges. The probability update is simulated for 50 time steps. The worst-case fusion rule is applied. The simulated data are then used to train the copula model and the AR models. After training, the calibrated weight coefficients in Eq. (19) are 0.04989131, 0.04896505, 0.05062734, 0.050125, and 0.80039129. Some examples of coefficients for the 40 AR models as well as the estimated variances after training are shown in Table I. Here the lag order is $L = 2$.

After the training, the 40 AR models are applied to predict the probability update for additional 30 time steps. In Fig. 5a, the original prediction probabilities and the forecasts are compared, where the solid lines are the original probabilities whereas the dashed lines are the mean values of forecasts. The shaded band indicates the bounds of one standard deviation ($\pm\sigma$) in the forecasts. The standard deviations of the marginal prediction probabilities are estimated from the standard deviations of the extremal probabilities such as the ones listed in Table I. It is assumed that the variances associated with the extremal probability values are the same for the joint probabilities. Thus the variances for the marginal probabilities are the sums of those ones for the joint probability values.

It is seen in Fig. 5a that the forecasts of the copula dynamics

TABLE I: Some examples of coefficients and variances out of the 40 AR models in the 3-node-6-edge network

	γ_0	γ_1	γ_2	σ_ϵ^2
example 1	0.43470733	0.15745345	-0.08249448	0.01222724
example 2	0.11957611	0.06856919	-0.18018992	0.01517523
example 3	0.11618891	0.06384009	-0.20648693	0.01410113
example 4	0.46474062	-0.19265025	-0.0755481	0.00337921
example 5	0.00392942	-0.0383012	-0.0383012	0.00011325
example 6	0.12958748	-0.02501151	-0.07006834	0.00051422

model match the general trend. When the lag order increases, the fluctuation in the forecasts will also increase. The forecasts of a higher-order AR model with $L = 12$ are shown in Fig. 5b, where local fluctuations appear in the forecasts.

The 3-node-3-edge network and 3-node-2-edge network in Fig. 4 are used for further tests. Simulation and model training are similarly done. The results of simulation and model forecast are shown in Fig. 6. Compared to the previous 3-node-6-edge case in Fig. 5, the variabilities of predictions by some nodes increase. The general trend is that when a node receives more information, its prediction capability increases with smaller fluctuation and variability. For instance, in the 3-node-3-edge and 3-node-2-edge networks, the probability of Node 0 fluctuates more than those of Nodes 1 and 2 since Node 0 does not receive information from others. Nevertheless, extensive information sharing such as in the fully connected 3-node-6-edge network also causes the synchronization of fluctuations for all nodes.

The copula dynamics model provides good estimations of the general trends of the prediction capabilities with the underlying linear AR models. Linear AR models can predict stable systems well. Here the variances are estimated as combinations of the ones from extremal probabilities. As a result, they are likely to be overestimated. Although the copula dynamics model explicitly captures the correlations between nodes, the major disadvantage is its computational complexity and lack of scalability for larger systems. As the number of nodes increases, the number of joint probabilities and copulas of extremal distributions will increase exponentially.

B. Demonstrations of The Functional Interdependency Models

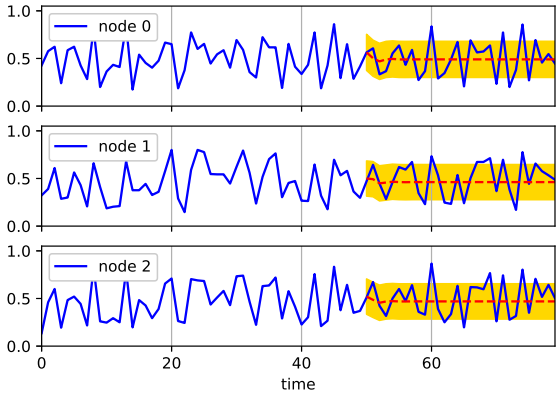
The functional interdependency model captures the correlations of prediction capabilities between nodes by functional relationships. This approach has the lower computational complexity than the copula dynamics model, since the marginal prediction probabilities are directly modeled. The 15-node-23-edge and 15-node-66-edge networks shown in Fig. 7 are used in the demonstration in addition to the previous ones in Fig. 4.

1) VAR models: The VAR model in Eq. (26) and the topology constrained VAR model in Eq. (27) are first demonstrated and compared with the 3-node-2-edge example. The simulation data are collected to train the VAR models with lag order $L = 2$. The training data and forecast results are shown in Fig. 8, where three VAR models are compared. For the first test, the worst-case fusion rule is applied. Fig. 8a shows the result of the original VAR model. Fig. 8b shows the value-constrained VAR model where the values of all coefficients are restricted between 0 and 1 and interpreted as probabilities. Fig. 8c shows the result of the topology-constrained VAR model where only the coefficients corresponding to the adjacent nodes are non-zero. For comparison, the same dataset is applied for all three VAR models. The calibrated model parameters or coefficients of the three VAR models are listed in Table II. The error bounds are defined as one standard deviation, which are directly obtained from the covariance matrix after the training procedure. For the second test, the best-case fusion rule is applied. The results of the VAR and constrained VAR models are shown in Figs. 8d-8f respectively. It is seen that all three models predict the trend well. When the value constraint for interpretability is imposed to the VAR model, the coefficients become physically meaningful as probabilities. When the topology constraint is further imposed, the number of functioning coefficients is significantly reduced. This sparse representation of VAR models makes training process easier. Less training data is required because of the reduced number of parameters to calibrate. The scalability of the VAR models is better than the copula dynamics model.

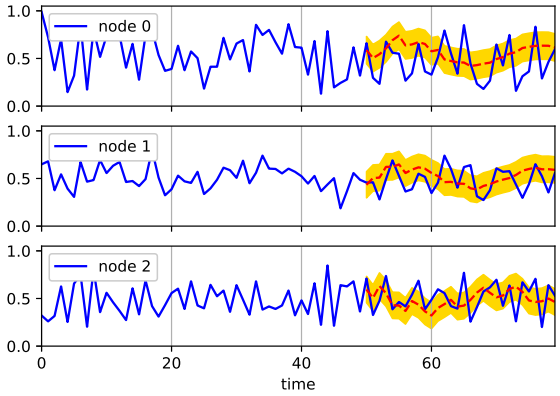
TABLE II: The coefficients of the VAR models in Fig. 8

VAR model			
$\boldsymbol{A}_0 =$	-1.32985007	0.5634467	0.39025738
$\mathbf{A}_1 =$	0.16732259	0.54656891	-0.05017579
	0.94741755	-0.05021139	0.14863032
	3.96408185	-0.88023708	0.12792812
$\mathbf{A}_2 =$	-0.78595986	0.04871532	0.12865764
	-1.36827169	0.28772943	-0.09098509
	0.76135351	-0.0757617	-0.04557109
Value-constrained VAR model			
$\boldsymbol{A}_0 =$	0.0	0.202493134	0.303131421
$\mathbf{A}_1 =$	0.0	0.559047466	0.0
	0.0	0.0	0.388614469
	0.425431619	0.0	0.0
$\mathbf{A}_2 =$	0.0	0.0392161135	0.0
	0.0	0.0	0.0
	0.612816961	0.0	0.0
Topology-constrained VAR model			
$\boldsymbol{A}_0 =$	0.429084615	0.319280699	0.498936303
$\mathbf{A}_1 =$	0.0325279063	0.0	0.0
	0.0	0.0	0.0
	0.0	0.0	0.0
$\mathbf{A}_2 =$	0.0	0.0	0.0
	0.0140168669	0.0	0.0
	0.0	0.391937688	0.0

Additional examples of constrained VAR models are shown in Fig. 9 with 15 nodes and 23 directed edges. It is seen that the forecast accuracy of the topology-constrained VAR model is similar to that of the traditional VAR model when $L = 2$. When the lag order is increased to $L = 4$, the traditional VAR model cannot provide meaningful predictions any more with only 50 steps of training data. This is because the increased number of coefficients with a larger L requires more training data for the fitting process. In contrast, the topology-constrained VAR model still provides reasonable forecasts for the larger L, since it has a much smaller set of non-

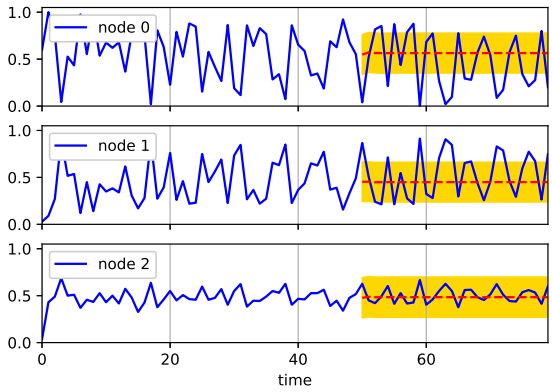


(a) prediction probabilities and AR model forecasts ($L = 2$)

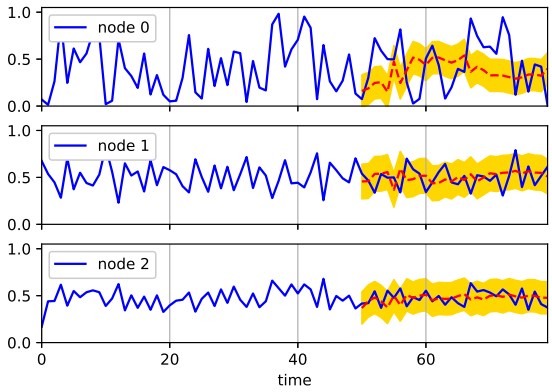


(b) prediction probabilities and AR model forecasts ($L = 12$)

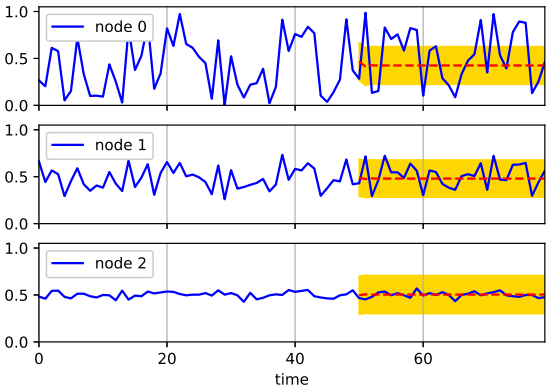
Fig. 5: Evolution of prediction probabilities in the 3-node-6-edge network and forecasts from the copula dynamics model



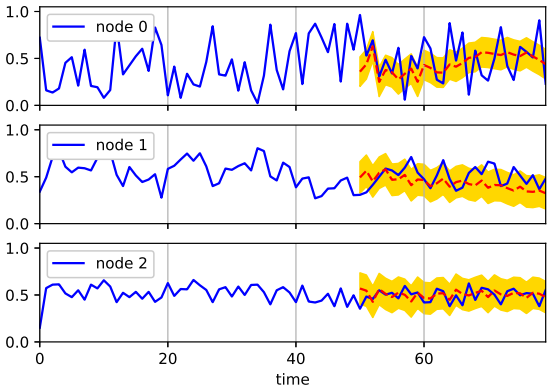
(a) AR model predictions ($L = 2$) for the 3-node-3-edge network



(b) AR model predictions ($L = 12$) for the 3-node-3-edge network



(c) AR model predictions ($L = 2$) for the 3-node-2-edge network



(d) AR model predictions ($L = 12$) for the 3-node-2-edge network

Fig. 6: Evolution of prediction probabilities and copula dynamics models

zero coefficients to train and therefore more efficient than the traditional VAR model.

2) GPR model: The proposed spatial-temporal GPR model is further demonstrated, where the correlation between nodes as well as temporal correlation in time series are simultaneously captured by the hybrid kernel function.

Some example results are shown in Figure 10, where the dynamics of prediction probabilities in the 3-node-6-edge, 3-node-3-edge, and 15-node-66-edge networks are modeled and predicted. Similar to the previous examples, 50 time steps of

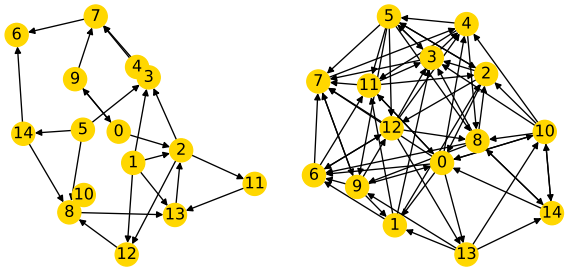


Fig. 7: Two additional examples of 15-node-23-edge and 15-node-66-edge networks to demonstrate functional interdependency models

simulation data are used to train the GPR model. Then the model is used to predict the next 30 time steps. Different from the linear VAR models, the local fluctuations of both means and variances are predicted by the GPR model. This is because predictions in GPR modeling are based on the local differences between samples, which is characterized by the distance functions in the kernels. In addition, the sinusoidal kernel is designed to capture the fluctuations.

One disadvantage of the GPR model however is the computational cost associated with the training and Bayesian update procedures, where the inverse of sample covariance matrix needs to be computed. This becomes very expensive as the number of training samples increases. The computational complexity prohibits GPR models from being applied if there are thousands of samples, as opposed to 80 samples in this example here.

VI. Concluding Remarks

In this paper, a statistical modeling approach is taken to model the information dynamics in CPSS networks, where the effect of information diffusion in a probabilistic graph model of network is captured. The effect is reflected as the changes of individual nodes' capabilities of prediction and decision making. The dynamics of capabilities for individual nodes becomes the unique characteristic in CPSS networks, which is different from traditional computer networks or social networks. Therefore, the fluctuations of these capabilities as a result of information diffusion in the network are modeled here, instead of tracking specific information elements as in other information dynamics models.

Three new statistical models are proposed to capture the strong dependencies or correlations between nodes' capabilities. Given the prior knowledge about the correlations, a gray-box approach is taken by incorporating them in model construction. The topology of the network can provide the insight into the correlations and simplify the models. The interpretability of the model parameters as probability values also provides constraints during the training, which results in better statistical models.

The proposed information dynamics models can be applied as the tool to design adaptable and scalable CPSS networks. For instance, it allows us to assess the impact of network topology changes upon disruption more efficiently than Monte Carlo simulations. The model predictions can help identify the ways to assign or allocate the most influential nodes to ensure resilience. The tool also allows us to perform sensitivity analyses to identify unreliable information sources, especially when the size of network grows, which helps design strategies for improving system reliability with necessary redundancy or control the extent of negative outcomes.

The limitations of the proposed approach are mainly from computational aspects. Explicit modeling of correlations with copulas makes the calculation cumbersome for large networks. With joint probabilities directly modeled, copula dynamics is conceptually appealing but practically difficult to apply. Linear vector autoregression has good track records in modeling time series. The predictions show the general trend whereas local information is missing. When significant fluctuations are present as in this information dynamics modeling problem, Gaussian process regression shows better performance. However, computational cost of constructing Gaussian process regression models with large sample sizes is a well-known issue.

Future studies of the data-driven approach for information dynamics modeling will need to tackle the computational cost issue. For instance, sparse Gaussian process methods can be applied to reduce the number of samples that are used in model construction and update. Some dimensionality reduction approaches such as latent variables can also be applied to improve the efficiency of modeling. In addition, the probabilistic graph model only captures the functions of sensing, computing, and communication for CPSS. The control capability and its effects on information propagation require further investigation.

References

[1] I. Horváth and B. H. Gerritsen, "Cyber-physical systems: Concepts, technologies and implementation principles," in *Proceedings of The 9th International Symposium on Tools and Methods of Competitive Engineering (TMCE2012)*, pp. 19–36, 2012.

[2] P. Maló, B. Almeida, R. Melo, K. Kalaboukas, and P. Cousin, "Self-organised middleware architecture for the internet-of-things," in *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, pp. 445–451, IEEE, 2013.

[3] M. Bernardo, R. De Nicola, and J. Hillston, "Formal methods for the quantitative evaluation of collective adaptive systems," *Lecture Notes in Computer Science*, vol. 9700, 2016.

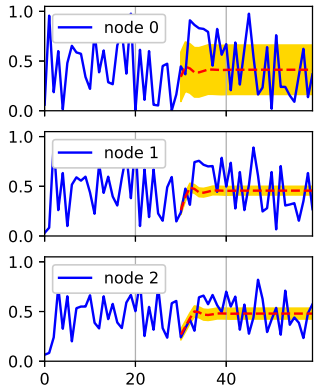
[4] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "Iot: Internet of threats? a survey of practical security vulnerabilities in real iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8182–8201, 2019.

[5] Y. Wang, "System resilience quantification for probabilistic design of internet-of-things architecture," in *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers Digital Collection, 2016.

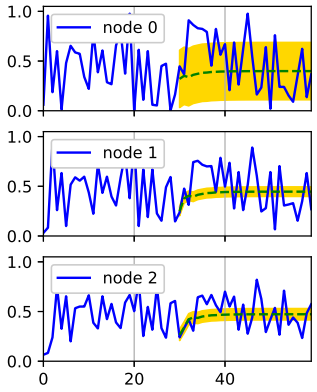
[6] Y. Wang, "Resilience quantification for probabilistic design of cyber-physical system networks," *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B*, vol. 4, no. 3, p. 031006, 2018.

[7] M. Viroli, J. Beal, F. Damiani, G. Audrito, R. Casadei, and D. Pianini, "From distributed coordination to field calculus and aggregate computing," *Journal of Logical and Algebraic Methods in Programming*, vol. 109, p. 100486, 2019.

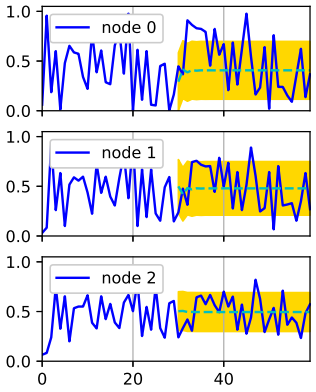
[8] I. Horváth and J. Wang, "Towards a comprehensive theory of multi-aspect interaction with cyber physical systems," in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers Digital Collection, 2015.



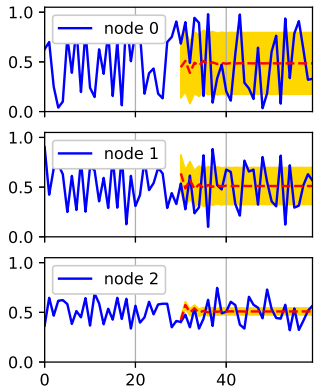
(a) VAR model (worst-case rule)



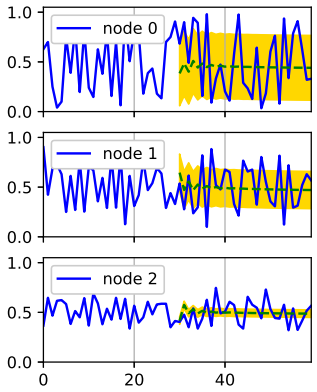
(b) Value-constrained VAR model (worst-case rule)



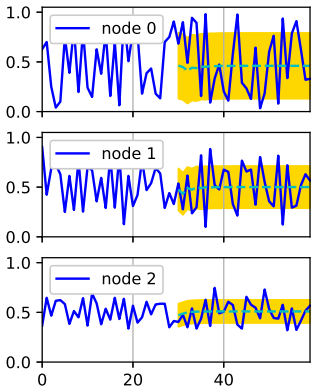
(c) Topology-constrained VAR model (worst-case rule)



(d) VAR model (best-case rule)



(e) Value-constrained VAR model (best-case rule)



(f) Topology-constrained VAR model (best-case rule)

Fig. 8: Evolution of prediction probabilities in the 3-node-2-edge network and forecasts from different VAR models, where the worst-case and best-case fusion rules are applied respectively

[9] Y. Li, I. Horváth, and Z. Rusák, "Constructing personalized messages for informing cyber-physical systems based on dynamic context information processing," in *Proceedings of The 12th International Symposium on Tools and Methods of Competitive Engineering (TMCE2018)*, pp. 105—-120, 2018.

[10] Z. Su, Q. Qi, Q. Xu, S. Guo, and X. Wang, "Incentive scheme for cyber physical social systems based on user behaviors," *IEEE Transactions on Emerging Topics in Computing*, 2017.

[11] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: Privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1868–1878, 2017.

[12] Y. Wang, "Trustworthiness in designing cyber-physical systems," in *Proceedings of The 12th International Symposium on Tools and Methods of Competitive Engineering (TMCE2018)*, pp. 27–40, 2018.

[13] Y. Wang, "Trust based cyber-physical systems network design," in *ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. V01AT02A037–V01AT02A037, American Society of Mechanical Engineers, 2018.

[14] F. Wu, B. A. Huberman, L. A. Adamic, and J. R. Tyler, "Information flow in social groups," *Physica A: Statistical Mechanics and its Applications*, vol. 337, no. 1-2, pp. 327–335, 2004.

[15] J. Yang and J. Leskovec, "Modeling information diffusion in implicit networks," in *2010 IEEE International Conference on Data Mining*, pp. 599–608, IEEE, 2010.

[16] A. Simma and M. I. Jordan, "Modeling events with cascades of poisson processes," in *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, pp. 546–555, 2010.

[17] H.-H. Jo, J. I. Perotti, K. Kaski, and J. Kertész, "Analytically solvable model of spreading dynamics with non-poissonian processes," *Physical Review X*, vol. 4, no. 1, p. 011041, 2014.

[18] Y. Wang, "Trust quantification for networked cyber-physical systems," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2055–2070, 2018.

[19] M. Kubo, K. Naruse, H. Sato, and T. Matubara, "The possibility of an epidemic meme analogy for web community population analysis," in *International Conference on Intelligent Data Engineering and Automated Learning*, pp. 1073–1080, Springer, 2007.

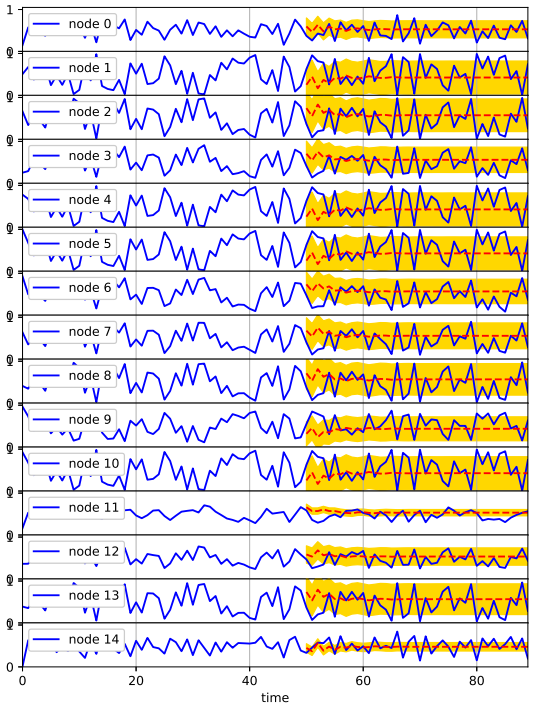
[20] J. Woo and H. Chen, "An event-driven sir model for topic diffusion in web forums," in *2012 IEEE International Conference on Intelligence and Security Informatics*, pp. 108–113, IEEE, 2012.

[21] F. Xiong, Y. Liu, Z.-j. Zhang, J. Zhu, and Y. Zhang, "An information diffusion model based on retweeting mechanism for online social media," *Physics Letters A*, vol. 376, no. 30-31, pp. 2103–2108, 2012.

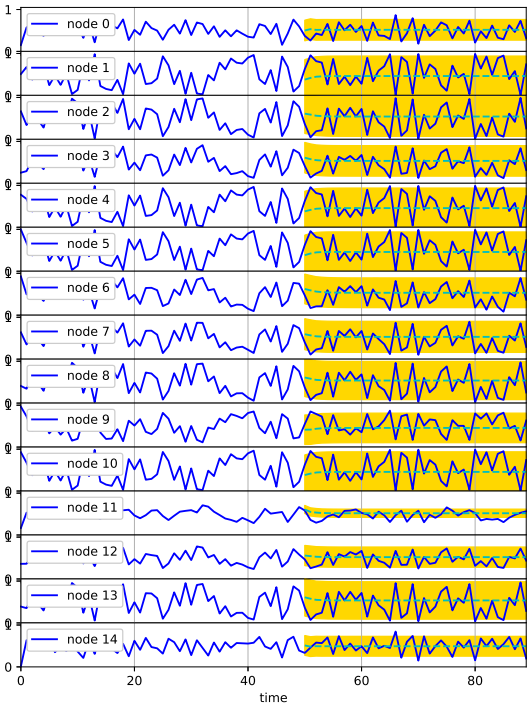
[22] Q. Liu, T. Li, and M. Sun, "The analysis of an seir rumor propagation model on heterogeneous network," *Physica A: Statistical Mechanics and its Applications*, vol. 469, pp. 372–380, 2017.

[23] J. Cannarella and J. A. Spechler, "Epidemiological modeling of online social network dynamics," *arXiv preprint arXiv:1401.4208*, 2014.

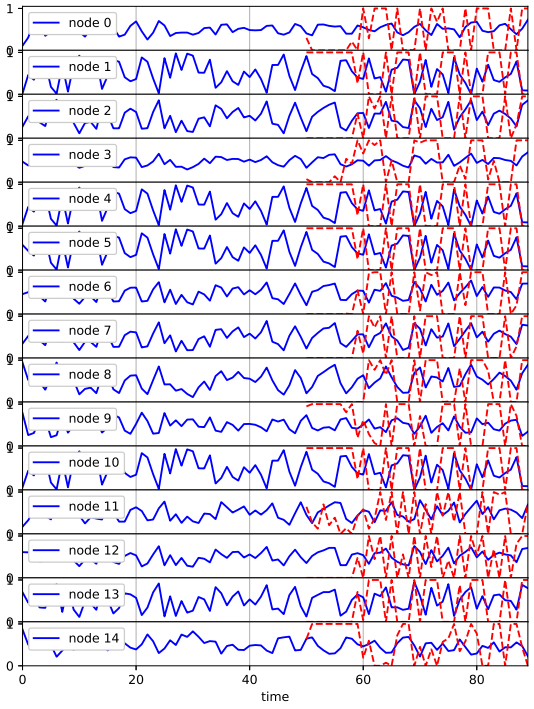
[24] Y. Yi, Z. Zhang, L. T. Yang, C. Gan, X. Deng, and L. Yi, "Reemergence modeling of intelligent information diffusion in heterogeneous social



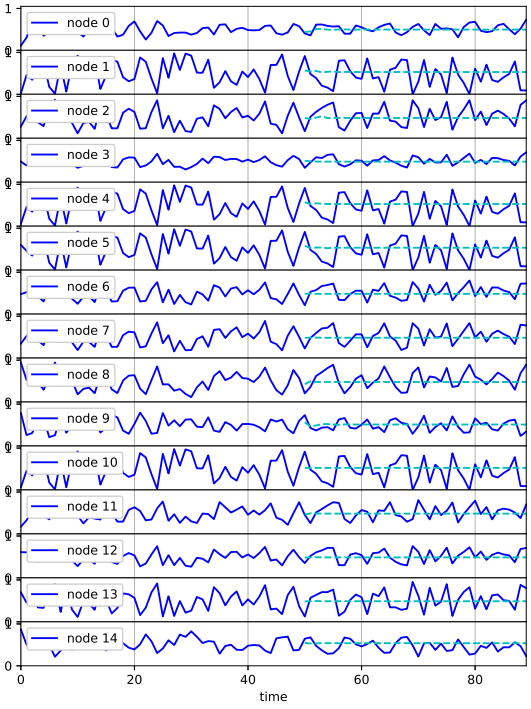
(a) VAR model forecasts ($L = 2$)



(b) Topology-constrained VAR model forecasts ($L = 2$)



(c) VAR model forecasts ($L = 4$)



(d) Topology-constrained VAR model forecasts ($L = 4$)

Fig. 9: Evolution of prediction probabilities in a 15-node-23-edge network and forecasts by VAR models

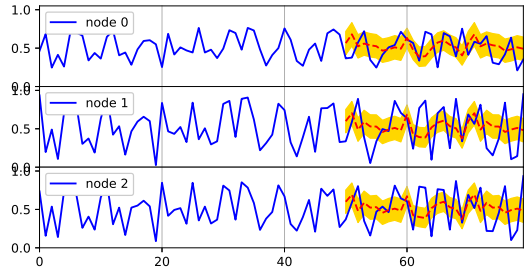
networks: the dynamics perspective," *IEEE Transactions on Network Science and Engineering*, 2020.

[25] T. Iwata, A. Shah, and Z. Ghahramani, "Discovering latent influence in online social activities via shared cascade poisson processes," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 266–274, 2013.

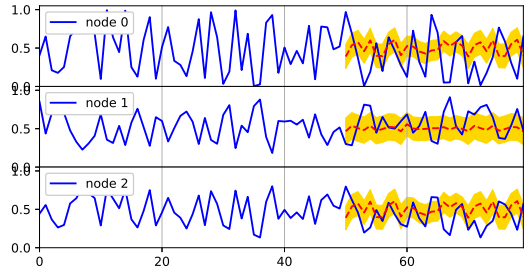
[26] R. Kobayashi and R. Lambiotte, "Tideh: Time-dependent hawkes process for predicting retweet dynamics," in *Tenth International AAAI Conference on Web and Social Media*, 2016.

[27] J. L. Iribarren and E. Moro, "Impact of human activity patterns on the dynamics of information diffusion," *Physical review letters*, vol. 103, no. 3, p. 038702, 2009.

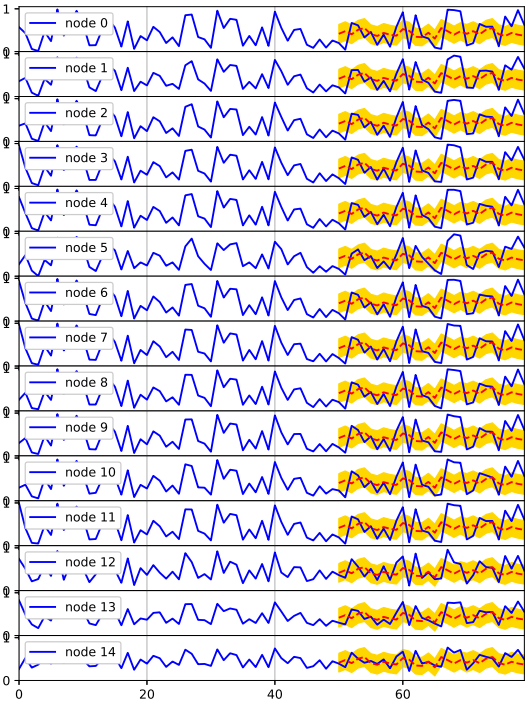
[28] N. Du, L. Song, H. Woo, and H. Zha, "Uncover topic-sensitive information diffusion networks," in *Artificial Intelligence and Statistics*,



(a) GPR model forecast of the 3-node-6-edge example



(b) GPR model forecast of the 3-node-3-edge example



(c) GPR model forecast of the 15-node-66-edge example

Fig. 10: The evolution of prediction probabilities in the 3-node-6-edge, 3-node-3-edge, and 15-node-66-edge network examples with GPR forecasts

pp. 229–237, 2013.

[29] Q. Yao, X. Wu, and X. Zhang, "Diffusion of information in mobile social networks: A brief survey," in *2015 IEEE International Conference on Mobile Services*, pp. 254–260, IEEE, 2015.

[30] M. Li, X. Wang, K. Gao, and S. Zhang, "A survey on information diffusion in online social networks: Models and methods," *Information*, vol. 8, no. 4, p. 118, 2017.

[31] F. Wang, H. Wang, and K. Xu, "Diffusive logistic model towards predicting information diffusion in online social networks," in *2012 32nd International Conference on Distributed Computing Systems Workshops*, pp. 133–139, IEEE, 2012.

[32] Y. Hu, R. J. Song, and M. Chen, "Modeling for information diffusion in online social networks via hydrodynamics," *IEEE Access*, vol. 5, pp. 128–135, 2017.

[33] H. Shen, D. Wang, C. Song, and A.-L. Barabási, "Modeling and predicting popularity dynamics via reinforced poisson processes," in *Twenty-eighth AAAI conference on artificial intelligence*, 2014.

[34] N. Alon, M. Feldman, A. D. Procaccia, and M. Tennenholtz, "A note on competitive diffusion through social networks," *Information Processing Letters*, vol. 110, no. 6, pp. 221–225, 2010.

[35] C. Jiang, Y. Chen, and K. R. Liu, "Evolutionary dynamics of information diffusion over social networks," *IEEE Transactions on Signal Processing*, vol. 62, no. 17, pp. 4573–4586, 2014.

[36] F. Chang, G. Yang, J. Qi, Y. Wang, S. Xu, J. Wang, and R. Wang, "Ipim: An effective contribution-driven information propagation incentive mechanism," *IEEE Access*, vol. 7, pp. 77362–77373, 2019.

[37] C. Li and Z. Ma, "Dynamic analysis of a spatial diffusion rumor propagation model with delay," *Advances in Difference Equations*, vol. 2015, no. 1, pp. 1–17, 2015.

[38] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information diffusion through blogspace," in *Proceedings of the 13th international conference on World Wide Web*, pp. 491–501, 2004.

[39] K.-K. Kleineberg and M. Boguná, "Evolution of the digital society reveals balance between viral and mass media influence," *Physical Review X*, vol. 4, no. 3, p. 031046, 2014.

[40] M. G. Rodriguez, J. Leskovec, D. Balduzzi, and B. Schölkopf, "Uncovering the structure and temporal dynamics of information propagation," *Network Science*, vol. 2, no. 1, pp. 26–65, 2014.

[41] B. Zong, Y. Wu, A. K. Singh, and X. Yan, "Inferring the underlying structure of information cascades," in *2012 IEEE 12th International Conference on Data Mining*, pp. 1218–1223, IEEE, 2012.

[42] Z. Lu, Y. Wen, W. Zhang, Q. Zheng, and G. Cao, "Towards information diffusion in mobile social networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1292–1304, 2015.

[43] S. Pei, L. Muchnik, J. S. Andrade Jr, Z. Zheng, and H. A. Makse, "Searching for superspreaders of information in real-world social media," *Scientific reports*, vol. 4, no. 1, pp. 1–12, 2014.

[44] O. Yagan, D. Qian, J. Zhang, and D. Cochran, "Conjoining speeds up information diffusion in overlaying social-physical networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 6, pp. 1038–1048, 2013.

[45] H. Lu, S. Lv, X. Jiao, X. Wang, and J. Liu, "Maximizing information diffusion in the cyber-physical integrated network," *Sensors*, vol. 15, no. 11, pp. 28513–28530, 2015.

[46] J. Wang, C. Jiang, Z. Han, T. Q. Quek, and Y. Ren, "Private information diffusion control in cyber physical systems: A game theory perspective," in *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, pp. 1–10, IEEE, 2017.

[47] Y. Yi, Z. Zhang, and C. Gan, "The outbreak threshold of information diffusion over social–physical networks," *Physica A: Statistical Mechanics and its Applications*, vol. 526, p. 121128, 2019.

[48] Y. Wang, "ProbNet: A cyber-physical-social systems modeling toolkit." https://github.com/GeorgiaTechMSSE/ProbNet/.

PLACE PHOTO HERE

Yan Wang is a Professor of Mechanical Engineering at Georgia Institute of Technology, Atlanta. He received his B.S. degree in electrical engineering from Tsinghua University, M.S. in electrical engineering from Chinese Academy of Sciences, Beijing, and Ph.D. in industrial engineering from the University of Pittsburgh. He is interested in modeling and simulation, cyber-physical systems, and uncertainty quantification. His research work was recognized with multiple best paper awards from conferences at American Society of Mechanical Engineers (ASME), Institute of Industrial & Systems Engineers (IISE), and The Minerals, Metals & Materials Society (TMS), as well as a National Science Foundation Early Career Award. He served as the Chair of the Executive Committee for ASME Computers & Information in Engineering Division and the Chair of ASME Advanced Modeling & Simulation Technical Committee.