

INFORMATION DYNAMICS IN THE NETWORK OF CYBER-PHYSICAL SYSTEMS

Yan Wang

Woodruff School of Mechanical Engineering
Georgia Institute of Technology
yan.wang@me.gatech.edu

ABSTRACT

Cyber-physical systems (CPS) are physical devices with highly integrated functionalities of sensing, computing, communication, and control. The levels of intelligence and functions that CPS can perform heavily rely on their intense collaboration and information sharing through networks. In this paper, the information propagation within CPS networks is studied. Information dynamics models are proposed to characterize the evolution of information processing capabilities of CPS nodes in networks. The models are based on a mesoscale probabilistic graph model, where the sensing and computing functions of CPS nodes are captured as the probabilities of correct predictions, whereas the communication functions are represented as the probabilities of mutual influences between nodes. In the proposed copula dynamics model, the information dependency among individuals is represented with joint prediction probabilities and estimated from copulas of extremal probabilities. In the proposed functional interdependency model, the correlations between prediction capabilities are captured with their functional relationships. A data-driven approach is taken to train the parameters of the information dynamics models with data from simulations. The information dynamics models are demonstrated with a simulator of CPS networks.

KEYWORDS

Cyber-Physical Systems, Probabilistic Graph Model, Copula, Time Series, Information Diffusion

1. INTRODUCTION

Cyber-physical systems (CPS) are physical devices with highly integrated functionalities 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 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 term cyber-physical-social systems (CPSS) is also used by researchers to describe the highly integrated systems with the social dimension.

In CPS, information collection, processing, and decision making are done in a decentralized fashion. The intelligence level of CPS is enhanced by intensive information sharing, as more high-quality information leads to better decisions. The level of dependencies among CPS devices for their computation and decision making is unprecedented. There is a strong need to understand the deep information dependency between devices in a CPS network. The knowledge about the behaviors of the complex system can help us to design more reliable and dependable systems.

Currently there is a lack of study of information propagation in CPS networks. Existing research on information diffusion focused on computer networks and social networks. The uniqueness of CPS networks is the sensing and control capabilities which do not exist in traditional computer networks. Information diffusion in traditional computer networks can be regarded as passive. In contrast, in CPS networks, information is generated and shared intensively among nodes. The risk that inaccurate information causes bad decisions is also much higher because more decisions are made locally. It is important to understand the evolution of information in CPS networks such that the adaptability and scalability of the systems can be properly engineered. Therefore, understanding how information propagates in the CPS networks and the sensitivity of the behavior with respect to network architecture and topology is a challenge for CPS design.

In a dynamically evolving CPS 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, which is useful to test systems engineering strategies towards

the system design. To model the effects of information exchange, here an information dynamics modeling approach is proposed to analyze the information interdependency among CPS nodes and subnetworks. The proposed model provides the insight of how nodes have influence on each other when information is exchanged and how the behavior of networks evolves dynamically. This dynamics analysis is useful for design of an open system with good adaptability, where nodes can be added and removed arbitrarily.

The proposed information dynamics modeling approach is based on a generic probabilistic graph model of networks ([1], [2]), where information exchange and processing at nodes are modeled at the mesoscale. In the probabilistic graph model, the sensing and computing capabilities of each node are characterized by a prediction probability, whereas the communication capabilities between nodes are characterized by pairwise reliance probabilities. The prediction probability measures how well a node can gather information and make sound judgement. The reliance probabilities capture the extent of influences for one node to another via information exchange. In the proposed information dynamics models, the evolution of prediction capabilities of nodes is captured, where the prediction accuracies of nodes are 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.

Two types of models are proposed here to capture the information dependency. The first one is called copula dynamics model, where correlations between prediction capabilities are explicitly modeled and represented by joint probabilities. The dynamics is modeled with the copulas of extremal probabilities. The second one is the functional interdependency model, where the correlations are captured by linear or nonlinear functional relationships. A data-driven approach is taken to build the information dynamics models, where simulations are performed and the models are trained based on the simulation data.

In the remainder of this paper, the background of CPS design, existing models of information diffusion in networks, and the probabilistic graph model is given in Section 2. The proposed information dynamics models are introduced in Section 3. The models are demonstrated and compared with a CPS simulator in Section 4.

2. BACKGROUND

2.1. CPS Design Challenges and Principles

There are unique engineering challenges in designing CPS. 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 [3][4]. Using new technologies as the augmentation to existing products can effectively enhance adaptability [5]. Second, the complexity of the CPS has significantly increased from traditional products and devices. The CPS 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. 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 realized [1][2]. 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 [6], how to enable context awareness and personalized communication between CPSS and humans [7], and how to quantify trustworthy strategic relationships for information sharing [8]-[10].

In general, the design factors and design principles to be considered for CPSS can be categorized in physical, virtual, and mental worlds. In the physical world, sustainability, reliability, resilience, interoperability, adaptability, biocompatibility, flexibility, and safety need to be considered. In the virtual world, we need to emphasize the principles of human-in-the-loop, data-driven design, co-design, scalability, usability, and security. In the mental world, the perceptions of risk, trust, and privacy, as well as memory capacity and emotion need to be incorporated in design. Those design factors and principles are summarized in **Table 1**. Note that not all factors need to be considered for every design. Some are more important than others for a particular CPS device.

Network connectivity is essential for CPS. A standalone CPS device cannot perform the functions which it is designed for. Compared to traditional products, the design of CPS requires engineers to have better understanding of the systems level behaviors, as

well as the new methodology for the optimization in large-scale networks. Systems level modeling methods and tools have been developed for CPS design and analysis, such as hybrid discrete-event and continuous simulations [11]–[13], inductive constraint logic programming [14] abductive reasoning [15], hybrid timed automaton [16], ontologies [17], information schema [18], UML [19], and SysML [20].

2.2. Information Diffusion in Networks

The information flow in computer networks and social networks has been studied. The propagation of information can be modeled in different ways. One of the most used approaches is the epidemic model of networks, where transmission probabilities of virus between nodes are mainly used to model the speed of

infection and the dynamics of outbreak and decay is captured with ordinary differential equations [21][22]. The epidemic model has been widely applied to study the propagation of keywords or phrases among blogs [23] and within social networks [24]. In the linear influence model [25], the propagation of information is modeled and parameterized by the influences of individual nodes in the network. In the event-driven modeling approaches, the adoption of new information by nodes is characterized by discrete Poisson processes [26][27] or continuous hazard function [28].

There is still no study of information diffusion in CPS networks, which have different behaviors from computer networks, given that CPS nodes possess complex functionalities of sensing, computing,

Table 1 Major design factors and principles to be incorporated in physical, cyber, and social design subspaces respectively.

Domain	Factor	Note
Physical World	Sustainability	The energy consumption of CPSS networks will be significant as the number of nodes grows exponentially.
	Reliability	The reliability CPSS components needs to be maintained as complexity increases with highly integrated functions.
	Resiliency	It is impossible to ensure millions of CPSS nodes be functioning at all time in large networks. Local disruption should not affect the overall functionality.
	Interoperability	Standards of communication for both hardware and software can continuously evolve while legacy systems need to be supported.
	Adaptability	Open architecture is needed for systems that can adapt to different working environments and conditions. The system may evolve by itself.
	Biocompatibility	CPSS should be compatible with biological systems of human or animal for medical purposes.
	Flexibility	CPSS materials that support sensing, computing, communication, and actuation can be soft materials.
	Safety	The operations of CPSS should not bring safety of human users and general public at risk.
Virtual World	Human-In-The-Loop	The level of automation can be very high in CPSS. Yet human involvement can improve the robustness of algorithms and decision making.
	Data Driven Design	The incorporation of product usage data can help identify design constraints and improve the product quality.
	Co-Design	Design of the complex systems requires the simultaneous consideration of software, algorithms, and hardware for better integration.
	Scalability	The software and algorithms should be functional when the complexity of applications increases with growing network and data sizes, and processing heterogeneous data.
	Usability	The design of software and user interface should maintain easy-to-use properties.
	Security	The information collected and shared by CPSS needs to be kept away from illegal access or use.
Mental World	Risk Perception	The perceived risks associated with CPSS usage are personal and vary by individual users.
	Trust	Trust is essential for using CPSS where intensive human entanglement for information gathering, sharing, augmentation, and automation will affect the daily lives of their users and customers.
	Privacy	The extent of data collection and sharing by CPSS needs to be customizable to individual users.
	Memory Capacity	The different mental capacities of human individuals when using and interaction with complex CPSS need to be considered.
	Emotion	In human-system interaction, the emotion of human can vary consistently along time, and the intention can be disguised and difficult to detect.

communication, and control.

2.3. Probabilistic graph model

A recently developed probabilistic graph model [2] for CPS networks, as the foundation of the information dynamics models, is introduced here. In the probabilistic graph model, each node has its own sensing, reasoning, and communication units. As illustrated in Figure 1, there are probabilities associated with information gathering and exchange between nodes. For each node, there is a prediction probability indicating the capabilities of information gathering and reasoning. For each directed edge indicating information exchange, there are two reliance probabilities associated with it. The three probabilities are defined as follows.

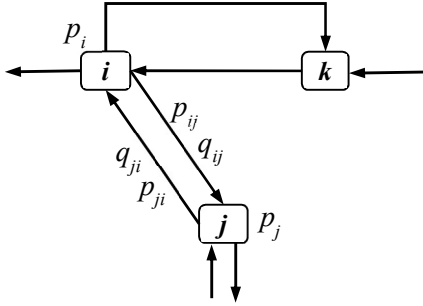


Figure 1 Probabilistic graph model of CPS systems

The *prediction probability* that the k^{th} node detects the true state of world θ is

$$\mathbb{P}(x_k = \theta) = p_k \quad (1)$$

where x_k is the state variable. For convenience, we denote $q_k = 1 - p_k$. The information dependency between nodes is modeled with probabilistic adjacency matrix $\mathbf{R}_P$ with elements as *P-reliance probabilities*

$$\mathbb{P}(x_j = \theta | x_i = \theta) = p_{ij} \quad (2)$$

which is the probability that the j^{th} node predicts the true state of world given that the i^{th} node predicts correctly. Similarly, we also have adjacency matrix $\mathbf{R}_Q$ with elements as *Q-reliance probabilities*

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

because nodes could be negatively correlated, or miscommunication between nodes could exist. The reliance probability can be used to model reliability of communication between nodes, e.g. in moving vehicles' ad hoc wireless networks, data packet loss is not uncommon.

Therefore, different from the adjacency matrix in traditional graph model with binary “yes-or-no” edge connection topology, there are reliance probabilities associated with each pair of nodes in the new probabilistic graph model. If the communication channel from node i and node j is disrupted, both p_{ij} and q_{ij} are zeros.

The random state variables with binary values ($= \theta$ or $\neq \theta$) can be extended to multiple values or continuous. For instance, one sensor measures a value (e.g. temperature or flow speed) which follows some distribution, as in prediction probability. If there are a finite set of possible values $\{\theta_1, \dots, \theta_N\}$ for state variables. The prediction probability $\mathbb{P}(x_k = \theta_n)$ and P-reliance probability $\mathbb{P}(x_j = \theta_n | x_i = \theta_m)$, where $1 \leq m, n \leq N$, can be enumerated similarly.

The edges in the probabilistic graph are directional. The neighbors of each node can be further differentiated as *source* nodes or *destination* nodes, as illustrated in Figure 2. For one node, its source nodes are those sending information to this node, whereas the destination nodes are those receiving information from it. When receiving different cues from source nodes, a CPS node can update its prediction probability to reflect its perception of the world. The aggregation of prediction probabilities sensitively depends on the rules of information fusion during the prediction update.

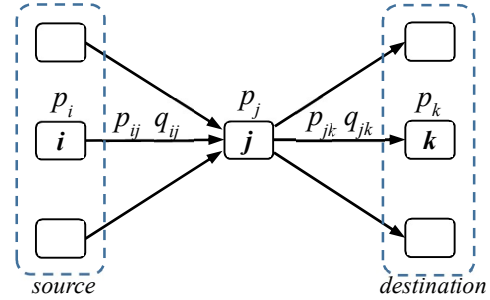


Figure 2 Source and destination nodes with respect to node j

If $P(x_k)$ and $P(x_k^C)$ denote the probabilities of a positive and a negative prediction from node k respectively, we define the *best-case* fusion rule as

$$P'(x_k) = 1 - (1 - P(x_k)) \prod_{i=1}^{M_P} P(x_i)(1 - P(x_k | x_i)) \prod_{j=1}^{M_N} P(x_j^C)(1 - P(x_k | x_j^C)) \quad (4)$$

where node k updates its prediction based on its own current prediction and those cues from its $M_P + M_N$ source nodes, out of which M_P of the source nodes provide positive predictions whereas M_N of them

provide negative predictions, $P(x_k|x_i)$ indicates the probability that a positive message from node i leads to a positive prediction of node k , and $P(x_k|x_j^c)$ is the probability that a negative message from node j leads to a positive prediction of node k . Therefore, if any of the cues from the source nodes is positive, the prediction of the node is positive. Some variations of this fusion rules exist. For instance, the previous prediction from itself can be either included or excluded during the update.

Similarly, the *worst-case* fusion rule can be defined as

$$P'(x_k) = P(x_k) \prod_{i=1}^{M_P} P(x_i) P(x_k|x_i) \prod_{j=1}^{M_N} P(x_j^c) P(x_k|x_j^c) \quad (5)$$

That is, if any of the cues from the source nodes is negative, the prediction of the node is negative. The *Bayesian* fusion rule is defined as

$$P'(x_k) = \frac{P(x_k) \max_P \{ (P(x_k))^r (1-P(x_k))^{S-r} \}}{\int (P(x_k))^r (1-P(x_k))^{S-r} dP} \quad (6)$$

where the prediction of the node is updated to P' from prior prediction P , and out of S cues that the neighboring nodes provide, r of them provide are positive, if the maximum likelihood principle is taken.

The probabilistic graph model provides a system level abstraction and a mesoscale description of CPS networks, where information exchange and aggregation are captured. More details about the probabilistic graph model can be found in Ref.[2].

3. THE INFORMATION DYNAMICS MODEL

Based on the probabilistic graph model, we propose two types of information dynamics models to capture the information diffusion in CPS networks based on the probabilities that the nodes produce meaningful information. The information dynamics models are to characterize and predict how information is produced and consumed in a networked CPS environment. In CPS networks, each node produces information by sensing and processing. Information is exchanged among 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. Thus the prediction probabilities of CPS nodes are dynamically updated with the mutual influences among each other. Therefore, strong dependencies exist among the prediction probabilities from different nodes.

To understand the propagation of information in CPS networks, the influences and interdependency among information producers and consumers need to be modeled. The high correlation between nodes need to be incorporated in modeling the dynamics. In the proposed information dynamics model, two approaches are taken to capture the interdependency. One is using joint probabilities, and the other is based on functional relationships.

In the first approach, the joint probabilities of prediction capabilities instead of the marginal probabilities are used to model the information dynamics. Joint probabilities explicitly capture the correlation among nodes. However, joint probabilities with a large number of variables are not easy to calculate, given that the number of nodes in a CPS network can be very large. Therefore copulas are proposed here to estimate the joint probabilities of prediction. The precise joint probabilities or copulas are unknown but can be estimated from the ones in some extreme scenarios, such as perfectly positive or negative correlation, or completely independence. The joint probabilities or copulas are known as extremal distributions. The extremal probabilities 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 joint probabilities can be estimated. From the joint probabilities, the marginal probabilities of predictions can be easily calculated. In the information dynamics model, the evolutions of extremal probabilities are explicitly modeled. Here, a data-driven approach is taken to model the dynamics of extremal probabilities. The interactions between nodes can be simulated from the probabilistic graph model. The copulas for extremal probabilities then can be obtained. Time series models can be used and trained from the 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, it will be computationally challenging to use joint probabilities to keep track of the evolution of large-scale networks.

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. That is, the prediction probability of one node is a function of the probabilities from its neighbors.

Here, the copula approach is introduced in Section 3.1, where the construction of extremal probabilities and

the estimation of joint probabilities are described. The functional approach to capture interdependency is described in Section 3.2.

3.1. Copula Dynamics

For n random variables $X_1, \dots, X_n$, each of which has the marginal cumulative distribution function $U_i(x) = \mathbb{P}(X_i \leq x)$, the copula $C: [0,1]^n \rightarrow [0,1]$ describes the joint cumulative distribution function of these random variables from their marginal distributions and is generally defined as

$$C(u_1, \dots, u_n) = \mathbb{P}[U_1 \leq u_1, \dots, U_n \leq u_n] \quad (7)$$

In a probabilistic graph model with n nodes, given the joint probability $P(x_1, \dots, x_n) = \mathbb{P}[x_1 = \theta, \dots, x_n = \theta]$ for the case that all predictions are positive, the corresponding copula is

$$\begin{aligned} C(p_1, \dots, p_n) \\ &= \mathbb{P}[\mathbb{P}(x_1) \leq P(x_1), \dots, \mathbb{P}(x_n) \leq P(x_n)] \\ &= P(x_1, \dots, x_n) \end{aligned} \quad (8)$$

This is due to the simplicity of $\mathbb{P}(x_i \leq \theta) = \mathbb{P}(x_i = \theta)$ 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.

Extremal Probabilities

The well-known Fréchet bounds of the copula are given as

$$\begin{aligned} \max\{0, \mathbb{P}(x_1) + \dots + \mathbb{P}(x_n) + 1 - n\} \leq \\ C(\mathbb{P}(x_1), \dots, \mathbb{P}(x_n)) \leq \min\{\mathbb{P}(x_1), \dots, \mathbb{P}(x_n)\} \end{aligned} \quad (9)$$

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

$$\max\{0, \mathbb{P}(x_1) + \mathbb{P}(x_2) - 1\} \leq C(\mathbb{P}(x_1), \mathbb{P}(x_2)) \leq \min\{\mathbb{P}(x_1), \mathbb{P}(x_2)\} \quad (10)$$

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 $\rho(x, y) = \text{Cov}(x, y) / \sqrt{\sigma^2(x)\sigma^2(y)}$, 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 $\rho_\tau(x, y) = \mathbb{P}[(x_1 - x_2)(y_1 - y_2) > 0] - \mathbb{P}[(x_1 - x_2)(y_1 - y_2) < 0]$ 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 $\mathcal{V}_1 = \{x_1, \dots, x_m\}$, all nodes are perfectly positively correlated. Within the second subset $\mathcal{V}_2 = \{x_{m+1}, \dots, x_n\}$, 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 $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2$, the extremal joint probability is

$$\begin{aligned} C_{\{\mathcal{V}_1, \mathcal{V}_2\}}(\mathbb{P}(x_1), \dots, \mathbb{P}(x_n)) &= \mathbb{P}(x_1, \dots, x_n | \{\mathcal{V}_1, \mathcal{V}_2\}) \\ &= \max\{0, \min_{i \in \mathcal{V}_1} \{\mathbb{P}(x_i)\} + \min_{j \in \mathcal{V}_2} \{\mathbb{P}(x_j)\} - 1\} \end{aligned} \quad (11)$$

when the correlation within either group is perfectly positive but perfectly negative between the two groups. When all nodes have perfectly positive correlation without partition, the extremal joint probability is

$$C_{\{\mathcal{V}+\}}(\mathbb{P}(x_1), \dots, \mathbb{P}(x_n)) = \mathbb{P}(x_1, \dots, x_n | \{\mathcal{V}+\}) = \min_{i \in \mathcal{V}} \{\mathbb{P}(x_i)\} \quad (12)$$

Another extremal joint probability is

$$C_{\{\mathcal{V}\perp\}}(\mathbb{P}(x_1), \dots, \mathbb{P}(x_n)) = \mathbb{P}(x_1, \dots, x_n | \{\mathcal{V}\perp\}) = \prod_{i \in \mathcal{V}} \mathbb{P}(x_i) \quad (13)$$

when all nodes are independent from each other.

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

$$C(\mathbb{P}(x_1), \dots, \mathbb{P}(x_n)) = \alpha C_{\{\mathcal{V}+\}} + \sum_{j=1}^N \beta_j C_{\{\mathcal{V}_1, \mathcal{V}_2\}} + (1 - \alpha - \sum_{j=1}^N \beta_j) C_{\{\mathcal{V}\perp\}} \quad (14)$$

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 $\beta_j = 0$ ($j = 1, \dots, N$) corresponds to the perfect positive correlation among all nodes,

whereas $\alpha = \beta_j = 0$ ($j = 1, \dots, N$) corresponds to the complete independence among all nodes. The copula $C(\mathbb{P}(x_1), \dots, \mathbb{P}(x_n))$ thus is located in the convex set formed by the extremal distributions, if all extremal distributions can be calculated. The challenge of estimating copulas however is to find out the weight coefficients.

The decomposition principle in Eq. (14) can be further generalized. If the nodes are self-organized into M different independent groups and nodes between groups are uncorrelated, then

$$C(\mathbb{P}(x_1), \dots, \mathbb{P}(x_n)) = \prod_{m=1}^M C_m \quad (15)$$

where copulas C_m 's are estimated according to Eq. (14) from the extremal distributions for partitions and independence within each group. Furthermore, if nodes are conditionally independent within each partition, the extremal distributions can also be estimated more accurately. In other words, additional dependency information among nodes helps identify the extremal probabilities more precisely.

Data-Driven Dynamics Modelling

The evolution of copulas with extremal probabilities can be generally modelled as

$$dC_{\{v_1, v_2\}}(t)/dt = f(C_{\{v_1, v_2\}}(t)) + \epsilon \quad (16)$$

where $\{v_1, v_2\}$ corresponds to any partition of nodes as in Eq. (11), and ϵ is the random noise term. Each copula in Eq. (14) has a respective dynamics model. A simple numerical approximation of Eq. (16) as time series autoregressive (AR) model is

$$C_{\{v_1, v_2\}}(k) = \gamma_0 + \sum_{l=1}^L \gamma_l C_{\{v_1, v_2\}}(k-l) + \epsilon \quad (17)$$

where the k -th time step value depends on the values of previous L steps, 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. (17) captures the time correlation of the expected values of copulas. Other more complex models such as autoregressive moving average (ARMA) can also be applied.

The data-driven approaches are necessary to calibrate the dynamics models. Based on the probabilistic graph model in Section 2.3, 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 modelling, two training procedures are needed. First, the weight coefficients α and β_j 's in Eq. (14) need to be trained and calibrated so that the actual joint probabilities of

state variables can be estimated from the extremal probabilities. Second, the parameters γ_0 and γ_l 's of the dynamics models of copulas in Eq. (17) also need to be calibrated. After parameter calibration, the models can be applied to predict the future values.

3.2. Functional Interdependency

The second approach to model the interdependency between predictions is to use analytical functions. The dynamics of prediction probabilities can be modelled by

$$\begin{aligned} \frac{dP(x_1, t)}{dt} &= f_1(P(x_1, t), \dots, P(x_n, t)) + \epsilon_1 \\ &\vdots \\ \frac{dP(x_n, t)}{dt} &= f_n(P(x_1, t), \dots, P(x_n, t)) + \epsilon_n \end{aligned} \quad (18)$$

where $f_1, \dots, f_n$ can be linear and nonlinear functions to capture the interdependency between prediction probabilities $P(x_k)$'s. If the prediction probabilities of all nodes are considered as a vector $(t) = [P(x_1, t), \dots, P(x_n, t)]^T$, the model is written as

$$\frac{d\mathbf{P}(t)}{dt} = \mathbf{F}(\mathbf{P}(t)) + \boldsymbol{\epsilon} \quad (19)$$

A simple linearized vector autoregression (VAR) model is

$$\mathbf{P}(k) = \mathbf{A}_0 + \sum_{l=1}^L \mathbf{A}_l \mathbf{P}(k-l) + \boldsymbol{\epsilon} \quad (20)$$

where the vector value at the k -th time step is related to the values at the previous L steps, $\mathbf{A}_0$ is the vector of intercepts, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon)$ 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. (20) captures the time and location dependencies of nodes simultaneously as the linear relationships.

More complex models can be applied to capture the functional interdependency between prediction probabilities. For instance, latent variables or hidden state variables can be introduced to capture the inherent correlations between prediction capabilities. An example of hidden state models can be defined as

$$\mathbf{P}(k) = \mathbf{B}_0 + \mathbf{B}\mathbf{V}(k) + \boldsymbol{\eta} \quad (21)$$

$$\mathbf{V}(k) = \mathbf{T}_0 + \mathbf{T}\mathbf{V}(k-1) + \boldsymbol{\epsilon} \quad (22)$$

where Eq. (21) captures the relation between observable $\mathbf{P}$ and hidden variables $\mathbf{V}$, the evolution of hidden state variables or state transition is modelled in Eq.(22). Here, $\mathbf{T}$ is the transition matrix, $\mathbf{T}_0$ is the vector of intercepts, $\mathbf{B}$ is the observation matrix, and $\mathbf{B}_0$ is the vector of observation bias. Notice that the

dimension of state vector $\mathbf{V}$ is not necessarily the same as the dimension of observable vector $\mathbf{P}$. $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}})$ and $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\varepsilon}})$ are associated with the noises of observation and state transition respectively. In the hidden state model, the interdependency among nodes is captured through the hidden state variables. The correlations between state variables or the common state variables corresponding to the observable variables represent the inherent correlations between the observables. More complex nonlinear models can be obtained similarly. The computational costs associate with nonlinear models however are higher than those with linear ones. Therefore nonlinear models are often approximated by linear models to reduce the computational complexity.

The parameter calibration process here will be similar to the ones for the copula dynamics models. In the VAR model in Eq. (20), the parameters to be calibrated are vector $\mathbf{A}_0$ and matrices $\mathbf{A}_l$'s. For the hidden state model in Eqs. (21) and (22), parameters $\mathbf{T}$, $\mathbf{T}_0$, $\mathbf{B}$, and $\mathbf{B}_0$ need to be calibrated.

4. DEMONSTRATIVE EXAMPLES

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. The VAR functional interdependency model are demonstrated with some larger networks. A CPS network simulator is developed to simulate the information update based on Monte Carlo sampling. The proposed information dynamics models were developed and compared with the Monte Carlo simulation results. Both the simulator and the information dynamics models were implemented in python programming language.

4.1. Demonstration of The Copula Dynamics Model

In the first example, a three-node random network is created where the nodes are connected at different probabilities. 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. The simulation algorithm is listed in Table 2. 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.

Table 2 The simulation algorithm to generate sequences of prediction and joint probabilities for nodes along time.

Input: A probabilistic graph model with initial prediction probabilities and reliance probabilities	
Output: Time sequences of prediction probabilities $P(x_k)$'s and joint probabilities $P(x_1, \dots, x_n)$'s.	
(1)	While maximum time steps is not reached:
(2)	Randomly generate the ground truth state value. Randomly generate a sample of correct observation for each node based on its prediction probability.
(3)	Based on the observations from its source nodes, each node obtains samples of observations based on P-reliance probabilities $P(x_k x_i)$'s if the source node predicts correctly, or based on Q-reliance probabilities $P(x_k x_j^c)$'s if the source node predicts incorrectly.
(4)	Update the prediction probability of each node according to a fusion rule (e.g. worst-case, best-case, Bayesian)
(5)	Repeat Steps (2-4) for N times.
(6)	Calculate the probabilities of correct prediction for individual nodes and the joint probabilities
(5)	Update the prediction probability for each node in the graph
(6)	Go to Step (1) for the next time step

After the simulation data are obtained, the extremal probabilities also need to be calculated based on Eqs. (11)-(13). To calibrate the weight coefficients α and β_j 's in Eq. (14), 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. (14). 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. (17) 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.

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. (14), the 8 joint probabilities for a future time step can be estimated. From the 8 joint probabilities, 3 marginal prediction probabilities can be easily obtained.

For the first three-node example, all nodes are fully connected with all 6 directional edges, as in Figure 3. The randomly generated initial prediction probabilities and reliance probabilities are listed in Table 3 to Table 5. The probability update is simulated for 60 time steps. The simulated data are used to train the copula model and the AR models. After training, the calibrated weight coefficients in Eq. (14) are shown in Table 6. Three examples of the calibrated coefficients for the 40 AR models are shown in Table 7. Here the lag order is $L=2$. The obtained standard deviations corresponding to the three models are listed in Table 8. The variances of the marginal probability values are the sum of the ones of the joint probability values, assuming that the joint probabilities are independent combinations.

Table 8. Sensitivity studies of choosing different lag order L 's were also conducted. The results showed that the model predictions are not sensitive to the choice of lag order in our model. Higher orders cause slightly higher computational costs. The lag order is also independent from the number of nodes. After training, the models are applied to predict the probability update for additional 30 time steps. In Figure 3, the simulated prediction probability update and the forecast are compared, where the mean and the bounds of two standard deviation are shown. The standard deviations of the marginal prediction probabilities shown in Figure 3 are estimated from the standard deviations of the extremal probabilities such as the ones listed in Table 8. The variances of the marginal probability values are the sum of the ones of the joint probability values, assuming that the joint probabilities are independent combinations.

Table 8. It is assumed that the variances associated with the extremal probability values are the same as the ones with joint probabilities.

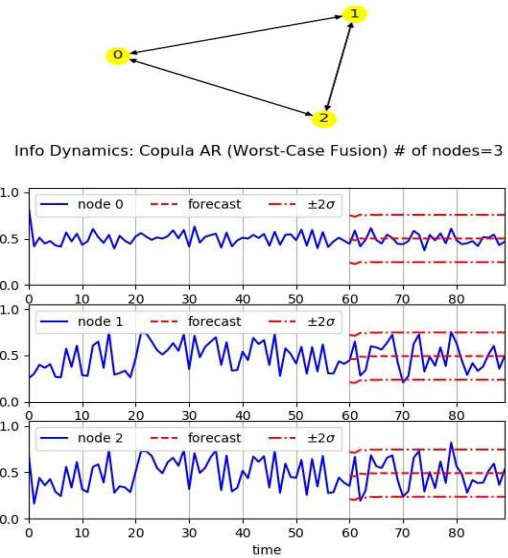


Figure 3 The simulated prediction probabilities and forecast starting from time step 60 by the copula dynamics model in a three-node-six-edge example.

Table 3 The initial prediction probabilities.

	Node 0	Node 1	Node 2
$P(x_k)$	0.223014	0.096656	0.321909

Table 4 The pairwise P-reliance probabilities.

$P(x_k x_i)$	Node 0	Node 1	Node 2
Node 0		0.171229	0.945277
Node 1	0.094149		0.265593
Node 2	0.257377	0.856983	

Table 5 The pairwise Q-reliance probabilities.

$P(x_k x_i^c)$	Node 0	Node 1	Node 2
Node 0		0.608827	0.548566
Node 1	0.917376		0.271680
Node 2	0.383766	0.364328	

Table 6 Calibrated weight coefficients in Eq. (14) for the 3-node example in Figure 3.

Index	Coefficient
1	0.171246
2	0.171111
3	0.171105
4	0.171483
5	0.315056

Table 7 Three examples of the calibrated coefficients for the AR models to predict future extremal probabilities.

	γ_0	γ_1	γ_2
example 1	0.0075325	0.27051428	-0.18069243

example 2	0.1269418	-0.03805253	-0.14354211
example 3	0.0243994	0.0312964	-0.14376424

The variances of the marginal probability values are the sum of the ones of the joint probability values, assuming that the joint probabilities are independent combinations.

Table 8 The standard deviations for the three calibrated AR models Table 7.

	σ_ϵ
example 1	0.0208359986
example 2	0.1614207112
example 3	0.0347577082

It is seen that the general trends of probabilities are predicted well by the models. The fluctuations are also enclosed by the error bounds. The bounds of two standard deviations are supposed to enclose 95% of samples in a normal distribution. The mean values of forecast are stabilized after a few steps, indicating that the system remains equilibrium in long term. The variances of the forecast for all three prediction probability values are similar. This is because all variances are estimated from those ones associated joint probabilities, which have all nodes involved.

To assess the robustness of the proposed copula dynamics model for the three-node network, sensitivity analyses are performed by reducing the number of edges. In Figure 4, the number of the directed edges is reduced to 2. The initial prediction probabilities and reliance probabilities are randomly generated. Simulation data collection and model training are similarly done. The results in Figure 4 show that the model can predict the trend well. Compared to the previous three-node-six-edge case in Figure 3, the variabilities of predictions by nodes increase. Therefore, the variances of forecast are also increased. The increased error bounds tend to overestimate the fluctuation range for those nodes which provide more stable predictions, such as Node 1 in Figure 4. Node 1 in this case receives information from Node 0 directly and Node 2 indirectly. The general trend is that when a node receives more information, its prediction capability increases with smaller fluctuations. Here, Node 2 fluctuates the most, since it does not receive information from others. Node 0 receives information from Node 2, and its prediction is more stable than the one by Node 2.

When the number of edges is further reduced to 1, the results of simulation and forecast are shown in Figure 5. Without receiving information, the prediction capabilities of Node 0 and Node 1 fluctuate significantly. As a result, the error bounds of the forecast further increase from the previous cases.

It is seen in this example that the copula dynamics model can provide accurate estimations of the trends and variances associated with prediction capabilities for the nodes. As the number of nodes increases, the number of joint probabilities and copulas of extremal distributions will increase exponentially. Therefore, the disadvantage of the copula dynamics model is the computational complexity for larger systems.

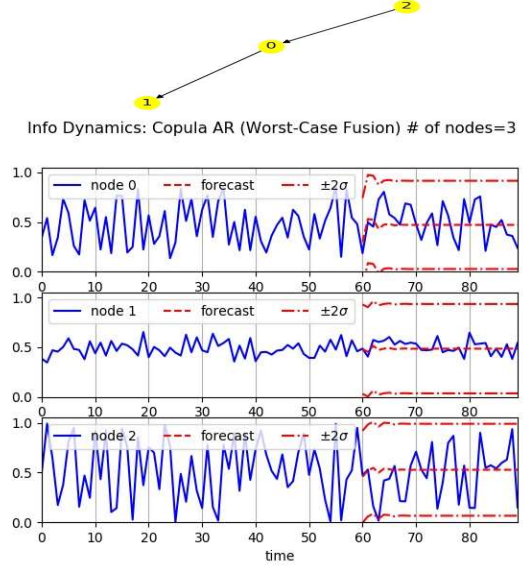


Figure 4 The simulated prediction probabilities and forecast starting from time step 60 by the copula dynamics model in a three-node-two-edge example.

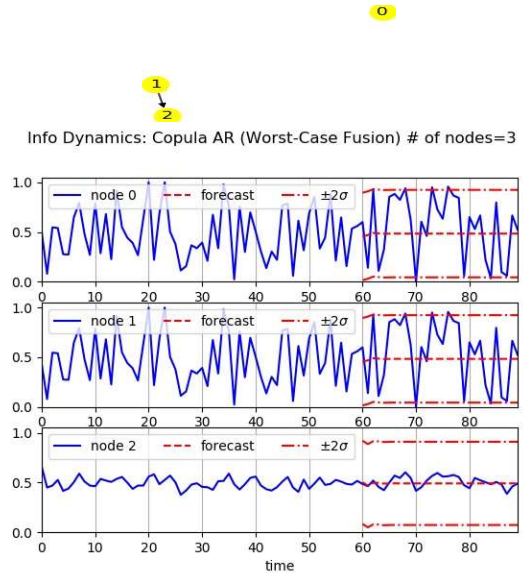


Figure 5 The simulated prediction probabilities and forecast starting from time step 60 by the copula dynamics model in a three-node-one-edge example.

4.2. Demonstration of The Functional Interdependency Model

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 prediction probability values are directly modelled.

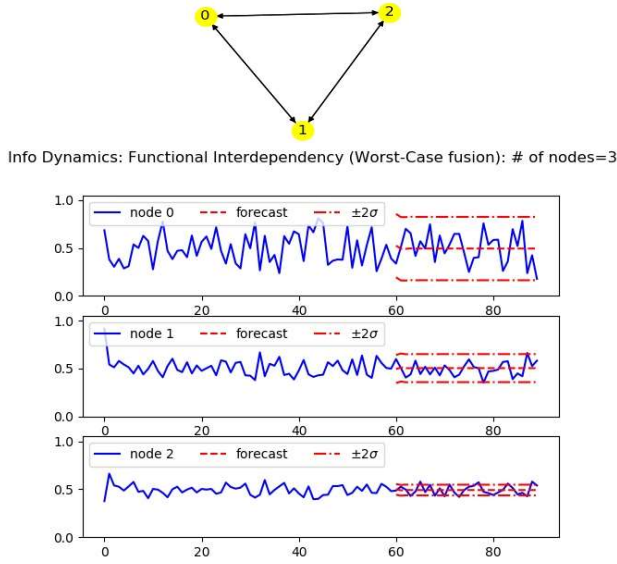


Figure 6 The simulated prediction probabilities and forecast starting from time step 60 by the VAR model in a three-node-six-edge

Table 9 The calibrated VAR model parameters in Figure 6 for the three-node-six-edge graph.

A_0	$[1.28702158 \quad 0.59495947 \quad 0.4537085]^T$
A_1	$\begin{bmatrix} -0.32078891 & 0.08630353 & -0.11113924 \\ -0.72776017 & 0.01507701 & 0.33916682 \\ -0.14554776 & -0.20091567 & -0.06757319 \end{bmatrix}$
A_2	$\begin{bmatrix} 0.0236274 & -0.04637095 & -0.03657765 \\ -0.15761436 & -0.00410451 & 0.0106602 \\ -0.25973278 & -0.03305852 & -0.06595382 \end{bmatrix}$
Σ_ϵ	$\begin{bmatrix} 0.02736594 & -0.01089281 & -0.00164103 \\ -0.01089281 & 0.00535025 & 0.00067239 \\ -0.00164103 & 0.00067239 & 0.00078254 \end{bmatrix}$

Here, the VAR model in Eq. (20) is demonstrated. The VAR model is applied to the previous three-node-six-edge example. The simulation data are collected to train the VAR model with lag order $L=2$. The training data and forecast results are shown in Figure 6. The calibrated model parameters are listed in Table 9. The error bounds are defined as two standard deviations, which are directly obtained from the covariance matrix in Table 9 after the training procedure. The model predicts the trend well. Compared to the copula dynamics model, the predicted error bounds are more precise and specific for different nodes associated

with their forecasted values. This indicates that the VAR model can provide more information about the dynamics of prediction capabilities than the copula dynamics model can.

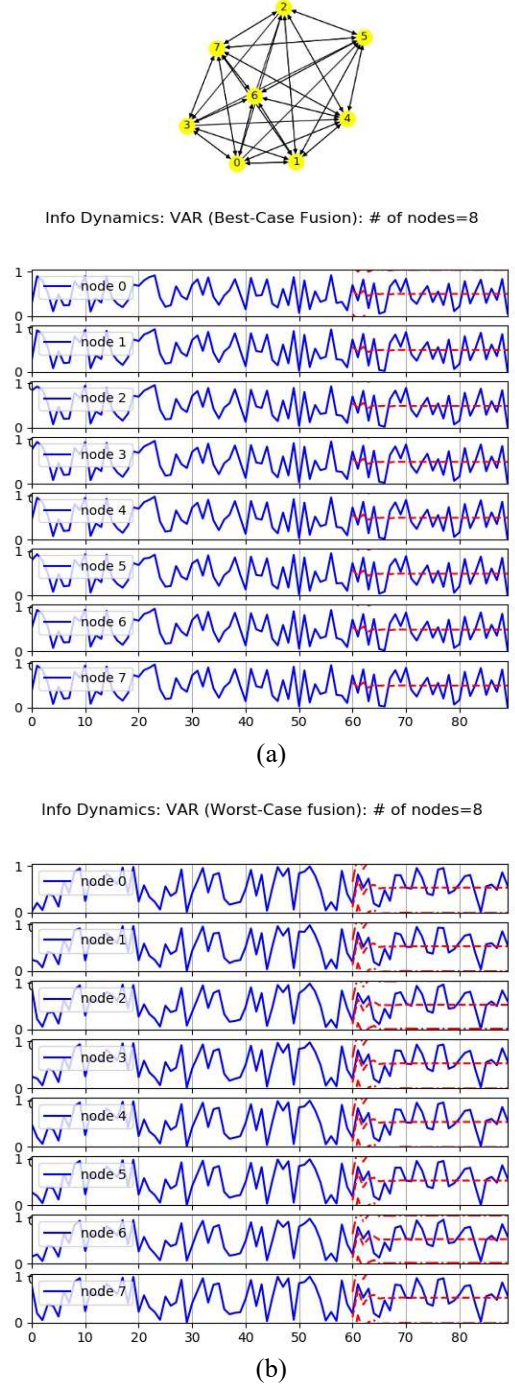
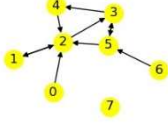
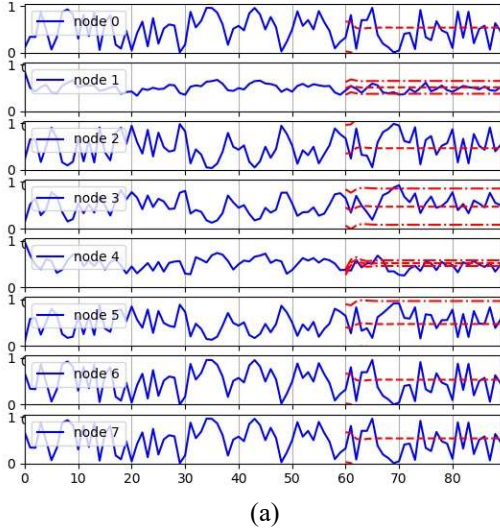


Figure 7 The simulated prediction probabilities and forecast starting from time step 60 by the VAR model in the 8-node-56-edge example. (a) with the best-case fusion rule; (b) with the worst-case fusion rule.

More examples are tested for the VAR model. In Figure 7, an 8-node-56-edge example is shown, where both best-case and worst-case fusion rules are applied. It is seen that the prediction capabilities of nodes fluctuate significantly.



Info Dynamics: VAR (Best-Case Fusion): # of nodes=8



Info Dynamics: VAR (Worst-Case fusion): # of nodes=8

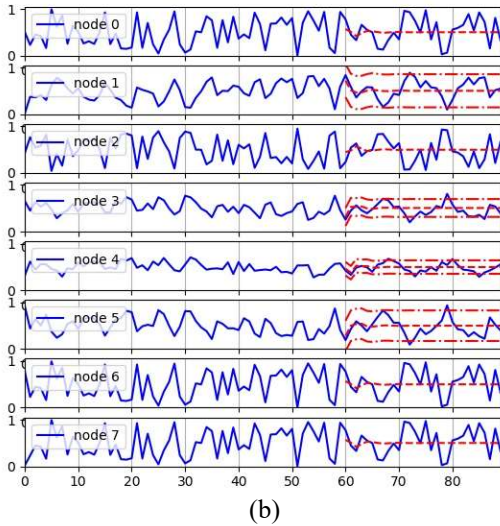


Figure 8 The simulated prediction probabilities and forecast starting from time step 60 by the VAR model in the 8-node-10-edge example. (a) with the best-case fusion rule; (b) with the worst-case fusion rule.

With the full connections between nodes, nodes share information among each other extensively. The mutual influences cause a large level of fluctuations. It is also seen that the fluctuations among nodes are synchronized. Strong correlations exist between them.

When the number of edges is reduced to 10 for the 8 nodes, the simulated and forecasted prediction capabilities are shown in Figure 8. Compared to the previous full-connection case in Figure 7, the extent of fluctuation is reduced. The forecasts are also more accurate with reduced error bounds.

5. CONCLUDING REMARKS

The analyses of the systems level behavior of CPS networks enable us to design better systems. How to design a CPS system which promotes effective information sharing is one of the major aspects of design. Therefore, we need models that can characterize and predict the information sharing behaviors of such systems.

In this paper, information dynamics models are proposed to predict the information propagation within a CPS network. Based on a recently developed mesoscale probabilistic graph model, two dynamics models are introduced to capture the mutual influences between nodes on their reasoning processes. The representation of correlations between nodes is the central theme in both models.

The first model represents the interdependency between nodes for their prediction capabilities explicitly with the joint probabilities of successful predictions. Theoretically the evolution of joint probabilities can capture the complete correlation information and help us to understand the information interdependency. However, the computational complexity increases exponentially as the number of nodes increases. Here, a data-driven copula dynamics model is proposed to capture the evolution of joint probabilities for the nodes' correct predictions via the copulas for extremal probabilities. Instead of modeling the dynamics of joint probabilities directly, the dynamics of their convex hulls known as the extremal probabilities are modeled. The advantage of this approach is that it is easier to estimate the extremal probabilities than the joint probabilities themselves in practice. Nevertheless, from the extremal probabilities, the joint probabilities still need to be estimated by interpolating the extremal probabilities. The data-driven approach demonstrated in this paper shows the possible ways of estimating the joint probabilities with the interpolation from their

bounds as well as predicting the dynamics of the bounds.

In the second model, the correlation between prediction capabilities is captured analytically with functional relationships, such as linear and nonlinear ones. This approach simplifies the dependencies as mathematical relations. Existing statistical models for time series analyses can be adopted for this purpose. The main limitation however is that the details of interdependency can be lost in the general mathematical models.

The prediction accuracy from the data-driven models sensitively relies on the training datasets. In general, larger datasets are always better for model training and calibration. For situations where there is a lack of training data, the proposed modeling approach will not be feasible. Alternative modeling approaches that are based more on the detailed knowledge about the systems will be needed. The current information diffusion models are only evaluated with a mesoscale network simulator. The comparisons with the detailed network simulators and actual experimental data are needed in further evaluations in the future.

ACKNOWLEDGMENTS

This work is supported in part by National Science Foundation under grant CMMI-1663227.

REFERENCES

- [1] Wang, Y. (2016) System resilience quantification for probabilistic design of Internet-of-Things architecture. In *Proc. 2016 ASME International Design Engineering Technical Conferences & The Computer and Information in Engineering Conference* (IDETC/CIE2016), Aug. 21-24, 2016, Charlotte, North Carolina, pp. V01BT02A011.
- [2] Wang, Y. (2018). Resilience Quantification for Probabilistic Design of Cyber-Physical System Networks. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*. 4(3), 031006.
- [3] Horváth, I., & Gerritsen, B. H. (2012). Cyber-physical systems: Concepts, technologies and implementation principles. In *Proc. 9th International Symposium on Tools and Methods of Competitive Engineering* (TMCE2012), pp. 19–36.
- [4] Tavčar, J., & Horváth, I. (2018). A review of the principles of designing smart cyber-physical systems for run-time adaptation: Learned lessons and open issues. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(1), 145-158.
- [5] Vroom, R. W., & Horváth, I. (2014). Cyber-physical augmentation: An exploration. *Proc. 10th International Symposium on Tools and Methods of Competitive Engineering* (TMCE2014), pp. 1509-1520.
- [6] Horváth, I., & Wang, J. (2015). Towards a comprehensive theory of multi-aspect interaction with cyber physical systems. In *Proc. ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (IDETC/CIE2015), pp. V01BT02A009.
- [7] Li, Y., Horváth, I., & Rusák, Z. (2018). Constructing Personalized Messages for Informing Cyber-Physical Systems based on Dynamic Context Information Processing. In *Proc. 12th International Symposium on Tools and Methods of Competitive Engineering* (TMCE2018), pp.105–120.
- [8] Wang, Y. (2018). Trustworthiness in designing cyber-physical systems. In *Proc. 12th International Symposium on Tools and Methods of Competitive Engineering* (TMCE2018), pp.27–40.
- [9] Wang, Y. (2018). Trust Based Cyber-Physical Systems Network Design. In *Proc. ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (IDETC/CIE2018), pp. V01AT02A037.
- [10] Wang, Y. (2018). Trust quantification for networked cyber-physical systems. *IEEE Internet of Things Journal*, 5(3), 2055-2070.
- [11] Jeon, J., Chun, I., & Kim, W. (2012). Metamodel-based CPS modeling tool. In *Embedded and Multimedia Computing Technology and Service*, Lecture Notes in Electrical Engineering, Vol. 181 (pp. 285-291). Springer.
- [12] Lee, K. H., Hong, J. H., & Kim, T. G. (2015). System of Systems Approach to Formal Modeling of CPS for Simulation-Based Analysis. *ETRI Journal*, 37(1), 175-185.
- [13] Lee, E. A., Niknami, M., Nouidui, T. S., & Wetter, M. (2015, October). Modeling and simulating cyber-physical systems using CyPhySim. In *Proceedings of the 12th IEEE International Conference on Embedded Software* (pp. 115-124). .
- [14] Saeedloei, N., & Gupta, G. (2011). A logic-based modeling and verification of CPS. *ACM SIGBED Review*, 8(2), 31-34.
- [15] Horváth, I. (2019). A Computational Framework for Procedural Abduction Done by Smart Cyber-Physical Systems. *Designs*, 3(1), 1.

- [16] Burmester, M., Magkos, E., & Chrissikopoulos, V. (2012). Modeling security in cyber-physical systems. *International Journal of Critical Infrastructure Protection*, 5(3-4), 118-126.
- [17] Petnga, L., & Austin, M. (2016). An ontological framework for knowledge modeling and decision support in cyber-physical systems. *Advanced Engineering Informatics*, 30(1), 77-94
- [18] Pourtalebi, S., & Horváth, I. (2017). Information schema constructs for instantiation and composition of system manifestation features. *Frontiers of Information Technology & Electronic Engineering*, 18(9), 1396-1415.
- [19] Magureanu, G., Gavrilesco, M., Pescaru, D., & Doboli, A. (2010, September). Towards UML modeling of cyber-physical systems: A case study for gas distribution. In *Proc. IEEE 8th International Symposium on Intelligent Systems and Informatics* (pp. 471-476).
- [20] Palachi, E., Cohen, C., & Takashi, S. (2013, April). Simulation of cyber physical models using SysML and numerical solvers. In *Proc. 2013 IEEE International Systems Conference (SysCon)* (pp. 671-675).
- [21] Newman, M. E. (2002). Spread of epidemic disease on networks. *Physical review E*, 66(1), 016128.
- [22] Wu, F., Huberman, B. A., Adamic, L. A., & Tyler, J. R. (2004). Information flow in social groups. *Physica A: Statistical Mechanics and its Applications*, 337(1-2), 327-335.
- [23] Gruhl, D., Guha, R., Liben-Nowell, D., & Tomkins, A. (2004, May). Information diffusion through blogspace. In *Proceedings of the 13th international conference on World Wide Web* (pp. 491-501).
- [24] Kleineberg, K. K., & Boguná, M. (2014). Evolution of the digital society reveals balance between viral and mass media influence. *Physical Review X*, 4(3), 031046.
- [25] Yang, J., & Leskovec, J. (2010, December). Modeling information diffusion in implicit networks. In *2010 IEEE International Conference on Data Mining* (pp. 599-608).
- [26] Simma, A., & Jordan, M. I. (2010, July). Modeling events with cascades of poisson processes. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence* (pp. 546-555).
- [27] Iwata, T., Shah, A., & Ghahramani, Z. (2013, August). 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).
- [28] Du, N., Song, L., Woo, H., & Zha, H. (2013, April). Uncover topic-sensitive information diffusion networks. In *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics* (pp. 229-237).