Influence Spread in the Heterogeneous Multiplex Linear Threshold Model

Yaofeng Desmond Zhong (钟耀锋), Vaibhav Srivastava, *Senior Member, IEEE*, and
Naomi Ehrich Leonard, *Fellow, IEEE*

Abstract—The linear threshold model (LTM) has been used to study spread on single-layer networks defined by one inter-agent sensing modality and agents homogeneous in protocol. We define and analyze the heterogeneous multiplex LTM to study spread on multi-layer networks with each layer representing a different sensing modality and agents heterogeneous in protocol. Protocols are designed to distinguish signals from different layers: an agent becomes active if a sufficient number of its neighbors in each of any a of the m layers is active. We focus on Protocol OR, when $a = 1$, and Protocol AND, when $a = m$, which model agents that are most and least readily activated, respectively. We develop theory and algorithms to compute the size of the spread at steady state for any set of initially active agents and to analyze the role of distinguished sensing modalities, network structure, and heterogeneity. We show how heterogeneity manages the tension in spreading dynamics between sensitivity to inputs and robustness to disturbances.

Index Terms—Cascade dynamics, heterogeneity, multi-agent systems, multi-layer networks, social networks, contagion

I. INTRODUCTION

THE spread of an activity or innovation across a population of agents that sense or communicate over a network has critical consequences for a wide range of systems from biology to engineering. The adoption of a strategy, such as wearing a face mask during a pandemic, can spread across a social network even when there are only a few early adopters. The observation and response to a threat by one or more vigilant animals can spread through a social animal group. A robot that detects a change in the environment and takes action can spread its behavior across a networked robot team.

To predict and control spread, we present and analyze a new model that captures the realities of *multiple inter-agent sensing modalities* and *heterogeneity in responsiveness of agents to others*. We develop and prove the validity of new algorithms that provide the means to systematically determine the spreading influence of a set of agents as a function of multi-layer network structure and agent heterogeneity.

The linear threshold model (LTM), from Granovetter [1] and Schelling [2], describes the spread of an activity as discrete-time, discrete-valued state dynamics where an agent adopts or rejects an activity by comparing the fraction of its neighbors that have adopted the activity to its individual threshold.

Kempe et al. [3] used the LTM with random thresholds to investigate spread of an activity over a population on a single-layer network. Lim et al. [4] introduced and analyzed the notion of cascade and contagion centralities in the model of [3]. The LTM on single-layer networks has also been studied in [5]–[9] and generalized to continuous-time, real-valued dynamics in [10].

The single-layer network in the LTM represents a single sensing modality or a projection of multiple sensing modalities. Yet, in real-world systems, agents may distinguish the different sensing modalities, rather than project them, in ways that impact spread. For example, someone deciding whether or not to wear a mask may consider as separate signals what they see others doing in the neighborhood and what they hear over social media that others are doing. And, how they act on the signals may differ from person to person. A more readily activated person starts wearing a mask when they observe enough of the first *or* second group wearing a mask. A less readily activated person starts wearing a mask only when they observe enough of the first *and* second groups wearing a mask.

The LTM has been used to study social and economic phenomena, ranging from spread of rioting to adoption of a convention (see [11] and references therein). Especially when time and resource are limited such that optimal decision-making is not feasible, agents tend to exhibit a threshold phenomenon: they adopt an alternative once sufficient neighbors have adopted the alternative [11]. Rosenthal *et al.* [12] have used the LTM to explain fast, coordinated collective response exhibited in group-living organisms. They also discuss how multiple sensing networks such as visual contact, metric distance, angular position, etc., aid the collective behavioral response. We are thus motivated to use the LTM to enable the rapid, adaptive behavior of social system in the design of technological systems, such as swarms of very small robots, where limits on computational resources prevent sophisticated fusion of information from multiple sensing networks.

We are further motivated to use multiplex (multi-layer) networks to model spread in a population of heterogeneous agents that interact through, and distinguish, multiple sensing modalities. Multiplex networks have been used to study consensus dynamics [13]–[17]. Yağan and Gligor [18] studied a multiplex LTM where an agent gets activated if a weighted average of active neighbors across layers is greater than its threshold. Other models of spread, such as epidemic models and game-theoretic models, in the case of multiple sensing modalities are reviewed in [19], but most restrict to homogeneous agents.

In a preliminary version of this work [20], we introduced the

LTM on multiplex networks with homogeneous agents, where the graph for each layer is associated with a different sensing modality, and Protocols OR and AND distinguish signals from different layers to model more and less readily activated agents, respectively. We analyzed the duplex (two-layer) LTM with agents that are homogeneous in protocol and showed how to compute cascade centrality, an agent's influence on the steady-state size of the cascade. Yang et al. [21] studied the influence minimization problem for the homogeneous model of [20]. In the present paper, the model and analysis from [20] are generalized to any finite number of network layers, and to heterogeneous agents. These generalizations have important implications in design, notably the opportunity to leverage individual differences in responsiveness and multiple sensing modalities to balance group-level flexibility and robustness. Additionally, we present a novel Bayesian network perspective on computation of influence spread, which reduces computational complexity.

Our contributions in the present paper are multifold. First, we define the heterogeneous multiplex LTM to analyze spreading dynamics on an arbitrary number of network layers with agents that employ protocols heterogeneously. Second, we define the heterogeneous multiplex live-edge model (LEM), which generalizes [3], and we introduce the live-edge tree to define reachability on this LEM. We prove a key result on equivalence of probabilities for the LTM and LEM.

Third, we derive Algorithms 1 and 2 to compute influence spread for the heterogeneous multiplex LTM. Algorithm 1 is provably correct and useful for small networks. Algorithm 2 maps the influence spread calculation to an inference problem in a Bayesian network and is efficient for large networks. We prove that calculating influence spread is #P-complete.

Fourth, we derive *analytical* expressions for influence spread in classes of multiplex networks. We show *how* ORs enhance and ANDs diminish spreading relative to the projected network. Fifth, we investigate heterogeneity in spreading and show how it can be used to manage the tradeoff between sensitivity to a reliable input and robustness to an unreliable signal.

The paper is organized as follows. Section II describes multiplex networks. Section III reviews the monoplex LTM and LEM. Sections IV and V introduce the heterogeneous multiplex LTM and LEM, respectively. We prove their equivalence in Section VI. Section VII defines multiplex influence spread and presents an algorithm to calculate it. Section VIII presents a Bayesian network approach to calculate multiplex influence spread. Heterogeneity is studied in Section IX. We conclude in Section X.

II. MONOPLEX AND MULTIPLEX NETWORKS

A *multiplex network* $\mathcal{G}$ is a family of $m \in \mathbb{N}$ directed weighted graphs $G_1, ..., G_m$. Each graph $G_k = (V, E^k)$, $k = 1, \ldots, m$, is a *layer* of the multiplex network. The agent set $V = \{1, 2, 3, ..., n\}$ is the same in all layers. The edge set of layer k is $E^k \subseteq V \times V$ and can be different in different layers. Each edge $e_{i,j}^k \in E^k$, pointing from i to j in layer k, is assigned a weight $w_{i,j}^k \in \mathbb{R}^+$. Here we adopt the "sensing" convention for edges: edge $e_{i,j}^k$ exists if agent i can sense agent

j in layer k. If edge $e_{i,j}^k$ exists, agent j is an *out-neighbor* of agent i in layer k. We denote the set of out-neighbors of i in layer k as N_i^k. We say that the weight of agent i's out-neighbor j in layer k is the weight $w_{i,j}^k$. We assume the weights of all out-neighbors for every agent sum up to 1, i.e., $\sum_{j \in N_i^k} w_{i,j}^k = 1$ for every agent i. A *monoplex network* is a multiplex network with $m = 1$, i.e., with only a single layer.

We call a graph *equal-neighbor* if every edge $e_{i,j}^k$ is assigned a weight $w_{i,j}^k = 1/d_i^k$, where d_i^k is the *out-degree* of node i in layer k and equal to the number of out-neighbors of node i in layer k. A *projection network* of $\mathcal{G}$ is the graph $\text{proj}(\mathcal{G}) = (V, E)$ where $E = \cup_{k=1}^m E^k$, i.e., the projection of all layers onto a single graph. The weight of an edge $e_{i,j}$ in E is defined to be $w_{i,j} = \psi(\mathbf{w}_{i,j}) / \sum_{j \in \cup_k N_i^k} \psi(\mathbf{w}_{i,j})$, where $\mathbf{w}_{i,j} = \{w_{i,j}^k | \forall k, w_{i,j}^k > 0\}$. Examples of function $\psi(\cdot)$ include summation, maximization, etc. In our examples, we choose $\psi(\cdot)$ as $\min(\cdot)$.

III. THE MONOPLEX LTM AND LEM

We consider the linear threshold model (LTM) and analysis presented in [3] for monoplex networks. As in [3], we assume the LTM is progressive, i.e., once an agent is active it stays active, and thresholds are random. The LTM is described by a discrete-time dynamical system in which the state $x_i(t) \in \{0, 1\}$ of each agent i at iteration t is inactive with $x_i(t) = 0$ or active with $x_i(t) = 1$. The LTM protocol determines how the active state spreads through the network. Our focus is on which agents will be active at steady state as a function of which agents are active initially. We define $\bar{x}_i = \lim_{t \to \infty} x_i(t)$.

In Section III-A and Section III-B, we review the LTM and live-edge model (LEM) for monoplex networks. In Section III-C, we recall the equivalence between LTM and LEM.

A. Monoplex LTM

Let S_t be the set of agents that are active by the end of iteration t. Once active, an agent remains active so that $S_{t-1} \subseteq S_t$. At $t = 0$, all agents are inactive except the initially active set S_0. Every agent in S_0 is called a *seed*. The LTM protocol determines when inactive agents at iteration $t - 1$ become active at iteration t. A steady state is reached when $S_{t-1} = S_t$.

The LTM protocol on a monoplex network is defined as follows (e.g., [3]). Each agent $i = 1, \ldots, n$ chooses a threshold μ_i randomly and independently from a uniform distribution $U(0, 1)$. An inactive agent i at iteration $t - 1$ becomes active at iteration t if the sum of weights of its active out-neighbors at $t - 1$ exceeds μ_i, that is, if $\mu_i < \sum_{j \in N_i \cap S_{t-1}} w_{i,j}$. For n agents, steady state is reached by $t \leq n$.

B. Monoplex LEM and Reachability

The LEM for a monoplex network is defined as follows [3]. Let S_0 be the set of seeds. Each unseeded agent randomly selects one of its outgoing edges with probability given by the edge weight. The selected edge is labeled as "live", while the unselected edges are labeled as "blocked". The seeds block all of their outgoing edges. Every directed edge will thus be either live or blocked. The choice of edges that are live is called a *selection of live edges*.

Let L be the set of all possible selections of live edges. The probability q_l of selection $l \in L$ is the product of the weights of the live edges in selection l. Because the selection of live edges can be done at the same time for every node, the LEM can be viewed as a static model. The LEM can alternatively be viewed as an iterative process in the case the live edges are selected sequentially.

A *live-edge path* [3] is a directed path that consists only of live edges. Let $\mathcal{L}_{ij}$ be the set of all possible distinct live-edge paths from agent $i \notin S_0$ to $j \in S_0$. The probability r_α of live-edge path $\alpha \in \mathcal{L}_{ij}$ is the product of the edge weights along the path. We say $i \notin S_0$ is *reachable from $j \in S_0$ by live-edge path α with probability r_α*, and $i \notin S_0$ is *reachable from $j \in S_0$ with probability r_{ij}*, where $r_{ij} = \sum_{\alpha \in \mathcal{L}_{ij}} r_\alpha$.

Alternatively, we can compute r_{ij} in terms of selections of live-edges. Let $L_{ij} \subseteq L$ be the set of all selections of live edges that contain a live-edge path from $i \notin S_0$ to $j \in S_0$. Then, $r_{ij} = \sum_{l \in L_{ij}} q_l$. Likewise, let $L_{iS_0} \subseteq L$ be the set of all selections of live edges that contain a live-edge path from $i \notin S_0$ to at least one node $j \in S_0$. Then, $i \notin S_0$ is *reachable from S_0 with probability r_{iS_0}*, where $r_{iS_0} = \sum_{l \in L_{iS_0}} q_l$.

C. Equivalence for Monoplex Networks

The monoplex LTM and LEM were proved to be equivalent in the following sense.

Lemma 1 ([3])**.** For a given monoplex network G with seed set S_0, the probabilities of the following two events for arbitrary agent $i \notin S_0$ are the same:

1) i is active at steady state for the LTM with random thresholds and initial active set S_0;
2) i is reachable from set S_0 under the random selection of live edges in the LEM.

IV. THE HETEROGENEOUS MULTIPLEX LTM

We introduce the LTM on a multiplex network with m layers by defining a family of protocols as follows. Each agent i chooses a threshold μ_i^k in each layer k for $i = 1, \ldots, n$ and $k = 1, \ldots, m$. Each μ_i^k is randomly and independently drawn from the uniform distribution $U(0,1)$. In general, each agent has different neighbors in different layers. If the sum of weights of active out-neighbors of agent i in layer k at $t - 1$ exceeds μ_i^k, that is, $\mu_i^k < \sum_{j \in N_i^k \cap S_{t-1}} w_{i,j}^k$, we say agent i receives a positive input $y_i^k(t) = 1$ from layer k at t. Otherwise, agent i receives a neutral input $y_i^k(t) = 0$.

The protocols that determine whether or not an inactive agent at $t - 1$ becomes active at t account for the possibility that the inputs it receives at t from the different layers may be conflicting. Let the average input agent i receives at t be $y_i(t) = \sum_{k=1}^m y_i^k(t)/m$.

Definition 1 (Multiplex LTM Protocol)**.** Given multiplex network $\mathcal{G}$ with seed set S_0, the *multiplex LTM protocol for agent i* is parametrized by $\delta_i \in [1/m, 1]$ as follows:

$$x_i(0) = 1, \quad \forall i \in S_0 \tag{1}$$

$$x_i(0) = 0, \quad \forall i \notin S_0 \tag{2}$$

$$x_i(t) = \begin{cases} 1, & \text{if } y_i(t) \geq \delta_i \text{ or } x_i(t-1) = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

We identify two protocols for the limiting values of δ_i:
Protocol OR: $\delta_i = 1/m$. Inactive agent i at iteration $t - 1$ becomes active at iteration t if it receives a positive input from *any* layer at t;
Protocol AND: $\delta_i = 1$. Inactive agent i at iteration $t - 1$ becomes active at iteration t if it receives positive inputs from *all* layers at t. $\quad\square$

Let $a_i \in \{1, \ldots, m\}$ refer to the number of layers that agent i needs to register as active in order to switch its own state from inactive to active. If δ_i is chosen such that $(a_i - 1)/m < \delta_i \leq a_i/m$, then the multiplex LTM protocol specifies exactly this condition on the state of agent i, i.e., that inactive agent i at iteration $t - 1$ becomes active at iteration t if it receives a positive input from any a_i of the m layers. Asymmetric sensitivity to layers can be modelled with y_i a convex combination of y_i^k.

In this paper, we examine the two limiting cases: Protocol OR, where $a_i = 1$, and Protocol AND, where $a_i = m$. Analysis in these cases is sufficient for understanding heterogeneity and spreading dynamics on multi-layer networks.

Remark 1. For clarity of exposition, we restrict the subsequent discussion to OR and AND protocols; however, the results and ideas extend, with appropriate modification of definitions and probability computation rules, to more general protocols discussed in Definition 1 as well as to other logical rules such as exclusive OR, etc.

Remark 2. Protocol OR models agents that are readily activated: there only needs to be sufficient activity among neighbors in one layer at $t - 1$ in order for agents to become active at t. Protocol AND models agents that are conservatively activated: there needs to be sufficient activity among neighbors in every layer at $t-1$ in order for agents to become active at t. Thus, agents with Protocol OR enhance spreading and agents with Protocol AND diminish spreading.

We study heterogeneous networks in which some agents use Protocol OR while the others use Protocol AND.

Definition 2 (Sequence of Protocols)**.** Let $u_i \in \{\text{OR}, \text{AND}\}$ be the protocol used by agent i. We define the *sequence of protocols* $\mathcal{U} = (u_1, u_2, ..., u_n)$ to be the protocols used by the n agents ordered from agent 1 to agent n.

Lemma 2. For a multiplex network $\mathcal{G}$ with n agents, the multiplex LTM converges in at most n iterations.

Proof. Assume the multiplex LTM converges in more than n iterations. Then at least one agent switches from inactive to active in each of the first n iterations and these agents are distinct. There is at least one agent in the initial active set that
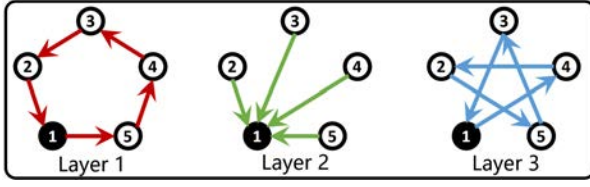
Fig. 1. An example of a three-layer multiplex network with five agents. Agent 1 is the seed, which is denoted by the black circle.
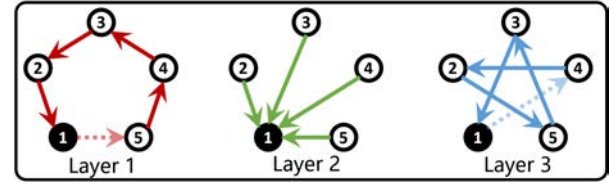

Fig. 2. The unique multiplex selection of live edges for the network in Fig. 1.

is not one of those n agents. This implies at least $n+1$ agents in the network, which is a contradiction. $\qquad\square$

V. The Heterogeneous Multiplex LEM

Our approach to analyzing the multiplex LTM generalizes the approach in [3], which uses the live-edge model (LEM) for monoplex networks to analyze the monoplex LTM. In Section V-A, we generalize the LEM to multiplex networks and introduce the notion of $\mathcal{U}$-reachability on the multiplex LEM. Unlike in our earlier work [20], the reachability we propose here allows for heterogeneous protocols among agents.

A. Multiplex LEM and Reachability

We introduce the LEM on a multiplex network as follows.

Definition 3 (Multiplex LEM). Consider a multiplex network $\mathcal{G}$ with seed set S_0. In each layer k, each unseeded agent i randomly selects one of its outgoing edges e^k_{i,j_k} with probability w^k_{i,j_k}. The selected edges are labeled as "live", while the unselected edges are labeled as "blocked". The seeds block all of their outgoing edges in every layer. The choice of edges that are live is a *multiplex selection of live edges*. Let L be the set of all possible multiplex selections of live edges. The probability q_l of selection $l \in L$ is the product of the weights of all live edges in selection l.

The challenge in generalizing the LEM to multiplex networks is in properly defining reachability. Here we introduce the live-edge tree, which we use to define reachability.

Definition 4 (Live-edge Tree). [1] Given a set of seeds S_0 and a multiplex selection of live edges $l \in L$, the *live-edge tree* T^l_i associated with agent $i \notin S_0$ is constructed as follows with agent i as the root node. Let e^k_{i,j_k} be the live edge of agent i in layer k, $k = 1, \ldots, m$. Then the children of the root node are agents $j_1, j_2, ..., j_m$, and the root node is connected to each child with the live edge in the corresponding layer. The tree is constructed recursively in this way for each child that itself has at least one child. Any agent in the network may appear multiple times as a node in the tree.

Fig. 1 provides an example of a three-layer multiplex network with five agents. The network has only one possible multiplex selection of live edges, given in Fig. 2. Fig. 3 shows the corresponding live-edge tree associated with agent 5.
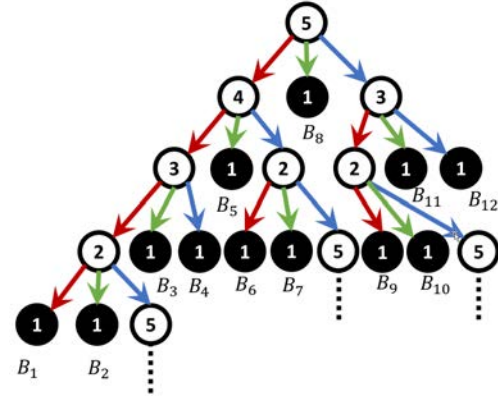

Fig. 3. The live-edge tree associated with agent 5 for the example three-layer multiplex network of Fig. 1 and the unique selection of live edges of Fig. 2. $\mathcal{B}_5 = \{B_1, B_2, ..., B_{12}\}$ is the set of distinct branches that end with a seed.

We next define reachability from S_0 of an unseeded agent in a multiplex network under a sequence of protocols $\mathcal{U}$, and then illustrate for the example in Fig. 3.

Definition 5 ($\mathcal{U}$-Reachability). Consider a multiplex network $\mathcal{G}$ with a seed set S_0 and a multiplex selection of live edges $l \in L$. Let T^l_i be the live-edge tree associated with agent $i \notin S_0$. Suppose there are b distinct branches in T^l_i indexed by $\beta = 1, \ldots, b$ and of the form: $B_\beta = (i, e^{k_0}_{i,i_1}, i_1, e^{k_1}_{i_1,i_2}, i_2, ..., i_s)$, where $i_j \in V$, $j = 1, \ldots, s$, $i_s \in S_0$, and each agent in V appears at most once in B_β. We call each B_β a *distinct branch that ends in a seed*. Denote the set of these branches as $\mathcal{B}^l_i = \{B_1, B_2, ..., B_b\}$. For any subset $\hat{\mathcal{B}} \subseteq \mathcal{B}^l_i$, let the set of agents in $\hat{\mathcal{B}}$ be $\hat{V}$ and the set of edges in $\hat{\mathcal{B}}$ be $\hat{E}$.

Given a sequence of protocols $\mathcal{U}$, we say that branch subset $\hat{\mathcal{B}} \subseteq \mathcal{B}^l_i$ is $\mathcal{U}$-*feasible for* i if $\hat{\mathcal{B}} \neq \varnothing$ and for every $\hat{i} \in \hat{V} \setminus S_0$ for which $u_{\hat{i}} = \text{AND}$, all of $\hat{i}$'s live edges belong to $\hat{E}$. Then, i is $\mathcal{U}$-*reachable from* S_0 *by the selection of live edges* l *with probability* q_l if there exists at least one $\hat{\mathcal{B}} \subseteq \mathcal{B}^l_i$ that is $\mathcal{U}$-feasible for i. Let $L^{\mathcal{U}}_{iS_0} \subseteq L$ be the set of all selections of live edges by which i is $\mathcal{U}$-reachable from S_0. Then, i is $\mathcal{U}$-*reachable from* S_0 *in* $\mathcal{G}$ *with probability* $r^{\mathcal{G},\mathcal{U}}_{iS_0}$, where $r^{\mathcal{G},\mathcal{U}}_{iS_0} = \sum_{l \in L^{\mathcal{U}}_{iS_0}} q_l$. $\qquad\square$

Remark 3. The condition for $\mathcal{U}$-feasibility for i of a branch subset $\hat{\mathcal{B}} \subseteq \mathcal{B}^l_i$ does not make explicit a condition on any $\hat{i} \in \hat{V} \setminus S_0$ for which $u_{\hat{i}} = \text{OR}$. This follows since for any such agent $\hat{i}$, the condition is that at least one of its live edges must be in $\hat{E}$ and this is always true by definition.

To illustrate $\mathcal{U}$-reachability, consider the live-edge tree associated with agent 5 in Fig. 3 for the unique selection of live edges in Fig. 2 for the multiplex network of Fig. 1 with seed

---

[1]To highlight key differences between multiplex and monoplex networks, we assume each $i \notin S_0$ has at least one neighbor in each layer. If not, with a slight modification of Defs. 4 - 5, the theory and computation are still valid.

set $S_0 = \{1\}$. Because the selection of live edges in Fig. 2 is unique, it is chosen with probability $q = 1$. Therefore, agent $i \notin S_0$ is $\mathcal{U}$-reachable from S_0 with probability 1 if there exists at least one $\hat{\mathcal{B}} \subseteq \mathcal{B}_i$ that is $\mathcal{U}$-feasible for i. For agent 5, there are 12 distinct branches that end in a seed, as shown in Fig. 3; thus, $\mathcal{B}_5 = \{B_1, B_2, ..., B_{12}\}$. For example, $B_8 = (5, e_{5,1}^2, 1)$.

We compute $\mathcal{U}$-reachability from S_0 for agent 5 for each of the following three sequences of protocols used by the five agents in the three-layer multiplex network:

$$\mathcal{U}_1 = (\text{OR}, \text{AND}, \text{AND}, \text{AND}, \text{OR}) \qquad (4)$$
$$\mathcal{U}_2 = (\text{OR}, \text{OR}, \text{AND}, \text{AND}, \text{AND}) \qquad (5)$$
$$\mathcal{U}_3 = (\text{OR}, \text{AND}, \text{AND}, \text{AND}, \text{AND}). \qquad (6)$$

1) Let $\mathcal{U} = \mathcal{U}_1$. Consider $\hat{\mathcal{B}} = \{B_8\} \subset \mathcal{B}_5$. Then, $\hat{V} = \{1, 5\}$ and $\hat{E} = \{e_{5,1}^2\}$. Since 5 is the only unseeded node in $\hat{V}$ and $u_5 = \text{OR}$, $\hat{\mathcal{B}}$ is $\mathcal{U}$-feasible for 5. Thus, agent 5 is $\mathcal{U}$-reachable from S_0 with probability 1.

2) Let $\mathcal{U} = \mathcal{U}_2$. Consider $\hat{\mathcal{B}} = \mathcal{B}_5$. Then, $\hat{V} = \{1, 2, 3, 4, 5\}$. The unseeded nodes $j \in \hat{V}$ for which $u_j = \text{AND}$ are $j = 3, 4, 5$. From Fig. 3, observe that all the live edges of nodes 3, 4, and 5, belong to $\hat{E}$, the edge set of $\hat{\mathcal{B}} = \mathcal{B}_5$. Thus, $\hat{\mathcal{B}}$ is $\mathcal{U}$-feasible for 5, and agent 5 is $\mathcal{U}$-reachable from S_0 with probability 1.

3) Let $\mathcal{U} = \mathcal{U}_3$. In this case there is no $\mathcal{U}$-feasible subset $\hat{\mathcal{B}} \subseteq \mathcal{B}_5$, since $u_2 = \text{AND}$ and agent 2 has a live edge $e_{2,5}^3$, which is not in the edge set of any branch in $\mathcal{B}_5$. Thus, agent 5 is not $\mathcal{U}$-reachable from S_0.

In the next section we prove the equivalence of the multiplex LTM and multiplex LEM. This implies that under $\mathcal{U}_1$ or $\mathcal{U}_2$, agent 5 will become active at steady state with probability 1 and under $\mathcal{U}_3$, agent 5 will remain inactive at steady state.

VI. EQUIVALENCE OF MULTIPLEX LTM AND LEM

We generalize the equivalence of LTM and LEM to multiplex networks in this section. Computing probabilities for the multiplex LTM is challenging because it requires solving over temporal iterations. However, leveraging the equivalence, the probability distributions can be computed without temporal iteration using the multiplex LEM as a static model, as was done in the monoplex case (see Section III-C and [3]).

First, we prove the following lemma that infers an agent's $\mathcal{U}$-reachability from the $\mathcal{U}$-reachability of its children in the live-edge tree. We then leverage this lemma to prove the equivalence in Theorem 1.

Lemma 3. Given a multiplex network $\mathcal{G}$ with seed set S_0, multiplex selection of live edges $l \in L$ and sequence of protocols $\mathcal{U}$, consider agent $i \notin S_0$ and its associated live-edge tree T_i^l. Assume i's live edge in layer k connects to agent i_1^k, $k = 1, \ldots, m$. Then the $\mathcal{U}$-reachability of i from S_0 by selection l can be inferred from the reachability of its children i_1^k and its protocol u_i as follows:

1) Let $u_i = \text{OR}$. Then, i is $\mathcal{U}$-reachable from S_0 by selection of live edges l if and only if at least one child i_1^k is $\mathcal{U}$-reachable from S_0 by selection of live edges l.

2) Let $u_i = \text{AND}$. Then, i is $\mathcal{U}$-reachable from S_0 by selection of live edges l if and only if every child i_1^k is $\mathcal{U}$-reachable from S_0 by selection of live edges l.

Proof. Let $T_{i_1^k}^l$ be the corresponding live-edge tree associated with agent i's child i_1^k for $k = 1, \ldots, m$.

1) Let $u_i = \text{OR}$ and suppose i_1^k is $\mathcal{U}$-reachable from S_0 by selection l for some k. By Definition 5, the set $\mathcal{B}_{i_1^k}^l$ of distinct branches that end with a seed in $T_{i_1^k}^l$ is nonempty and there exists a subset $\hat{\mathcal{B}}_k \subseteq \mathcal{B}_{i_1^k}^l$ that is $\mathcal{U}$-feasible for i_1^k such that for every $\hat{i} \in \hat{V}_k \setminus S_0$ for which $u_{\hat{i}} = \text{AND}$, all of $\hat{i}$'s live edges belong to $\hat{E}_k$, where $\hat{V}_k$ and $\hat{E}_k$ are the sets of agents and edges in $\hat{\mathcal{B}}_k$, respectively.

For every branch $B_r^k = (i_1^k, e_{i_1^k, i_2}^{k'}, i_2, ..., i_s) \in \hat{\mathcal{B}}_k$, there exists a branch $B_r^{k0} = (i, e_{i, i_1^k}^k, i_1^k, e_{i_1^k, i_2}^{k'}, i_2, ..., i_s)$ in T_i^l. Let $\hat{\mathcal{B}}_{k0} \subseteq \mathcal{B}_i^l$ be the set of all branches in T_i^l that correspond to branches in $\hat{\mathcal{B}}_k$. Then, $\hat{V}_{k0} = \hat{V}_k \cup \{i\}$ is the set of agents in $\hat{\mathcal{B}}_{k0}$ and $\hat{E}_{k0} = \hat{E}_k \cup \{e_{i, i_1^k}^k\}$ is the set of edges in $\hat{\mathcal{B}}_{k0}$. Thus, since $u_i = \text{OR}$ and $\hat{\mathcal{B}}_k$ is $\mathcal{U}$-feasible for i_1^k, by Definition 5, $\hat{\mathcal{B}}_{k0}$ must be $\mathcal{U}$-feasible for i. Agent i is therefore $\mathcal{U}$-reachable from S_0 by selection l. This proves the "if" part of the statement.

If no child i_1^k is $\mathcal{U}$-reachable from S_0 by selection l, then there exists no nonempty $\hat{\mathcal{B}}_k$ in $T_{i_k}^l$ that is $\mathcal{U}$-feasible for i_1^k. This implies there is no nonempty $\hat{\mathcal{B}}_{k0}$ in T_i^l that is $\mathcal{U}$-feasible for i. Therefore, agent i cannot be $\mathcal{U}$-reachable from S_0 by selection l. This proves the "only if" part of the statement.

2) Let $u_i = \text{AND}$ and suppose i_1^k is $\mathcal{U}$-reachable from S_0 by selection l, for every $k = 1, \ldots, m$. By Definition 5, for every k, there exists a subset $\hat{\mathcal{B}}_k \subseteq \mathcal{B}_{i_1^k}^l$ in $T_{i_1^k}^l$ that is $\mathcal{U}$-feasible for i_1^k. For every k let $\hat{\mathcal{B}}_{k0} \subseteq \mathcal{B}_i^l$ be the set of all branches in T_i^l that correspond to branches in $\hat{\mathcal{B}}_k$ as defined in the proof of 1). Let $\hat{\mathcal{B}}_0 = \cup_{k=1}^m \hat{\mathcal{B}}_{k0} \subseteq \mathcal{B}_i^l$ with agent set $\hat{V}_0$ and edge set $\hat{E}_0$. By construction, $i \in \hat{V}_0$ and all of agent i's live edges belong to $\hat{E}_0$. It follows that $\hat{\mathcal{B}}_0$ is $\mathcal{U}$-feasible for i and thus agent i is $\mathcal{U}$-reachable from S_0 by selection l. This proves the "if" part of the statement.

If there is one child i_1^k that is not $\mathcal{U}$-reachable from S_0 by selection l, then there exists no set $\hat{\mathcal{B}}_k$ in $T_{i_k}^l$ that is $\mathcal{U}$-feasible for i_1^k. Suppose there is a set $\hat{\mathcal{B}}_0 \subseteq \mathcal{B}_i^l$ in T_i^l that is $\mathcal{U}$-feasible for i. Since $u_i = \text{AND}$, by Definition 5, it follows that edge $e_{i, i_1^k}^k \in \hat{E}_0$. For every branch in $\hat{\mathcal{B}}_0$ that starts with i and edge $e_{i, i_1^k}^k$, i.e., $B_r^{k0} = (i, e_{i, i_1^k}^k, i_1^k, e_{i_1^k, i_2}^{k'}, i_2, ..., i_s)$, we denote the set of all corresponding branches $B_r^k = (i_1^k, e_{i_1, i_2}^{k'}, i_2, ..., i_s)$ as $\hat{\mathcal{B}}_k \subseteq \mathcal{B}_{i_1^k}^l$. It follows that $\hat{\mathcal{B}}_k$ is $\mathcal{U}$-feasible for i_1^k in $T_{i_1^k}^l$ and i_1^k is reachable from S_0. This is a contradiction. Thus, there is no set $\hat{\mathcal{B}}_0$ that is $\mathcal{U}$-feasible for i and agent i cannot be $\mathcal{U}$-reachable from S_0 by selection l. This proves the "only if" part of the statement. $\square$

We state the equivalence of multiplex LTM and multiplex LEM in the following theorem.

Theorem 1. *For a multiplex network $\mathcal{G}$ with a seed set S_0, a multiplex selection of live edges $l \in L$ and a sequence of protocols $\mathcal{U}$, the probabilities of the following two events regarding an arbitrary agent $i \notin S_0$ are the same:*

1) *i is active at steady state for the multiplex LTM under $\mathcal{U}$ with random thresholds and initial active set S_0;*

2) *i is $\mathcal{U}$-reachable from the set S_0 under random selection of live edges in the multiplex LEM.*

Proof. We prove by mathematical induction.

i) Probability in the first statement (multiplex LTM)

We first define some LTM-related events and probabilities. Let X_i^k and Z_i^k, respectively, be the events that the sum of weights of active out-neighbors of i in layer k does and does not exceed μ_i^k at t: $X_i^k = \{\mu_i^k < \sum_{j \in N_i^k \cap S_t} w_{i,j}^k\}$ and $Z_i^k = \{\mu_i^k \geq \sum_{j \in N_i^k \cap S_t} w_{i,j}^k\}$. Let Y_i^k be the event that the sum of weights of active out-neighbors of i in layer k does not exceed μ_i^k at $t-1$, i.e., $Y_i^k = \{\mu_i^k \geq \sum_{j \in N_i^k \cap S_{t-1}} w_{i,j}^k\}$. For agent i that is inactive at t, we define the probability that i becomes active at $t+1$ as $\mathbb{P}_{i(\text{OR})}^{t+1}$ $\left(\mathbb{P}_{i(\text{AND})}^{t+1}\right)$ if i uses Protocol OR (AND).

If i uses Protocol OR, i being inactive at t means none of i's thresholds is exceeded at $t-1$ and i being active at $t+1$ means i's thresholds in at least one layer is exceeded at t. In this case, the probability that μ_i^k is exceeded at t is $\mathbb{P}_i^k = \mathbb{P}(X_i^k | Y_i^1, Y_i^2, ..., Y_i^m) = \mathbb{P}(X_i^k | Y_i^k)$. The last equality holds because random variables $\mu_i^1, \mu_i^2, ..., \mu_i^m$ are independent. Then $\mathbb{P}_{i(\text{OR})}^{t+1}$ is the complement of the probability that its threshold in none of the layers are exceeded, i.e., $\mathbb{P}_{i(\text{OR})}^{t+1} = 1 - \prod_{k=1}^m (1 - \mathbb{P}_i^k)$.

If i uses Protocol AND, i being inactive at t means at least one of its thresholds is not exceeded at $t-1$ and being active at $t+1$ means all of i's thresholds are exceeded at t. In this case, there are $2^m - 1$ possible events, with probabilities denoted as $\mathbb{P}_{m+1}, ..., \mathbb{P}_{2^m+m-1}$. We have that $\mathbb{P}_i^{m+1} = \mathbb{P}(X_i^1, X_i^2, ..., X_i^m | Y_i^1, Y_i^2, ..., Y_i^m)$. The other probabilities have a similar form but with one or more, but not all, of the Y_i^k replaced by Z_i^k. Since Y_i^k and Z_i^k are mutually exclusive for all k, we have $\mathbb{P}_{i(\text{AND})}^{t+1} = \sum_{l=m+1}^{2^m+m-1} \mathbb{P}_i^l$. Since the thresholds are independent of one another, $\mathbb{P}_i^l$ can be separated into the product of m terms, each of which involves events associated to one layer only: $\mathbb{P}_i^l = \prod_{k=1}^m \mathbb{P}_i^{0k}$, where $\mathbb{P}_i^{0k}$ is either $\mathbb{P}(X_i^k, Y_i^k)/\mathbb{P}(Y_i^k) = \mathbb{P}(X_i^k | Y_i^k)$ or $\mathbb{P}(X_i^k, Z_i^k)/\mathbb{P}(Z_i^k) = 1$.

ii) Probability in the second statement (multiplex LEM)

While the multiplex LEM and $\mathcal{U}$-reachability is defined in a static way, it also describes $\mathcal{U}$-reachability of agents as an iterative process over time [3] as follows. First, using Lemma 3, determine the $\mathcal{U}$-reachability of the agents with at least one edge coming from initial set $S_0' = S_0$. If an agent is determined to be $\mathcal{U}$-reachable from S_0', add it to S_0' to get a new reachable set S_1'. In the next iteration, follow the same procedure and get a sequence of reachable sets $S_0', S_1', S_2', ...$. The process ends at iteration t if $S_t' = S_{t-1}'$, where S_t' is the set of agents that are $\mathcal{U}$-reachable from S_0.

We next define some LEM-related events and probabilities. Let $X_i'^k$ and $Z_i'^k$, respectively, be the events that agent i's live edge in layer k does and does not connect to the reachable set S_t' at t. We define $Y_i'^k$ to be the event that agent i's live edge in layer k does not connect to the reachable set S_{t-1}' at $t-1$. Consider the LEM as an iterative process. If agent $i \notin S_t'$, then we define the probability that $i \in S_{t+1}'$ as $\mathbb{P}_{i(\text{OR})}'^{t+1} (\mathbb{P}_{i(\text{AND})}'^{t+1})$ if i uses Protocol OR (AND).

If i uses Protocol OR, by Lemma 3, $i \notin S_t'$ means all of i's live edges are not connected to S_{t-1}' and $i \in S_{t+1}'$ means at least one live edge of i is connected to S_t'. In this case, the probability that i's live edge in layer k connects to S_t' is

$\mathbb{P}_i'^k = \mathbb{P}(X_i'^k | Y_i'^1, Y_i'^2, ..., Y_i'^m) = \mathbb{P}(X_i'^k | Y_i'^k)$. Then $\mathbb{P}_{i(\text{OR})}'^{t+1}$ is the complement of the probability that none of i's live edges connects to S_t', i.e., $\mathbb{P}_{i(\text{OR})}'^{t+1} = 1 - \prod_{k=1}^m (1 - \mathbb{P}_i'^k)$.

If i uses Protocol AND, by Lemma 3, $i \notin S_t'$ means at least one of i's live edges are not connected to S_{t-1}' and $i \in S_{t+1}'$ means all of i's live edges are connected to S_t'. In this case, there are $2^m - 1$ possible events, with probabilities denoted as $\mathbb{P}_i'^{m+1}, ..., \mathbb{P}_i'^{2^m+m-1}$. We have that $\mathbb{P}_i'^{m+1} = \mathbb{P}(X_i'^1, X_i'^2, ..., X_i'^m | Y_i'^1, Y_i'^2, ..., Y_i'^m)$. The other probabilities have similar form but with one or more, but not all, of the $Y_i'^k$ replaced by $Z_i'^k$. Since $Y_i'^k$ and $Z_i'^k$ are mutually exclusive for all k, we have $\mathbb{P}_{i(\text{AND})}'^{t+1} = \sum_{l=m+1}^{2^m+m-1} \mathbb{P}_i'^l$. Since the live edges that each agent uses are independent of one another $\mathbb{P}_i'^l = \prod_{k=1}^m \mathbb{P}_i'^{0k}$, where $\mathbb{P}_i'^{0k}$ is either $\mathbb{P}(X_i'^k, Y_i'^k)/\mathbb{P}(Y_i'^k) = \mathbb{P}(X_i'^k | Y_i'^k)$ or $\mathbb{P}(X_i'^k, Z_i'^k)/\mathbb{P}(Z_i'^k) = 1$.

iii) Equivalence

Similar to [3], we have $\mathbb{P}(X_i^k | Y_i^k) = \frac{\sum_{j \in N_i^k \cap S_t \setminus S_{t-1}} w_{i,j}^k}{1 - \sum_{j \in N_i^k \cap S_{t-1}} w_{i,j}^k}$ and $\mathbb{P}(X_i'^k | Y_i'^k) = \frac{\sum_{j \in N_i^k \cap S_t' \setminus S_{t-1}'} w_{i,j}^k}{1 - \sum_{j \in N_i^k \cap S_{t-1}'} w_{i,j}^k}$. We also observe $\mathbb{P}_{i(\text{OR})}^{t+1}$ and $\mathbb{P}_{i(\text{AND})}^{t+1}$ only depend on $\mathbb{P}(X_i^k | Y_i^k), \forall k$, and $\mathbb{P}_{i(\text{OR})}'^{t+1}$ and $\mathbb{P}_{i(\text{AND})}'^{t+1}$ only depends on $\mathbb{P}(X_i'^k | Y_i'^k), \forall k$. Thus, by induction over iterations, we have $\mathbb{P}_{i(\text{OR})}^{t+1} = \mathbb{P}_{i(\text{OR})}'^{t+1}, \forall t$ and $\mathbb{P}_{i(\text{AND})}^{t+1} = \mathbb{P}_{i(\text{AND})}'^{t+1}, \forall t$, which concludes the proof. $\square$

The proof of Theorem 1 follows the spirit of the monoplex case [3], but it does not make direct use of previous results.

Theorem 1 generalizes the equivalence of LTM and LEM to multiplex networks, and thus provides the means to calculate an agent's influence, in terms of spreading information through the network, without needing to simulate the multiplex LTM.

VII. MULTIPLEX INFLUENCE SPREAD

In this section we define multiplex influence spread and multiplex cascade centrality for the LTM. We derive and prove the validity of an algorithm to compute them using the equivalence of Theorem 1. We prove analytical expressions for influence spread in some special cases of multiplex networks; these expressions provide a systematic way to evaluate spread.

A. Monoplex Influence Spread and Cascade Centrality

The *monoplex influence spread of agents in S_0*, denoted $\sigma_{S_0}^G$, is defined as the expected number of active agents at steady state for the monoplex LTM given the network G and initial active set S_0 [3]. The *monoplex cascade centrality of agent j*, denoted $\mathcal{C}_j^G$, is the influence spread of agent j defined in [4] as $\mathcal{C}_j^G = \sigma_j^G$, the expected number of active agents at steady state for the monoplex LTM given G and $S_0 = \{j\}$.

B. Multiplex Influence Spread and Cascade Centrality

Influence spread and cascade centrality are naturally generalized to the multiplex setting as follows.

Definition 6 (Multiplex influence spread)**.** The *multiplex influence spread of agents in S_0*, denoted $\sigma_{S_0}^{\mathcal{G},\mathcal{U}}$, is defined as the expected number of active agents at steady state for the

multiplex LTM given the network $\mathcal{G}$, sequence of protocols $\mathcal{U}$, and initial active set S_0. Let $\mathbb{E}_{S_0}^{\mathcal{G},\mathcal{U}}$ and $\mathbb{P}_{S_0}^{\mathcal{G},\mathcal{U}}$ be expected value and probability, respectively, conditioned on $\mathcal{G},\mathcal{U},S_0$. Then

$$\sigma_{S_0}^{\mathcal{G},\mathcal{U}} = \mathbb{E}_{S_0}^{\mathcal{G},\mathcal{U}}\Big(\sum_{i=1}^{n}\bar{x}_i\Big) = \sum_{i=1}^{n}\mathbb{P}_{S_0}^{\mathcal{G},\mathcal{U}}(\bar{x}_i=1). \qquad (7)$$

Definition 7 (Multiplex cascade centrality). The *multiplex cascade centrality of agent j*, denoted $\mathcal{C}_j^{\mathcal{G},\mathcal{U}}$, is defined as

$$\mathcal{C}_j^{\mathcal{G},\mathcal{U}} = \sigma_j^{\mathcal{G},\mathcal{U}}. \qquad (8)$$

When $\mathcal{U} = (u, ..., u)$, we replace $\mathcal{U}$ with u in the superscript. For example, when $u = \mathrm{OR}$, we write $\mathbb{P}_{S_0}^{\mathcal{G},\mathrm{OR}}(\bar{x}_i=1)$ and $\mathcal{C}_j^{\mathcal{G},\mathrm{OR}}$. When $\mathcal{G}$ is understood we drop it from the superscript.

C. Computing Multiplex Influence Spread and Centrality

We can directly compute multiplex influence spread and multiplex cascade centrality by computing probabilities of $\mathcal{U}$-reachability for the LEM, which is much easier than computing probabilities of agents being active at steady state for the LTM. We summarize in a corollary to Theorem 1.

Corollary 1. Given multiplex network $\mathcal{G}$ and sequence of protocols $\mathcal{U}$, multiplex influence spread of agents in S_0 and multiplex cascade centrality of agent j can be determined as

$$\sigma_{S_0}^{\mathcal{G},\mathcal{U}} = \sum_{i=1}^{n} r_{iS_0}^{\mathcal{G},\mathcal{U}}, \quad \mathcal{C}_j^{\mathcal{G},\mathcal{U}} = \sum_{i=1}^{n} r_{ij}^{\mathcal{G},\mathcal{U}}. \qquad (9)$$

Proof. By Definition 5, $r_{iS_0}^{\mathcal{G},\mathcal{U}}$ is the probability that i is $\mathcal{U}$-reachable from S_0 in the multiplex LEM of Definition 3. By Theorem 1, $r_{iS_0}^{\mathcal{G},\mathcal{U}} = \mathbb{P}_{S_0}^{\mathcal{G},\mathcal{U}}(\bar{x}_i=1)$. Thus, by Definition 6, $\sum_{i=1}^{n} r_{iS_0}^{\mathcal{G},\mathcal{U}} = \sum_{i=1}^{n}\mathbb{P}_{S_0}^{\mathcal{G},\mathcal{U}}(\bar{x}_i=1) = \sigma_{S_0}^{\mathcal{G},\mathcal{U}}$. In case $S_0 = \{j\}$, by Definition 7, $\mathcal{C}_j^{\mathcal{G},\mathcal{U}} = \sigma_j^{\mathcal{G},\mathcal{U}} = \sum_{i=1}^{n} r_{ij}^{\mathcal{G},\mathcal{U}}$. $\qquad\square$

Algorithm 1 uses (9) to compute multiplex influence spread and can be specialized to compute cascade centrality.

Algorithm 1 (Compute multiplex influence spread $\sigma_{S_0}^{\mathcal{G},\mathcal{U}}$). Given multiplex network $\mathcal{G}$ and sequence of protocols $\mathcal{U}$:

1) Find the set L of all possible selections of live edges for multiplex network $\mathcal{G}$ and initially active set S_0. Calculate the probability q_l of each $l \in L$.
2) For each agent i find $L_{iS_0}^{\mathcal{U}} \subseteq L$, the set of all $l \in L$ such that i is $\mathcal{U}$-reachable from S_0 by selection l.
3) Calculate $r_{iS_0}^{\mathcal{G},\mathcal{U}} = \sum_{l \in L_{iS_0}^{\mathcal{U}}} q_l$.
4) Calculate $\sigma_{S_0}^{\mathcal{G},\mathcal{U}} = \sum_{i=1}^{n} r_{iS_0}^{\mathcal{G},\mathcal{U}}$.

We can compute the *computational complexity of Algorithm 1* as follows. Denote the geometric mean of the out-degree of all agents in a multiplex network as $\bar{d}$. Then $|L| = \bar{d}^{mn}$. Determining the reachability of agents has a time cost of $O(n)$. Thus, Algorithm 1 has a time cost of $O(n\bar{d}^{mn})$.

Computing $\sigma_{S_0}^{\mathcal{G},\mathcal{U}}$ using the multiplex LTM requires performing simulation of each combination of thresholds and also has a time cost of $O(n\bar{d}^{mn})$. Although Algorithm 1 does not improve efficiency, it does provide the means to derive analytical expressions for networks with certain structure, and thus insights into the role of multiple sensing modalities.

D. Computing Multiplex Influence Spread Analytically

We derive analytical expressions for multiplex cascade centrality for two illustrative classes of the LTM with a two-layer (duplex) network and N agents. In Section VII-D1, each of the layers is the same path network. This means that each agent has the same neighbors for each sensing modality; for example, each agent can see and hear its neighbors. Our results reveal how the cascade is affected when agents distinguish between signals rather than project them. If we view the path network as a cycle network with one link missing, then the duplex permutation network we study in Section VII-D2 has one layer missing the link between agents 1 and N and the other layer missing the link between agents $N-1$ and N.

1) Duplex Repeated Path Network: Let $\mathcal{G}_R$ be the duplex repeated path network of Fig. 4, which has the monoplex equal-neighbor path network $G_{Pa} = \mathrm{proj}(\mathcal{G}_R)$ on each of its two layers. For any N and any agent j, monoplex cascade centrality $\mathcal{C}_j^{G_{Pa}}$ and multiplex cascade centralities $\mathcal{C}_j^{\mathcal{G}_R,\mathrm{OR}}$ and $\mathcal{C}_j^{\mathcal{G}_R,\mathrm{AND}}$ can be expressed analytically as follows.
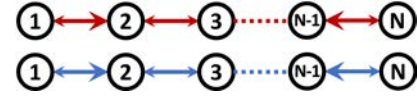

Fig. 4. Duplex repeated path network $\mathcal{G}_R$ has equal-neighbor path graph G_{Pa} as each layer.

Proposition 1 (Multiplex cascade centrality for $\mathcal{G}_R$). Consider the monoplex equal-neighbor path network G_{Pa} and duplex repeated path network $\mathcal{G}_R$ for N agents with $u_i = u \in \{\mathrm{OR}, \mathrm{AND}\}$. Then

$$\mathcal{C}_j^{G_{Pa}} = h_j(.5), \quad \mathcal{C}_j^{\mathcal{G}_R,\mathrm{OR}} = h_j(.75), \quad \mathcal{C}_j^{\mathcal{G}_R,\mathrm{AND}} = h_j(.25),$$

$$h_j(p_0) = \begin{cases} \sum_{l=0}^{N-2} p_0^l + p_0^{N-2}, & j \in \{1,N\} \\ 1 + \sum_{l=0}^{N-3} p_0^l + p_0^{N-3}, & j \in \{2,N-1\} \\ \sum_{l=0}^{j-1} p_0^l + p_0^{j-1} + \sum_{l=1}^{N-j-1} p_0^l + p_0^{N-j-1}, & \text{o.w.} \end{cases}$$

Moreover,

$$\mathcal{C}_j^{\mathcal{G}_R,\mathrm{OR}} > \mathcal{C}_j^{G_{Pa}} > \mathcal{C}_j^{\mathcal{G}_R,\mathrm{AND}}.$$

Proof. The result can be derived from Algorithm 1. $\qquad\square$

Proposition 1 provides a systematic way to evaluate spread for any number of agents N in the multiplex LTM on $\mathcal{G}_R$. Because the expression is a geometric series, the multiplex cascade centralities converge exponentially to fixed values as N increases. The inequality is consistent with the intuition in Remark 2: when agents can distinguish signals from different sensing modalities and use Protocol OR (AND), they are more (less) easily activated, and the cascade is enhanced (diminished) relative to when agents cannot distinguish signals.

2) Duplex Permutation Networks: Let $\mathcal{G}_P$ be the duplex permutation network of Fig. 5. Then $\mathrm{proj}(\mathcal{G}_P) = G_C$, the monoplex equal-neighbor cycle network. We derive analytical expressions for $\mathbb{P}_j^{\mathcal{G}_P,u}(\bar{x}_i=1)$, $u \in \{\mathrm{OR}, \mathrm{AND}\}$, as follows.

Proposition 2 (Probabilities for multiplex cascade centrality for $\mathcal{G}_P$). Consider the duplex permutation network $\mathcal{G}_P$ for N
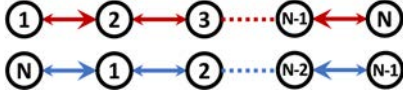
Fig. 5. Duplex permutation network $\mathcal{G}_P$.

agents with $u_i = u \in \{OR, AND\}$ and the monoplex equal-neighbor cycle network $G_C = \mathrm{proj}(\mathcal{G}_P)$. Then

$$\mathbb{P}_j^{\mathcal{G}_P, OR}(\bar{x}_i = 1) = (.75)^{|i-j|} + .5(.75)^{N-|i-j|-3} - .5(.75)^{N-5}$$
$$\mathbb{P}_j^{\mathcal{G}_P, AND}(\bar{x}_i = 1) = (.25)^{|i-j|} \qquad (10)$$
$$\mathbb{P}_j^{G_C}(\bar{x}_i = 1) = (.5)^{|i-j|} + (.5)^{N-|i-j|}$$

where $3 \leq i \leq N - 3$ and $j = 2, ..., i - 2, i + 2, ..., N - 2$. Moreover,

$$\mathcal{C}_j^{\mathcal{G}_P, OR} > \mathcal{C}_j^{G_C} > \mathcal{C}_j^{\mathcal{G}_P, AND}$$

$$\mathbb{P}_j^{\mathcal{G}_P, AND}(\bar{x}_i = 1) = \mathbb{P}_j^{\mathcal{G}_R, AND}(\bar{x}_i = 1) = (0.25)^{|i-j|}$$

$$\mathbb{P}_j^{\mathcal{G}_P, OR}(\bar{x}_i = 1) > \mathbb{P}_j^{\mathcal{G}_R, OR}(\bar{x}_i = 1).$$

Proof. The probabilities derive from Algorithm 1. The first inequality follows since $\mathbb{P}_j^{\mathcal{G}_P, OR}(\bar{x}_i = 1) > \mathbb{P}_j^{G_C}(\bar{x}_i = 1) > \mathbb{P}_j^{\mathcal{G}_P, AND}(\bar{x}_i = 1)$. The rest follows from Proposition 1. □

As in Proposition 1, the first inequality in Proposition 2 is consistent with the intuition in Remark 2.

When $u_N = AND$, the activity can only spread in $\mathcal{G}_P$ from j to i along the path between j and i on G_C that does not contain N. This explains the equality of probabilities for $\mathcal{G}_P$ and $\mathcal{G}_R$. However, when $u_i = u = OR$, the activity can spread in $\mathcal{G}_P$ from j to i along either path between j and i on G_C. This explains the last inequality, i.e., that the cascade is greater in $\mathcal{G}_P$ than in $\mathcal{G}_R$. This is an interesting result as it says that cascade size can be increased merely by designing the graph in one layer to be a permutation of the graph in the other.

Remark 4. The multiplex influence spread is said to be *submodular* if

$$\sigma_{S \cup \{i\}}^{\mathcal{G}, \mathcal{U}} - \sigma_S^{\mathcal{G}, \mathcal{U}} \geq \sigma_{T \cup \{i\}}^{\mathcal{G}, \mathcal{U}} - \sigma_T^{\mathcal{G}, \mathcal{U}}, \qquad (11)$$

for all agents i and all pairs of agent sets $S \subseteq T$. The submodularity property ensures that a greedy selection of seed set will achieve a near-optimal influence spread [3].
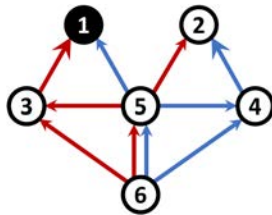

Fig. 6. Multiplex network $\mathcal{G}_s$ with two equal-neighbor layers and six agents. Red (blue) arrows represent edges in layer 1 (layer 2). $S_0 = \{1\}$.

If all the agents use Procotol OR, the multiplex influence spread can be shown to be submodular. However, we use the duplex network in Fig. 6 to show that if one or more agents use Protocol AND, the multiplex influence spread is

not submodular. In particular, the multiplex influence spread is not submodular with agents 5 and 6 using Protocol AND. Note that the protocols of other agents do not affect influence spread since agents 1 and 2 have no out-neighbors, and agents 3 and 4 only have out-neighbors in one layer. In Table I, we show the influence spread of four seed sets and a breakdown of the probabilities of each agent being active in each case.

TABLE I
INFLUENCE SPREAD OF FOUR DIFFERENT SEED SETS OF MULTIPLEX NETWORK IN FIG. 6 WITH AGENTS 5 AND 6 USING PROTOCOL AND.

Seed Set	$\{3\}$	$\{3, 1\}$	$\{3, 4\}$	$\{3, 4, 1\}$
Probabilities of agent 1 becoming active	0	1	0	1
Probabilities of agent 2 becoming active	0	0	0	0
Probabilities of agent 3 becoming active	1	1	1	1
Probabilities of agent 4 becoming active	0	0	1	1
Probabilities of agent 5 becoming active	0	0	1/4	1/2
Probabilities of agent 6 becoming active	0	0	7/16	5/8
Influence Spread	1	2	2.6875	4.125

We observe that

$$\sigma_{\{3\} \cup \{1\}}^{\mathcal{G}, \mathcal{U}} - \sigma_{\{3\}}^{\mathcal{G}, \mathcal{U}} = 1 < 1.4375 = \sigma_{\{3,4\} \cup \{1\}}^{\mathcal{G}, \mathcal{U}} - \sigma_{\{3,4\}}^{\mathcal{G}, \mathcal{U}},$$

which violates the submodularity condition.

VIII. A BAYESIAN NETWORK APPROACH

In this section, we map the problem of computing multiplex influence spread into a problem of probabilistic inference in Bayesian networks (BN). This means we can compute multiplex influence spread by using an appropriate algorithm for inference in BNs, such as the loopy belief propagation algorithm. We first define directed acyclic graphs.

Definition 8 (Directed acyclic graph). A *directed acyclic graph* (DAG) is a directed graph with no directed cycles.

Let the undirected graph of a DAG be obtained by replacing each directed edge with an undirected edge. If this undirected graph is a tree, the DAG is called a *polytree*. If this undirected graph contains cycles, the DAG is called a *DAG with loops*. A DAG with loops is not the same as a cyclic graph.

Definition 9 (Bayesian network and CPTs). Let $G = (V, E)$, where $V = 1, 2, ..., n$ and $E \subset V \times V$, be a directed acyclic graph (DAG). Each node $i \in V$ is associated with a random variable $x_i' \in \mathcal{X}_i'$. Denote the set of out-neighbors of $i \in V$ as N_i'. Let $\mathbb{P}(x_i' | x_{N_i'}')$ be the probability of x_i' conditioned on the states of nodes in N_i'. Then G is a *Bayesian network* if the joint distribution of the random variables is factorized into conditional probabilities: $\mathbb{P}(x_1', x_2', ..., x_n') = \prod_{i=1}^n \mathbb{P}(x_i' | x_{N_i'}')$. When every x_i' has finite support, then $\mathbb{P}(x_i' | x_{N_i'}')$ can be listed in a table, called a *conditional probability table (CPT)*, for each realization of $x_{N_i'}'$.

Probabilistic inference in a Bayesian network refers to calculating the marginal probability of the state of each unobserved node, conditioned on the states of the observed nodes. The belief propagation (BP) algorithm was first proposed by Pearl [22] to solve probabilistic inference in Bayesian networks. Pearl [22] showed that the algorithm is exact on DAGs without loops, i.e., trees and polytrees [22]. It is not guaranteed

to converge when applied to DAGs with loops. However, Murphy et al. [23] showed that loopy belief propagation (LBP) - the application of Pearl's algorithm to DAGs with loops - provides a good approximation when it converges.

In Algorithm 2, we show that if the projection network $\mathrm{proj}(\mathcal{G})$ is a DAG (where $\mathrm{proj}(\mathcal{G})$ projects all the layers onto a single graph as defined in Section II), the joint distribution of the probability of activation of agents for the multiplex LTM admits the graphical structure of a Bayesian network.

Algorithm 2 (Bayesian network from multiplex LTM)**.** Given multiplex network $\mathcal{G}$ for which $\mathrm{proj}(\mathcal{G})$ is a DAG and sequence of protocols $\mathcal{U}$:

1) Let $G = \mathrm{proj}(\mathcal{G})$ be the underlying DAG for the Bayesian network. Then $N_i' = N_i = \cup_{k=1}^m N_i^k$.
2) Let the random variable x_i' of node i in the Bayesian network be $\bar{x}_i$, the steady-state value of agent i for the multiplex LTM on $\mathcal{G}$. Then $x_i' \in \mathcal{X}_i' = \{0,1\}$. The observed nodes include the nodes j in the seed set S_0 (for which $x_j' = 1$) and the nodes h not in the seed set that have no out-neighbors (for which $x_h' = 0$).
3) Construct CPTs for unobserved nodes from conditional probabilities for the multiplex LTM: $\mathbb{P}(x_i'|x_{N_i'}') = \mathbb{P}^{\mathcal{G},u_i}(\bar{x}_i|\bar{x}_{N_i})$, where
$\mathbb{P}^{\mathcal{G},\mathrm{OR}}(\bar{x}_i=1|\bar{x}_{N_i}) = 1 - \prod_{k=1}^m (\sum_{j \in N_i^k \cap \{\bar{x}_{N_i}=1\}} w_{i,j}^k)$,
$\mathbb{P}^{\mathcal{G},\mathrm{AND}}(\bar{x}_i=1|\bar{x}_{N_i}) = \prod_{k=1}^m (1-\sum_{j \in N_i^k \cap \{\bar{x}_{N_i}=1\}} w_{i,j}^k)$,
$\mathbb{P}^{\mathcal{G},\mathrm{OR}}(\bar{x}_i=0|\bar{x}_{N_i}) = 1 - \mathbb{P}^{\mathcal{G},\mathrm{OR}}(\bar{x}_i=1|\bar{x}_{N_i})$, and
$\mathbb{P}^{\mathcal{G},\mathrm{AND}}(\bar{x}_i=0|\bar{x}_{N_i}) = 1 - \mathbb{P}^{\mathcal{G},\mathrm{AND}}(\bar{x}_i=1|\bar{x}_{N_i})$.

We use the example of Fig. 6 to show how to construct CPTs. For agent 6, $N_6 = \{3,4,5\}$, $\bar{x}_{N_6} = \{\bar{x}_3, \bar{x}_4, \bar{x}_5\}$ and its CPT has $2^{|N_6|} = 8$ rows. Table II shows CPTs for agent 6 using Protocol OR and Protocol AND.

TABLE II
CPT OF AGENT 6 IN FIG. 6, THE CONDITIONAL PROBABILITIES ARE
$\mathbb{P}_0^{\mathrm{OR}} = \mathbb{P}^{\mathcal{G},\mathrm{OR}}(\bar{x}_6=0|\bar{x}_3,\bar{x}_4,\bar{x}_5)$, $\mathbb{P}_1^{\mathrm{OR}} = \mathbb{P}^{\mathcal{G},\mathrm{OR}}(\bar{x}_6=1|\bar{x}_3,\bar{x}_4,\bar{x}_5)$,
$\mathbb{P}_0^{\mathrm{AND}} = \mathbb{P}^{\mathcal{G},\mathrm{AND}}(\bar{x}_6=0|\bar{x}_3,\bar{x}_4,\bar{x}_5)$, $\mathbb{P}_1^{\mathrm{AND}} = \mathbb{P}^{\mathcal{G},\mathrm{AND}}(\bar{x}_6=1|\bar{x}_3,\bar{x}_4,\bar{x}_5)$.

$\bar{x}_3$	$\bar{x}_4$	$\bar{x}_5$	$\mathbb{P}_0^{\mathrm{OR}}$	$\mathbb{P}_1^{\mathrm{OR}}$	$\mathbb{P}_0^{\mathrm{AND}}$	$\mathbb{P}_1^{\mathrm{AND}}$
0	0	0	1.00	0.00	1.00	0.00
0	0	1	0.25	0.75	0.75	0.25
0	1	0	0.50	0.50	1.00	0.00
0	1	1	0.00	1.00	0.50	0.50
1	0	0	0.50	0.50	1.00	0.00
1	0	1	0.00	1.00	0.50	0.50
1	1	0	0.25	0.75	0.75	0.25
1	1	1	0.00	1.00	0.00	1.00

We focus on the case when $\mathrm{proj}(\mathcal{G})$ is a DAG. The case when $\mathrm{proj}(\mathcal{G})$ is cyclic can be handled by combining the junction tree algorithm [24] and belief propagation. The cyclic graph is first transformed into a junction tree by adding additional edges. In a junction tree, nodes are grouped into junction nodes and marginalization (BP) is performed on the junction node level. After marginal probabilities of junction nodes are obtained, a subsequent marginalization within the junction node of interest is required to obtain the desired marginal probability.

Theorem 2. *Given a multiplex network $\mathcal{G}$ for which $\mathrm{proj}(\mathcal{G})$ is a DAG, with seed set S_0 and sequence of protocols $\mathcal{U}$, the following two probabilities are the same:*

1) $\mathbb{P}_{S_0}^{\mathcal{G},\mathcal{U}}(\bar{x}_i = 1)$, *the probability that agent i is active at steady state for the multiplex LTM.*
2) $\mathbb{P}(x_i' = 1|x_j' = 1, x_h' = 0, j \in S_0, h \notin S_0, N_h = \varnothing)$, *the marginal probability of node i in the corresponding Bayesian network of Algorithm 2, conditioned on observed nodes in the seed set and those not in the seed set that have no out-neighbors.*

Proof. The activation of node i in the multiplex LTM, conditioned on the state of node i's out-neighbors in $\mathrm{proj}(\mathcal{G})$, is independent of the state of all other nodes. Thus, the joint probability of activation of each node i factors into components based on the graphical structure of the Bayesian network constructed in Algorithm 2. The probability of activation of node i, conditioned on the state of each of node i's out-neighbors in $\mathrm{proj}(\mathcal{G})$, is precisely the corresponding probability constructed in the CPT for node i associated with the multiplex LTM. Nodes in the seed set and those not in the seed set that have no out-neighbors are encoded as observed nodes in Algorithm 2. Thus, the probability that agent i is active at steady state for the multiplex LTM (probability in the first statement) can be computed by probabilistic inference on the corresponding Bayesian network (the marginal probability in the second statement). □

The equivalence in Theorem 2 implies that we can compute multiplex influence spread using probabilistic inference in BNs, which can be solved with BP algorithms.

Corollary 2. *Given a multiplex network $\mathcal{G}$ for which $\mathrm{proj}(\mathcal{G})$ is a DAG, with seed set S_0 and sequence of protocols $\mathcal{U}$, multiplex influence spread $\sigma_{S_0}^{\mathcal{G},\mathcal{U}}$ can be computed as*

$$\sum_{i=1}^n \mathbb{P}(x_i' = 1|x_j' = 1, x_h' = 0, j \in S_0, h \notin S_0, N_h = \varnothing).$$

Proof. This follows from Theorem 2 and Definition 6. □

Remark 5. The *computational complexity of Corollary 2 is dominated by the LBP algorithm, which has a complexity of* $O(n2^{\bar{d}})$. As $2 \le \bar{d} < mn$, the time cost of our Bayesian network approach is lower than that of Algorithm 1, which is $O(n\bar{d}^{mn})$.

Remark 6. Here, we provide an alternative proof of Proposition 1 based on probabilistic inference in BNs. Table III shows the CPT of agent $i \in \{2, \ldots, N-1\}$ for G_{Pa} and for $\mathcal{G}_R$ with $u = \mathrm{OR}$ and $u = \mathrm{AND}$. With one initially active

TABLE III
CPT OF AGENT $i \in \{2, 3, ..., N-1\}$. FOR G_{Pa}, $p_0 = .5$; FOR $\mathcal{G}_R$ WITH $u = \mathrm{OR}$, $p_0 = .75$; FOR $\mathcal{G}_R$ AND $u = \mathrm{AND}$, $p_0 = .25$.

$\bar{x}_{i-1}$	$\bar{x}_{i+1}$	$\mathbb{P}^{\mathcal{G},u}(\bar{x}_i=0|\bar{x}_{i-1},\bar{x}_{i+1})$	$\mathbb{P}^{\mathcal{G},u}(\bar{x}_i=1|\bar{x}_{i-1},\bar{x}_{i+1})$
0	0	1	0
0	1	$1 - p_0$	p_0
1	0	$1 - p_0$	p_0
1	1	0	1

agent j, the activity can only spread from an agent closer to j to an agent farther from j. Assume $1 < j < i < N$, for all three networks, the probability that agent i is active at steady state for the multiplex LTM can be factorized as follows:

$$\mathbb{P}_j^{\mathcal{G},u}(\bar{x}_i=1) = \mathbb{P}^{\mathcal{G},u}(\bar{x}_i=1|\bar{x}_{i-1}=1,\bar{x}_{i+1}=0) \qquad (12)$$
$$\times \mathbb{P}^{\mathcal{G},u}(\bar{x}_{i-1}=1|\bar{x}_{i-2}=1,\bar{x}_i=0) \times \cdots$$
$$\times \mathbb{P}^{\mathcal{G},u}(\bar{x}_{j+1}=1|\bar{x}_j=1,\bar{x}_{j+2}=0) = p_0^{i-j}.$$

The last equality holds since each agent uses the same protocol and so each conditional probability is p_0. The cascade centralities follow by Definitions 6 and 7. Cases $i,j \in \{1,N\}$ are calculated similarly. The inequality follows since $\mathbb{P}_j^{\mathcal{G}_R,\mathrm{OR}}(\bar{x}_i=1) > \mathbb{P}_j^{\mathcal{G}_{Pa}}(\bar{x}_i=1) > \mathbb{P}_j^{\mathcal{G}_R,\mathrm{AND}}(\bar{x}_i=1)$. $\qquad\square$

Nguyen and Zheng [25] showed that computing influence spread in a monoplex DAG with the Independent Cascade Model (ICM) is #P-complete. Here, we prove a similar result for computing influence spread for the multiplex LTM. The implication is that approximating influence spread for the multiplex LTM is the best we can do for large networks.

Theorem 3. *Consider a multiplex network $\mathcal{G}$ for which* $\mathrm{proj}(\mathcal{G})$ *is a DAG, with seed set S_0 and sequence of protocols* $\mathcal{U}$. *Computing $\sigma_{S_0}^{\mathcal{G},\mathcal{U}}$ for the multiplex LTM is #P-complete.*

Proof. The multiplex LTM problem, i.e., computing $\sigma_{S_0}^{\mathcal{G},\mathcal{U}}$ for the multiplex LTM, is #P-complete if (i) it is #P-hard and (ii) it is in #P. We first prove (ii). By Corollary 2, every instance of the multiplex LTM problem can be reduced to a marginalization problem. Since the marginalization problem is #P-complete, the multiplex LTM problem is in #P.

We prove (i) by showing that the multiplex LTM problem is a reduction from the ICM problem, which has been shown to be #P-complete (Theorem 1 of [25]). The ICM problem refers to computing the influence spread $\sigma_{S_0}^{G,\mathrm{ICM}}$ for the ICM on a monoplex DAG $G = (V,E)$ with seed set S_0 and probability $w_{j,i}$ assigned to each edge $e_{j,i} \in E$.

Let m be the largest number of out-neighbors over all nodes in V. Consider a multiplex network with m layers $G_1,...,G_m$ where $G_k = (V,E^k)$. To define edge sets E^k, assign all edges in G and to the multiplex network such that for each node there is at most one outgoing edge in each layer k. Let $w_{j,i}$ be the weight of edge from i to j in the multiplex network. Define a set V' with a node $i' \in V'$ for each node $i \in V$. For each i and k, compute the sum of weights of i's outgoing edges in layer k. If the sum is not 1, create an edge $e_{i,i'}^k \in E'^k$ from i to i' and assign to the edge a weight that makes the sum equal 1. Let multiplex network $\mathcal{G}'$ have m layers $G_1',...,G_m'$ where $G_k' = (V \cup V', E^k \cup E'^k)$. Then $\mathrm{proj}(\mathcal{G}')$ is a DAG. Further, every node $i' \in V'$ has no out-neighbors and $i' \notin S_0$. So in the multiplex LTM $i' \in V'$ remains inactive. Let $\mathcal{U} = \{\mathrm{OR},...\mathrm{OR}\}$. By construction, $\sigma_{S_0}^{\mathcal{G}',\mathcal{U}} = \sigma_{S_0}^{G,\mathrm{ICM}}$. $\qquad\square$

IX. HETEROGENEITY IN PROTOCOL

A. Small Heterogeneous Multiplex Networks

We compute multiplex cascade centrality to evaluate for the LTM the role of heterogeneity in the tradeoff between sensitivity of the cascade to a reliable input and robustness of the cascade to an unreliable signal. We consider a signal to be reliable if at least one agent can observe it in all sensing modalities; otherwise it is unreliable. For example, if the signal is only observed by means of a camera, it might be an optical illusion, whereas if the signal is also observed with a laser scanner it is likely real. Consider a team of small robots involved in collective transport that cascade to form a bridge when one robot observes a gap in the terrain. Forming the bridge may require the robots to abandon their current task, which could be counterproductive if the gap isn't really there.

Knowing that agents that use Protocol OR enhance the cascade and agents that use AND diminish the cascade, we examine how to leverage heterogeneity in protocol to advantage. Parametrizing the tradeoff by c, we solve as a function of c for the optimal heterogeneous distribution of agents using OR and agents using AND.

We investigate with the duplex network of Fig. 7, which is small enough that we can compute cascade centrality with Algorithm 1. There are six agents (nodes 1 to 6) and a seventh node that represents an external signal. When node 7 appears in both layers, as in Fig. 7, we interpret it as reliable. When node 7 appears in only one layer, we interpret it as unreliable. We assume that only one agent (e.g., agent 1 in Fig. 7) senses node 7, whether or not it is reliable, and that it is equally likely to be any of the six agents that sense it. We assume the edge pointing to the signal has a weight of 1.

The graph in layer 1 (red edges) in Fig. 7 represents a *directed sensing* modality, e.g., a team of robots with front and side facing cameras or a school of fish that see poorly to their rear. The graph in layer 2 (blue edges) represents a *proximity sensing* modality, e.g., robots that receive local broadcasts or fish that detect local movement with their lateral line.
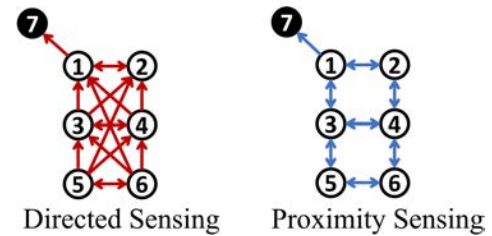

Fig. 7. Duplex network with agents 1 to 6, layer 1 (red), and layer 2 (blue). Node 7, the external signal, is reliable since it appears in both layers.

As each agent can use either protocol, there are $2^6 = 64$ different possible sequences of protocols $\mathcal{U}$ in total. We define the utility function Q as a function of $\mathcal{U}$ and $c \geq 0$ to measure the benefit of cascades that result from reliable signals less the cost of cascades that result from unreliable signals:

$$Q(\mathcal{U},c) = \frac{1}{6}\sum_{l=1}^{6}\left(\mathcal{C}_7^{\mathcal{G}_r^l,\mathcal{U}} - c\frac{1}{2}(\mathcal{C}_7^{\mathcal{G}_{u1}^l,\mathcal{U}} + \mathcal{C}_7^{\mathcal{G}_{u2}^l,\mathcal{U}})\right). \qquad (13)$$

Superscript l indexes the agent sensing node 7. Subscripts "r", "u1" and "u2" index the networks where node 7 appears in both layers, layer 1 only, and layer 2 only, respectively. Increasing c increases cost of response to unreliable signals relative to benefit of response to reliable signals. Given c, the optimal sequence of protocols is $\mathcal{U}^c = \mathrm{argmax}_{\mathcal{U}} Q(\mathcal{U},c)$.

Fig. 8 illustrates $\mathcal{U}^c$ (with symmetry implied) on a plot of the optimal fraction of agents using AND as a function of c. A white (gray) circle represents an agent using OR (AND). When c is 0 or small, responding to reliable signals dominates and all agents use OR. When c is increased towards 3 and beyond, avoiding unreliable signals dominates and all agents use AND. For c in between, the optimal solution is heterogeneous with more agents using AND as c increases: first agent 1 or 2, then agents 1 and 2, then {1,2,5} or {1,2,6}, then {1,2,3,6} or {1,2,4,5}, and then {1,2,3,5,6} or {1,2,4,5,6}.

It is intuitive that as the cost c of responding to unreliable signals grows, the ratio of the number of agents using AND (i.e., less readily activated agents) to the number of agents using OR (i.e., more readily activated agents) should grow. What this example shows beyond that intuition is how our methodology provides the systematic means to compute, and develop insight on, how many and where in the multiplex network it is optimal, as a function of c, to locate agents using AND versus agents using OR. We can observe in Fig. 8 a pattern that results from the directed sensing network (red) in Fig. 7: the first agents to become less responsive, as c grows, should be those in the "front", and each subsequent agent to become less responsive should be furthest from the other less responsive agents.
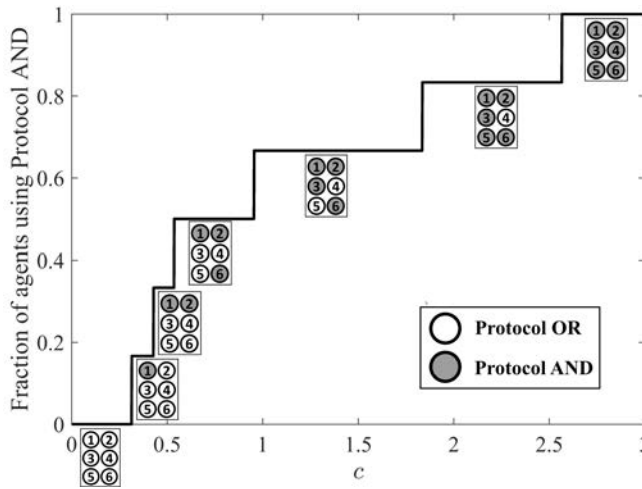

Fig. 8. The optimal fraction of agents using Protocol AND with illustration of optimal solution $\mathcal{U}^c$ as c varies from 0 to 3. Symmetry is implied.

B. Large Heterogeneous Multiplex Networks

We apply Corollary 2 to study multiplex cascade centrality for a random multiplex network with 20 agents and homogeneous and heterogeneous protocols. We randomly generate duplex networks, for which the projection networks are DAGs, by fixing a topological order of nodes and assigning edges randomly with probability p_e. A higher probability p_e means agents sense a greater number of the other agents. We consider five scenarios, where each agent randomly and independently chooses Protocol OR with probability 0% (homogeneous AND), 25%, 50%, 75%, 100% (homogeneous OR).

We let the root node be the initial active agent and study how the cascade centrality of the root changes as we vary p_e

from 0 to 1. For every value of p_e, we randomly generate 400 networks. Fig. 9 shows how the cascade centrality, averaged over the random networks, changes as a function of p_e.
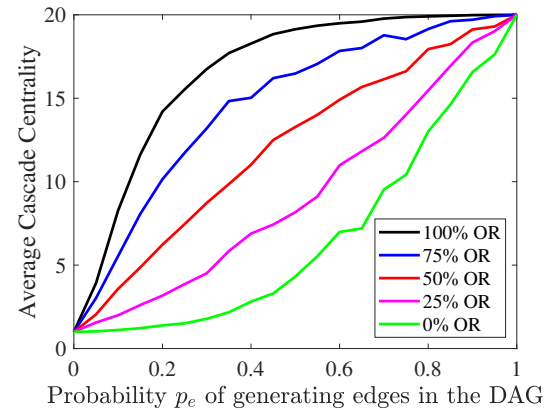

Fig. 9. Multiplex cascade centrality of root node, averaged over 400 networks, as a function of probability p_e of edges in the DAG.

Regardless of the protocol, as $p_e \to 0$, the DAG become disconnected and the cascade centrality goes to 1 (only the root node is active at steady state). As $p_e \to 1$, the root node activates every other agent and the cascade centrality goes to 20. For p_e in between, Fig. 9 shows that the homogeneous groups with $u = $ OR are most readily activated and cascade size is sensitive in the range $p_e \in (0, .5)$. Homogeneous groups with $u = $ AND are least readily activated and cascade size is sensitive in the range $p_e \in (.5, 1)$. Fig. 9 shows further the implications of heterogeneity in protocol. For example, it can be seen how the percentage of agents using OR (versus AND) can be selected and modulated to tune the sensitivity of cascade size as a function of p_e.

X. CONCLUSION

We have extended the LTM to multiplex networks where agents use different protocols that distinguish signals from multiple sensing modalities. We have derived algorithms to compute influence spread accurately using the multiplex LEM and approximately using probabilistic inference. We have shown how multiple sensing modalities affect spread and how heterogeneity trades off sensitivity and robustness of spread. Our methods provide systematic means for designing heterogeneous protocols on multiplex networks of resource constrained systems, such as teams of nano-robots, to control the dynamics of spread.

Our analyses naturally extend to more general protocols and alternative logical rules. The non-progressive LTM, where active agents don't necessary stay active, has been studied on monoplex networks (see, e.g., [26]). Understanding the non-progressive LTM on multiplex networks presents another compelling direction for future work.

REFERENCES

[1] M. Granovetter, "Threshold models of collective behavior," *American Journal of Sociology*, vol. 83, no. 6, pp. 1420–1443, 1978.
[2] T. C. Schelling, *Micromotives and Macrobehavior*. Norton, 1978.

[3] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2003, p. 137–146.

[4] Y. Lim, A. Ozdaglar, and A. Teytelboym, "A simple model of cascades in networks," MIT, Tech. Rep., 2015.

[5] D. Acemoglu, A. Ozdaglar, and E. Yildiz, "Diffusion of innovations in social networks," in *IEEE Conf. Decision and Control*, 2011, pp. 2329–2334.

[6] D. Rosa and A. Giua, "A non-progressive model of innovation diffusion in social networks," in *IEEE Conf. Decision and Control*, 2013, pp. 6202–6207.

[7] A. Garulli, A. Giannitrapani, and M. Valentini, "Analysis of threshold models for collective actions in social networks," in *European Control Conference*, 2015, pp. 211–216.

[8] M. Fardad and G. Kearney, "On a linear programming approach to the optimal seeding of cascading failures," in *IEEE Conf. Decision and Control*, 2017, pp. 102–107.

[9] W. S. Rossi, G. Como, and F. Fagnani, "Threshold models of cascades in large-scale networks," *IEEE Transactions on Network Science and Engineering*, vol. 6, no. 2, pp. 158–172, 2019.

[10] Y. D. Zhong and N. E. Leonard, "A continuous threshold model of cascade dynamics," in *IEEE Conf. Decision and Control*, 2019, pp. 1704–1709.

[11] D. J. Watts, "A simple model of global cascades on random networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 9, pp. 5766–5771, 2002.

[12] S. B. Rosenthal, C. R. Twomey, A. T. Hartnett, H. S. Wu, and I. D. Couzin, "Revealing the hidden networks of interaction in mobile animal groups allows prediction of complex behavioral contagion," *Proceedings of the National Academy of Sciences*, vol. 112, no. 15, pp. 4690–4695, 2015.

[13] S. Gomez, A. Diaz-Guilera, J. Gomez-Gardeñes, C. J. Perez-Vicente, Y. Moreno, and A. Arenas, "Diffusion dynamics on multiplex networks," *Physical Review Letters*, vol. 110, no. 2, 2013.

[14] I. Trpevski, A. Stanoev, A. Koseska, and L. Kocarev, "Discrete-time distributed consensus on multiplex networks," *New Journal of Physics*, vol. 16, no. 11, p. 113063, 2014.

[15] H. Shao, Y. Xi, M. Mesbahi, D. Li, Y. Xu, and Z. Gan, "Relative tempo of consensus dynamics on multiplex networks," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5184–5189, Jul. 2017.

[16] C. G. Antonopoulos and Y. Shang, "Opinion formation in multiplex networks with general initial distributions," *Scientific Reports*, vol. 8, no. 1, 2018.

[17] V. V. Vasconcelos, S. A. Levin, and F. L. Pinheiro, "Consensus and polarization in competing complex contagion processes," *Journal of the Royal Society Interface*, vol. 16, no. 155, p. 20190196, 2019.

[18] O. Yağan and V. Gligor, "Analysis of complex contagions in random multiplex networks," *Physical Review E*, vol. 86, no. 3, p. 036103, 2012.

[19] M. Salehi, R. Sharma, M. Marzolla, M. Magnani, P. Siyari, and D. Montesi, "Spreading processes in multilayer networks," *IEEE Transactions on Network Science and Engineering*, vol. 2, no. 2, pp. 65–83, 2015.

[20] Y. D. Zhong, V. Srivastava, and N. E. Leonard, "On the linear threshold model for diffusion of innovations in multiplex social networks," in *IEEE Conf. Decision and Control*, 2017, pp. 2593–2598.

[21] L. Yang, Z. Yu, M. A. El-Meligy, A. M. El-Sherbeeny, and N. Wu, "On multiplexity-aware influence spread in social networks," *IEEE Access*, vol. 8, pp. 106 705–106 713, 2020.

[22] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[23] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *Proc. 15th Conf. Uncertainty in Artificial Intelligence*, 1999, pp. 467–475.

[24] S. L. Lauritzen and D. J. Spiegelhalter, "Local computations with probabilities on graphical structures and their application to expert systems," *J. Royal Statistical Society, Series B*, pp. 157–224, 1988.

[25] H. Nguyen and R. Zheng, "Influence spread in large-scale social networks–a belief propagation approach," in *Machine Learning and Knowledge Discovery in Databases*, 2012, pp. 515–530.

[26] D. Rosa and A. Giua, "A non-progressive model of innovation diffusion in social networks," in *52nd IEEE Conference on Decision and Control*, 2013, pp. 6202–6207.

Yaofeng Desmond Zhong received the B.E degree in civil engineering from Tsinghua University, Beijing, China, in 2015, and the M.A. and Ph.D. degree in mechanical engineering from Princeton University, Princeton, NJ, USA, in 2017 and 2020, respectively.

He is currently a Research Scientist at Siemens Technology, Princeton, NJ, USA. His research interests include modeling and analysis of collective behavior in biological systems and designing bio-inspired multi-agent engineering systems.

Vaibhav Srivastava (Senior Member, IEEE) received the B.Tech. degree (2007) in mechanical engineering from the Indian Institute of Technology Bombay, Mumbai, India; the M.S. degree in mechanical engineering (2011), the M.A. degree in statistics (2012), and the Ph.D. degree in mechanical engineering (2012) from the University of California at Santa Barbara, Santa Barbara, CA. He served as a Lecturer and Associate Research Scholar with the Mechanical and Aerospace Engineering Department, Princeton University, Princeton, NJ from 2013-2016.

He is currently an Assistant Professor with the Electrical and Computer Engineering at Michigan State University. He is also affiliated with Mechanical Engineering, Cognitive Science Program, and Connected and Autonomous Networked Vehicles for Active Safety (CANVAS) Program. His research focuses on Cyber Physical Human Systems with emphasis on mixed human-robot systems, networked multi-agent systems, aerial robotics, and connected and autonomous vehicles.

Naomi Ehrich Leonard (Fellow, IEEE) received the B.S.E. degree in mechanical engineering from Princeton University, Princeton, NJ, USA, in 1985, and the M.S.and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, MD, USA, in 1991 and 1994, respectively. From 1985 to 1989, she was an Engineer in the electric power industry.

She is currently the Edwin S. Wilsey Professor with the Department of Mechanical and Aerospace Engineering and the Director of the Council on Science and Technology at Princeton University. She is also an Associated Faculty of Princeton's Program in Applied and Computational Mathematics. Her research and teaching are in control and dynamical systems with current interests in multi-agent systems on networks, distributed decision making, spreading processes, multi-robot teams, and collective behavior.