Competition, Collaboration, and Optimization in Multiple Interacting Spreading Processes

Hanlin Sun[*]

School of Mathematical Sciences, Queen Mary University of London,
Mile End Road, London E1 4NS, United Kingdom

David Saad[†]

Non-linearity and Complexity Research Group, Aston University, Birmingham, B4 7ET, United Kingdom

Andrey Y. Lokhov[‡]

Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

Competition and collaboration are at the heart of multiagent probabilistic spreading processes. The battle for public opinion and competitive marketing campaigns are typical examples of the former, while the joint spread of multiple diseases such as HIV and tuberculosis demonstrates the latter. These spreads are influenced by the underlying network topology, the infection rates between network constituents, recovery rates, and, equally important, the interactions between the spreading processes themselves. Here, for the first time, we derive dynamic message-passing equations that provide an exact description of the dynamics of two, interacting, unidirectional spreading processes on tree graphs, and we develop systematic low-complexity models that predict the spread on general graphs. We also develop a theoretical framework for the optimal control of interacting spreading processes through optimized resource allocation under budget constraints and within a finite time window. Derived algorithms can be used to maximize the desired spread in the presence of a rival competitive process and to limit the spread through vaccination in the case of coupled infectious diseases. We demonstrate the efficacy of the framework and optimization method on both synthetic and real-world networks.

Subject Areas: Complex Systems,
Interdisciplinary Physics,
Statistical Physics

I. INTRODUCTION

Spreading processes have become increasingly more important in the fast-moving modern world, where physical mobility is cheaper and easier than ever before, and where information is passed instantaneously on multilayered interwoven webs of contacts. Consequently, pandemics— whether physical or virtual, in the form of computer viruses, internet rumors, or marketing campaigns—spread very rapidly. For instance, the ongoing COVID-19 pandemic [1,2] caused an unprecedented disruption in the world's functioning. Previously, an outbreak of African swine fever caused by *Asfivirus* in China [3] in 2018 posed the risk of spreading globally. In 2017, a worldwide cyberattack by the WannaCry ransomware cryptoworm was estimated to have affected more than 200,000 computers across 150 countries, with total damages ranging from hundreds of millions to billions of dollars [4].

Very often, spreading processes do not diffuse on their own but instead show a complex dynamics, characterized by a collaboration or competition between them. An example of a collaborative spreading process is the coinfection of HIV and tuberculosis, the latter being a major factor that influences death rates of AIDS patients [5]. The risk of developing tuberculosis is estimated to be 16–27 times higher in people living with HIV than among uninfected individuals [6]. AIDS patients are more susceptible to tuberculosis, due to their weakened immune system, and tuberculosis can also activate the replication of the HIV virus. Epidemiological studies have also shown that coinfection exists between HIV and many other diseases, including malaria parasites [7], herpes [8], fungal [9] and bacteria [10], and between Zika and dengue viruses [11]. An example of asymmetric collaborative spreading is given by hepatitis D, which can only transmit to people who are already infected with hepatitis B.

[*]hanlin.sun@qmul.ac.uk
[†]d.saad@aston.ac.uk
[‡]lokhov@lanl.gov

The spreading of rumors is an example of a competitive process. Recently, the "antivaxx movement" in the U.S. has attracted the public's attention—for instance, through Twitter messages—leading to a growing number of parents who believe that vaccination is a violation of human rights, and that vaccines cause autism and brain damage and do not benefit the health and safety of society. This belief has been spreading rapidly through social media. As a result, the measles virus, which was declared to be eliminated in 2000, is making a comeback [12]; consequently, the World Health Organization named vaccine hesitancy as one of the top 10 global threats [13], and social network platforms have been requested to block the spread of related information [14]. A realistic competitive scenario is given by a situation where "valid information" and "unsubstantiated rumors" spread on the network simultaneously, and where individuals exposed to one tend to believe in its content and are less susceptible to the other.

Two important problems that recurrently arise in the analysis of spreading processes are (1) forecasting the dynamics and (2) the optimal use of resources to control the dynamics, for instance, to maximize or minimize the spread. Forecasting is based on a probabilistic modeling and inference of the system state, such as the prediction of infection probability over time for given initial conditions and interaction type. Optimization is often referred to resource allocation tasks such as the initial choice of best spreaders or to the best vaccination strategy to contain an outbreak of a disease. In this paper, we develop novel methods that address current gaps pertaining to both inference and optimization of interaction spreading processes.

There exists a large body of work on probabilistic modeling of spreading processes. The most commonly studied models include single spreading processes that follow the susceptible-infected-recovered/removed (SIR), susceptible-infected (SI), and susceptible-infected-susceptible (SIS) dynamics [15,16], where variables can have a small set of statuses, such as S, I, and R, and transition from one status to another depending on their original status and that of their neighbors. The exact prediction of the spread within these models is NP-hard [17], and therefore, the dynamics has been approximately analyzed using a variety of mean-field methods (see Refs. [15,18–23] for a review). A mean-field method of the message-passing type that is particularly suited for approximating dynamics of continuous and discrete SIR-type models on sparse networks has been introduced in Refs. [24–26]; in particular, it gives an *exact* prediction of the spread on tree graphs. When averaged over an ensemble of graphs, this method is equivalent to the edge-based compartmental modeling (EBCM) method [27–29] derived using the cavity-method-type arguments and the correct choice of dynamic variables that allows one to close the system of equations. Yet another equivalent representation is given in terms of dynamic belief propagation (DBP) equations on time

trajectories, which was presented in Refs. [30–32]. A framework introduced in Ref. [32] showed how starting from a DBP representation allows one to systematically derive closed-form dynamic message-passing (DMP) equations for any models with unidirectional dynamics (so that variable statuses cannot be revisited). In particular, the method of Ref. [32] not only recovers previously known DMP equations for simple SIR-type models [24,25,28,33] but also allows one to derive DMP equations for more complex models with multiple neighbor-dependent transitions, where guessing the correct dynamic variables becomes incomparably harder.

The analysis of multiagent spreading is much more involved because of the interaction between processes and its impact on the spreading dynamics. Numerical studies of multiagent processes [34] have revealed the existence of new transitions, as a result of the cross-process interaction, and an aggressive spreading mode, which points to a percolation transition. These results highlight the risk of an unpredictable and violent outbreak in cooperative spreading scenarios. The most relevant studies to the current work focus on the analysis of multiagent spreading in a competitive scenario on a specific network, using continuous equations similar to those of dynamic message passing [35], and on a two-stage infection process, which is a specific case for multiagent spreading processes [36]. Also relevant to our work are studies of complex contagions [37,38], characterized by the requirement for multiple transmissions before a network node changes status. While this scenario is not exactly the same as the interacting processes that we analyze in this paper, its dynamics depend on the infection history, similarly to the scenarios we examine. In this case, the interplay between topology and initial conditions may give rise to hybrid phase transitions when cascades are only possible for sufficiently prevalent initial infections. We can envisage similar phenomena in the scenarios studied here for some infection probabilities, but have not observed them in the experiments carried out here as they are not the focus of our study. Interestingly, recent work [39] shows a mapping between interacting multiagent spreading processes and social reinforcement infections through multiple transmissions.

Competitive [35,40–45] and collaborative [46–56] spreading processes have been studied in different contexts and in a variety of scenarios. The foci of many of these studies have been the fixed-point properties of the system, such as a phase diagram [40,41,44,49], describing regimes where one spreading process dominates the infection map or where both processes coinfect the system nodes and the type of transition between phases, epidemic thresholds, and the infection cluster size [35,42,45,46,48,50–54]. These analyses, for the most part, do not require a full solution of the dynamics. Other studies focus on dynamical properties of the fraction of infected network nodes, by one of the processes or both [42,44], by investigating the

corresponding differential equations to identify phenomena such as hysteresis [56] (particularly in the SIS model scenario), the relation between topology and dynamics [50,52], and the emergence of infected clusters [57] linked to temporal correlations. Additionally, most studies focus on analyzing networks of different degree distributions [35,41,42,49–51,53,54], rather than specific network instances.

A recent attempt to extend message-passing equations to the case of cooperative epidemic spreading [58], which is most relevant to the current study, only focused on a particular case where transmission is independent of the status of the target node. Additionally, this work studies different degree distributions, rather than specific instances, and concentrates on the fixed point properties such as the phase diagram and infected cluster size, falling short of a complete description of the dynamics. Hence, unlike in the case of single dynamics, exact equations for describing the complete dynamics of general interacting processes on tree graphs remain unknown. In this work, our first major contribution consists in a derivation of DMP equations for interacting unidirectional spreading processes that are exact on tree graphs, including low-complexity message-passing equations for the case of collaborative interactions. Moreover, we study approximate schemes to these equations that result in simplified expressions that can be applied on general sparse networks.

Optimal resource deployment in various spreading settings has mostly been investigated in the case of a single spreading process. One of the most commonly studied problems is identifying the most influential spreaders, on which the deployment of resources at time zero would maximize the spread at a given end time. Most of these studies rely on the network's topological properties, and selection strategies are based on high-degree nodes [59], neighbors of randomly selected vertices [60], betweenness centrality [61], random walk [62], graph partitioning [63], and K-shell decomposition [64]. These approaches mostly ignore important dynamical aspects that impact performance [65,66]. A related approach, termed network dismantling [67–69], aims at identifying the nodes that, if removed, lead to the fragmentation of the giant component and prevent the global percolation. The optimal deployment of immunization has been addressed using a belief propagation algorithm [70], based on cavity method techniques developed previously for deterministic threshold models [30,31]. Several scenarios that incorporate the dynamical properties of the spreading process—such as the optimal seeding problem, where one allocates the set of initially infected nodes that maximize the spread asymptotically—have been studied and analyzed [71–73]. The optimal seeding problem has been analyzed for the simple diffusion models of independent cascade (IC) and linear threshold types, and the optimal seeding problem has been shown to be NP-hard [73] for both; i.e., there are no

deterministic algorithmic solutions that grow polynomially with the system size. A different perspective is given by the study of scenarios with a finite time horizon, as studied for the IC [74] and other spreading models [75]. Most relevant to the current study is the application of a recurrent optimization framework [76] to the DMP-based probabilistic modeling of spreading processes [32]. The framework also facilitates both open-loop resource allocation (a preplanned assignment) and a closed-loop (dynamical resource deployment with feedback) under a limited remedial budget. We utilize a similar framework to investigate and optimize the dynamics of multiple spreading processes.

To the best of our knowledge, no analysis or optimization algorithms have been offered to address the *general* case of multi-process modeling and optimization, namely, incorporating both detailed topologies and dynamical properties within a fixed time window for both inference and optimization. Special cases, such as optimal seeding, have been addressed mostly via linearized fixed-point analysis [58] and for a simple dynamic that lends itself to single time-step optimization. Moreover, most optimization algorithms for single-agent processes follow the spread on a static network topology and cannot fully capture the intricate dynamics of multiple spreading processes; they are therefore less effective for the optimization tasks we aim to solve. As a second major contribution, we build an optimization framework for both competitive and collaborative scenarios based on the derived DMP equations that enable an accurate probabilistic forecasting.

We demonstrate that the inference method that we construct in this paper provides an accurate dynamical description of both competitive and collaborative scenarios on both toy and large-scale problems; it is asymptotically exact on treelike networks and provides a good approximation on networks with loops. We develop a related optimization algorithm for maximizing the spread within a given time window against a competing spread, as well as the containment of spreads in a collaborative spreading scenario through an optimized vaccination strategy that curbs one of the spreading processes. We demonstrate the efficacy of the suggested algorithm, offering excellent results with a scalable computational complexity.

The paper is organized as follows: In Sec. II, we derive exact and approximate DMP equations for general models of multiple interacting spreading processes. In Sec. III, we validate the efficacy of the probabilistic modeling by comparing the results with Monte Carlo simulations on synthetic and real networks. The optimization algorithm is introduced in Sec. IV for both competitive and cooperative scenarios and is tested on synthetic networks in Sec. V A. In Sec. V B, we apply the optimization algorithm to real-world networks for demonstrating its usefulness in more realistic scenarios, including both competitive and collaborative cases, and the optimal deployment of vaccines to contain an epidemic. A summary and outlook are provided in Sec. VI.

II. MODEL AND DYNAMIC MESSAGE-PASSING EQUATIONS

Spreading models studied in this work are based on the discrete-time SI process, where a couple of spreading agents are active in parallel and they interact with one another. The implication is that the status of a network vertex determines its susceptibility to be infected by either (or both) of the spreading processes. For instance, in mutually exclusive competitive processes that describe, for instance, the battle for public opinion, once a vertex has been infected by one process, it cannot be infected by any other and retains its status, which is also termed a "cross-immunity" [36]. In a collaborative spreading scenario that describes, for example, the spread of multiple diseases, being infected by one process increases the susceptibility of being infected by another according to some predefined conditional probability. Another variant of the model considered here is that of vaccination in the presence of collaboratively spreading diseases, where vaccination against one agent affects the spread of both processes. Although the framework for the various scenarios is similar, it includes some important differences and will therefore be developed separately. It is important to point out that the introduction of cooperative or competitive spreading processes cannot simply be reduced to a stochastic process with more statuses, which would include coinfection statuses; the new interactions between statuses complicate the dynamics due to the dependence of the interaction probability on the status of neighboring variables.

In both competitive and collaborative scenarios, we assume that two SI-type processes are spreading in discrete time on a graph $G = (V, E)$ comprising the set of vertices V and edges E such that each node i can generally be found in one of four statuses at any time step t: susceptible [$\sigma_i(t) = S$)], infected by disease A [$\sigma_i(t) = A$], infected by disease B [$\sigma_i(t) = B$], or activated by both processes A and B [$\sigma_i(t) = AB$]. In what follows, we define the spreading model in two scenarios and derive the corresponding DMP equations.

A. From dynamic belief propagation to dynamic message passing

DMP equations can be hard to guess beyond the simplest models, but they can be systematically derived starting from the general dynamic belief propagation algorithm that approximates the probabilities of time trajectories of individual nodes [32]. We use this approach here to obtain exact equations on tree graphs. For the two-process dynamics considered in this work, the dynamics of a single node i is fully described by a pair of activation times, (τ_i^A, τ_i^B), where τ_i^A denotes the first time when node i is found in the status A, and similarly for τ_i^B. For instance, $\tau_i^A = 0$ means that node i was initially in the active status A,

and we denote by $\tau_i^A = *$ the situation where node i did not get A-activated before some final observation time, in other words $*$ absorbs all the history that happens after the end of the observation window. For the convenience of presentation, in what follows, we consider two separate "observation windows," for the A and B processes.

We start our derivations with the general DBP equations [32,77,78] on the interaction graph, where the goal is to approximate the probability $m_{T_A+1,T_B+1}^i(\tau_i^A, \tau_i^B)$ that node i exhibits a trajectory (τ_i^A, τ_i^B) during the observation time window of length T_A for process A and T_B for process B (we keep the flexibility of having two separate time windows, although in most situations we use $T_A = T_B = t$). Exact equations that compute the probability $m_{T_A+1,T_B+1}^i(\tau_i^A, \tau_i^B)$ of a given time trajectory (τ_i^A, τ_i^B) of node i are explained in Appendix A. Because of the properties of the belief propagation algorithm [79,80], the fixed-point solution of the DBP equations is guaranteed to be exact on tree graphs, and it provides good estimates of marginal probabilities on loopy but sparse graphs [32].

Given the computed value of the marginals $m_{T_A,T_B}^i(\tau_i^A, \tau_i^B)$, one can straightforwardly define quantities of interest, such as probabilities for a given node i to be found in a given status:

$$P_S^i(t) = \sum_{\tau_i^A > t} \sum_{\tau_i^B > t} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B), \qquad (1)$$

$$P_A^i(t) = \sum_{\tau_i^A \leq t} \sum_{\tau_i^B} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B), \qquad (2)$$

$$P_B^i(t) = \sum_{\tau_i^B \leq t} \sum_{\tau_i^A} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B), \qquad (3)$$

$$P_{AB}^i(t) = \sum_{\tau_i^A \leq t} \sum_{\tau_i^B \leq t} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B). \qquad (4)$$

The definition of $P_A^i(t)$ [$P_B^i(t)$] indicates that at time t, the vertex i is A activated (B activated) irrespective of the other process. Therefore, $P_A^i(t)$ [$P_B^i(t)$] represents the probability that at time t, vertex i is in status A (B) alone or in status AB. Initial condition probabilities where the node is exclusively found as status A (B) will be denoted as $P_{A^*}^i(0)$ [$P_{B^*}^i(0)$]. In principle, one can solve the DBP equations to obtain $m_{T_A+1,T_B+1}^i(\tau_i^A, \tau_i^B)$ and then use the expressions (A5)–(A8) to obtain final aggregated expressions for the dynamic messages $P_S^i(t)$, $P_A^i(t)$, $P_B^i(t)$, and $P_{AB}^i(t)$. However, because of the generality of the DBP equations that are valid for any dynamics, this approach may not be the most efficient one: Computing a single marginal $m_{t+1,t+1}^i(\tau_i^A, \tau_i^B)$ may require as many as $O(t^{2d})$ operations for a single marginal, where d is the degree of the node i. On the other hand, for the concrete dynamics

such as the processes considered here, which often have a special structure, it is beneficial, from a computational point of view, to explore this structure in order to drastically reduce the complexity of computing the dynamic marginals $P_S^i(t)$, $P_A^i(t)$, $P_B^i(t)$, and $P_{AB}^i(t)$. When this special structure is exploited to produce closed-form algebraic equations for computing $P_S^i(t)$, $P_A^i(t)$, $P_B^i(t)$, and $P_{AB}^i(t)$ with low algorithmic complexity, the resulting computational scheme will be referred to as dynamic message-passing equations for the given process.

Therefore, the procedure we adopt below for deriving DMP equations will be as follows: (i) Specify DBP equations based on a given two-process dynamics; (ii) where possible, exploit the structure in the dynamics to derive low-complexity closed-form DMP equations that iteratively compute the quantities of interest $P_S^i(t)$, $P_A^i(t)$, $P_B^i(t)$, and $P_{AB}^i(t)$ starting with the algebraic definitions (A5)–(A8), which will inherit the exactness of prediction on tree graphs; and (iii) to further reduce the computational complexity or the algebraic form of the exact equations, derive *approximate* DMP equations that could be used as an algorithm for inference or optimization problems on general graphs.

B. Mutually exclusive competitive processes

The dynamics in mutually exclusive competitive processes can be made explicit by listing the allowed transitions and their respective probabilities at every discrete time step:

$$S(i) + A(j) \xrightarrow{\alpha_{ji}^A} A(i) + A(j),$$
$$S(i) + B(j) \xrightarrow{\alpha_{ji}^B} B(i) + B(j). \qquad (5)$$
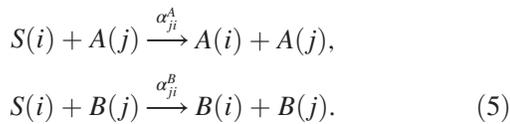
In other words, the two infection processes A and B are mutually exclusive: Any node can be infected by a neighboring node, assuming one of the two statuses, but once infected by one of the two processes, it cannot change its status. Since the infection is based on a two-vertex interaction through an edge, the processes in Eq. (5) seem deceptive as two completely independent parallel processes; however, they clearly interact through the graph topology and the exclusivity of the adopted statuses. Indeed, we assume that at any given time step, the infection probabilities of process A (denoted α_{ji}^A) or process B (denoted α_{ji}^B) are treated as independent, but the probability of being infected by both A and B simultaneously is forced to be 0 (thus creating probabilistic dependence between the two processes). For closing the dynamic update rule, we need to define what happens in the case where both processes jointly infect the vertex in the same time step, resulting in an invalid status. There are many possible ways to deal with this case, which could be accommodated in both analysis and simulations, depending on the needs of a

particular application. For the sake of simplicity, we consider the rule where the probabilities of transitioning to either status A or B, or staying in status S is proportionally renormalized in such a way that they sum to one. Alternatively, in simulation, one could think of this procedure as resampling in the case where the joint infection occurs: If a joint infection status by both processes A and B is sampled, it is rejected, and resampling is carried out. This process is done repeatedly until a valid status without progressing the dynamics, such that no spurious probabilistic dependencies emerge. According to the dynamic rules defined above, the probability of the transition to the status A from status S for a node i is given by

$$v_A^i(t) = 1 - \prod_{j \in \partial i}(1 - \alpha_{ji}^A \mathbb{1}[\sigma_j(t) = A]), \qquad (6)$$

where ∂i denotes the set of neighbors of node i, and $\mathbb{1}$ is an indicator function. Similarly, define

$$v_B^i(t) = 1 - \prod_{j \in \partial i}(1 - \alpha_{ji}^B \mathbb{1}[\sigma_j(t) = B]), \qquad (7)$$

$$Z_i = 1 - v_A^i(t) v_B^i(t). \qquad (8)$$

Then, under the resampling procedure explained above, the final renormalized transition probabilities at time step t read

$$q_{S \to A}^i(t) = \frac{v_A^i(t)(1 - v_B^i(t))}{Z_i(t)}, \qquad (9)$$

$$q_{S \to B}^i(t) = \frac{v_B^i(t)(1 - v_A^i(t))}{Z_i(t)}, \qquad (10)$$

$$q_{S \to S}^i(t) = \frac{(1 - v_A^i(t) - v_B^i(t) + v_A^i(t) v_B^i(t))}{Z_i(t)}, \qquad (11)$$

where the notation $\sigma_i(t) = A/B/S$ refers to a node i at time t, being in *one* of the statuses A, B, or S.

Given expressions (B1)–(B3), we can straightforwardly specify the respective DBP equations for the mutually exclusive competing dynamics of the two processes that are given in Appendix B. In Table I, we numerically verify that the DBP equations are exact on tree graphs. With a naive implementation, DBP marginals can be computed explicitly in time $\max(O(|E|T^{2(c-1)}), O(NT^{2c}))$, where c is the maximum degree of the graph, and T is the final observation time. It is important to note that transition probabilities defined as in Eqs. (B1)–(B3) reflect the complexity of interactions between processes that result from the renormalization of probabilities. Indeed, these expressions depend on the particular realization of statuses for all neighbors and hence lack any iterative structure at each time step. Because of this lack of structure, exact

TABLE I. Demonstration of exactness of DBP equations for mutually exclusive competitive processes via numerical comparison with the results from 10^8 Monte Carlo simulations on random trees. Case 1: Four-node random tree with uniform parameters $\alpha_A = 0.5$ and $\alpha_B = 0.5$ and initial conditions $P_A^2(0) = 1$ and $P_B^3(0) = 1$. Marginal probabilities for node $i = 0$ at time $t = 3$ are presented. Case 2: Four-node random tree with uniform parameters $\alpha_A = 0.2$ and $\alpha_B = 0.8$ and the initial conditions $P_A^2(0) = 1$ and $P_B^3(0) = 1$. Marginal probabilities for node $i = 0$ at time $t = 3$ are presented. Case 3: Five-node random tree with uniform parameters $\alpha_A = 0.4$ and $\alpha_B = 0.6$ and the initial conditions $P_A^2(0) = 1$ and $P_B^3(0) = 1$. Marginal probabilities for node $i = 0$ at time $t = 4$ are presented. Here and in the following tables and figures, the index of the chosen node has no significance but emphasizes that a *specific* node is investigated.

	Case 1		Case 2		Case 3	
	DMP	MC	DMP	MC	DMP	MC
$P_S^i(t)$	0.07986111	0.0799122	0.0077903	0.0078026	0.02302213	0.0230145
$P_A^i(t)$	0.13194444	0.1318928	0.00257234	0.0025947	0.00967836	0.0096883
$P_B^i(t)$	0.78819444	0.788195	0.98963737	0.9896027	0.967299512	0.9672972

low-complexity DMP equations cannot be straightforwardly derived for the chosen dynamics.

Although DBP equations for the mutually exclusive competing processes are exact on trees (see Table I), their polynomial but potentially high computational complexity makes them a less attractive choice in applications. To address this issue, we notice that real application problems are typically defined on sparse but nontree graphs. Even if low-complexity DMP equations were available, they would still yield only approximate solutions in general loopy networks. This observation motivates us to search for a tractable approximation of the message-passing equations on tree graphs: If the approximation is good, the resulting error on treelike loopy networks may be similar to the application of exact DBP or DMP equations. In Appendix C, we implement this strategy and derive approximate DMP equations for the mutually exclusive competing scenario with a computational complexity $O(|E|T)$, where $|E|$ is the number of edges in the graph and T is the observation window, which makes them scalable for very large sparse networks with millions of nodes. The approximation that we use is inspired by the fact that, in the absence of renormalization, the dynamics of each process follows the dynamics of the usual SI-type process, and hence it is natural to try to perform the renormalization procedure at the level of dynamic marginals. In what follows, we present numerical tests that illustrate the accuracy of the employed approximation.

C. Collaborative process

In the two-process collaborative scenario considered here, when a node is infected by one process, its susceptibility to activation by another process increases (or decreases). Unlike in the mutually exclusive case, a node can be infected by both processes, and the influence of the different processes is not necessarily symmetric. The dynamics in collaborative processes can be made explicit by listing the possible transitions and their respective probabilities at every discrete time step:

$$S(i) + A(j) \xrightarrow{\alpha_{ji}^A} A(i) + A(j), \tag{12}$$

$$S(i) + B(j) \xrightarrow{\alpha_{ji}^B} B(i) + B(j), \tag{13}$$

$$A(i) + B(j) \xrightarrow{\alpha_{ji}^{BA}} AB(i) + B(j), \tag{14}$$

$$A(i) + B(j) \xrightarrow{\alpha_{ij}^{AB}} A(i) + AB(j). \tag{15}$$

In this scenario, status AB can be regarded as a combination of status A and status B. The nontriviality of the interaction between both processes comes from the fact that α_{ij}^{AB} and α_{ji}^{BA} are different from α_{ji}^A and α_{ji}^B, respectively: when a node is infected by one process and becomes more (or less) vulnerable to another, and vice versa. Notice that unlike the mutually exclusive scenario where the status AB is forbidden, under the general collaborative scenario, the process $S(i) \to AB(i)$ is allowed, and in discrete time the rule

$$S(i) + AB(j) \xrightarrow{\alpha_{ji}^A \times \alpha_{ji}^B} AB(i) + AB(j) \tag{16}$$

follows from the transition rules above, simply as coactivation that happens at the same time.

Following the scheme outlined above, we can start by forming a dynamic transition kernel that encapsulates the various transition rules and allows us to write the DBP equations for this spreading model. The resulting DBP equations are given in Appendix D. The special structure of the dynamic kernel is written as a sum of possible transition sequences factorized over the neighbors of a given node, making it possible to derive low-complexity and *exact* DMP equations for the collaborative model. The algebraic form of equations for the dynamic marginals is given a simple and intuitive meaning. The probability of finding node i in status S can be written as

TABLE II.　Demonstrating the exactness of the DMP equations for collaborative processes via numerical comparison against results from 10^8 Monte Carlo simulations on random trees. Case 1: Six-node random tree with uniform parameters $\alpha_A = 0.1$, $\alpha_B = 0.2$, $\alpha_{AB} = 0.8$, and $\alpha_{BA} = 0.9$, and the initial conditions $P_A^2(0) = 1$ and $P_B^0(0) = 1$. Marginal probabilities for node $i = 3$ at time $t = 5$ are presented. Case 2: Six-node random tree with uniform parameters $\alpha_A = 0.6$, $\alpha_B = 0.5$, $\alpha_{AB} = 0.2$, and $\alpha_{BA} = 0.3$, and the initial conditions $P_A^2(0) = 1$ and $P_B^0(0) = 1$. Marginal probabilities for node $i = 3$ at time $t = 5$ are presented. Case 3: Five-node random tree with uniform parameters $\alpha_A = 0.5$, $\alpha_B = 0.4$, $\alpha_{AB} = 0.8$, and $\alpha_{BA} = 0.6$, and the initial conditions $P_A^2(0) = 1$ and $P_B^0(0) = 1$. Marginal probabilities for node $i = 3$ at time $t = 3$ are presented.

	Case 1		Case 2		Case 3	
	DMP	MC	DMP	MC	DMP	MC
$P_S^i(t)$	0.22111846	0.22111299	0.0152	0.0152037	0.06912	0.06908835
$P_A^i(t)$	0.02838733	0.02837872	0.02859	0.02859528	0.11808	0.1180704
$P_B^i(t)$	0.07848602	0.07845333	0.67248	0.67248319	0.12416	0.12412187
$P_{AB}^i(t)$	0.67200819	0.67205496	0.28373	0.28371783	0.68864	0.68871938

$$P_S^i(t) = P_S^i(0) \prod_{k \in \partial i \setminus j} \theta_{A,B}^{k \to i}(t, t), \qquad (17)$$

where $\theta_{A,B}^{k \to i}(t, t)$ is an aggregated dynamic message defined through the fundamental messages on time trajectories:

$$\theta_{A,B}^{k \to i}(t, t) = \sum_{\tau_k^A} \sum_{\tau_k^B} \prod_{t'=0}^{t-1} (1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t'])$$

$$\times \prod_{t''=0}^{t-1} (1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']) m_{T_A, T_B}^{k \to i}(\tau_k^A, \tau_k^B). \quad (18)$$

From the definition of $\theta_{A,B}^{k \to i}(t, t)$, it is easy to "read off" its physical meaning: It corresponds to the probability that node k did not send activation signals A or B before time t, while i follows a fixed $(\tau_i^A, \tau_i^B) = (*, *)$ dynamics; i.e., it does not activate until time t (see Appendixes A and D for more details). This case conveys the following meaning to the expression (17): $P_S^i(t)$ is given by the probability that i is in status S at the initial time, times the probability that none of its neighbors has activated it with any of the processes until time t (which exactly factorizes over neighbors on a tree graph).

In a similar way, the marginals corresponding to statuses A and B can be expressed as activations by time t:

$$P_A^i(t) = \sum_{t' \leq t} \mu_A^i(t'), \qquad (19)$$

$$P_B^i(t) = \sum_{t' \leq t} \mu_B^i(t'), \qquad (20)$$

where reduced marginals $\mu_A^i(t')$ and $\mu_B^i(t')$ are defined as follows:

$$\mu_A^i(t) = \sum_{\tau_i^B} m_{t,t}^i(t, \tau_i^B) = P_A^i(t) - P_A^i(t-1), \quad (21)$$

$$\mu_B^i(t) = \sum_{\tau_i^A} m_{t,t}^i(\tau_i^A, t) = P_B^i(t) - P_B^i(t-1). \quad (22)$$

Finally, using the normalization of probabilities [notice that, by definition, $P_{AB}^i(t)$ is contained in both $P_A^i(t)$ and $P_B^i(t)$], we finally get

$$P_{AB}^i(t) = P_A^i(t) + P_B^i(t) + P_S^i(t) - 1. \qquad (23)$$

The exact forms of the DMP equations for collaborative processes, along with detailed derivations, are provided in Appendix D. Because of the exploitation of the structure properties, the DMP equations have a much lower computational complexity $O(|E|T^2)$ for a final observation time T compared to the DBP equations while still providing exact predictions on tree graphs. We numerically verify this fact in a number of instances as shown in Table II.

In the search for simplified equations that are easier to use for inference and optimization purposes, as well as reduced computational complexity in time, we implement a strategy similar to the one used for the mutually exclusive scenario and derive approximate equations that could be used in lieu of exact DMP equations on a general graph. To do so, we notice that the special case of transmission probabilities $\alpha_{ij}^{AB} = \alpha_{ij}^A$ and $\alpha_{ij}^{BA} = \alpha_{ij}^B$ for collaborative dynamics (12)–(15) corresponds to noninteracting spreading processes: Activation by one process does not change the activation dynamics for the other. In this case, the DMP equations should simplify into the product of two independent SI-like processes:

$$m_{t,t}^i(\tau_i^A, \tau_i^B) = \mu_A^i(\tau_i^A) \mu_B^i(\tau_i^B) \qquad (24)$$

$$\text{if} \quad \alpha_{ij}^{AB} = \alpha_{ij}^A \quad \text{and} \quad \alpha_{ij}^{BA} = \alpha_{ij}^B. \qquad (25)$$

We use this observation to produce a simplified version of DMP equations, expanding the exact equations for interacting spreading processes around the noninteracting point. We keep certain first-order corrections in the update equations only, so the resulting equations are similar to the SI-type equations. This approximate version is expected to be good as long as α_{ij}^{AB} and α_{ij}^{A} and α_{ij}^{BA} and α_{ij}^{B} are similar. It would clearly break down if $\alpha_{ij}^{A} = 0$ or $\alpha_{ij}^{B} = 0$ while the other infection probabilities remain finite. The full derivation and expressions for the approximate DMP equations are provided in Appendix E.

III. INFERENCE USING APPROXIMATE DYNAMIC MESSAGE-PASSING EQUATIONS

In Sec. II, we considered DBP and DMP equations that are exact on tree graphs, which follows from their derivation and supporting numerical checks. In this section, our goal is to numerically establish the validity of approximate DMP equations introduced in the previous section.

These equations enjoy an improved computational complexity and a simpler algebraic form, and are expected to provide a good approximation in the regimes discussed above.

To validate the approximate DMP equations obtained for competitive or collaborative processes, we test the accuracy of the inferred node values against numerical results obtained via Monte Carlo simulations. Testing is carried out on both synthetically generated networks and real instances.

A. Inference in competitive spreading

Validation is carried out on two synthetic networks generated by the package NetworkX, a tree network and a network with loops, and on a real-world undirected benchmark Polbooks [81] network. The latter is provided as an example of a sparse network. With the given initial condition, we apply both Monte Carlo simulation and the DMP method to all models.

Exhaustive numerical experiments reveal that DMP-based inference provides accurate marginal posterior probabilities for the variable statuses, with the exception of very-high-infection parameter values (very close to 1). Therefore, we do not examine cases with extreme infection parameter values in the examples provided. The location of seeds initializing the processes and target nodes to be observed are chosen randomly and have no significance.

In Fig. 1, we show three synthetically generated networks: (a) a toy tree network of 10 nodes, (b) a network of 10 nodes with loops, and (c) the Polbooks [81] network with 105 nodes. The choice of network has no significance; it is a standard benchmark network used in the literature (of books about U.S. politics sold by Amazon) [81]. The network comprises two sparsely connected hubs, where edges represent books (vertices) bought jointly by the same individuals. To validate the efficacy of DMP in modeling competitive scenarios, we compare results obtained from running Eqs. (A4)–(A9) against results obtained using Monte Carlo simulations. Simulations are carried out 10 times for gathering statistics; each round includes 10^3 samples per node (about 10^5 samplings in total, depending on the network size). The parameters used in the toy model tests are $\alpha_A = 0.3$ and $\alpha_B = 0.7$, and we observe the marginal posterior probabilities $P_A^{i=3}(t)$ in both Figs. 2(a) and 2(c), and $P_S^{i=3}(t)$ in Figs. 2(b) and 2(d) for the tree and loopy graphs, respectively. The seeds initializing the processes are placed on nodes 7 for process A and on nodes 2 for process B. The choice of these particular nodes is arbitrary and has no significance. The results obtained show excellent agreement between theory and simulations. We also test the accuracy of the method on the benchmark Polbooks network as shown in Figs. 2(e) and 2(f) for the parameters $\alpha_A = 0.2$ and $\alpha_B = 0.2$. In this case, process A starts from nodes 1 and 2, and process B from nodes 4 and 37 (again, both are arbitrary choices). The observed probabilities $P_A^{i=0}(t)$ in Fig. 2(e) and $P_S^{i=0}(t)$ in Fig. 2(e) show good agreement between theory and simulations.
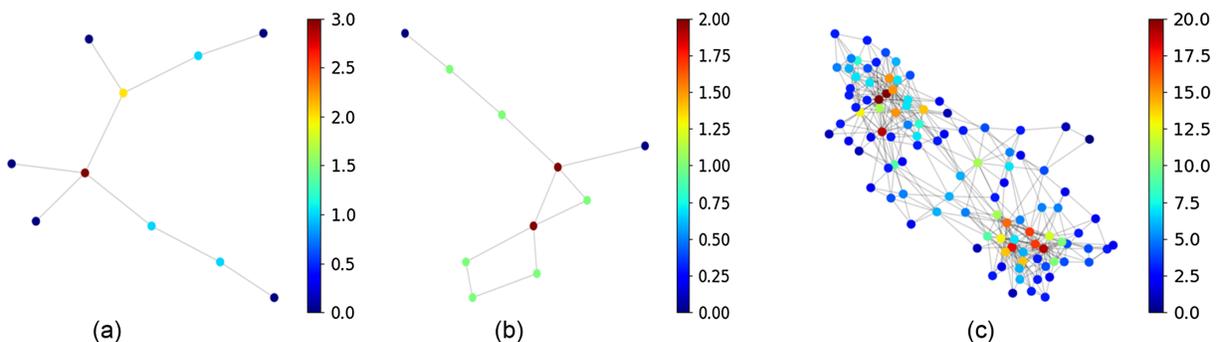

FIG. 1. Networks used for validation. (a) Toy tree network of 10 nodes. (b) Toy network with loops of 10 nodes. (c) Polbooks network [81] with 105 nodes. The color scale represents the degree of nodes.
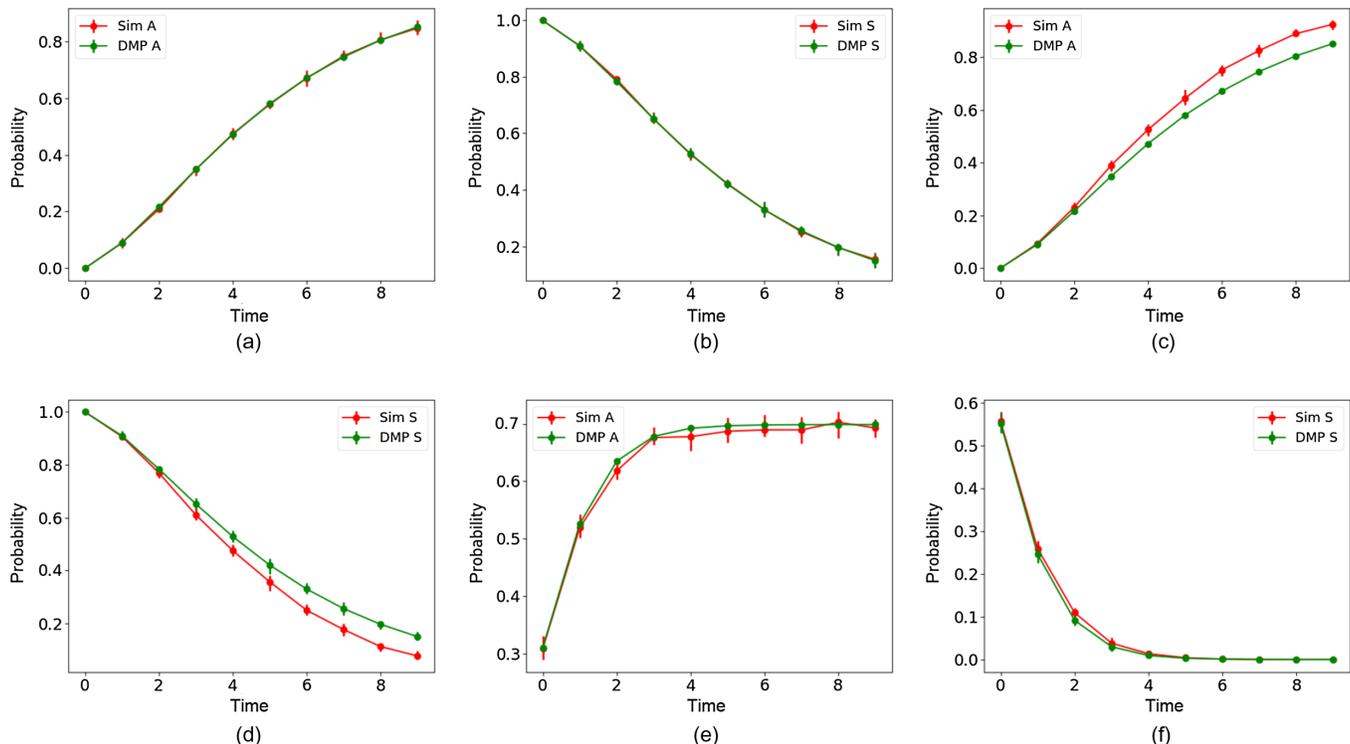
FIG. 2.   Comparison of DMP-based results and Monte Carlo simulations for competitive processes. For each of the graphs, 10^5 samples have been used, and both mean values and error bars are shown, unless they are smaller than the symbol size. (a) Competitive process on a treelike network using the parameters $\alpha_A = 0.3$, $\alpha_B = 0.7$ and observing $P_A^{i=3}(t)$. The seeds initializing the processes are placed in nodes 7 for process A and in nodes 2 for process B. (b) Same as in panel (a) but observing $P_S^{i=3}(t)$. (c) Comparing results from the DMP method and Monte Carlo simulation for the network with loops using the same parameters as in panel (a) and observing $P_A^{i=3}(t)$. (d) Same as panel (c) but observing $P_S^{i=3}(t)$. For the Polbooks experiment, we observe (e) $P_A^{i=0}(t)$ and (f) $P_S^{i=0}(t)$; the parameters used are $\alpha_A = 0.2$ and $\alpha_B = 0.2$, and the seeds initializing the processes are placed in nodes 1 and 2 for process A and in nodes 4 and 37 for process B.

B. Inference in collaborative spreading

Similar experiments were run for a collaborative process on the toy tree network: graphs with loops and the benchmark football network. The latter is an undirected network of American football games between colleges during the 2020 fall season [82], which are quite uniformly connected.

The results are shown in Fig. 3 for the various cases.

The experimental results indicate that modeling based on approximate DMP equations is very accurate on treelike networks, as expected for message-passing algorithms. It is less accurate on small loopy graphs at longer times, as expected, because of the small loops that violate the cavity method's assumption (the specific 10-node network used includes two small loops). This effect is suppressed to some extent on larger networks, where loops are typically longer as demonstrated in the real benchmark network examples.

As discussed in Appendix E, the approximation quality is expected to degrade when α_A and α_B are very different from α_{AB} and α_{BA}, respectively. Indeed, in this case, the prediction of the dynamics by the approximate DMP equations can become inaccurate. To illustrate this point,

let us consider an extreme example on a chain of three nodes and two links, with node 2 connected to nodes 1 and 3. Let us assume that $\alpha_A = 1$, $\alpha_B = 0$, and $\alpha_{AB} = \alpha_{BA} = 1$, i.e., that the infection B can be transmitted only to nodes that are already infected with A. Interestingly, this scenario is relevant for several disease pairs, such as hepatitis D, which can only be transmitted to individuals already infected with hepatitis B. Given the initial conditions $P_A^1(0) = 1$, $P_S^2(0) = 1$, and $P_B^3(0) = 1$, node 1 will become infected with disease B at time $t = 3$: It first infects node 2 with process A at time $t = 1$; this leads to infection of node 2 by infection B coming from node 3 at time $t = 2$; and finally, node 2 transmits infection B to node 1 at time $t = 3$. This example represents an extreme case where the approximate DMP equations are not valid and indeed preclude node 1 from being infected by process B, which illustrates that they may not be exact even on tree graphs when the approximation criterion is not satisfied. However, it is easy to check that exact DMP equations provide an accurate answer in this case as well, as it should. This counterexample reiterates the trade-off between the exactness and the computational complexity between exact and
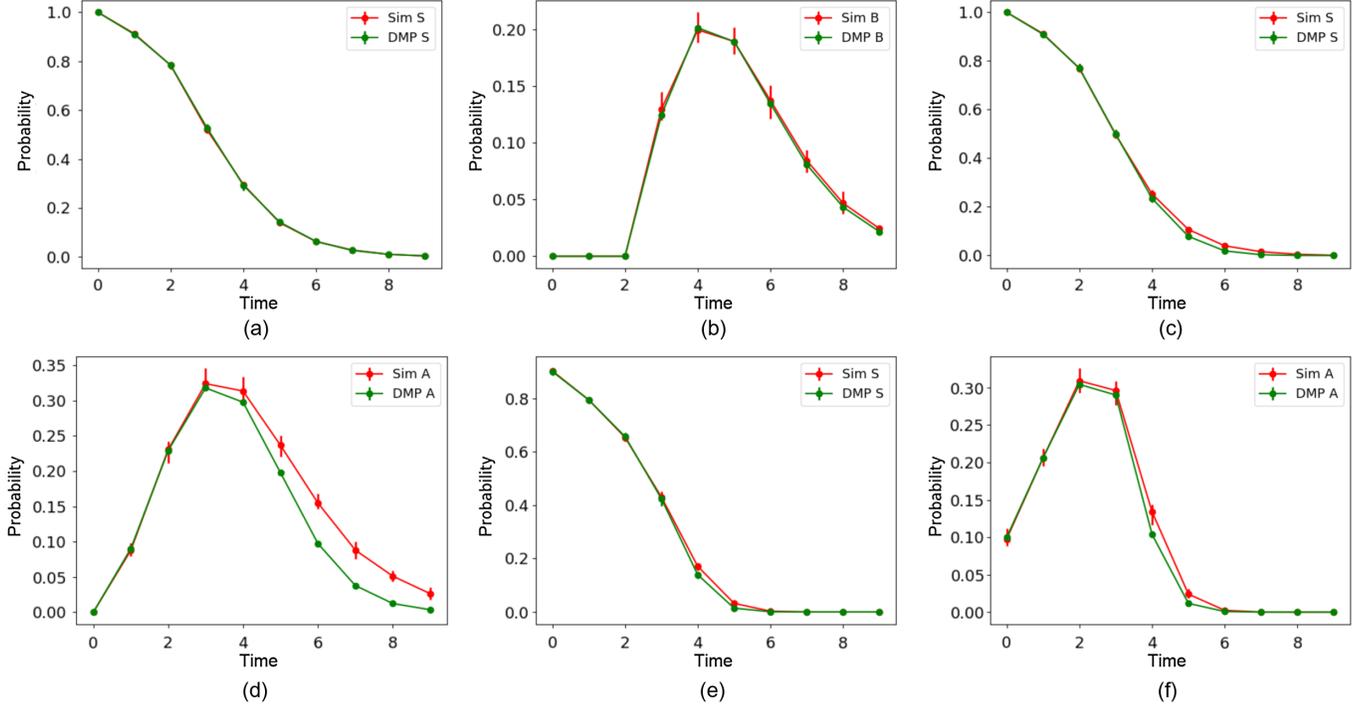
FIG. 3.   Comparison of DMP-based results and Monte Carlo simulations for collaborative processes. For each of the graphs, 10^3 samples have been used per node (about 10^5 samples in total), and both mean values and error bars are shown (unless they are smaller than the symbol size). (a) Collaborative process on a treelike network using the parameters $\alpha_A = 0.3$, $\alpha_B = 0.7$, $\alpha_{AB} = 0.6$, and $\alpha_{BA} = 0.6$, observing $P_S^{i=3}(t)$ (the node to monitor has been selected arbitrarily). Process A was seeded at node 7 and process B at node 2. (b) Same as in panel (a) but observing $P_B^{i=3}(t)$. (c) Comparing results for the network with loops using the same parameters as in panel (a), observing $P_S^{i=3}(t)$ and (d) $P_A^{i=3}(t)$. (e) Comparing results obtained from the DMP method and Monte Carlo simulations of a collaborative process on the football network. The parameters used are $\alpha_A = 0.1$, $\alpha_B = 0.2$, $\alpha_{AB} = 0.3$, and $\alpha_{BA} = 0.4$. The probabilities represent observations of node 2, while process A was seeded at nodes 3 and 4, and process B at nodes 0 and 1 (there is no significance to any of these choices).

approximate DMP equations discussed in the previous section. However, as illustrated in numerical examples, in applications with plausible parameters, one could expect that the approximate DMP method provides a good description for cooperative spreading processes on sparse networks.

IV. DMP-BASED OPTIMIZATION METHOD

Competition and collaboration of spreading processes on graphs can be optimized through the judicious use of resources. We demonstrate how managing a spreading process against an adversarial competing agent can be optimized within a given time frame and how joint collaborative processes can be affected through the best use of resources. The latter can take the form of spreading maximization while making use of the process interdependencies, or of containment through vaccination to impact on the spread of both processes.

We outline a general procedure for optimization in this section. Details for the specific optimization problems we address here are given in Appendix F for competitive processes and in Appendix H for collaborative processes.

The core approach for optimization is based on a discretized variational method, whereby a functional over a time window (Lagrangian) is optimized through changes in control parameters throughout the time interval. The dynamics, resource constraints, initial conditions, and other restrictions on the parameters used are enforced through the use of Lagrange multipliers. A similar method is used in optimal control.

We denote the components of the Lagrangian function used in a way similar to [76]:

$$\mathcal{L} = \underbrace{\mathcal{O}}_{\text{objective}} + \underbrace{\mathcal{B} + \mathcal{P} + \mathcal{I} + \mathcal{D}}_{\text{constraints}}, \qquad (26)$$

where $\mathcal{L}$ is the Lagrangian function, $\mathcal{O}$ is the objective to be optimized, $\mathcal{B}$ is the budget or constraints on the resource used, $\mathcal{I}$ represents the component that forces initial conditions, $\mathcal{P}$ are restrictions on the probabilities used, and $\mathcal{D}$ represents the dynamical constraints that take the form of approximate DMP equations. All of the terms $\mathcal{B}$, $\mathcal{I}$, $\mathcal{P}$, and $\mathcal{D}$ are forced through the use of Lagrange multipliers.

For different problems, the constraints and objectives vary. We take the competitive process as an example. In this problem, we want to find an optimal allocation of a limited number of spreaders (representing a budget, potentially time dependent) for process A, which minimizes the spreading of process B. These objectives could represent a competition in a political or commercial setting. The objective function in this case is

$$\mathcal{O} = \sum_i (1 - P_i^B(T)), \tag{27}$$

where T is the end of the set time window. Our goal is to maximize this objective function, thus minimizing the spread of process B. The resources at our disposal for seeding nodes with process A are represented by the budget constraint at time zero (although more elaborate budget constraints could be accommodated as in Ref. [76] and in some of the examples that follow):

$$B_\nu = \sum_i \nu^i(0), \tag{28}$$

where $\nu^i(0)$ is the deployment of a fraction of the budget for process A on node i. The budget constraint is forced through the Lagrange multiplier λ^{bu}, such that

$$\mathcal{B} = \lambda^{\text{bu}}\left(B_\nu - \sum_i \nu^i(0)\right). \tag{29}$$

The fraction or probability ν^i is kept within a certain range, determined by the upper and lower bounds $\bar{\nu}$ and $\underline{\nu}$, respectively. This process is also enforced using a Lagrange multiplier

$$\mathcal{P} = \epsilon \sum_i \left(\log(\bar{\nu} - \nu^i(0)) + \log\left(\nu^i(0) - \underline{\nu}\right)\right). \tag{30}$$

The term $\mathcal{D}$ is given by enforcing the approximate DMP equations for the competitive case using a set of Lagrange multipliers, as detailed in Appendix H for the competitive case. The remaining term $\mathcal{I}$ forces the initial conditions for the dynamics.

The extremization of the Lagrangian (26) is performed as follows. Variation of $\mathcal{L}$ with respect to the dual variables (Lagrange multipliers) results in the DMP equations starting from the given initial conditions, while derivation with respect to the primal variables (control and dynamic parameters) results in a second set of equations, coupling the Lagrange multipliers and the primal variable values at different times. Ending conditions for the forward dynamics provide the initial conditions for the backward dynamics.

We solve the coupled system of equations by forward-backward propagation, a widely used control method detailed in Ref. [76]. This method has a number of advantages compared to other localized optimization procedures. It is simple to implement, of modest computational complexity $O(ET)$, where E is the number of edges in the graph and T the time window, and does not require any adjustable parameters. The forward-backward optimization provides resource (budget) values to be placed at time zero (or at any time within the time window if we so wish) in order to optimize the objective function, e.g., that of Eq. (27). One potential drawback of the method is the possible nonconvergence of the dynamics to an optimal solution. This drawback can be mitigated, to some extent, by solving the equations for the backwards dynamics using other available solvers and by storing the best solutions found over time, or approaching a fixed point via gradient descent. In general, as the functions used become more nonlinear, it will become more difficult to obtain optimal solutions, although we have not experienced significant problems in the cases studied here.

V. NUMERICAL STUDY OF THE OPTIMIZATION ALGORITHM

To validate and demonstrate the efficacy of the optimization method, we carry out experiments on both synthetic and realistic networks. Before embarking on a large-scale application, we study the performance of the derived method on a treelike network of 30 nodes.

A. Validation of the optimization algorithm

To validate the DMP-optimization algorithm on a problem that could be exhaustively studied and intuitively presented, we restrict the study to a small, exemplar, treelike synthetic model. Moreover, we select a small number of nodes (3) on which resources could be deployed. The objective is to maximize the spread of both agents or to minimize the spreading of one of them (the disease-control scenario) in both competitive and collaborative processes.
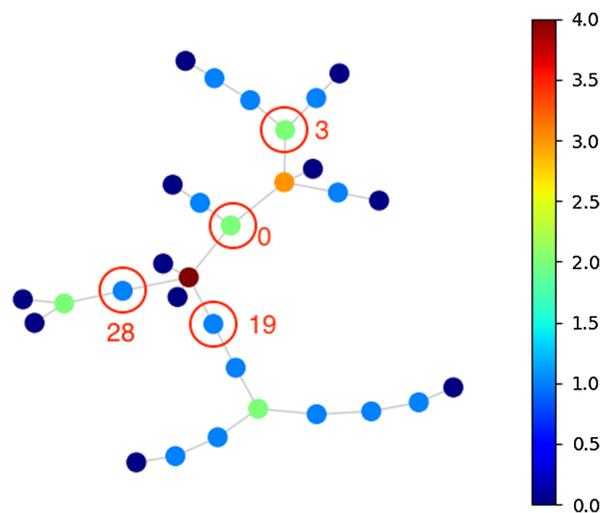

FIG. 4. Tree network of 30 nodes used for carrying out the experiments. The color scale represents the degree of nodes. Controllable nodes in the various experiments are marked.

The 30-node network used for carrying out these experiments is presented in Fig. 4. Comparison between the DMP-based optimization algorithm and the exhaustive search is implemented in the following way: We consider a scenario where the entire budget is available at time $t = 0$. The optimization problem minimizes the spreading of process B in a competitive process through judicious budget allocation of the seeds for process A; i.e., we aim at minimizing $\sum_i P_i^B(T)$, where T is the end time of the process.

In the experiments, resources for process A were deployed on three nodes: 0, 19, and 28 (determined by the choices for nodes 0 and 19 because of the total budget constraint). The fixed seed for process B is node 3; this choice is arbitrary and insignificant. The objective function landscape has been explored by sampling for different parameter values as denoted by the green points in Fig. 5(a). The collaborative scenario with specific infection parameters is plotted in Fig. 5(b). A different set of controllable nodes is presented in Figs. 5(c) and 5(d) for the competitive and collaborative cases, respectively. The maximum value obtained for the objective function is contrasted with the results obtained using the DMP-based optimization procedure, marked by the red dot, in all cases.

(a) Competitive process – minimizing the spreading of process B.

(b) Collaborative process – minimizing the spreading of process B.

(c) Competitive process – minimizing the susceptible nodes.

(d) Collaborative process – minimizing the susceptible nodes.
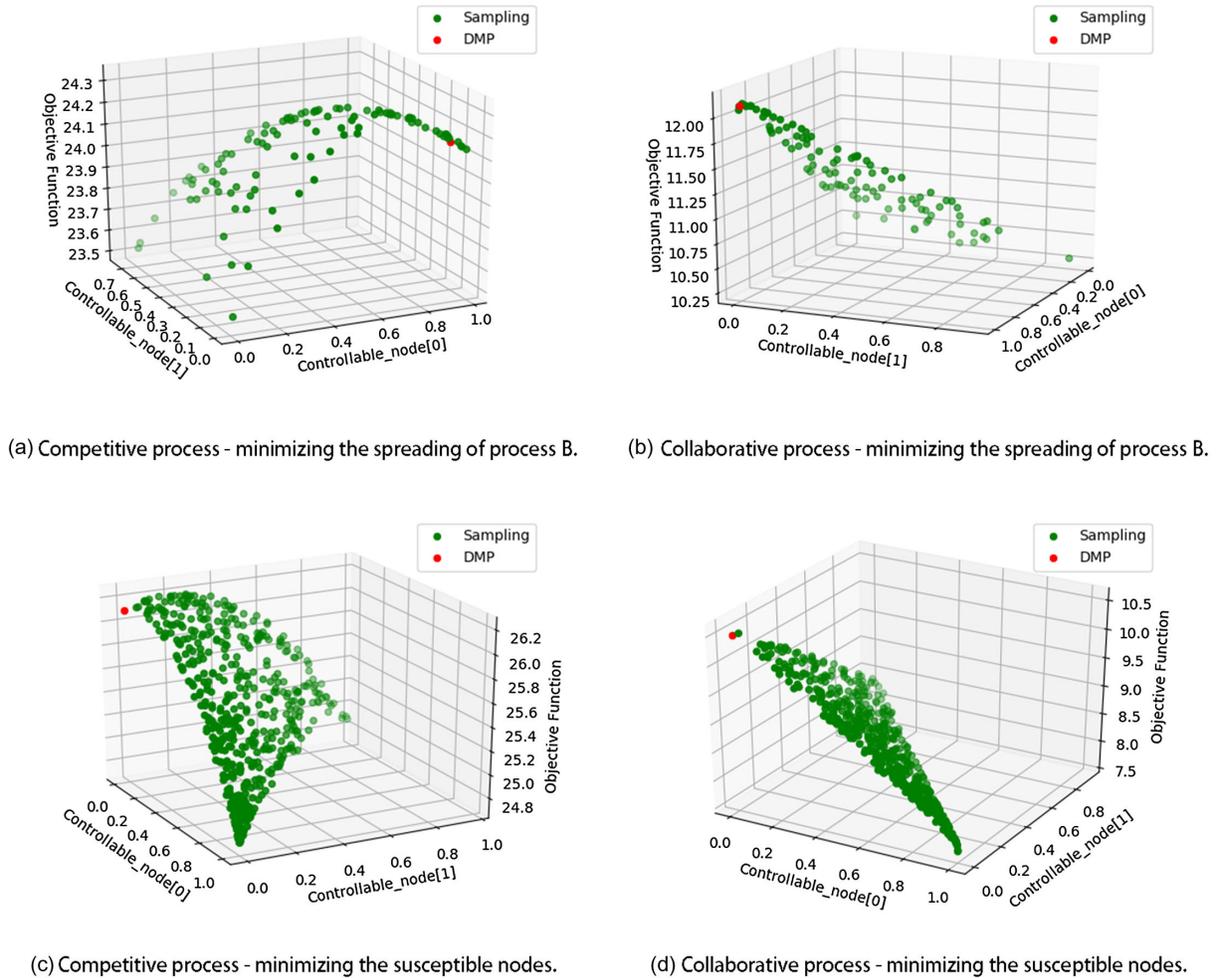
FIG. 5. Comparison of results obtained using the DMP-based optimization method and random sampling in parameter space. (a) Competitive process with a time window $T = 3$ and infection probabilities $\alpha_A = 0.5$ and $\alpha_B = 0.5$. Sampled values for the various control parameters are marked by green points, while the DMP-optimal value is marked by a red point. Nodes 0, 28, and 19 were infected by process A as controllable nodes, with a total budget of one, while the fixed seed for process B is node 3. The objective, in this case, is to contain the spread of process B, namely, minimize $\sum_i P_i^B(T)$. The DMP-based optimization maximizes the objective function as 24.17, while the optimal sampling result is 24.1. (b) Similar experiment to panel (a), with the same conditions and objective, for the collaborative process with double infection parameters $\alpha_{AB} = 0.8$ and $\alpha_{BA} = 0.8$. The DMP-optimal objective value is 12.11 against 12.1 from sampling. (c) Competitive process using the time window and infection parameters as in panel (a) but a different objective function—maximizing the infection $\sum_i (1 - P_i^S(T))$. Controllable nodes are 6, 8, and 15, and node 3 is infected by process B. The optimal DMP objective function value is 26.11 against 26.2 from sampling. (d) Similar experiment to panel (c) for the collaborative process with double infection parameters as in panel (b). The DMP-optimal result is 10.32 and 10.3 from sampling.

It is clear that there is good agreement between the DMP-optimal values obtained and the optima discovered through sampling, although in some of the cases they are not identical. The corresponding objective function values for Fig. 5 are denoted in the various subfigures.

B. Results on real networks

1. Competitive processes

We study the performance of our optimization algorithm on networks assuming that all resources are available at time zero. Other optimization strategies where resources' availability is time dependent could also be considered [76], as shown later. One of the problems in comparing the performance of our method to other approaches is the limited number of dedicated, competing optimization methods for the multiple-agent scenario, especially in cases where resource availability and its deployment are spread over the whole time window. Therefore, we compare the DMP-optimized spreading process with known heuristics for the single-agent scenario such as a uniform allocation of resources over all nodes (except seed nodes induced with process B), the high degree deployment strategy [83] HDA, and K-shell decomposition [64]. "Free spreading" refers to an uncontrolled spread of process B. "Blocking" refers to the allocation of a competing agent that is noninfectious ($p = 0$). In all cases, resources are used to contain the spreading of process B.

Multiagent spreading processes exist in different settings—from social networks, energy grids, and road networks to the graph of random interactions between individuals. There is no specific network type that is more relevant to the scenarios we examine. Therefore, we choose a set of sparsely connected benchmark networks of different characteristics to test the efficacy of our algorithm compared to competing approaches. The description of the different networks and their specific properties appear in the corresponding references. We aim to show that the suggested methods work well on all of the sparse networks examined, and we therefore expect it to work well on most other sparse networks.

In these experiments, we allocate seeds of process B on $0.05N$ nodes at time zero, where N is the total number of nodes in the network. The same amount of resource is allocated to the controlled competing process A. Infection parameters are $\alpha_A = 0.2$ and $\alpha_B = 0.3$. This choice of parameters is arbitrary, and the performance for other parameter choices provides qualitatively similar results. The free parameter that forces the upper or lower limits of the resource variables (F2) is set to $\epsilon = 0.1$ initially and decays exponentially with iteration steps. Experimental results show that decaying ϵ in this manner leads to an improved performance, arguably since it allows one to obtain solutions that are closer to the limit values. The optimization procedure is iterated 10 times, and the best result is selected. The objective is to minimize the spreading of process B, and the normalized total spreading $\sum_i P_B^i(T)/N$ at time $T = 3$ is shown in Table III. The short time window used is due to the small diameter of the networks.

In competitive processes, we observe that the containment of process B is carried out effectively by optimal deployment of spreading agents of process A, compared to static blocking, HDA deployment, K-shell, and uniform seeding. This scenario corresponds to marketing, rumor spreading or fake news, and opinion setting. We also evaluate the improvement obtained for a given budget in blocking the spread, which allows one to allocate the appropriate budget for containment (e.g., addressing the spread of fake news or antivaxxing rumors by releasing verifiable information). Clearly, key factors in determining the spread are the actual infection parameters associated with the various processes, which can be obtained through data analysis. The football network is highly connected, and we speculate that this is the reason for the success of uniformly allocating the spreading agents.

When the budget allocation is performed dynamically at different times, several optimization procedures can be used. Conventional stochastic optimal control [87] is based

TABLE III. Comparing different deployment methods for a competitive scenario on various networks. In each network, we randomly choose $0.05N$ nodes as seeds of process B and the same total budget for the competitive process A, where N is the number of nodes in the network. The infection parameters are $\alpha_A = 0.2$ and $\alpha_B = 0.3$. The initial parameter ϵ, which forces the budget limits per node as in Eq. (F2), is initially set to 0.1 but decays exponentially with the iteration steps. The optimization procedure is iterated 10 times, and the best result is selected. The objective is to minimize the spreading of process B; the normalized total spreading $\sum_i P_B^i(T)/N$ at time $T = 3$ is shown. In all methods, budgets are available at time zero. Uniform allocation assumes that all the budgets are uniformly allocated to all nodes at time zero (except those infected with process B). In free spreading, no agents are allocated for process A. In blocking, competitive agents are not infectious ($p = 0$); all the budget is used to contain the spreading of B. The best results are denoted by bold font.

Network	Number of nodes	DMP	Uniform	K-shell	HDA	Blocking	Free spreading
Football [82]	115	0.5829	**0.5674**	0.8264	0.7731	0.7860	0.8264
Lesmis [84]	77	**0.0834**	0.2211	0.1962	0.0990	0.1016	0.3549
Karate [85]	35	**0.4206**	0.4902	0.5470	0.4212	0.4966	0.5472
Power [86]	4941	**0.0500**	0.0621	0.0641	0.0632	0.0501	0.0644
Polbooks [81]	105	**0.2243**	0.3733	0.3102	0.2779	0.4316	0.5744

on planning ahead for the entire time window, taking into account future uncertainties. An optimal solution in discrete time can be stated through a solution of the Bellman equation that, in our setting, would result in an algorithm with a high computational complexity due to the curse of dimensionality [88]. To mitigate this issue, we adopt a procedure that is similar to the one used in the closed loop control setting, which updates the resource allocation at every step based on the information on the realization of the stochastic dynamics. This approach does not strictly guarantee the solution optimality, but it quantitatively takes into account realization of uncertainties while keeping computational complexity under control. We carry out the optimization for the end time T at each step, on a shrinking time window, incorporating the newly available information. We consider a case where one process spreads randomly, while for the other, a unit budget is available per time step, to be optimally deployed. Given the information at hand for each time step, one may then consider the following dynamic allocation strategies: (a) The DMP-greedy strategy deploys the unit to optimize the objective function for the next time step; (b) the DMP-optimal strategy optimizes the objective function for the end time T while incorporating information available at each time

step. Note that the latter represents a closed-loop-like optimization in the sense that updated information on realization of dynamics at each time step is incorporated to produce a better resource allocation plan.

To demonstrate the efficacy of our method and the differences between the DMP-greedy and DMP-optimal resource deployment, we use the football network [82] and infect node 1 at time 0 (blue point to the left of center; total budget for B is 1); we then optimally deploy a budget of 1 for process A at each time step $t = 1, 2, \ldots, 5$. The infection parameters used are $\alpha_A = \alpha_B = 0.7$. The results are shown in Fig. 6, where the heat bar represents the dominating process per node through the value $P_A^i(t) - P_B^i(t)$. Red and blue represent dominating processes A and B, respectively. It is clear that the DMP-optimal process (b) is much more effective than the DMP-greedy one (a) in restricting the spread of process B by maximizing the spread of process A. Numerical comparisons between the two methods for the same network and conditions are presented in Table IV. It is clear that while the DMP-greedy process is successful at earlier time steps, the DMP-optimal process minimizes the spread of process B at $T = 5$. A second example on a more densely connected network is given in Appendix G.
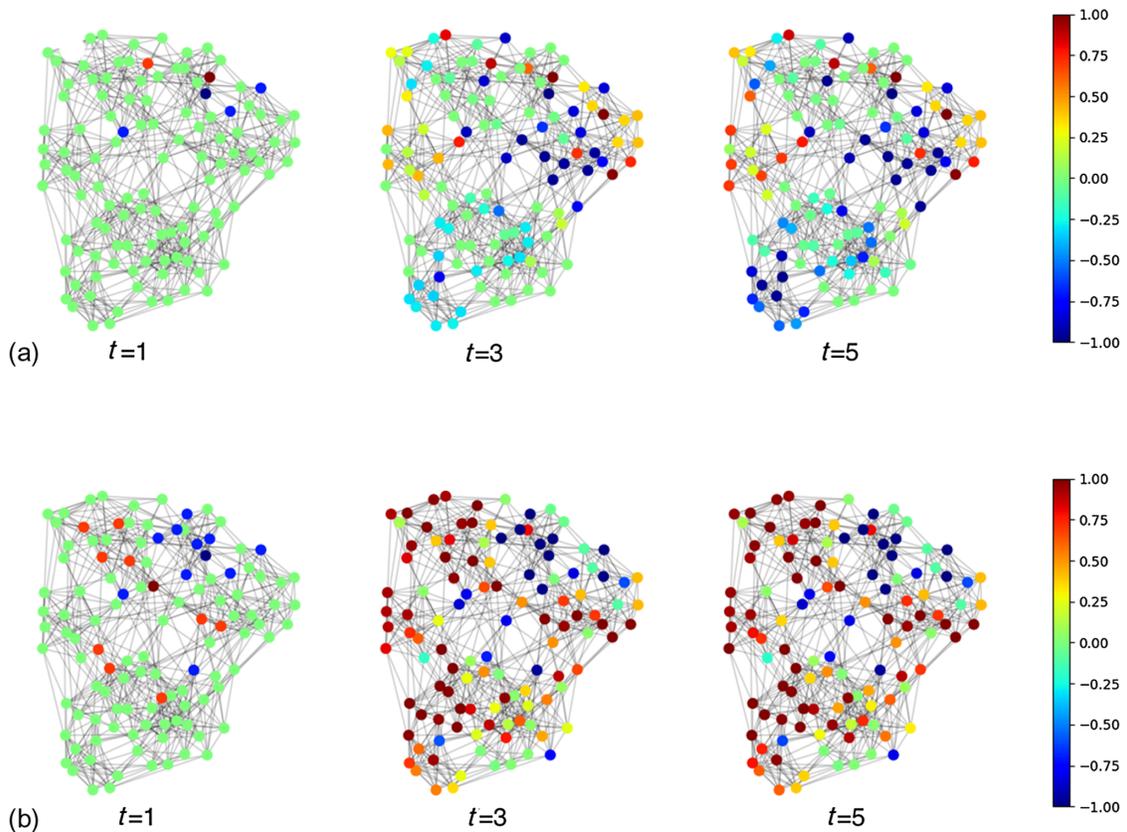
FIG. 6.   Football network with infection parameters $\alpha_A = \alpha_B = 0.7$. The budget for B is 1, allocated on node 1 at $t = 0$, and a budget of 1 per time step is optimally assigned for process A. The figures represent the containment of process B at different times $t = 1, 2, \ldots, 5$ due to the judicious allocation of resources of process A. (a) DMP-greedy strategy; (b) DMP-optimal strategy. The heat bar represents the dominating process per node through the value $P_A^i(t) - P_B^i(t)$. Red and blue represent dominating processes A and B, respectively.

TABLE IV.   DMP-based resource deployment methods for the football network under the same conditions as in Fig. 6. The objective is to minimize the spreading of B, i.e., minimize the fraction of B nodes at time $T = 5$, $\sum_i P_B^i(T)/N$.

Time	DMP optimal	DMP greedy
$t = 1$	0.0767	**0.0556**
$t = 2$	0.2157	**0.1871**
$t = 3$	**0.3109**	0.4336
$t = 4$	**0.3211**	0.5615
$t = 5$	**0.3211**	0.5628

To demonstrate the improvement in reducing the spread of agent B given a budget b_B against an optimally deployed budget b_A of process A, we plot the ratio of infected A and B statuses at time $T = 5$ against the budget ratio b_A/b_B. This process has been used for the Lesmis network [84] (77 nodes, representing the coappearance of characters in the novel *Les Miserables*) with infection probabilities $\alpha_A = \alpha_B = 0.5$ and time window $T = 5$. The results shown in Fig. 7(a) have been averaged over five instances for both uniform and optimal DMP-based deployment. It is clear from the figure that the optimal DMP-based deployment provides much better results than uniform deployment, which exhibits a linear increase. The saturation for high ratios is limited by the size of the graph.

We observe several instances in which the ratio of infected probabilities exhibits a fast transition at specific points. Figure 7(b) shows a fast transition towards a process-A-dominated network as the ratio between the budgets allocated exceeds a certain value $b_A/b_B \approx 0.28$. The y axis represents the ratio of process probabilities $\sum_{i=1}^N P_A^i(T)/\sum_{k=1}^N P_B^k(T)$. These results clearly depend

on the topology, budgets, and infection rates used. The example given here is based on the Lesmis network with infection probabilities $\alpha_A = 0.5$ and $\alpha_B = 0.7$, fixed budget $b_B = 1$, time window $T = 5$, and optimal DMP-based deployment.

To evaluate the interplay between infection parameter values and budget allocated to each of the processes, we study a competitive case on the Lesmis network, with a varying ratio between budgets ($b_B = 1$, $b_A = 0.5, …, 4.5$) and infection probabilities ($\alpha_B = 0.7$, $\alpha_A = 0.1, …, 0.7$). The points where the processes end up with equal probabilities at $T = 5$ are plotted in Fig. 7(c) for the DMP-optimized deployment of resource A (green line) and for uniform deployment (red line). The results are averaged over five randomly chosen initial positions for the seed of B. From the figure, it is clear that, in this case, DMP-optimized deployment can effectively mitigate significantly inferior infection rates or budget ratios (area above the green curve); only when both ratios are very low does the B process dominate the network after T steps (yellow area). Uniform deployment results in much inferior performance (area above the red line).

2. Collaborative processes

Results obtained for collaborative scenarios, shown in Table V, exhibit a similar behavior, showing that collaborative processes can be optimized to spread quickly due to the mutually supportive role played by the two processes. In this case, some nodes ($0.05N$) are infected by process B, and we allocate a given budget of process A such that the joint spreading will be maximized and the number of noninfected nodes (in status S) minimized. This case could represent, for instance, the spread of opinions on the basis
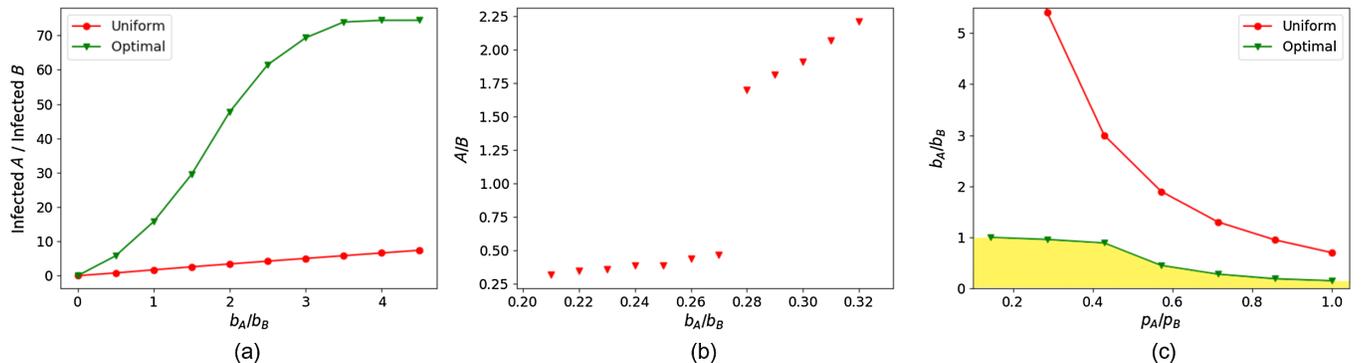

FIG. 7.   Optimized competitive scenario on the Lesmis network. (a) Ratio of nodes infected by processes A and B after $T = 5$ time steps for a given initial budget ratio b_A/b_B. The infection probabilities used are $\alpha_A = \alpha_B = 0.5$. The green curve represents the ratio in the case of DMP-optimized deployment of the A budget, while the red curve represents the uniform deployment case. (b) Fast transition in the ratio of expected infected statuses $\sum_{i=1}^N P_A^i(T)/\sum_{k=1}^N P_B^k(T)$ for infection probabilities $\alpha_A = 0.5$ and $\alpha_B = 0.7$ and time window $T = 5$. The budget for B is fixed at $b_B = 1$, and the budget deployment for A is optimized using DMP. (c) Interplay between infection parameter values and budget allocated to each of the processes on the Lesmis network. The x axis represents the ratio between infection parameters α_A/α_B and the y axis the ratio between budgets b_A/b_B. The green line represents values for which the two processes have equal probabilities in the network at $T = 5$ when DMP-optimized deployment of resource A is used; the red line represents the same line for uniform deployment. Results are averaged over five randomly chosen initial positions for the seed of B.

TABLE V.   Comparing different resource-allocation approaches for collaborative processes on various networks. On each network, we randomly select $0.05N$ nodes as seeds of process B and the same total budget for process A to be optimally allocated, where N is the total number of nodes in the network. The infection parameters chosen are $\alpha_A = 0.2$, $\alpha_B = 0.3$, $\alpha_{AB} = 0.4$, and $\alpha_{BA} = 0.5$. The initial parameter ϵ, which forces the budget limits per node as in Eq. (F2), is initially set to 0.1 but decays exponentially with the iteration steps. The optimization procedure is iterated 10 times, and the best result is selected. The objective function, in this case, is to maximize the spreading of processes A and B, i.e., minimize the fraction of susceptible nodes $\sum_i P_S^i(T)/N$ at time $T = 3$. In DMP allocation-based optimization, we assume the seeding budget for B is available at time zero. Uniform allocation assumes that the budget for A is initially allocated uniformly to all free nodes. HDA deployment [83] and K-shell [64] seeding are used as before; in free spreading, no collaborative spreading budget A is allocated. The lowest fraction of susceptible nodes is denoted by bold.

Network	Number of nodes	DMP allocation	Uniform allocation	K-shell	HDA	Free spreading
Football [82]	115	**0.0536**	0.0582	0.1736	0.1543	0.1735
Lesmis [84]	77	**0.2222**	0.3832	0.3151	0.3051	0.6451
Karate [85]	35	**0.2771**	0.3743	0.4527	0.2481	0.4529
Power [86]	4941	**0.7652**	0.7930	0.8434	0.9024	0.9355
Polbooks [81]	105	**0.1524**	0.2065	0.3347	0.2204	0.4255

of political affiliation. In this case, the DMP-based optimization algorithm also works well, with the exception of the football network where uniform spreading seems to be successful, arguably due to the same reasons as in the competitive case.

The final set of experiments we carry out relates to the vaccination policy in collaborative spreading processes. In this case, two collaborative processes A and B spread throughout the system, and the task is to minimize the spread through a vaccination process to one of the two, say, B. The role of vaccination in this case is in to reduce the infection probability by the level of vaccination deployed. For instance, we adopt a simplistic model whereby if the vaccine deployed at a given node is b, its probability of being infected will reduce to $\alpha_B - b$. Similarly, the

TABLE VI.   Different vaccine-allocation policies in a collaborative process on different benchmark networks. On each network, we randomly choose $0.05N$ of the nodes as seeds for process B and $0.01N$ nodes as seeds for process A; the total vaccination budget is $0.05N$, where N is the number of nodes in the network. Infection parameters are arbitrarily set to $\alpha_A = 0.4$, $\alpha_B = 0.4$, $\alpha_{AB} = 0.9$, and $\alpha_{BA} = 0.9$. The parameter ϵ in Eq. (F2) is set to 0.01. The optimization procedure is iterated 10 times, and the best result is selected. The objective is to minimize the spreading of processes A and B, i.e., maximize the fraction of susceptible nodes $\sum_i P_S^i(T)/N$ at the end of the process (time $T = 3$). In DMP-based optimal allocation, we assume that all budgets are available at time zero. Uniform allocation assumes all budgets to be uniformly allocated to all nodes at time zero. In free spreading, no vaccine is allocated. A detailed description of the vaccine allocation problem can be found in Appendix I.

Network	Number of nodes	DMP allocation	Uniformed allocation	Free spreading
Football [82]	115	**0.1887**	0.0607	0.0294
Lesmis [84]	77	**0.9455**	0.5519	0.4941
Karate [85]	35	**0.4676**	0.3491	0.3123
Power [86]	4941	**0.8891**	0.8650	0.8356
Polbooks [81]	105	**0.5483**	0.3071	0.2354

infection parameter α_{BA} will also decrease by the same amount to $\alpha_{BA} - b$. Other infection models can be easily accommodated. The optimization task is in the deployment of a given vaccination budget such that the fraction of noninfected sites $\sum_i P_S^i(T)/N$ at time T is maximized.

The experimental results shown in Table VI demonstrate that the DMP-based optimization shows excellent performance.

Generally, for the different network topologies and size, DMP-based optimization appears to be more effective than other methods. The main cases where it does not offer the best result are in complex networks with bounded connectivity fluctuation, where it has been shown that uniformly applied immunization strategies are highly effective [83].

To demonstrate the efficacy of DMP-optimal resource deployment for vaccination in the case of a collaborative spreading process, we use the main road network of England [89]. Livestock epidemics often spread through the transport of infected livestock on the road network (as was the case in the 2001 Foot and Mouth epidemic in the UK). We consider the spread of highly infectious coupled processes through the road network starting from the areas of Greater London (node 10, process A) and Leeds (node 11, process B); the budget for both A and B is one unit, and infection parameters are $\alpha_A = \alpha_B = 0.2$, $\alpha_{BA} = \alpha_{AB} = 0.99$. The process runs for 20 steps. The vaccination budget is one unit per time step and is effective against process B only while being ineffective for process A. The results shown in Fig. 8 demonstrate the efficacy of the DMP-optimal vaccination strategy aimed at minimizing $\sum_i (1 - P_S^i(t))$ (b) in contrast to the free spreading of both infections (a). Blue and red represent uninfected and infected statuses, respectively; more specifically, the node color represents $1 - P_S^i(t)$, where red and blue correspond to 1 and 0, respectively. As we can see in Fig. 8, the infection spreading around London remains the same, with or without the deployment of a vaccine, as it is mainly infected by process A; hence, the vaccination has no effect while the spread emanating from the Leeds area is effectively blocked.
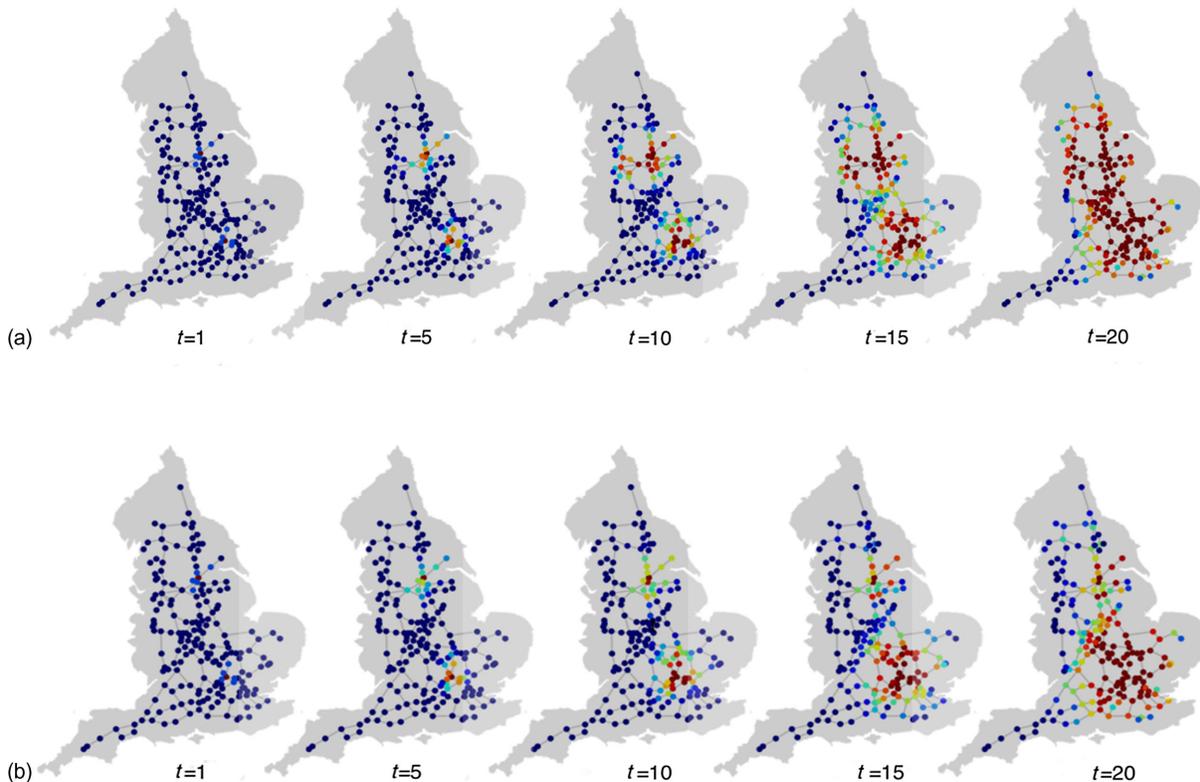
FIG. 8. England road network—the collaborative spread of highly infectious coupled processes. We consider the spread of highly infectious coupled processes through the road network starting from the areas of Greater London (node 10, process A) and Leeds (node 11, process B); the budget for both A and B is one unit, and infection parameters are $\alpha_A = \alpha_B = 0.2$, $\alpha_{BA} = \alpha_{AB} = 0.99$. The process runs for 20 steps. The vaccination budget is one unit per time step and is effective to suppress process B only while being ineffective for process A. The node color represents the value of $1 - P_S^i(t)$, where red and blue correspond to 1 and 0, respectively (the heat bar is similar to that of Fig. 6). (a) Free spread of the epidemics with no vaccination. (b) Vaccination budget of one unit per time step against process B deployed at each time step using the DMP-optimal algorithm. We see that in panel (b), the infection in the London area spreads unhindered in both (a) and (b), as it is mainly infected by process A, while spread emanating from the Leeds area is effectively blocked when the vaccination is deployed.

VI. SUMMARY AND FUTURE WORK

Competition and collaboration between spreading processes are prevalent on social and information networks and on interaction networks between humans or livestock, to name but a few. To better understand the expected spread and dynamics of diseases, marketing material, opinions, and information, it is essential to infer and forecast the spreading dynamics. Moreover, the judicious use of limited resources will help contain the spread of epidemics, win political and marketing campaigns, and better inform the public in the battle against unsubstantiated or misleading information.

Most current studies of multispreading processes, such as Ref. [34], heavily rely on computer simulations and heuristics with varying results from one instance to another. Obtaining reliable estimates therefore comes at a high computational cost in order to obtain meaningful statistics. We advocate the use of a principled probabilistic technique for the analysis and employ the same framework to develop

and offer an optimization algorithm for the best use of the resource available.

Both competitive and collaborative spreading processes are investigated here using the probabilistic DMP framework, and they provide an accurate description of the spreading dynamics, as validated on both synthetic and real networks. For the first time, we derive exact message-passing equations for two interacting unidirectional dynamic processes and study approximate DMP equations that benefit from a simpler form and a lower algorithmic complexity; this derivation represents the first major novel contribution of this work. The probabilistic description facilitates the study of multiagent spreading processes on general sparse networks at a modest computational cost, opening up new possibilities for obtaining insights into the characteristics of such scenarios.

As a second major contribution in this work, we employ a scalable optimization framework that incorporates the DMP dynamics to deploy limited resource for an objective

function to be optimized at the end of the time window. This optimization task is a difficult hard-computational problem since the resource is deployed at earlier times; the two processes dynamically interact throughout the time window, and the aim is to optimize the objective function at its end.

One of the scenarios we consider is that of resource deployment in a mutually exclusive competitive case where the deployment of an agent's resources aims to maximize its spread at the end of the time window with respect to the spread of an adversarial process. This scenario can be seen as a competition between adversarial agents, each of which aims to contain the spread of the other, making it highly relevant in the battle for public opinion. The collaborative scenarios we optimize include constructive interaction between the spreading processes, such that infection by one process facilitates the infection by another. In this case, the aim is to exploit the exposure optimally in order to maximize the spread of both processes. Finally, we also examine optimal vaccination strategies in the case of collaborative spreading processes, where vaccination is effective with respect to only one process but helps to reduce the spread of both due to the interaction between them.

The optimization algorithm has been tested in several different scenarios and on a range of small-scale and real networks, showing excellent performance with respect to the existing alternatives of high-degree allocation, K-shell-based approaches, free spreading, and uniform budget allocation. We demonstrate that the optimized deployment makes a *huge* difference in the use of limited resources and allows for a balanced competition even with significantly inferior resource availability or lower infection rates. We see a significant potential in the use of such principled algorithms to make the most of limited resources.

Moreover, the suggested framework is highly adaptive and can accommodate targeted spreading [76], where only specific vertices are available and at specific times (as in the case of critical vote and time-sensitive campaigns); temporal deployment of resources, either of the process of interest or of the adversarial process, where resources are available at different times within the time window (e.g., because of limited production or shipment restrictions);

determining vaccination policies; and for more complex collaborative scenarios. We anticipate that the developed methods will also be useful for development of scalable [90] algorithms for learning [91,92] of spreading models in the context of multiple interacting spreading processes.

The application of our method for suggesting vaccination policy to contain the spread of collaborative health conditions, for process containment (e.g., the spread of fake news and antivaxxing messages), and for effective information dissemination (e.g., marketing or opinion-setting material) is promising, and timely applications that should be explored.

APPENDIX A: DYNAMIC BELIEF PROPAGATION

In the two-process dynamics, the dynamics of a single node i is described by a pair of activation times, (τ_i^A, τ_i^B), where τ_i^A denotes the first time when node i is found in status A, and similarly for τ_i^B. For instance, $\tau_i^A = 0$ means that node i was initially in the active status A, and we denote by $\tau_i^A = *$ the situation where node i did not get A-activated before some final observation time; i.e., in some sense $*$ absorbs all the history that happens after the end of the observation window. For the convenience of presentation, in what follows, we consider two separate "observation windows," for A and B processes.

We start our derivations with the general DBP equations [32,77,78] on the interaction graph, where the goal is to approximate the probability $m_{T_A+1,T_B+1}^i(\tau_i^A, \tau_i^B)$ that node i has a trajectory (τ_i^A, τ_i^B) during the observation time window of length T_A for process A and T_B for process B:

$$m_{T_A+1,T_B+1}^i(\tau_i^A, \tau_i^B) = \sum_{\{\tau_k^A,\tau_k^B\}_{k\in\partial i}} W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k\in\partial i}) \prod_{k\in\partial i} m_{T_A,T_B}^{k\to i}(\tau_k^A, \tau_k^B \| \tau_i^A, \tau_i^B), \tag{A1}$$

where ∂i denotes neighbors of i on the graph and the sum runs over all possible values of τ_k^A and τ_k^B, i.e., $\{0, 1, \dots, t_{A/B}, *\}$. The transition kernel W^i is defined through the dynamical rules for a given dynamics; we provide an explicit expression for the kernel below. The probabilities $m_{T_A,T_B}^{i\to j}(\tau_i^A, \tau_i^B \| \tau_j^A, \tau_j^B)$ have a sense of the probability that node i has a trajectory of length T_A for process A and T_B for process B [in what follows, we refer to this as the (T_A, T_B) trajectory] equal to (τ_i^A, τ_i^B), subject to a fixed trajectory (τ_j^A, τ_j^B) of node j, satisfying the following consistency relations:

$$m_{T_A+1,T_B+1}^{i\to j}(\tau_i^A, \tau_i^B \| \tau_j^A, \tau_j^B) = \sum_{\{\tau_k^A,\tau_k^B\}_{k\in\partial i\backslash j}} W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k\in\partial i}) \prod_{k\in\partial i\backslash j} m_{T_A,T_B}^{k\to i}(\tau_k^A, \tau_k^B \| \tau_i^A, \tau_i^B). \tag{A2}$$

The fixed-point solution of Eqs. (A1) and (A2) is guaranteed to be exact on tree graphs and provides good estimates of the marginal probabilities on loopy but sparse graphs [32,79].

Notice that $m_{T_A,T_B}^{i \to j}(\tau_i^A, \tau_i^B \| \tau_j^A, \tau_j^B)$ is not a conditional probability (e.g., Bayes rule does not apply). Let us state the following fact that will be crucial for the derivations below. (This fact is due to the properties of the DBP equations and the dynamics specified by W^i.)

Fact 1: (Fundamental property of messages) For $\tau_i^A \neq *$ and $\tau_i^B \neq *$,

$$m_{T_A,T_B}^i(\tau_i^A, \tau_i^B) = m_{\tau_i^A, \tau_i^B}^i(\tau_i^A, \tau_i^B) \quad \forall \; T_A > \tau_i^A \;\; \text{and} \;\; T_B > \tau_i^B.$$

A similar statement holds for messages.

Thus, future events cannot affect current or past events. Because of Fact 1, the DBP equations can be considerably simplified. Indeed, Eqs. (A1) and (A2) basically state that the right-hand sides do not depend on the final times in the observation windows, i.e., on times $(T_A + 1, T_B + 1)$. In particular, for $\tau_i^A = 0$ or $\tau_i^B = 0$, the right-hand side does not depend on τ_i^A or τ_i^B, respectively. For other cases, choosing $T_A = \tau_i^A - 1$ and $T_B = \tau_i^B - 1$, we see that since the time window of messages is $(\tau_i^A - 1, \tau_i^B - 1)$, they do not depend on the precise value of τ_i^A and τ_i^B. Thus, the following fact holds.

Fact 2: (Simplified DBP equations) For the dynamics considered, the following DBP equations hold:

$$m_{T_A+1,T_B+1}^i(\tau_i^A, \tau_i^B) = \sum_{\{\tau_k^A, \tau_k^B\}_{k \in \partial i}} W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$
$$\times \prod_{k \in \partial i} m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B \| *, *), \quad (A3)$$

$$m_{T_A+1,T_B+1}^{i \to j}(\tau_i^A, \tau_i^B \| \tau_j^A, \tau_j^B)$$
$$= \sum_{\{\tau_k^A, \tau_k^B\}_{k \in \partial i \setminus j}} W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$
$$\times \prod_{k \in \partial i \setminus j} m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B \| *, *). \quad (A4)$$

To simplify the notations, in what follows, we use the notation $m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B) \equiv m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B \| *, *)$. Given the computed value of the marginals $m_T^i(\tau_i^A, \tau_i^B)$, one can straightforwardly define quantities of interest, such as probabilities for a given node i to be found in a given status:

$$P_S^i(t) = \sum_{\tau_i^A > t} \sum_{\tau_i^B > t} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B) = m_{t,t}^i(*, *), \quad (A5)$$

$$P_A^i(t) = \sum_{\tau_i^A \leq t} \sum_{\tau_i^B} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B)$$
$$= \sum_{\tau_i^A \leq t} \sum_{\tau_i^B \in \{0,1,\dots,t,*\}} m_{t,t}^i(\tau_i^A, \tau_i^B)$$
$$= \sum_{\tau_i^A \leq t} (m_{t,0}^i(\tau_i^A, 0) + m_{t,0}^i(\tau_i^A, *)), \quad (A6)$$

$$P_B^i(t) = \sum_{\tau_i^B \leq t} \sum_{\tau_i^A} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B)$$
$$= \sum_{\tau_i^B \leq t} \sum_{\tau_i^A \in \{0,1,\dots,t,*\}} m_{t,t}^i(\tau_i^A, \tau_i^B)$$
$$= \sum_{\tau_i^B \leq t} (m_{0,t}^i(0, \tau_i^B) + m_{0,t}^i(*, \tau_i^B)), \quad (A7)$$

$$P_{AB}^i(t) = \sum_{\tau_i^A \leq t} \sum_{\tau_i^B \leq t} m_{T_A,T_B}^i(\tau_i^A, \tau_i^B) = \sum_{\tau_i^A \leq t, \tau_i^B \leq t} m_{t,t}^i(\tau_i^A, \tau_i^B). \quad (A8)$$

In the last expression, we used Fact 1, and in the previous expressions, we used the definition of $*$ and hence the equalities of the type

$$m_{T_A,T_B}^i(\tau_i^A, *) = m_{T_A,T_B+1}^i(\tau_i^A, *) + m_{T_A,T_B+1}^i(\tau_i^A, T_B + 1). \quad (A9)$$

Hence, in principle, to each node i, one can associate a $(t + 1) \times (t + 1)$ matrix of probabilities $m_{t,t}^i(\tau_i^A, \tau_i^B)$, and to each edge (ij), one can associate a fourth-order tensor of probabilities $m_{t,t}^{i \to j}(\tau_i^A, \tau_i^B \| \tau_j^A, \tau_j^B)$; one can then solve the fixed-point equations (A4), e.g., through iteration using some initialization for the conditional probabilities; next, one uses Eq. (A3) to compute the quantities $m_{t,t}^i(\tau_i^A, \tau_i^B)$; and finally, one uses Eqs. (A5)–(A8) to assemble the probabilities of finding node i in one of the statuses S, A, B, or AB. However, the naive implementation of this scheme would require as many as $O(t^{2(d-1)})$ operations for an estimation of a single message in Eq. (A4) and $O(t^{2d})$ operations for a single marginal in Eq. (A3), which still produces a polynomial-time algorithm for bounded-degree graphs but can quickly become prohibitive for large d.

On the other hand, the argument above applies to general dynamics parametrized by the flipping times τ_i^A, τ_i^B for node i; however, dynamics of interest (such as processes considered here) often have a special structure, and it is beneficial, from the computational point of view, to explore this structure in order to drastically reduce the complexity of the computation of the marginal probabilities (A5)–(A8). Therefore, while the problem is conceptually solvable in polynomial time at the level of the basic DBP equations (A3) and (A4), below we derive a computational scheme that would allow for an efficient computation of marginal probabilities (A5)–(A8). We refer to this scheme as the DMP equations.

APPENDIX B: EXACT DBP EQUATIONS FOR MUTUALLY EXCLUSIVE COMPETITIVE PROCESSES

As explained in the main text, the transition rules for competitive dynamics at time step t are defined as follows:

$$S(i) \to A(i), \quad \text{with probability } q_{S \to A}^i(t) = v_A^i(t)(1 - v_B^i(t))/Z_i(t), \tag{B1}$$

$$S(i) \to B(i), \quad \text{with probability } q_{S \to B}^i(t) = v_B^i(t)(1 - v_A^i(t))/Z_i(t), \tag{B2}$$

$$S(i) \to S(i), \quad \text{with probability } q_{S \to S}^i(t) = (1 - v_A^i(t) - v_B^i(t) + v_A^i(t)v_B^i(t))/Z_i(t), \tag{B3}$$

where

$$v_A^i(t) = 1 - \prod_{j \in \partial i}(1 - \alpha_{ji}^A \mathbb{1}[\sigma_j(t) = A]), \tag{B4}$$

$$v_B^i(t) = 1 - \prod_{j \in \partial i}(1 - \alpha_{ji}^B \mathbb{1}[\sigma_j(t) = B]), \tag{B5}$$

$$Z_i = 1 - v_A^i(t)v_B^i(t), \tag{B6}$$

and $\sigma_i(t)$ denotes the status of node i at time t. For two subsets $\Theta_A \subset \partial i$ and $\Theta_B \subset \partial i$, the transmission probabilities introduced in Eqs. (B1)–(B3) can be explicitly written as follows:

$$q_{S \to A}^i(t; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$
$$= \frac{(1 - \prod_{j \in \Theta_A}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t]))\prod_{l \in \Theta_B}(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t])}{\prod_{j \in \Theta_A}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t]) + \prod_{l \in \Theta_B}(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t]) - \prod_{j \in \Theta_A, l \in \Theta_B}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t])(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t])}, \tag{B7}$$

$$q_{S \to B}^i(t; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$
$$= \frac{\prod_{j \in \Theta_A}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t])(1 - \prod_{l \in \Theta_B}(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t]))}{\prod_{j \in \Theta_A}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t]) + \prod_{l \in \Theta_B}(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t]) - \prod_{j \in \Theta_A, l \in \Theta_B}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t])(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t])}, \tag{B8}$$

$$q_{S \to S}^i(t; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$
$$= \frac{\prod_{j \in \Theta_A}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t])\prod_{l \in \Theta_B}(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t])}{\prod_{j \in \Theta_A}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t]) + \prod_{l \in \Theta_B}(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t]) - \prod_{j \in \Theta_A, l \in \Theta_B}(1 - \alpha_{ji}^A \mathbb{1}[\tau_j^A \le t])(1 - \alpha_{li}^B \mathbb{1}[\tau_l^B \le t])}. \tag{B9}$$

Given the usual parametrization of marginals and messages $m^i(\tau_i^A, \tau_i^B)$, only the following messages can be nonzero for this dynamics: $m^i(\tau_i^A, *)$, $m^i(*, \tau_i^B)$, and $m^i(*, *)$. The dynamic messages are then defined as follows:

$$P_A^i(t) = \sum_{\tau_i^A \le t} m_{t,t}^i(\tau_i^A, *) = P_A^i(t - 1) + m_{t,t}^i(t, *), \tag{B10}$$

$$P_B^i(t) = \sum_{\tau_i^B \le t} m_{t,t}^i(*, \tau_i^B) = P_B^i(t - 1) + m_{t,t}^i(*, t), \tag{B11}$$

$$P_S^i(t) = m_{t,t}^i(*, *), \tag{B12}$$

with the initial conditions $m_{t,t}^i(0, *) = P_A^i(0)$ and $m_{t,t}^i(*, 0) = P_B^i(0)$.

As discussed in Appendix A, the complexity of a general DBP message evaluation requires $O(t^{2d})$ steps, where d is the node degree. Under the mutually exclusive scenario, some of the combinations of flipping times are forbidden since a node cannot be simultaneously activated under both A and B processes. To reflect this structure, we rewrite DBP equations for $t > 0$ in a computationally suboptimal but compact way:

$$m_{t,t}^i(t,*) = P_S^i(0) \sum_{\substack{\Theta_A \subset \partial i; \\ \Theta_B \subset \partial i; \\ \Theta_S = \partial i \backslash (\Theta_A \cup \Theta_B)}} \sum_{\{\tau_k^A\}_{k \in \Theta_A}} \sum_{\{\tau_l^B\}_{l \in \Theta_B}} \left(\prod_{t'=0}^{t-2} q_{S \to S}^i(t'; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i}) \right) q_{S \to A}^i(t-1; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$

$$\times \prod_{k \in \Theta_A} m_{t-1,t-1}^{k \to i}(\tau_k^A, *) \prod_{l \in \Theta_B} m_{t-1,t-1}^{l \to i}(*, \tau_l^B) \prod_{n \in \Theta_S} m_{t-1,t-1}^{n \to i}(*, *), \tag{B13}$$

$$m_{t,t}^i(*,t) = P_S^i(0) \sum_{\substack{\Theta_A \subset \partial i; \\ \Theta_B \subset \partial i; \\ \Theta_S = \partial i \backslash (\Theta_A \cup \Theta_B)}} \sum_{\{\tau_k^A\}_{k \in \Theta_A}} \sum_{\{\tau_l^B\}_{l \in \Theta_B}} \left(\prod_{t'=0}^{t-2} q_{S \to S}^i(t'; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i}) \right) q_{S \to B}^i(t-1; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$

$$\times \prod_{k \in \Theta_A} m_{t-1,t-1}^{k \to i}(\tau_k^A, *) \prod_{l \in \Theta_B} m_{t-1,t-1}^{l \to i}(*, \tau_l^B) \prod_{n \in \Theta_S} m_{t-1,t-1}^{n \to i}(*, *), \tag{B14}$$

$$m_{t,t}^i(*,*) = P_S^i(0) \sum_{\substack{\Theta_A \subset \partial i; \\ \Theta_B \subset \partial i; \\ \Theta_S = \partial i \backslash (\Theta_A \cup \Theta_B)}} \sum_{\{\tau_k^A\}_{k \in \Theta_A}} \sum_{\{\tau_l^B\}_{l \in \Theta_B}} \left(\prod_{t'=0}^{t-1} q_{S \to S}^i(t'; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i}) \right)$$

$$\times \prod_{k \in \Theta_A} m_{t-1,t-1}^{k \to i}(\tau_k^A, *) \prod_{l \in \Theta_B} m_{t-1,t-1}^{l \to i}(*, \tau_l^B) \prod_{n \in \Theta_S} m_{t-1,t-1}^{n \to i}(*, *). \tag{B15}$$

DBP messages in the expressions above are computed in a similar way, where subsets Θ_A and Θ_B are chosen from $\partial i \backslash j$:

$$m_{t,t}^{i \to j}(t,*) = P_S^i(0) \sum_{\substack{\Theta_A \subset \partial i \backslash j; \\ \Theta_B \subset \partial i \backslash j; \\ \Theta_S = \partial i \backslash (j \cup \Theta_A \cup \Theta_B)}} \sum_{\{\tau_k^A\}_{k \in \Theta_A}} \sum_{\{\tau_l^B\}_{l \in \Theta_B}} \left(\prod_{t'=0}^{t-2} q_{S \to S}^i(t'; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i}) \right) q_{S \to A}^i(t-1; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$

$$\times P_S^i(0) \prod_{k \in \Theta_A} m_{t-1,t-1}^{k \to i}(\tau_k^A, *) \prod_{l \in \Theta_B} m_{t-1,t-1}^{l \to i}(*, \tau_l^B) \prod_{n \in \Theta_S} m_{t-1,t-1}^{n \to i}(*, *), \tag{B16}$$

$$m_{t,t}^{i \to j}(*,t) = P_S^i(0) \sum_{\substack{\Theta_A \subset \partial i \backslash j; \\ \Theta_B \subset \partial i \backslash j; \\ \Theta_S = \partial i \backslash (j \cup \Theta_A \cup \Theta_B)}} \sum_{\{\tau_k^A\}_{k \in \Theta_A}} \sum_{\{\tau_l^B\}_{l \in \Theta_B}} \left(\prod_{t'=0}^{t-2} q_{S \to S}^i(t'; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i}) \right) q_{S \to B}^i(t-1; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$

$$\times \prod_{k \in \Theta_A} m_{t-1,t-1}^{k \to i}(\tau_k^A, *) \prod_{l \in \Theta_B} m_{t-1,t-1}^{l \to i}(*, \tau_l^B) \prod_{n \in \Theta_S} m_{t-1,t-1}^{n \to i}(*, *), \tag{B17}$$

$$m_{t,t}^{i \to j}(*,*) = P_S^i(0) \sum_{\substack{\Theta_A \subset \partial i \backslash j; \\ \Theta_B \subset \partial i \backslash j; \\ \Theta_S = \partial i \backslash (j \cup \Theta_A \cup \Theta_B)}} \sum_{\{\tau_k^A\}_{k \in \Theta_A}} \sum_{\{\tau_l^B\}_{l \in \Theta_B}} \left(\prod_{t'=0}^{t-1} q_{S \to S}^i(t'; \Theta_A, \Theta_B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i}) \right)$$

$$\times \prod_{k \in \Theta_A} m_{t-1,t-1}^{k \to i}(\tau_k^A, *) \prod_{l \in \Theta_B} m_{t-1,t-1}^{l \to i}(*, \tau_l^B) \prod_{n \in \Theta_S} m_{t-1,t-1}^{n \to i}(*, *). \tag{B18}$$

In Sec. II of the main text, we numerically show that these equations are exact on tree graphs.

APPENDIX C: APPROXIMATE DMP EQUATIONS FOR MUTUALLY EXCLUSIVE COMPETITIVE PROCESSES

In the search of low-complexity equations that could be used on general graphs, in this section we derive approximate DMP equations for mutually exclusive competitive processes. As outlined in the main text, our approximation scheme is based on the idea that in the absence of renormalization, the dynamics of each process follows the dynamics of a simple SI process. Hence, we derive equations that perform the renormalization procedure at the level of dynamic marginals.

Let us start by reminding the equations for the SI model. We closely follow the notations used in Refs. [32,93]. For a single process A, let $\theta_A^{i \to j}(t)$ denote the probability that no infection message A has been passed from node i to node j until time t, and let $\phi_A^{i \to j}(t)$ denote the probability that no infection message A has been passed until time $t - 1$, while

node i is infected with A at time t. Exact DMP equations for a single-process SI model read [32]

$$P_S^i(t) = P_S^i(0) \prod_{k \in \partial i} \theta_A^{k \to i}(t), \tag{C1}$$

$$P_A^i(t) = 1 - P_S^{i \to j}(t), \tag{C2}$$

$$P_S^{i \to j}(t) = \prod_{k \in \partial i \setminus j} \theta_A^{k \to i}(t), \tag{C3}$$

$$P_A^{i \to j}(t) = 1 - P_S^{i \to j}(t), \tag{C4}$$

$$\theta_A^{i \to j}(t) = \theta_A^{i \to j}(t-1) - \alpha_{ji}^A \phi_A^{i \to j}(t-1), \tag{C5}$$

$$\phi_A^{i \to j}(t) = (1 - \alpha_{ji}^A)\phi_A^{i \to j}(t-1) + [P_A^{i \to j}(t) - P_A^{i \to j}(t-1)]. \tag{C6}$$

For the two competitive and mutually exclusive processes A and B, we approximate the total spreading as a product of two independent spreading processes that follow Eqs. (C1)–(C6). Our goal is to derive approximate dynamic marginals, which we denote by $\hat{P}_S^i(t)$, $\hat{P}_A^i(t)$, and $\hat{P}_B^i(t)$. For simplicity, let us assume that $\alpha_{ij}^A < 1$, and hence $\theta_A^{i \to j}$ type messages are nonzero (treatment of the case of deterministic spreading is straightforward and results in several additional equations). Under the chosen approximation, the *non-normalized* probability of staying in the susceptible status can be written as

$$\tilde{P}_S^{i \to j}(t) = \hat{P}_S^{i \to j}(t-1) \prod_{k \in \partial i \setminus j} \frac{\theta_A^{k \to i}(t) \theta_B^{k \to i}(t)}{\theta_A^{k \to i}(t-1)\theta_B^{k \to i}(t-1)}. \tag{C7}$$

Under this approximation, the transition to status A happens when the target node is susceptible, and the B infection is not transmitted:

$$\tilde{P}_A^{i \to j}(t) = \hat{P}_S^{i \to j}(t-1)\left[\prod_{k \in \partial i \setminus j}\left(1 - \frac{\alpha_B \phi_B^{k \to i}(t-1)}{\theta_B^{k \to i}(t-1)} \right) \right.$$
$$\left. \times \left(1 - \prod_{k \in \partial i \setminus j}\left(1 - \frac{\alpha_A \phi_A^{k \to i}(t-1)}{\theta_A^{k \to i}(t-1)} \right) \right) \right]$$
$$+ \hat{P}_A^{i \to j}(t-1),$$
$$\tilde{P}_B^{i \to j}(t) = \hat{P}_S^{i \to j}(t-1)\left[\prod_{k \in \partial i \setminus j}\left(1 - \frac{\alpha_A \phi_A^{k \to i}(t-1)}{\theta_A^{k \to i}(t-1)} \right) \right.$$
$$\left. \times \left(1 - \prod_{k \in \partial i \setminus j}\left(1 - \frac{\alpha_B \phi_B^{k \to i}(t-1)}{\theta_B^{k \to i}(t-1)} \right) \right) \right]$$
$$+ \hat{P}_B^{i \to j}(t-1). \tag{C8}$$

We then compute the renormalized messages for the three statuses $\hat{P}_{S/A/B}^{i \to j}(t)$:

$$\hat{P}_{S/A/B}^{i \to j}(t) = \frac{\tilde{P}_{S/A/B}^{i \to j}(t)}{\tilde{P}_S^{i \to j}(t) + \tilde{P}_A^{i \to j}(t) + \tilde{P}_B^{i \to j}(t)}. \tag{C9}$$

The dynamic marginals are calculated iteratively in a similar fashion:

$$\tilde{P}_S^i(t) = \hat{P}_S^i(t-1)\prod_{k \in \partial i}\frac{\theta_A^{k \to i}(t)\theta_B^{k \to i}(t)}{\theta_A^{k \to i}(t-1)\theta_B^{k \to i}(t-1)},$$
$$\tilde{P}_A^i(t) = \hat{P}_S^i(t-1)\left[\prod_{k \in \partial i}\left(1 - \frac{\alpha_B \phi_B^{k \to i}(t-1)}{\theta_B^{k \to i}(t-1)} \right) \right.$$
$$\left. \times \left(1 - \prod_{k \in \partial i}\left(1 - \frac{\alpha_A \phi_A^{k \to i}(t-1)}{\theta_A^{k \to i}(t-1)} \right) \right) \right]$$
$$+ \hat{P}_A^i(t-1),$$
$$\tilde{P}_B^i(t) = \hat{P}_S^i(t-1)\left[\prod_{k \in \partial i}\left(1 - \frac{\alpha_A \phi_A^{k \to i}(t-1)}{\theta_A^{k \to i}(t-1)} \right) \right.$$
$$\left. \times \left(1 - \prod_{k \in \partial i}\left(1 - \frac{\alpha_B \phi_B^{k \to i}(t-1)}{\theta_B^{k \to i}(t-1)} \right) \right) \right]$$
$$+ \hat{P}_B^i(t-1),$$
$$P_{S/A/B}^i(t) = \frac{\tilde{P}_{S/A/B}^i(t)}{\tilde{P}_S^i(t) + \tilde{P}_A^i(t) + \tilde{P}_B^i(t)}. \tag{C10}$$

We numerically study the performance of these approximate DMP equations in the main text.

APPENDIX D: EXACT AND DMP EQUATIONS FOR COLLABORATIVE PROCESSES

1. DBP equations for collaborative processes

Let us start by recalling the transition rules for the collaborative spreading, as discussed in the main text:

$$S(i) + A(j) \xrightarrow{\alpha_{ji}^A} A(i) + A(j), \tag{D1}$$

$$S(i) + B(j) \xrightarrow{\alpha_{ji}^B} B(i) + B(j), \tag{D2}$$

$$A(i) + B(j) \xrightarrow{\alpha_{ji}^{BA}} AB(i) + B(j), \tag{D3}$$

$$A(i) + B(j) \xrightarrow{\alpha_{ij}^{AB}} A(i) + AB(j). \tag{D4}$$

We start by specifying the transition kernel $W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$ based on the dynamic rules (D1)–(D4):

$$W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k\in\partial i}) = \underbrace{P_S^i(0)\mathbb{1}[\tau_i^A < \tau_i^B]\{S(i)\xrightarrow{\tau_i^A} A(i)\xrightarrow{\tau_i^B} AB(i)\}}_{(1)} \tag{D5}$$

$$+\underbrace{P_S^i(0)\mathbb{1}[\tau_i^B < \tau_i^A]\{S(i)\xrightarrow{\tau_i^B} B(i)\xrightarrow{\tau_i^A} AB(i)\}}_{(2)} \tag{D6}$$

$$+\underbrace{P_S^i(0)\mathbb{1}[\tau_i^A = \tau_i^B]\{S(i)\xrightarrow{\tau_i^A=\tau_i^B} AB(i)\}}_{(3)} \tag{D7}$$

$$+\underbrace{P_{A^*}^i(0)\mathbb{1}[\tau_i^A = 0]\mathbb{1}[\tau_i^B > 0]\{A(i)\xrightarrow{\tau_i^B} AB(i)\}}_{(4)} \tag{D8}$$

$$+\underbrace{P_{B^*}^i(0)\mathbb{1}[\tau_i^B = 0]\mathbb{1}[\tau_i^A > 0]\{B(i)\xrightarrow{\tau_i^A} AB(i)\}}_{(5)} \tag{D9}$$

$$+\underbrace{P_{AB}^i(0)\mathbb{1}[\tau_i^A = \tau_i^B = 0]}_{(6)}. \tag{D10}$$

In the expression above, it is assumed that when $\tau_i^A = *$ or $\tau_i^B = *$, the corresponding transition does not happen during the given observation window, or, alternatively, it happens *eventually*, beyond the observation horizon. Here, $P_S^i(0)$, $P_{A^*}^i(0)$, $P_{B^*}^i(0)$, and $P_{AB}^i(0)$ denote probabilities that at initial time, node i is initialized in the statuses S, A exclusively, B exclusively, or AB, respectively.

We can now provide explicit forms of the expressions that correspond to the transitions inside curly brackets; specifically, in the following, we refer to the numbered items in (D5)–(D10). In the case $\tau_i^A = *$ and $\tau_i^B = *$, we have

$$(1) = P_S^i(0)\mathbb{1}[\tau_i^A < \tau_i^B]\mathbb{1}[\tau_i^A > 0]\prod_{t'=0}^{\tau_i^A-2}\prod_{k\in\partial i}(1-\alpha_{ki}^A\mathbb{1}[\tau_k^A\le t'])\underbrace{\left(1-\prod_{k\in\partial i}(1-\alpha_{ki}^A\mathbb{1}[\tau_k^A\le \tau_i^A-1])\right)}_{A\text{-activation with prob }\alpha_{ki}^A\text{ at time }\tau_i^A}$$

$$\times\underbrace{\prod_{t''=0}^{\tau_i^A-1}\prod_{k\in\partial i}(1-\alpha_{ki}^B\mathbb{1}[\tau_k^B\le t''])}_{\text{no }B\text{-activation with prob }\alpha_{ki}^B\text{ by }\tau_i^A}\underbrace{\prod_{t'''=\tau_i^A}^{\tau_i^B-2}\prod_{k\in\partial i}(1-\alpha_{ki}^{BA}\mathbb{1}[\tau_k^B\le t'''])\left(1-\prod_{k\in\partial i}(1-\alpha_{ki}^{BA}\mathbb{1}[\tau_k^B\le \tau_i^B-1])\right)}_{B\text{-activation with prob }\alpha_{ki}^{BA}\text{ at time }\tau_i^B}; \tag{D11}$$

$$(2) = P_S^i(0)\mathbb{1}[\tau_i^B < \tau_i^A]\mathbb{1}[\tau_i^B > 0]\prod_{t'=0}^{\tau_i^B-2}\prod_{k\in\partial i}(1-\alpha_{ki}^B\mathbb{1}[\tau_k^B\le t'])\underbrace{\left(1-\prod_{k\in\partial i}(1-\alpha_{ki}^B\mathbb{1}[\tau_k^B\le \tau_i^B-1])\right)}_{B\text{-activation with prob }\alpha_{ki}^B\text{ at time }\tau_i^B}$$

$$\times\underbrace{\prod_{t''=0}^{\tau_i^B-1}\prod_{k\in\partial i}(1-\alpha_{ki}^A\mathbb{1}[\tau_k^A\le t''])}_{\text{no }A\text{-activation with prob }\alpha_{ki}^A\text{ by }\tau_i^B}\underbrace{\prod_{t'''=\tau_i^B}^{\tau_i^A-2}\prod_{k\in\partial i}(1-\alpha_{ki}^{AB}\mathbb{1}[\tau_k^A\le t'''])\left(1-\prod_{k\in\partial i}(1-\alpha_{ki}^{AB}\mathbb{1}[\tau_k^A\le \tau_i^A-1])\right)}_{A\text{-activation with prob }\alpha_{ki}^{AB}\text{ at time }\tau_i^A}; \tag{D12}$$

$$(3) = P_S^i(0)\mathbb{1}[\tau_i^A = \tau_i^B]\mathbb{1}[\tau_i^A > 0]\prod_{t'=0}^{\tau_i^A-2}\prod_{k\in\partial i}(1 - \alpha_{ki}^A\mathbb{1}[\tau_k^A \le t'])\underbrace{\left(1 - \prod_{k\in\partial i}(1 - \alpha_{ki}^A\mathbb{1}[\tau_k^A \le \tau_i^A - 1])\right)}_{A\text{-activation with prob } \alpha_{ki}^A \text{ at time } \tau_i^A}$$

$$\times \prod_{t'=0}^{\tau_i^A-2}\prod_{k\in\partial i}(1 - \alpha_{ki}^B\mathbb{1}[\tau_k^B \le t'])\underbrace{\left(1 - \prod_{k\in\partial i}(1 - \alpha_{ki}^B\mathbb{1}[\tau_k^B \le \tau_i^A - 1])\right)}_{B\text{-activation with prob } \alpha_{ki}^B \text{ at time } \tau_i^B}; \tag{D13}$$

$$(4) = P_{A^*}^i(0)\mathbb{1}[\tau_i^B > 0]\mathbb{1}[\tau_i^A = 0]\prod_{t'=0}^{\tau_i^B-2}\prod_{k\in\partial i}(1 - \alpha_{ki}^{BA}\mathbb{1}[\tau_k^B \le t'])\underbrace{\left(1 - \prod_{k\in\partial i}(1 - \alpha_{ki}^{BA}\mathbb{1}[\tau_k^B \le \tau_i^B - 1])\right)}_{B\text{-activation with prob } \alpha_{ki}^{BA} \text{ at time } \tau_i^B}; \tag{D14}$$

$$(5) = P_{B^*}^i(0)\mathbb{1}[\tau_i^A > 0]\mathbb{1}[\tau_i^B = 0]\prod_{t'=0}^{\tau_i^A-2}\prod_{k\in\partial i}(1 - \alpha_{ki}^{AB}\mathbb{1}[\tau_k^A \le t'])\underbrace{\left(1 - \prod_{k\in\partial i}(1 - \alpha_{ki}^{AB}\mathbb{1}[\tau_k^A \le \tau_i^A - 1])\right)}_{A\text{-activation with prob } \alpha_{ki}^{AB} \text{ at time } \tau_i^A}; \tag{D15}$$

$$(6) = P_{AB}^i(0)\underbrace{\mathbb{1}[\tau_i^A = 0]\mathbb{1}[\tau_i^B = 0]}_{\text{no activation in status } AB}. \tag{D16}$$

If either $\tau_i^A = *$ or $\tau_i^B = *$, the activation above is replaced by an absence of activation. For instance, in the case of $\tau_i^B = *$, the last term in Eq. (D11) would read

$$\underbrace{\prod_{t'''=\tau_i^A}^{t}\prod_{k\in\partial i}(1 - \alpha_{ki}^{BA}\mathbb{1}[\tau_k^B \le t'''])}_{\text{absence of } B\text{-activation with prob } \alpha_{ki}^{BA} \text{ at time } \tau_i^B} \tag{D17}$$

for the $(t+1, t+1)$ trajectory of node i.

The DBP equations for collaborative processes are given by Eqs. (A3) and (A4) using the $W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k\in\partial i})$ defined above. In what follows, we explain how to derive the equivalent low-complexity DMP equations for the collaborative processes from the DBP equations (A3) and (A4), state their full iterative form, and finally expand the exact DMP equations around the noninteracting point, which yields approximate DMP equations that have a more compact form and a lower algorithmic complexity.

2. Definitions of dynamic messages

Let us introduce the mathematical definitions for the dynamic messages as the functions of fundamental probabilities on time trajectories, $m_{T_A,T_B}^{i\to j}(\tau_i^A, \tau_i^B) \equiv m_{T_A,T_B}^{i\to j}(\tau_i^A, \tau_i^B\|*, *)$, which will be used in the computations scheme below. First, it is useful to consider reduced marginals:

$$\mu_A^i(t) = \sum_{\tau_i^B}m_{t,t}^i(t, \tau_i^B) = P_A^i(t) - P_A^i(t-1), \tag{D18}$$

$$\mu_B^i(t) = \sum_{\tau_i^A}m_{t,t}^i(\tau_i^A, t) = P_B^i(t) - P_B^i(t-1). \tag{D19}$$

They have the physical meaning of probabilities of individual activation with one of the processes, when we are not concerned with the activation by the other process. In a similar way, we also define reduced messages:

$$\mu_A^{i\to j}(t) = \sum_{\tau_i^B}m_{t,t}^{i\to j}(t, \tau_i^B) = P_A^{i\to j}(t) - P_A^{i\to j}(t-1), \tag{D20}$$

$$\mu_B^{i\to j}(t) = \sum_{\tau_i^A}m_{t,t}^{i\to j}(\tau_i^A, t) = P_B^{i\to j}(t) - P_B^{i\to j}(t-1). \tag{D21}$$

Similarly to dynamic marginals (A5)–(A8), one can define the aggregated dynamic messages. In principle, we can define them for a general fixed dynamics (τ_j^A, τ_j^B) of the "cavity" node j, but in what follows, we only use the messages for the fixed dynamics $(\tau_j^A, \tau_j^B) = (*, *)$:

$$P_S^{i\to j}(t) = \sum_{\tau_i^A>t}\sum_{\tau_i^B>t}m_{T_A,T_B}^{i\to j}(\tau_i^A, \tau_i^B) = m_{t,t}^{i\to j}(*, *), \tag{D22}$$

$$P_A^{i\to j}(t) = \sum_{\tau_i^B}\sum_{\tau_i^A\le t}m_{T_A,T_B}^{i\to j}(\tau_i^A, \tau_i^B) = \sum_{\tau_i^A\le t}\mu_A^{i\to j}(\tau_i^A), \tag{D23}$$

$$P_B^{i\to j}(t) = \sum_{\tau_i^B}\sum_{\tau_i^B\le t}m_{T_A,T_B}^{i\to j}(\tau_i^A, \tau_i^B) = \sum_{\tau_i^B\le t}\mu_B^{i\to j}(\tau_i^B), \tag{D24}$$

$$P_{AB}^{i \to j}(t) = \sum_{\tau_i^A \leq t} \sum_{\tau_i^B \leq t} m_{t,t}^{i \to j}(\tau_i^A, \tau_i^B). \tag{D25}$$

These quantities have the same physical interpretation, except that they are defined on the "cavity" graph where node j follows a fixed (τ_j^A, τ_j^B) dynamics.

Let us also define the following dynamic messages that are weighted combinations of messages:

$$\theta_{A,B}^{k \to i}(t_A, t_B) = \sum_{\tau_k^A} \sum_{\tau_k^B} \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) \prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']) m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B), \tag{D26}$$

$$\begin{aligned} \theta_{A,B,AB}^{k \to i}(t_A, t_B, t_{AB}; \tau_i^A) = \sum_{\tau_k^A} \sum_{\tau_k^B} \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) \prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']) \\ \times \prod_{t'''=\tau_i^A}^{t_{AB}-1}(1 - \alpha_{ki}^{BA} \mathbb{1}[\tau_k^B \leq t''']) m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B), \end{aligned} \tag{D27}$$

$$\begin{aligned} \theta_{B,A,BA}^{k \to i}(t_B, t_A, t_{BA}; \tau_i^B) = \sum_{\tau_k^A} \sum_{\tau_k^B} \prod_{t'=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t']) \prod_{t''=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t'']) \\ \times \prod_{t'''=\tau_i^B}^{t_{BA}-1}(1 - \alpha_{ki}^{AB} \mathbb{1}[\tau_k^A \leq t''']) m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B), \end{aligned} \tag{D28}$$

$$\theta_{AB}^{k \to i}(t_B) = \theta_{A,B,AB}^{k \to i}(0, 0, t_B; 0) = \sum_{\tau_k^B} \prod_{t'=0}^{t_B-1}(1 - \alpha_{ki}^{BA} \mathbb{1}[\tau_k^B \leq t']) \mu_B^{k \to i}(\tau_k^B), \tag{D29}$$

$$\theta_{BA}^{k \to i}(t_A) = \theta_{B,A,BA}^{k \to i}(0, 0, t_A; 0) = \sum_{\tau_k^A} \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^{AB} \mathbb{1}[\tau_k^A \leq t']) \mu_A^{k \to i}(\tau_k^A). \tag{D30}$$

These messages have the meaning of the probabilities that node k did not send A and B activation signals to node i on the "cavity" graph where node i follows a fixed $(*, *)$ dynamics.

Next, we define the following quantities:

$$\phi_A^{k \to i}(t_A, t_B) = \sum_{\tau_k^A} \sum_{\tau_k^B} \mathbb{1}[\tau_k^A \leq t_A] \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) \prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']) m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B), \tag{D31}$$

$$\phi_B^{k \to i}(t_A, t_B) = \sum_{\tau_k^A} \sum_{\tau_k^B} \mathbb{1}[\tau_k^B \leq t_B] \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) \prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']) m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B), \tag{D32}$$

$$\phi_{A,B}^{k \to i}(t_A, t_B) = \sum_{\tau_k^A} \sum_{\tau_k^B} \mathbb{1}[\tau_k^A \leq t_A] \mathbb{1}[\tau_k^B \leq t_B] \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) \prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']) m_{T_A,T_B}^{k \to i}(\tau_k^A, \tau_k^B), \tag{D33}$$

$$\phi_{A,B,AB}^{k\rightarrow i}(t_A, t_B, t_{AB}; \tau_i^A) = \sum_{\tau_k^A} \sum_{\tau_k^B} \mathbb{1}[\tau_k^B \leq t_{AB}] \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) \prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t''])$$

$$\times \prod_{t'''=\tau_i^A}^{t_{AB}-1}(1 - \alpha_{ki}^{BA} \mathbb{1}[\tau_k^B \leq t''']) m_{T_A,T_B}^{k\rightarrow i}(\tau_k^A, \tau_k^B), \tag{D34}$$

$$\phi_{B,A,BA}^{k\rightarrow i}(t_B, t_A, t_{BA}; \tau_i^B) = \sum_{\tau_k^A} \sum_{\tau_k^B} \mathbb{1}[\tau_k^A \leq t_{BA}] \prod_{t'=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t']) \prod_{t''=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t''])$$

$$\times \prod_{t'''=\tau_i^B}^{t_{BA}-1}(1 - \alpha_{ki}^{AB} \mathbb{1}[\tau_k^A \leq t''']) m_{T_A,T_B}^{k\rightarrow i}(\tau_k^A, \tau_k^B), \tag{D35}$$

$$\phi_{AB}^{k\rightarrow i}(t_B) = \sum_{\tau_k^B} \mathbb{1}[\tau_k^B \leq t_B] \prod_{t'=0}^{t_B-1}(1 - \alpha_{ki}^{BA} \mathbb{1}[\tau_k^B \leq t']) \mu_B^{k\rightarrow i}(\tau_k^B), \tag{D36}$$

$$\phi_{BA}^{k\rightarrow i}(t_A) = \sum_{\tau_k^A} \mathbb{1}[\tau_k^A \leq t_A] \prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^{AB} \mathbb{1}[\tau_k^A \leq t']) \mu_A^{k\rightarrow i}(\tau_k^A). \tag{D37}$$

These messages have the meaning of the probabilities that node k did not send A and B activation signals to node i on the "cavity" graph where node i follows a fixed $(*, *)$ dynamics, but node k is in an active status itself.

Finally, we define auxiliary expressions

$$\theta_B^{k\rightarrow i}(t, t) = \sum_{\tau_k^B} \prod_{t''=0}^{t-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']) m_{T_A,T_B}^{k\rightarrow i}(t, \tau_k^B), \quad \text{(D38)}$$

$$\theta_A^{k\rightarrow i}(t, t) = \sum_{\tau_k^A} \prod_{t'=0}^{t-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) m_{T_A,T_B}^{k\rightarrow i}(\tau_k^A, t). \quad \text{(D39)}$$

3. Update equations for dynamic messages

We start with the definition

$$P_S^i(t+1) = m_{t+1,t+1}^i(*, *). \tag{D40}$$

From the DBP equations on time trajectories (A3), we get

$$m_{t+1,t+1}^i(*, *)$$

$$= \sum_{\{\tau_k^A, \tau_k^B\}_{k\in\partial i}} W^i(*, *; \{\tau_k^A, \tau_k^B\}_{k\in\partial i})$$

$$\times \prod_{k\in\partial i} m_{t,t}^{k\rightarrow i}(\tau_k^A, \tau_k^B \| *, *), \tag{D41}$$

where

$$W^i(*, *; \{\tau_k^A, \tau_k^B\}_{k\in\partial i})$$

$$= \prod_{k\in\partial i} \prod_{t'=0}^{t}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \leq t']) \prod_{t''=0}^{t}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \leq t'']). \tag{D42}$$

Using the definition (D26),

$$P_S^i(t+1) = P_S^i(0) \prod_{k\in\partial i\backslash j} \theta_{A,B}^{k\rightarrow i}(t+1, t+1\|*, *). \tag{D43}$$

For simplicity, let us denote $\theta_{A,B}^{k\rightarrow i}(t+1, t+1) \equiv \theta_{A,B}^{k\rightarrow i}(t+1, t+1\|*, *)$. With this simplified notation, we finally write

$$P_S^i(t+1) = P_S^i(0) \prod_{k\in\partial i\backslash j} \theta_{A,B}^{k\rightarrow i}(t+1, t+1). \tag{D44}$$

We use the same simplified notation for all quantities of the type $\#^{k\rightarrow i}(\ldots) \equiv \#^{k\rightarrow i}(\ldots\|*, *)$. Using the previously defined quantities ϕ_A, ϕ_B, and $\phi_{A,B}$, through Eqs. (D31)–(D33), we derive the following update equations:

$$\theta_{A,B}^{k\rightarrow i}(t+1, t+1) = \theta_{A,B}^{k\rightarrow i}(t, t) - \alpha_{ki}^A \phi_A^{k\rightarrow i}(t, t)$$

$$- \alpha_{ki}^B \phi_B^{k\rightarrow i}(t, t) + \alpha_{ki}^A \alpha_{ki}^B \phi_{A,B}^{k\rightarrow i}(t, t). \tag{D45}$$

Using the equality $\mathbb{1}[\tau_k^A \leq t_A] = \mathbb{1}[\tau_k^A \leq t_A - 1] + \mathbb{1}[\tau_k^A = t_A]$, we further get

$$\phi_A^{k\rightarrow i}(t, t) = (1 - \alpha_{ki}^A) \phi_A^{k\rightarrow i}(t-1, t) + \theta_B^{k\rightarrow i}(t, t), \tag{D46}$$

$$\phi_B^{k\rightarrow i}(t, t) = (1 - \alpha_{ki}^B) \phi_B^{k\rightarrow i}(t-1, t-1) + \theta_A^{k\rightarrow i}(t, t), \tag{D47}$$

where θ_A and θ_B have been defined above. Finally, using

$$\phi_A^{k \to i}(t-1, t) = \phi_A^{k \to i}(t-1, t-1) + (1 - \alpha_{ki}^B)\phi_{A,B}^{k \to i}(t-1, t-1), \tag{D48}$$

$$\phi_B^{k \to i}(t, t-1) = \phi_B^{k \to i}(t-1, t-1) + (1 - \alpha_{ki}^A)\phi_{A,B}^{k \to i}(t-1, t-1), \tag{D49}$$

we obtain

$$\phi_A^{k \to i}(t, t) = (1 - \alpha_{ki}^A)\phi_A^{k \to i}(t-1, t-1) - \alpha_{ki}^B(1 - \alpha_{ki}^A)\phi_{A,B}^{k \to i}(t-1, t-1) + \theta_B^{k \to i}(t, t), \tag{D50}$$

$$\phi_B^{k \to i}(t, t) = (1 - \alpha_{ki}^B)\phi_B^{k \to i}(t-1, t-1) - \alpha_{ki}^A(1 - \alpha_{ki}^B)\phi_{A,B}^{k \to i}(t-1, t-1) + \theta_A^{k \to i}(t, t). \tag{D51}$$

Additionally,

$$\phi_{A,B}^{k \to i}(t, t) = (1 - \alpha_{ki}^A)(1 - \alpha_{ki}^B)\phi_{A,B}^{k \to i}(t-1, t-1) - m_{t,t}^{k \to i}(t, t) \tag{D52}$$

$$+ \sum_{\tau_k^A \le t} \prod_{t'=0}^{t-1}(1 - \alpha_{ki}^A \mathbb{1}[t' \ge \tau_k^A]) m_{t,t}^{k \to i}(\tau_k^A, t) \tag{D53}$$

$$+ \sum_{\tau_k^B \le t} \prod_{t'=0}^{t-1}(1 - \alpha_{ki}^B \mathbb{1}[t' \ge \tau_k^B]) m_{t,t}^{k \to i}(t, \tau_k^B), \tag{D54}$$

where $m_{t,t}^{k \to i}(\tau_k^A, \tau_k^B) \equiv m_{t,t}^{k \to i}(\tau_k^A, \tau_k^B \| *, *)$. Finally, we need to compute, iteratively, $\theta_A^{k \to i}(t, t)$ and $\theta_B^{k \to i}(t, t)$:

$$\theta_A^{k \to i}(t, t) = \theta_A^{k \to i}(t-1, t) - \alpha_{ki}^A \sum_{\tau_k^A \le t-1} \prod_{t'=0}^{t-2}(1 - \alpha_{ki}^A \mathbb{1}[t' \ge \tau_k^A]) m_{t,t}^{k \to i}(\tau_k^A, t), \tag{D55}$$

$$\theta_B^{k \to i}(t, t) = \theta_B^{k \to i}(t, t-1) - \alpha_{ki}^B \sum_{\tau_k^B \le t-1} \prod_{t'=0}^{t-2}(1 - \alpha_{ki}^B \mathbb{1}[t' \ge \tau_k^B]) m_{t,t}^{k \to i}(t, \tau_k^B), \tag{D56}$$

where, by definition, $\theta_A^{k \to i}(0, t) = \mu_B^{k \to i}(t) = P_B^{k \to i}(t) - P_B^{k \to i}(t-1)$, and similarly $\theta_B^{k \to i}(t, 0) = \mu_A^{k \to i}(t) = P_A^{k \to i}(t) - P_A^{k \to i}(t-1)$. Initial and border conditions of that type can be directly obtained from the definitions of the dynamic messages and their connections to messages on time trajectories.

From the scheme above, we still need to compute the update equations for the marginals and the messages on time trajectories of the type $m_{t,t}^i(\tau_i^A, \tau_i^B)$ and $m_{t,t}^{i \to j}(\tau_i^A, \tau_i^B)$. The derivation of the equations for these messages is given below. Once the full or reduced marginals on time trajectories are computed, the dynamic marginals of interest are easy to compute according to the following equations:

$$P_A^i(t) = \sum_{t' \le t} \mu_A^i(t'), \tag{D57}$$

$$P_B^i(t) = \sum_{t' \le t} \mu_B^i(t'), \tag{D58}$$

$$P_{AB}^i(t) = \sum_{t' \le t, t'' \le t} m_{t',t''}^i(t', t'') = P_A^i(t) + P_B^i(t) + P_S^i(t) - 1. \tag{D59}$$

4. Computation of $m_{t,t}^i(\tau_i^A, \tau_i^B)$ and $m_{t,t}^{i \to j}(\tau_i^A, \tau_i^B)$

It is convenient to break down the computation depending on a particular combination of (τ_i^A, τ_i^B), including the cases of times 0 and $*$. We provide a detailed derivation of one of the most general cases; all the others can be treated similarly. Let us consider the case of finite-time (τ_i^A, τ_i^B). Several cases of the type depicted in Fig. 9 need to be considered. For the sake of illustration, we consider the subcase $* \ne \tau_i^B > \tau_i^A > 0$.

Using Fact 2, the simplified DBP equations read

$$m_{t,t}^i(\tau_i^A, \tau_i^B) = \sum_{\{\tau_k^A, \tau_k^B\}_{k \in \partial i}} W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i})$$

$$\times \prod_{k \in \partial i} m_{t-1,t-1}^{k \to i}(\tau_k^A, \tau_k^B) \tag{D60}$$

and

$$m_{t,t}^{i \to j}(\tau_i^A, \tau_i^B) = \sum_{\{\tau_k^A, \tau_k^B\}_{k \in \partial i \backslash j}} W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k \in \partial i \backslash j}, \tau_j^A = *,$$

$$\tau_j^B = *) \prod_{k \in \partial i \backslash j} m_{t-1,t-1}^{k \to i}(\tau_k^A, \tau_k^B). \tag{D61}$$
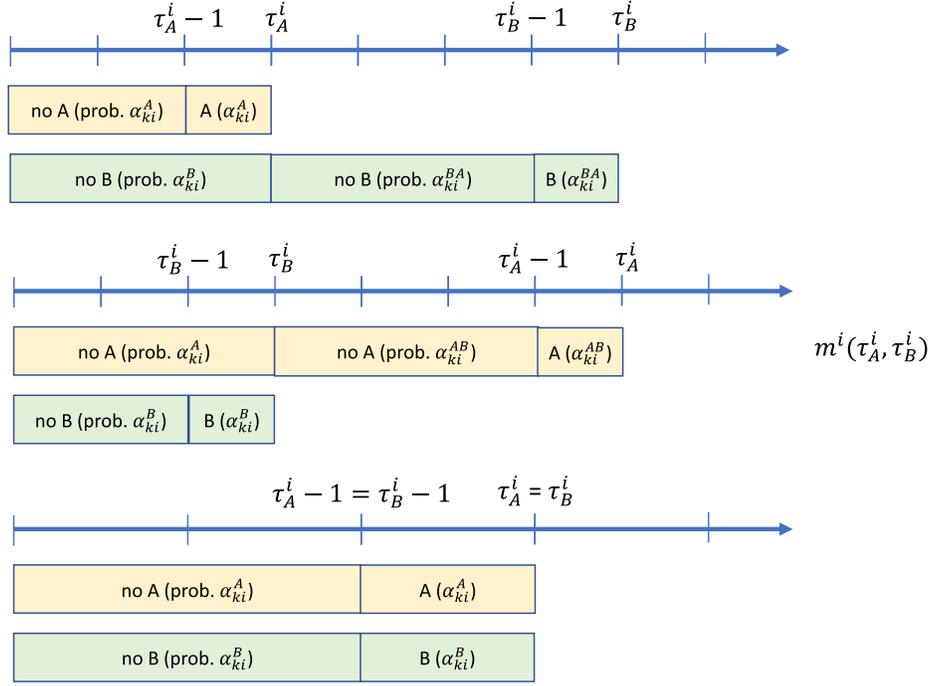
FIG. 9.   Three cases to consider for the marginals and messages for the time trajectory (τ_i^A, τ_i^B).

Using the definitions for the dynamics in the case considered, i.e., term (1) in $W^i(\tau_i^A, \tau_i^B; \{\tau_k^A, \tau_k^B\}_{k\in\partial i})$, as well as for the dynamic messages (26), we obtain

$$
m_{t,t}^i(\tau_i^A, \tau_i^B) = P_S^i(0)\Bigg[\prod_{k\in\partial i}\theta_{A,B,AB}^{k\to i}(\tau_i^A - 1, \tau_i^A, \tau_i^B - 1; \tau_i^A) - \prod_{k\in\partial i}\theta_{A,B,AB}^{k\to i}(\tau_i^A - 1, \tau_i^A, \tau_i^B; \tau_i^A)
$$
$$
- \prod_{k\in\partial i}\theta_{A,B,AB}^{k\to i}(\tau_i^A, \tau_i^A, \tau_i^B - 1; \tau_i^A) + \prod_{k\in\partial i}\theta_{A,B,AB}^{k\to i}(\tau_i^A, \tau_i^A, \tau_i^B; \tau_i^A)\Bigg],   \tag{D62}
$$

and, in the same way,

$$
m_{t,t}^{i\to j}(\tau_i^A, \tau_i^B) = P_S^i(0)\Bigg[\prod_{k\in\partial i\backslash j}\theta_{A,B,AB}^{k\to i}(\tau_i^A - 1, \tau_i^A, \tau_i^B - 1; \tau_i^A) - \prod_{k\in\partial i\backslash j}\theta_{A,B,AB}^{k\to i}(\tau_i^A - 1, \tau_i^A, \tau_i^B; \tau_i^A)
$$
$$
- \prod_{k\in\partial i\backslash j}\theta_{A,B,AB}^{k\to i}(\tau_i^A, \tau_i^A, \tau_i^B - 1; \tau_i^A) + \prod_{k\in\partial i\backslash j}\theta_{A,B,AB}^{k\to i}(\tau_i^A, \tau_i^A, \tau_i^B; \tau_i^A)\Bigg].   \tag{D63}
$$

Through the definition, we immediately get

$$
\theta_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB} + 1; \tau_i^A) = \theta_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB}; \tau_i^A) - \mathbb{1}[\tau_i^A \le t_{AB}]\alpha_{ki}^{BA}\phi_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB}; \tau_i^A),   \tag{D64}
$$

where the former quantity is defined through Eq. (D34). Working with this definition, we get the following update equation for $\phi_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB}; \tau_i^A)$:

$$
\phi_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB}; \tau_i^A\|\tau_i^A, \tau_i^B) = (1 - \alpha_{ki}^{BA})\phi_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB} - 1; \tau_i^A\|\tau_i^A, \tau_i^B)
$$
$$
+ \sum_{\tau_k^A}\prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A\mathbb{1}[\tau_k^A \le t'])\prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B\mathbb{1}[\tau_k^B \le t''])m_{t_{AB},t_{AB}}^{k\to i}(\tau_k^A, t_{AB}).   \tag{D65}
$$

The term $\prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B\mathbb{1}[\tau_k^B \le t''])$ is not active for the times considered here and is equal to 1 because $\tau_k^B = t_{AB}$ and $t_{AB} \ge t_B$; hence, we get the final update equation

$$\phi_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB}; \tau_i^A) = (1 - \alpha_{ki}^{BA})\phi_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB} - 1; \tau_i^A)$$

$$+ \sum_{\tau_k^A \le t_{AB}} \prod_{t'=0}^{t_A - 1} (1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \le t'])m_{t_{AB},t_{AB}}^{k\to i}(\tau_k^A, t_{AB}) + m_{t_{AB},t_{AB}}^{k\to i}(*, t_{AB}), \tag{D66}$$

where all $m_{t_{AB},t_{AB}}^{k\to i}(t', t'')$ are known from previous step computations.

5. Exact DMP equations for collaborative processes

In this section, we summarize the derivation procedure outlined above and state the full form of the DMP equations for collaborative processes.

(i) Initial conditions from the problem definition:

$$P_S^i(0), P_{A^*}^i(0), P_{B^*}^i(0), P_{AB}^i(0). \tag{D67}$$

(ii) Initialization of dynamic messages:

$$m_{0,0}^i(0,0) = P_{AB}^i(0), \qquad m_{0,0}^i(0,*) = P_{A^*}^i(0), \qquad m_{0,0}^i(*,0) = P_{B^*}^i(0), \tag{D68}$$

$$m_{0,0}^{i\to j}(0,0) = P_{AB}^i(0), \qquad m_{0,0}^{i\to j}(0,*) = P_{A^*}^i(0), \qquad m_{0,0}^{i\to j}(*,0) = P_{B^*}^i(0), \tag{D69}$$

$$\mu_A^i(0) = P_A^i(0), \qquad \mu_B^i(0) = P_B^i(0), \tag{D70}$$

$$\theta_{A,B}^{k\to i}(0,0) = \theta_{A,B,AB}^{k\to i}(0,0) = \theta_{B,A,BA}^{k\to i}(0,0) = \theta_{AB}^{k\to i}(0,0) = \theta_{BA}^{k\to i}(0,0) = 1, \tag{D71}$$

$$\phi_A^{k\to i}(0,0) = P_A^k(0), \qquad \phi_B^{k\to i}(0,0) = P_B^k(0), \qquad \phi_{A,B}^{k\to i}(0,0) = P_{AB}^k(0), \tag{D72}$$

$$\phi_{AB}^{k\to i}(0) = P_B^k(0), \qquad \phi_{BA}^{k\to i}(0) = P_A^k(0). \tag{D73}$$

(iii) Border conditions for dynamic messages:

$$\theta_A^{k\to i}(0, t) = \mu_B^{k\to i}(t), \qquad \theta_B^{k\to i}(t, 0) = \mu_A^{k\to i}(t), \tag{D74}$$

$$\theta_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB}; \tau_i^A) = \theta_{A,B}^{k\to i}(t_A, t_B) \quad \text{for } t_{AB} \le \tau_i^A, \tag{D75}$$

$$\theta_{B,A,BA}^{k\to i}(t_B, t_A, t_{BA}; \tau_i^B) = \theta_{A,B}^{k\to i}(t_A, t_B) \quad \text{for } t_{BA} \le \tau_i^B, \tag{D76}$$

$$\phi_{A,B,AB}^{k\to i}(t_A, t_B, t_{AB}; \tau_i^A) = \phi_B^{k\to i}(t_A, t_B) \quad \text{for } t_{AB} = t_B \quad \text{and} \quad t_{AB} \le \tau_i^A, \tag{D77}$$

$$\phi_{B,A,BA}^{k\to i}(t_B, t_A, t_{BA}; \tau_i^B) = \phi_A^{k\to i}(t_A, t_B) \quad \text{for } t_{BA} = t_A \quad \text{and} \quad t_{BA} \le \tau_i^B,$$

(true but no need to enforce) $\theta_{AB}^{k\to i}(t_B) = \theta_{A,B,AB}^{k\to i}(0, 0, t_B; 0), \qquad \theta_{BA}^{k\to i}(t_A) = \theta_{B,A,BA}^{k\to i}(0, 0, t_A; 0). \tag{D78}$

(iv) Expression for the dynamic marginals:

$$P_A^i(t) = \sum_{t' \le t} \mu_A^i(t'), \tag{D79}$$

$$P_B^i(t) = \sum_{t' \le t} \mu_B^i(t'), \tag{D80}$$

$$P_S^i(t) = m_{t,t}^i(*, *) = P_S^i(0) \prod_{k \in \partial i \backslash j} \theta_{A,B}^{k\to i}(t, t), \tag{D81}$$

$$P^i_{AB}(t) = \sum_{t' \le t, t'' \le t} m^i_{t',t''}(t', t'') = P^i_A(t) + P^i_B(t) + P^i_S(t) - 1. \tag{D82}$$

(v) Update equations:

$$\mu^i_A(t) = \sum_{t' \le t} m^i_{t,t}(t, t') + m^i_{t,t}(t, *), \tag{D83}$$

$$\mu^i_B(t) = \sum_{t' \le t} m^i_{t,t}(t', t) + m^i_{t,t}(*, t), \tag{D84}$$

$$\mu^{i \to j}_A(t) = \sum_{t' \le t} m^{i \to j}_{t,t}(t, t') + m^{i \to j}_{t,t}(t, *), \tag{D85}$$

$$\mu^{i \to j}_B(t) = \sum_{t' \le t} m^{i \to j}_{t,t}(t', t) + m^{i \to j}_{t,t}(*, t), \tag{D86}$$

$$
m^i_{t,t}(\tau^A_i, \tau^B_i) =
\begin{cases}
P^i_{A^*}(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{AB}(\tau^B_i - 1) - \prod_{k \in \partial i} \theta^{k \to i}_{AB}(\tau^B_i) \right] & \text{if } \tau^A_i = 0, \tau^B_i \ne 0 \\[2ex]
P^i_{B^*}(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{BA}(\tau^A_i - 1) - \prod_{k \in \partial i} \theta^{k \to i}_{BA}(\tau^A_i) \right] & \text{if } \tau^B_i = 0, \tau^A_i \ne 0 \\[2ex]
P^i_S(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{A,B,AB}(\tau^A_i - 1, \tau^A_i, \tau^B_i - 1; \tau^A_i) - \prod_{k \in \partial i} \theta^{k \to i}_{A,B,AB}(\tau^A_i - 1, \tau^A_i, \tau^B_i; \tau^A_i) \right. \\[1ex]
\left. - \prod_{k \in \partial i} \theta^{k \to i}_{A,B,AB}(\tau^A_i, \tau^A_i, \tau^B_i - 1; \tau^A_i) + \prod_{k \in \partial i} \theta^{k \to i}_{A,B,AB}(\tau^A_i, \tau^A_i, \tau^B_i; \tau^A_i) \right] & \text{if } \tau^B_i > \tau^A_i > 0 \\[2ex]
P^i_S(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(\tau^B_i - 1, \tau^B_i, \tau^A_i - 1; \tau^B_i) - \prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(\tau^B_i - 1, \tau^B_i, \tau^A_i; \tau^B_i) \right. \\[1ex]
\left. - \prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(\tau^B_i, \tau^B_i, \tau^A_i - 1; \tau^B_i) + \prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(\tau^B_i, \tau^B_i, \tau^A_i; \tau^B_i) \right] & \text{if } \tau^A_i > \tau^B_i > 0 \\[2ex]
P^i_S(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{A,B}(\tau^A_i - 1, \tau^A_i - 1) - \prod_{k \in \partial i} \theta^{k \to i}_{A,B}(\tau^A_i - 1, \tau^A_i) \right. \\[1ex]
\left. - \prod_{k \in \partial i} \theta^{k \to i}_{A,B}(\tau^A_i, \tau^A_i - 1) + \prod_{k \in \partial i} \theta^{k \to i}_{A,B}(\tau^A_i, \tau^A_i) \right] & \text{if } \tau^A_i = \tau^B_i \ne 0 \\[2ex]
P^i_{A^*}(0) \prod_{k \in \partial i} \theta^{k \to i}_{AB}(t) & \text{if } \tau^A_i = 0, \tau^B_i = * \\[2ex]
P^i_{B^*}(0) \prod_{k \in \partial i} \theta^{k \to i}_{BA}(t) & \text{if } \tau^B_i = 0, \tau^A_i = * \\[2ex]
P^i_S(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{A,B,AB}(\tau^A_i - 1, \tau^A_i, t; \tau^A_i) - \prod_{k \in \partial i} \theta^{k \to i}_{A,B,AB}(\tau^A_i, \tau^A_i, t; \tau^A_i) \right] & \text{if } \tau^A_i > 0, \tau^B_i = * \\[2ex]
P^i_S(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(\tau^B_i - 1, \tau^B_i, t; \tau^B_i) - \prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(\tau^B_i, \tau^B_i, t; \tau^B_i) \right] & \text{if } \tau^B_i > 0, \tau^A_i = * \\[2ex]
P^i_S(0) \prod_{k \in \partial i} \theta^{k \to i}_{A,B}(t, t) & \text{if } \tau^A_i = *, \tau^B_i = *.
\end{cases}
\tag{D87}
$$

The update expressions for the messages $m_{t,t}^{i \to j}(\tau_i^A, \tau_i^B)$ are similar to the expressions for the marginals $m_{t,t}^i(\tau_i^A, \tau_i^B)$ above, except ∂i is replaced with $\partial i \backslash j$:

$$
m_{t,t}^{i \to j}(\tau_i^A, \tau_i^B) =
\begin{cases}
P_{A^*}^i(0)\left[\displaystyle\prod_{k \in \partial i \backslash j} \theta_{AB}^{k \to i}(\tau_i^B - 1) - \prod_{k \in \partial i \backslash j} \theta_{AB}^{k \to i}(\tau_i^B) \right] & \text{if } \tau_i^A = 0, \tau_i^B \neq 0 \\[3em]

P_{B^*}^i(0)\left[\displaystyle\prod_{k \in \partial i \backslash j} \theta_{BA}^{k \to i}(\tau_i^A - 1) - \prod_{k \in \partial i \backslash j} \theta_{BA}^{k \to i}(\tau_i^A) \right] & \text{if } \tau_i^B = 0, \tau_i^A \neq 0 \\[3em]

P_S^i(0)\left[\displaystyle\prod_{k \in \partial i \backslash j} \theta_{A,B,AB}^{k \to i}(\tau_i^A - 1, \tau_i^A, \tau_i^B - 1; \tau_i^A) - \prod_{k \in \partial i \backslash j} \theta_{A,B,AB}^{k \to i}(\tau_i^A - 1, \tau_i^A, \tau_i^B; \tau_i^A) \right. & \\[2em]
\left. - \displaystyle\prod_{k \in \partial i \backslash j} \theta_{A,B,AB}^{k \to i}(\tau_i^A, \tau_i^A, \tau_i^B - 1; \tau_i^A) + \prod_{k \in \partial i \backslash j} \theta_{A,B,AB}^{k \to i}(\tau_i^A, \tau_i^A, \tau_i^B; \tau_i^A) \right] & \text{if } \tau_i^B > \tau_i^A > 0 \\[3em]

P_S^i(0)\left[\displaystyle\prod_{k \in \partial i \backslash j} \theta_{B,A,BA}^{k \to i}(\tau_i^B - 1, \tau_i^B, \tau_i^A - 1; \tau_i^B) - \prod_{k \in \partial i \backslash j} \theta_{B,A,BA}^{k \to i}(\tau_i^B - 1, \tau_i^B, \tau_i^A; \tau_i^B) \right. & \\[2em]
\left. - \displaystyle\prod_{k \in \partial i \backslash j} \theta_{B,A,BA}^{k \to i}(\tau_i^B, \tau_i^B, \tau_i^A - 1; \tau_i^B) + \prod_{k \in \partial i \backslash j} \theta_{B,A,BA}^{k \to i}(\tau_i^B, \tau_i^B, \tau_i^A; \tau_i^B) \right] & \text{if } \tau_i^A > \tau_i^B > 0 \\[3em]

P_S^i(0)\left[\displaystyle\prod_{k \in \partial i \backslash j} \theta_{A,B}^{k \to i}(\tau_i^A - 1, \tau_i^A - 1) - \prod_{k \in \partial i \backslash j} \theta_{A,B}^{k \to i}(\tau_i^A - 1, \tau_i^A) \right. & \\[2em]
\left. - \displaystyle\prod_{k \in \partial i \backslash j} \theta_{A,B}^{k \to i}(\tau_i^A, \tau_i^A - 1) + \prod_{k \in \partial i \backslash j} \theta_{A,B}^{k \to i}(\tau_i^A, \tau_i^A) \right] & \text{if } \tau_i^A = \tau_i^B \neq 0 \\[3em]

P_{A^*}^i(0) \displaystyle\prod_{k \in \partial i \backslash j} \theta_{AB}^{k \to i}(t) & \text{if } \tau_i^A = 0, \tau_i^B = * \\[2em]

P_{B^*}^i(0) \displaystyle\prod_{k \in \partial i \backslash j} \theta_{BA}^{k \to i}(t) & \text{if } \tau_i^B = 0, \tau_i^A = * \\[2em]

P_S^i(0)\left[\displaystyle\prod_{k \in \partial i \backslash j} \theta_{A,B,AB}^{k \to i}(\tau_i^A - 1, \tau_i^A, t; \tau_i^A) - \prod_{k \in \partial i \backslash j} \theta_{A,B,AB}^{k \to i}(\tau_i^A, \tau_i^A, t; \tau_i^A) \right] & \text{if } \tau_i^A > 0, \tau_i^B = * \\[3em]

P_S^i(0)\left[\displaystyle\prod_{k \in \partial i \backslash j} \theta_{B,A,BA}^{k \to i}(\tau_i^B - 1, \tau_i^B, t; \tau_i^B) - \prod_{k \in \partial i \backslash j} \theta_{B,A,BA}^{k \to i}(\tau_i^B, \tau_i^B, t; \tau_i^B) \right] & \text{if } \tau_i^B > 0, \tau_i^A = * \\[3em]

P_S^i(0) \displaystyle\prod_{k \in \partial i \backslash j} \theta_{A,B}^{k \to i}(t, t) & \text{if } \tau_i^A = *, \tau_i^B = *.
\end{cases}
$$

$$\tag{D88}$$

The dynamic messages entering the expressions above are updated as follows:

$$
\theta_{A,B}^{k \to i}(t+1, t+1) = \theta_{A,B}^{k \to i}(t, t) - \alpha_{ki}^A \phi_A^{k \to i}(t, t) - \alpha_{ki}^B \phi_B^{k \to i}(t, t) + \alpha_{ki}^A \alpha_{ki}^B \phi_{A,B}^{k \to i}(t, t),
\tag{D89}
$$

$$
\theta_{A,B}^{k \to i}(t+1, t) = \theta_{A,B}^{k \to i}(t, t) - \alpha_{ki}^A \phi_A^{k \to i}(t, t),
\tag{D90}
$$

$$
\theta_{A,B}^{k \to i}(t, t+1) = \theta_{A,B}^{k \to i}(t, t) - \alpha_{ki}^B \phi_B^{k \to i}(t, t),
\tag{D91}
$$

$$\phi_{A,B}^{k\to i}(t,t) = (1-\alpha_{ki}^A)(1-\alpha_{ki}^B)\phi_{A,B}^{k\to i}(t-1,t-1) - m_{t,t}^{k\to i}(t,t)$$
$$+ \sum_{\tau_k^A \le t}\prod_{t'=0}^{t-1}(1-\alpha_{ki}^A\mathbb{1}[t' \ge \tau_k^A])m_{t,t}^{k\to i}(\tau_k^A,t) + \sum_{\tau_k^B \le t}\prod_{t'=0}^{t-1}(1-\alpha_{ki}^B\mathbb{1}[t' \ge \tau_k^B])m_{t,t}^{k\to i}(t,\tau_k^B), \quad \text{(D92)}$$

$$\phi_A^{k\to i}(t-1,t) = \phi_A^{k\to i}(t-1,t-1) + (1-\alpha_{ki}^B)\phi_{A,B}^{k\to i}(t-1,t-1), \quad \text{(D93)}$$

$$\phi_B^{k\to i}(t,t-1) = \phi_B^{k\to i}(t-1,t-1) + (1-\alpha_{ki}^A)\phi_{A,B}^{k\to i}(t-1,t-1), \quad \text{(D94)}$$

$$\phi_A^{k\to i}(t,t) = (1-\alpha_{ki}^A)\phi_A^{k\to i}(t-1,t-1) - \alpha_{ki}^B(1-\alpha_{ki}^A)\phi_{A,B}^{k\to i}(t-1,t-1) + \theta_B^{k\to i}(t,t), \quad \text{(D95)}$$

$$\phi_B^{k\to i}(t,t) = (1-\alpha_{ki}^B)\phi_B^{k\to i}(t-1,t-1) - \alpha_{ki}^A(1-\alpha_{ki}^B)\phi_{A,B}^{k\to i}(t-1,t-1) + \theta_A^{k\to i}(t,t), \quad \text{(D96)}$$

$$\theta_A^{k\to i}(t,t) = \theta_A^{k\to i}(t-1,t) - \alpha_{ki}^A\sum_{\tau_k^A \le t-1}\prod_{t'=0}^{t-2}(1-\alpha_{ki}^A\mathbb{1}[t' \ge \tau_k^A])m_{t,t}^{k\to i}(\tau_k^A,t), \quad \text{(D97)}$$

$$\theta_B^{k\to i}(t,t) = \theta_B^{k\to i}(t,t-1) - \alpha_{ki}^B\sum_{\tau_k^B \le t-1}\prod_{t'=0}^{t-2}(1-\alpha_{ki}^B\mathbb{1}[t' \ge \tau_k^B])m_{t,t}^{k\to i}(t,\tau_k^B), \quad \text{(D98)}$$

$$\theta_{AB}^{k\to i}(t+1) = \theta_{AB}^{k\to i}(t) - \alpha_{ki}^{BA}\phi_{AB}^{k\to i}(t), \quad \text{(D99)}$$

$$\theta_{BA}^{k\to i}(t+1) = \theta_{BA}^{k\to i}(t) - \alpha_{ki}^{AB}\phi_{BA}^{k\to i}(t), \quad \text{(D100)}$$

$$\phi_{AB}^{k\to i}(t) = (1-\alpha_{ki}^{BA})\phi_{AB}^{k\to i}(t-1) + \mu_B^{k\to i}(t), \quad \text{(D101)}$$

$$\phi_{BA}^{k\to i}(t) = (1-\alpha_{ki}^{AB})\phi_{BA}^{k\to i}(t-1) + \mu_A^{k\to i}(t), \quad \text{(D102)}$$

$$\theta_{A,B,AB}^{k\to i}(t_A,t_B,t_{AB}+1;\tau_i^A) = \theta_{A,B,AB}^{k\to i}(t_A,t_B,t_{AB};\tau_i^A) - \mathbb{1}[\tau_i^A \le t_{AB}]\alpha_{ki}^{BA}\phi_{A,B,AB}^{k\to i}(t_A,t_B,t_{AB};\tau_i^A), \quad \text{(D103)}$$

$$\theta_{B,A,BA}^{k\to i}(t_B,t_A,t_{BA}+1;\tau_i^B) = \theta_{B,A,BA}^{k\to i}(t_B,t_A,t_{BA};\tau_i^B) - \mathbb{1}[\tau_i^B \le t_{BA}]\alpha_{ki}^{AB}\phi_{B,A,BA}^{k\to i}(t_B,t_A,t_{BA};\tau_i^B), \quad \text{(D104)}$$

$$\phi_{A,B,AB}^{k\to i}(t_A,t_B,t_{AB};\tau_i^A) = (1-\alpha_{ki}^{BA})\phi_{A,B,AB}^{k\to i}(t_A,t_B,t_{AB}-1;\tau_i^A$$
$$+ \sum_{\tau_k^A \le t_{AB}}\prod_{t'=0}^{t_A-1}(1-\alpha_{ki}^A\mathbb{1}[\tau_k^A \le t'])m_{t_{AB},t_{AB}}^{k\to i}(\tau_k^A,t_{AB}) + m_{t_{AB},t_{AB}}^{k\to i}(*,t_{AB}), \quad \text{(D105)}$$

$$\phi_{B,A,BA}^{k\to i}(t_B,t_A,t_{BA};\tau_i^B) = (1-\alpha_{ki}^{AB})\phi_{B,A,BA}^{k\to i}(t_B,t_A,t_{BA}-1;\tau_i^B$$
$$+ \sum_{\tau_k^B \le t_{BA}}\prod_{t'=0}^{t_B-1}(1-\alpha_{ki}^B\mathbb{1}[\tau_k^B \le t'])m_{t_{BA},t_{BA}}^{k\to i}(t_{BA},\tau_k^B) + m_{t_{BA},t_{BA}}^{k\to i}(t_{BA},*). \quad \text{(D106)}$$

(vi) Order of the updates:
(1) Only once, at time zero: initialize messages.
(2) If initialization is not given for a message, its initial value is fixed through a border condition (reduction to a different message) and will need to be initialized in this way for each update.
(3) Update ϕ's.
(4) Update θ's.
(5) Update m messages and marginals.
(6) Update μ messages and marginals.
(7) Compute dynamic marginals $P_S^i(t), P_A^i(t), P_B^i(t), P_{AB}^i(t)$.

APPENDIX E: APPROXIMATE DMP EQUATIONS FOR COLLABORATIVE PROCESSES

From the dynamic rules for the interacting spreading processes (D1)–(D4), we notice that the special case of transmission probabilities $\alpha_{ij}^{AB} = \alpha_{ij}^{A}$ and $\alpha_{ij}^{BA} = \alpha_{ij}^{B}$ corresponds to noninteracting spreading processes: Activation by one process does not change the activation dynamics for the other. In this case, the DMP equations should simplify into the product of two independent SI-like processes:

$$m_{t,t}^{i}(\tau_i^A, \tau_i^B) = \mu_A^i(\tau_i^A)\mu_B^i(\tau_i^B). \qquad (\text{E1})$$

We use this observation to produce a simplified version of DMP equations, expanding exact equations for interacting spreading processes around the noninteracting point. Several expansions are possible, including the most straightforward one where we expand to a certain order in $\alpha_{ij}^{A} - \alpha_{ij}^{AB}$ and $\alpha_{ij}^{B} - \alpha_{ij}^{BA}$, which will result in bulky expressions. Instead, here we produce a simplified set of DMP equations by keeping certain first-order corrections in the update equations only, so the resulting equations are similar to the SI-type equations. Using the approximation

$$m_{t,t}^{i}(\tau_i^A, \tau_i^B) \approx \mu_A^i(\tau_i^A)\mu_B^i(\tau_i^B) \qquad (\text{E2})$$

and the general expression

$$\mu_A^i(t) = \hat{P}_A^{i\to j}(t) - \hat{P}_A^{i\to j}(t-1) = \sum_{t' \le t} m_{t,t}^{i}(t, t') + m_{t,t}^{i}(t, *) = \sum_{t' \le t-1} m_{t,t-1}^{i}(t, t') + m_{t,t-1}^{i}(t, *), \qquad (\text{E3})$$

we compute the corresponding contribution of each term. For simplicity, in what follows, we assume that $\alpha_{ij} < 1$, and hence $\theta^{k\to i}$-type messages are nonzero (the treatment of the case of deterministic spreading is straightforward and results in several additional equations).

(i) Simplified approximate computation of $m_{t,t-1}^{i}(t, *)$: We start with the following approximations in the dynamic messages:

$$\theta_{A,B}^{k\to i}(t_A, t_B) = \sum_{\tau_k^A}\sum_{\tau_k^B}\prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \le t'])\prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \le t''])m_{T_A,T_B}^{k\to i}(\tau_k^A, \tau_k^B) \approx \hat{\theta}_A^{k\to i}(t_A)\hat{\theta}_B^{k\to i}(t), \qquad (\text{E4})$$

$$\phi_A^{k\to i}(t_A, t_B)$$
$$= \sum_{\tau_k^A}\sum_{\tau_k^B}\mathbb{1}[\tau_k^A \le t_A]\prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^A \mathbb{1}[\tau_k^A \le t'])\prod_{t''=0}^{t_B-1}(1 - \alpha_{ki}^B \mathbb{1}[\tau_k^B \le t''])m_{T_A,T_B}^{k\to i}(\tau_k^A, \tau_k^B) \approx \hat{\phi}_A^{k\to i}(t_A)\hat{\theta}_B^{k\to i}(t). \qquad (\text{E5})$$

The update equation

$$\theta_{A,B}^{k\to i}(t+1, t) = \theta_{A,B}^{k\to i}(t, t) - \alpha_{ki}^A \phi_A^{k\to i}(t, t) \qquad (\text{E6})$$

becomes, under this approximation,

$$\hat{\theta}_A^{k\to i}(t+1)\hat{\theta}_B^{k\to i}(t) = \hat{\theta}_A^{k\to i}(t)\hat{\theta}_B^{k\to i}(t) - \alpha_{ki}^A \hat{\phi}_A^{k\to i}(t)\hat{\theta}_B^{k\to i}(t), \qquad (\text{E7})$$

or simply

$$\hat{\theta}_A^{k\to i}(t+1) = \hat{\theta}_A^{k\to i}(t) - \alpha_{ki}^A \hat{\phi}_A^{k\to i}(t). \qquad (\text{E8})$$

Using these simplified expressions, after some algebra, we get

$$m_{t,t-1}^{i}(t, *) = P_S^i(0)\left[\prod_{k\in\partial i}\theta_{A,B}^{k\to i}(t-1, t-1) - \prod_{k\in\partial i}\theta_{A,B}^{k\to i}(t, t-1)\right] \qquad (\text{E9})$$

$$\approx P_S^i(0)\left[\prod_{k\in\partial i}\hat{\theta}_A^{k\to i}(t-1)\hat{\theta}_B^{k\to i}(t-1) - \prod_{k\in\partial i}\hat{\theta}_A^{k\to i}(t)\hat{\theta}_B^{k\to i}(t-1)\right] \qquad (\text{E10})$$

$$\approx P_S^i(0)\left[\prod_{k\in\partial i}\hat{\theta}_A^{k\to i}(t-1)\hat{\theta}_B^{k\to i}(t-1) - \prod_{k\in\partial i}\hat{\theta}_A^{k\to i}(t)\hat{\theta}_B^{k\to i}(t-1)\right] \tag{E11}$$

$$\approx P_S^i(0)\prod_{k\in\partial i}\hat{\theta}_A^{k\to i}(t-1)\hat{\theta}_B^{k\to i}(t-1)\left[1 - \prod_{k\in\partial i}\frac{\hat{\theta}_A^{k\to i}(t)}{\hat{\theta}_A^{k\to i}(t-1)}\right] \tag{E12}$$

$$\approx P_S^i(t-1)\left[1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_{ij}^A\hat{\phi}_A^{i\to j}(t-1)}{\hat{\theta}_A^{k\to i}(t-1)}\right)\right]. \tag{E13}$$

(ii) Simplified approximate computation of $\sum_{t'\le t-1}m_{t,t-1}^i(t,t')$: Similarly, we start with the approximations in the dynamic messages:

$$\theta_{B,A,BA}^{k\to i}(t_B,t',t_{BA};t') = \sum_{\tau_k^A}\sum_{\tau_k^B}\prod_{t=0}^{t_B-1}(1 - \alpha_{ki}^B\mathbb{1}[\tau_k^B\le t])\prod_{t''=0}^{t'-1}(1 - \alpha_{ki}^A\mathbb{1}[\tau_k^A\le t''])$$
$$\times\prod_{t'''=t'}^{t_{BA}-1}(1 - \alpha_{ki}^{AB}\mathbb{1}[\tau_k^A\le t'''])m_{T_A,T_B}^{k\to i}(\tau_k^A,\tau_k^B) \tag{E14}$$

$$\approx \hat{\theta}_A^{k\to i}(t_{BA})\hat{\theta}_B^{k\to i}(t_B), \tag{E15}$$

$$\phi_{B,A,BA}^{k\to i}(t_B,t',t_{BA};t') = \sum_{\tau_k^A}\sum_{\tau_k^B}\mathbb{1}[\tau_k^A\le t_{BA}]\prod_{t=0}^{t_B-1}(1 - \alpha_{ki}^B\mathbb{1}[\tau_k^B\le t])\prod_{t''=0}^{t'-1}(1 - \alpha_{ki}^A\mathbb{1}[\tau_k^A\le t''])$$
$$\times\prod_{t'''=t'}^{t_{BA}-1}(1 - \alpha_{ki}^{AB}\mathbb{1}[\tau_k^A\le t'''])m_{T_A,T_B}^{k\to i}(\tau_k^A,\tau_k^B) \tag{E16}$$

$$\approx \hat{\phi}_A^{k\to i}(t_{BA})\hat{\theta}_B^{k\to i}(t_B),, \tag{E17}$$

$$\theta_{BA}^{k\to i}(t_A) = \sum_{\tau_k^A}\prod_{t'=0}^{t_A-1}(1 - \alpha_{ki}^{AB}\mathbb{1}[\tau_k^A\le t'])\mu_A^{k\to i}(\tau_k^A) \approx \hat{\theta}_A^{k\to i}(t_A). \tag{E18}$$

The approximate version of the update equations,

$$\theta_{B,A,BA}^{k\to i}(t_B,t',t_{BA}+1;t') = \theta_{B,A,BA}^{k\to i}(t_B,t',t_{BA};t') - \mathbb{1}[t'\le t_{BA}]\alpha_{ki}^{AB}\phi_{B,A,BA}^{k\to i}(t_B,t',t_{BA};t'), \tag{E19}$$

then reads

$$\hat{\theta}_A^{k\to i}(t_{BA}+1)\hat{\theta}_B^{k\to i}(t_B) = \hat{\theta}_A^{k\to i}(t_{BA})\hat{\theta}_B^{k\to i}(t_B) - \mathbb{1}[t'\le t_{BA}]\alpha_{ki}^{AB}\hat{\phi}_A^{k\to i}(t_{BA})\hat{\theta}_B^{k\to i}(t_B), \tag{E20}$$

or simply

$$\hat{\theta}_A^{k\to i}(t_{BA}+1) = \hat{\theta}_A^{k\to i}(t_{BA}) - \mathbb{1}[t'\le t_{BA}]\alpha_{ki}^{AB}\hat{\phi}_A^{k\to i}(t_{BA}). \tag{E21}$$

Using these simplified expressions, after some algebra, we get

$$
\sum_{t' \leq t-1} m^i_{t,t-1}(t,t') = \sum_{0 < t' \leq t-1} P^i_S(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(t'-1,t',t-1;t') - \prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(t'-1,t',t;t') \right.
$$
$$
\left. - \prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(t',t',t-1;t') + \prod_{k \in \partial i} \theta^{k \to i}_{B,A,BA}(t',t',t;t') \right]
$$
$$
+ P^i_{B^*}(0) \left[\prod_{k \in \partial i} \theta^{k \to i}_{BA}(t-1) - \prod_{k \in \partial i} \theta^{k \to i}_{BA}(t) \right] \tag{E22}
$$

$$
= \sum_{0 < t' \leq t-1} P^i_S(0) \left[\prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1)\hat\theta^{k \to i}_B(t'-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t)\hat\theta^{k \to i}_B(t'-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1)\hat\theta^{k \to i}_B(t') \right.
$$
$$
\left. + \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t)\hat\theta^{k \to i}_B(t') + P^i_{B^*}(0) \left[\prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t) \right] \right] \tag{E23}
$$

$$
= \sum_{0 < t' \leq t-1} P^i_S(0) \left(\prod_{k \in \partial i} \hat\theta^{k \to i}_B(t'-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_B(t') \right) \left(\prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t) \right)
$$
$$
+ P^i_{B^*}(0) \left[\prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t) \right] \tag{E24}
$$

$$
= \left[\prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t) \right] \left[\sum_{0 < t' \leq t-1} P^i_S(0) \left(\prod_{k \in \partial i} \hat\theta^{k \to i}_B(t'-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_B(t') \right) + P^i_{B^*}(0) \right] \tag{E25}
$$

$$
= \left[\prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t) \right] \left[P^i_S(0) \left(\prod_{k \in \partial i} \hat\theta^{k \to i}_B(0) - \prod_{k \in \partial i} \hat\theta^{k \to i}_B(t-1) \right) + P^i_{B^*}(0) \right] \tag{E26}
$$

$$
\approx P^i_{B^*}(t-1) \left[\prod_{k \in \partial i} \hat\theta^{k \to i}_A(t-1) - \prod_{k \in \partial i} \hat\theta^{k \to i}_A(t) \right]. \tag{E27}
$$

(iii) *Final form of the approximate DMP equations:* The final update equations that we need,

$$
\phi^{k \to i}_A(t,t) = (1 - \alpha^A_{ki})\phi^{k \to i}_A(t-1,t) + \theta^{k \to i}_B(t,t), \tag{E28}
$$

with

$$
\theta^{k \to i}_B(t,t) \approx \mu^{k \to i}_A(t)\hat\theta^{k \to i}_B(t), \tag{E29}
$$

give

$$
\hat\phi^{i \to j}_A(t) = \hat\phi^{i \to j}_A(t-1) - \alpha^A_{ij}\hat\phi^{i \to j}_A(t-1) + \hat P^{i \to j}_A(t) - \hat P^{i \to j}_A(t-1). \tag{E30}
$$

Combining all of the computations above, we finally obtain the following approximate form of the DMP equations, which should provide a good approximation around the noninteracting point:

$$
\hat P^i_S(t) = P^i_S(0) \prod_{k \in \partial i \backslash j} \hat\theta^{k \to i}_A(t)\hat\theta^{k \to i}_B(t), \tag{E31}
$$

$$
\hat\theta^{i \to j}_A(t) = \hat\theta^{i \to j}_A(t-1) - \alpha^A_{ij}\hat\phi^{i \to j}_A(t-1), \tag{E32}
$$

$$
\hat\theta^{i \to j}_B(t) = \hat\theta^{i \to j}_B(t-1) - \alpha^B_{ij}\hat\phi^{i \to j}_B(t-1), \tag{E33}
$$

$$\hat{\phi}_A^{i\to j}(t) = \hat{\phi}_A^{i\to j}(t-1) - \alpha_{ij}^A\hat{\phi}_A^{i\to j}(t-1) + \hat{P}_A^{i\to j}(t) - \hat{P}_A^{i\to j}(t-1), \tag{E34}$$

$$\hat{\phi}_B^{i\to j}(t) = \hat{\phi}_B^{i\to j}(t-1) - \alpha_{ij}^B\hat{\phi}_B^{i\to j}(t-1) + \hat{P}_B^{i\to j}(t) - \hat{P}_B^{i\to j}(t-1), \tag{E35}$$

$$\hat{P}_A^{i\to j}(t) = \hat{P}_A^{i\to j}(t-1) + \hat{P}_S^i(t-1)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_{ij}^A\hat{\phi}_A^{i\to j}(t-1)}{\hat{\theta}_A^{k\to i}(t-1)}\right)\right]$$

$$+ \hat{P}_{B*}^{i\to j}(t-1)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_{ij}^{AB}\hat{\phi}_A^{i\to j}(t-1)}{\hat{\theta}_A^{k\to i}(t-1)}\right)\right], \tag{E36}$$

$$\hat{P}_B^{i\to j}(t) = \hat{P}_B^{i\to j}(t-1) + \hat{P}_S^i(t-1)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_{ij}^B\hat{\phi}_B^{i\to j}(t-1)}{\hat{\theta}_B^{k\to i}(t-1)}\right)\right]$$

$$+ \hat{P}_{A*}^{i\to j}(t-1)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_{ij}^{BA}\hat{\phi}_B^{i\to j}(t-1)}{\hat{\theta}_B^{k\to i}(t-1)}\right)\right]. \tag{E37}$$

In the main text, we numerically establish the approximation power of these equations. In what follows, we use these approximate DMP equations for performing optimization due to their simpler form and lower computational complexity.

APPENDIX F: OPTIMIZATION OF MUTUALLY EXCLUSIVE COMPETITIVE PROCESSES

For competitive processes, we study two optimization problems—the multiagent seeding problem and disease containment. The only difference between the two is the objective function. For disease containment, the objective function to be maximized is $\mathcal{O} = \sum_i (1 - P_i^B(T))$, and for the multiagent seeding problem, $\mathcal{O} = \sum_i (1 - P_i^S(T))$. The budget constraint at time zero is enforced by the Lagrange multiplier $\lambda^{\rm bu}$ as

$$\mathcal{B} = \lambda^{\rm bu}\left(B_\nu - \sum_i \nu^i(0)\right), \tag{F1}$$

where permitted $\underline{\nu} < \nu^i < \bar{\nu}$ value restrictions are enforced by the term

$$\mathcal{P} = \epsilon\sum_i \left(\log(\bar{\nu} - \nu^i(0)) + \log(\nu^i(0) - \underline{\nu})\right). \tag{F2}$$

In this case, the restrictions used are $\bar{\nu} = 1$ and $\underline{\nu} = 0$.

Initial conditions are forced through a set of Lagrange multipliers λ for the various parameter values:

$$\mathcal{I} = \sum_i \lambda_i^{\hat{A}}(0)(\hat{P}_A^i(0) - \nu^i(0)(1 - \delta_{\sigma_i(0),B})) + \sum_i \lambda_i^A(0)(P_A^i(0) - \nu^i(0)(1 - \delta_{\sigma_i(0),B})) + \sum_i \lambda_i^{\hat{B}}(0)(\hat{P}_B^i(0) - \delta_{\sigma_i(0),B})$$

$$+ \sum_i \lambda_i^B(0)(P_B^i(0) - \delta_{\sigma_i(0),B}) + \sum_i \lambda_i^{\hat{S}}(0)(\hat{P}_S^i(0) - 1 + v^i(0)(1 - \delta_{\sigma_i(0),B}) + \delta_{\sigma_i(0),B})$$

$$+ \sum_i \lambda_i^S(0)(P_S^i(0) - 1 + v^i(0)(1 - \delta_{\sigma_i(0),B}) + \delta_{\sigma_i(0),B}) + \sum_{ij} \lambda_{ij}^{\theta_B}(0)(\theta_B^{i\to j}(0) - 1)$$

$$+ \sum_{ij} \lambda_{ij}^{\theta_B}(0)(\theta_B^{i\to j}(0) - 1) + \sum_{ij} \lambda_{ij}^{\phi_A}(0)(\phi_A^{i\to j}(0) - \nu^i(0)(1 - \delta_{\sigma_i(0),B}))$$

$$+ \sum_{ij} \lambda_{ij}^{\phi_B}(0)(\phi_B^{i\to j}(0) - \delta_{\sigma_i(0),B}) + \sum_{ij} \lambda_{ij}^{\hat{A}}(0)(\hat{P}_A^{i\to j}(0) - \nu^i(0)(1 - \delta_{\sigma_i(0),B}))$$

$$+ \sum_{ij} \lambda_{ij}^{\hat{B}}(0)(\hat{P}_B^{i\to j}(0) - \delta_{\sigma_i(0),B}) + \sum_{ij} \lambda_{ij}^A(0)(P_A^{i\to j}(0) - \nu^i(0)(1 - \delta_{\sigma_i(0),B}))$$

$$+ \sum_{ij} \lambda_{ij}^B(0)(P_B^{i\to j}(0) - \delta_{\sigma_i(0),B}) + \sum_{ij} \lambda_{ij}^{\hat{S}}(0)(\hat{P}_S^{i\to j}(0) - 1 + \delta_{\sigma_i(0),B} + \nu^i(0)(1 - \delta_{\sigma_i(0),B}))$$

$$+ \sum_{ij} \lambda_{ij}^S(0)(P_S^{i\to j}(0) - 1 + \delta_{\sigma_i(0),B} + \nu^i(0)(1 - \delta_{\sigma_i(0),B})). \tag{F3}$$

The DMP equations for the dynamics are forced through a set of Lagrange multipliers,

$$
\begin{aligned}
\mathcal{D} = &\sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{\theta_A}(t+1)[\theta_A^{i\to j}(t+1) - \theta_A^{i\to j}(t) + \alpha_A \phi_A^{i\to j}(t)] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{\theta_B}(t+1)[\theta_B^{i\to j}(t+1) - \theta_B^{i\to j}(t) + \alpha_B \phi_B^{i\to j}(t)] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{\phi_A}(t+1)[\phi_A^{i\to j}(t+1) + (\alpha_A - 1)\phi_A^{i\to j}(t) - [P_A^{i\to j}(t+1) - P_A^{i\to j}(t)]] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{\phi_B}(t+1)[\phi_B^{i\to j}(t+1) + (\alpha_B - 1)\phi_B^{i\to j}(t) - [P_B^{i\to j}(t+1) - P_B^{i\to j}(t)]] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{\hat{S}}(t+1)\left[\hat{P}_S^{i\to j}(t+1) - P_S^{i\to j}(t)\prod_{k\in\partial i\backslash j} \frac{\theta_A^{k\to i}(t+1)\theta_B^{k\to i}(t+1)}{\theta_A^{k\to i}(t)\theta_B^{k\to i}(t)}\right] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{\hat{A}}(t+1)\left[\hat{P}_A^{i\to j}(t+1) - P_A^{i\to j}(t) - P_S^{i\to j}(t)\prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_B \phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\left(1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_A \phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\right)\right] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{\hat{B}}(t+1)\left[\hat{P}_B^{i\to j}(t+1) - P_B^{i\to j}(t) - P_S^{i\to j}(t)\prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_A \phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\left(1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_B \phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right)\right] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{S}(t+1)\left[P_S^{i\to j}(t+1) - \frac{\hat{P}_S^{i\to j}(t+1)}{\hat{P}_S^{i\to j}(t+1) + \hat{P}_A^{i\to j}(t+1) + \hat{P}_B^{i\to j}(t+1)}\right] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{A}(t+1)\left[P_A^{i\to j}(t+1) - \frac{\hat{P}_A^{i\to j}(t+1)}{\hat{P}_S^{i\to j}(t+1) + \hat{P}_A^{i\to j}(t+1) + \hat{P}_B^{i\to j}(t+1)}\right] \\
&+ \sum_{ij}\sum_{t=0}^{T-1} \lambda_{ij}^{B}(t+1)\left[P_B^{i\to j}(t+1) - \frac{\hat{P}_B^{i\to j}(t+1)}{\hat{P}_S^{i\to j}(t+1) + \hat{P}_A^{i\to j}(t+1) + \hat{P}_B^{i\to j}(t+1)}\right] \\
&+ \sum_{i}\sum_{t=0}^{T-1} \lambda_i^{\hat{S}}(t+1)\left[\hat{P}_S^{i}(t+1) - P_S^{i}(t)\prod_{k\in\partial i} \frac{\theta_A^{k\to i}(t+1)\theta_B^{k\to i}(t+1)}{\theta_A^{k\to i}(t)\theta_B^{k\to i}(t)}\right] \\
&+ \sum_{i}\sum_{t=0}^{T-1} \lambda_i^{\hat{A}}(t+1)\left[\hat{P}_A^{i}(t+1) - P_A^{i}(t) - P_S^{i}(t)\prod_{k\in\partial i}\left(1 - \frac{\alpha_B \phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_A \phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\right)\right] \\
&+ \sum_{i}\sum_{t=0}^{T-1} \lambda_i^{\hat{A}}(t+1)\left[\hat{P}_A^{i}(t+1) - P_A^{i}(t) - P_S^{i}(t)\prod_{k\in\partial i}\left(1 - \frac{\alpha_B \phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_A \phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\right)\right] \\
&+ \sum_{i}\sum_{t=0}^{T-1} \lambda_i^{\hat{B}}(t+1)\left[\hat{P}_B^{i}(t+1) - P_B^{i}(t) - P_S^{i}(t)\prod_{k\in\partial i}\left(1 - \frac{\alpha_A \phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_B \phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right)\right] \\
&+ \sum_{i}\sum_{t=0}^{T-1} \lambda_i^{S}(t+1)\left[P_S^{i}(t+1) - \frac{\hat{P}_S^{i}(t+1)}{\hat{P}_S^{i}(t+1) + \hat{P}_A^{i}(t+1) + \hat{P}_B^{i}(t+1)}\right] \\
&+ \sum_{i}\sum_{t=0}^{T-1} \lambda_i^{A}(t+1)\left[P_A^{i}(t+1) - \frac{\hat{P}_A^{i}(t+1)}{\hat{P}_S^{i}(t+1) + \hat{P}_A^{i}(t+1) + \hat{P}_B^{i}(t+1)}\right] \\
&+ \sum_{i}\sum_{t=0}^{T-1} \lambda_i^{B}(t+1)\left[P_B^{i}(t+1) - \frac{\hat{P}_B^{i}(t+1)}{\hat{P}_S^{i}(t+1) + \hat{P}_A^{i}(t+1) + \hat{P}_B^{i}(t+1)}\right].
\end{aligned}
\tag{F4}
$$

Derivatives with respect to the dynamics parameters give rise to the optimization (dynamical) equations for the Lagrange multipliers. The case described here is that of the containment problem (minimizing spread of an adversarial process):

$$\frac{\partial \mathcal{L}}{\partial \hat{P}_S^{i\rightarrow j}(t)} = \lambda_{ij}^{\hat{S}}(t) + 1[t \neq 0]\left[-\lambda_{ij}^S(t)\frac{\hat{P}_A^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t)}{(\hat{P}_S^{i\rightarrow j}(t) + \hat{P}_A^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t))^2} + \lambda_{ij}^A(t)\frac{\hat{P}_A^{i\rightarrow j}(t)}{(\hat{P}_S^{i\rightarrow j}(t) + \hat{P}_A^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t))^2}\right.$$
$$\left. + \lambda_{ij}^B(t)\frac{\hat{P}_B^{i\rightarrow j}(t)}{(\hat{P}_S^{i\rightarrow j}(t) + \hat{P}_A^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t))^2}\right],$$

$$\frac{\partial \mathcal{L}}{\partial \hat{P}_S^i(t)} = \lambda_i^{\hat{S}}(t) + 1[t \neq 0]\left[-\lambda_i^S(t)\frac{\hat{P}_A^i(t) + \hat{P}_B^i(t)}{(\hat{P}_S^i(t) + \hat{P}_A^i(t) + \hat{P}_B^i(t))^2} + \lambda_i^A(t)\frac{\hat{P}_A^i(t)}{(\hat{P}_S^i(t) + \hat{P}_A^i(t) + \hat{P}_B^i(t))^2}\right.$$
$$\left. + \lambda_i^B(t)\frac{\hat{P}_B^i(t)}{(\hat{P}_S^i(t) + \hat{P}_A^i(t) + \hat{P}_B^i(t))^2}\right],$$

$$\frac{\partial \mathcal{L}}{\partial P_S^{i\rightarrow j}(t)} = \left[-\lambda_{ij}^{\hat{S}}(t+1)\prod_{k\in\partial i\setminus j}\frac{\theta_A^{k\rightarrow i}(t+1)\theta_B^{k\rightarrow i}(t+1)}{\theta_A^{k\rightarrow i}(t)\theta_B^{k\rightarrow i}(t)} - \lambda_{ij}^{\hat{A}}(t+1)\prod_{k\in\partial i\setminus j}\left(1 - \frac{\alpha_B\phi_B^{k\rightarrow i}(t)}{\theta_B^{k\rightarrow i}(t)}\right)\left(\left(1 - \prod_{k\in\partial i\setminus j}\left(1 - \frac{\alpha_A\phi_A^{k\rightarrow i}(t)}{\theta_A^{k\rightarrow i}(t)}\right)\right)\right)\right.$$
$$\left. - \lambda_{ij}^{\hat{B}}(t+1)\prod_{k\in\partial i\setminus j}\left(1 - \frac{\alpha_A\phi_A^{k\rightarrow i}(t)}{\theta_A^{k\rightarrow i}(t)}\right)\left(1 - \prod_{k\in\partial i\setminus j}\left(1 - \frac{\alpha_B\phi_B^{k\rightarrow i}(t)}{\theta_B^{k\rightarrow i}(t)}\right)\right)\right]1[t\neq T] + \lambda_{ij}^S(t),$$

$$\frac{\partial \mathcal{L}}{\partial P_S^i(t)} = \left[-\lambda_i^{\hat{S}}(t+1)\prod_{k\in\partial i}\frac{\theta_A^{k\rightarrow i}(t+1)\theta_B^{k\rightarrow i}(t+1)}{\theta_A^{k\rightarrow i}(t)\theta_B^{k\rightarrow i}(t)} - \lambda_i^{\hat{A}}(t+1)\prod_{k\in\partial i}\left(1 - \frac{\alpha_B\phi_B^{k\rightarrow i}(t)}{\theta_B^{k\rightarrow i}(t)}\right)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_A\phi_A^{k\rightarrow i}(t)}{\theta_A^{k\rightarrow i}(t)}\right)\right)\right.$$
$$\left. - \lambda_i^{\hat{B}}(t+1)\prod_{k\in\partial i}\left(1 - \frac{\alpha_A\phi_A^{k\rightarrow i}(t)}{\theta_A^{k\rightarrow i}(t)}\right)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_B\phi_B^{k\rightarrow i}(t)}{\theta_B^{k\rightarrow i}(t)}\right)\right)\right]1[t\neq T] + \lambda_i^S(t),$$

$$\frac{\partial \mathcal{L}}{\partial P_A^{i\rightarrow j}(t)} = [\lambda_{ij}^{\phi_A}(t+1) - \lambda_{ij}^{\hat{A}}(t+1)]1[t\neq T] + \lambda_{ij}^A(t) - 1[t\neq 0][\lambda_{ij}^{\phi_A}(t)],$$

$$\frac{\partial \mathcal{L}}{\partial P_A^i(t)} = [-\lambda_i^{\hat{A}}(t+1)]1[t\neq T] + \lambda_i^A(t),$$

$$\frac{\partial \mathcal{L}}{\partial \hat{P}_A^{i\rightarrow j}(t)} = \lambda_{ij}^{\hat{A}}(t) + 1[t\neq 0]\left[-\lambda_{ij}^A(t)\frac{\hat{P}_S^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t)}{(\hat{P}_S^{i\rightarrow j}(t) + \hat{P}_A^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t))^2} + \lambda_{ij}^S(t)\frac{\hat{P}_S^{i\rightarrow j}(t)}{(\hat{P}_S^{i\rightarrow j}(t) + \hat{P}_A^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t))^2}\right.$$
$$\left. + \lambda_{ij}^B(t)\frac{\hat{P}_B^{i\rightarrow j}(t)}{(\hat{P}_S^{i\rightarrow j}(t) + \hat{P}_A^{i\rightarrow j}(t) + \hat{P}_B^{i\rightarrow j}(t))^2}\right],$$

$$\frac{\partial \mathcal{L}}{\partial \hat{P}_A^i(t)} = \lambda_i^{\hat{A}}(t) + 1[t\neq 0]\left[-\lambda_i^A(t)\frac{\hat{P}_S^i(t) + \hat{P}_B^i(t)}{(\hat{P}_S^i(t) + \hat{P}_A^i(t) + \hat{P}_B^i(t))^2} + \lambda_i^S(t)\frac{\hat{P}_S^i(t)}{(\hat{P}_S^i(t) + \hat{P}_A^i(t) + \hat{P}_B^i(t))^2}\right.$$
$$\left. + \lambda_i^B(t)\frac{\hat{P}_B^i(t)}{(\hat{P}_S^i(t) + \hat{P}_A^i(t) + \hat{P}_B^i(t))^2}\right],$$

$$\frac{\partial \mathcal{L}}{\partial \theta_A^{i\rightarrow j}(t)} = \lambda_{ij}^{\theta_A}(t) - 1[t\neq T](\lambda_{ij}^{\theta_A}(t+1)) + \sum_{a\in\partial j\setminus i}\lambda_{ja}^{\hat{S}}(t+1)P_S^{j\rightarrow a}(t)\frac{1}{\theta_A^{i\rightarrow j}(t)}\prod_{l\in\partial j\setminus a}\frac{\theta_A^{l\rightarrow j}(t+1)\theta_B^{l\rightarrow j}(t+1)}{\theta_A^{l\rightarrow j}(t)\theta_B^{l\rightarrow j}(t)}1[t\neq T]$$
$$- \sum_{a\in\partial j\setminus i}\lambda_{ja}^{\hat{S}}(t)P_S^{j\rightarrow a}(t-1)\frac{1}{\theta_A^{i\rightarrow j}(t)}\prod_{l\in\partial j\setminus a}\frac{\theta_A^{l\rightarrow j}(t)\theta_B^{l\rightarrow j}(t)}{\theta_A^{l\rightarrow j}(t-1)\theta_B^{l\rightarrow j}(t-1)}1[t\neq 0] + \lambda_j^{\hat{S}}(t+1)P_S^j(t)\frac{1}{\theta_A^{i\rightarrow j}(t)}$$
$$\times \prod_{l\in\partial j}\frac{\theta_A^{l\rightarrow j}(t+1)\theta_B^{l\rightarrow j}(t+1)}{\theta_A^{l\rightarrow j}(t)\theta_B^{l\rightarrow j}(t)}1[t\neq T] - \lambda_j^{\hat{S}}(t)P_S^j(t-1)\frac{1}{\theta_A^{i\rightarrow j}(t)}\prod_{l\in\partial j}\frac{\theta_A^{l\rightarrow j}(t)\theta_B^{l\rightarrow j}(t)}{\theta_A^{l\rightarrow j}(t-1)\theta_B^{l\rightarrow j}(t-1)}1[t\neq 0]$$
$$- \sum_{a\in\partial j\setminus i}\lambda_{ja}^{\hat{A}}(t+1)P_S^{j\rightarrow a}(t)\left[1 - \frac{\alpha_A\phi_A^{i\rightarrow j}}{(\theta_A^{i\rightarrow j}(t))^2}\prod_{l\in\partial j\setminus a,i}\left(1 - \frac{\alpha_A\phi_A^{l\rightarrow j}(t)}{\theta_A^{l\rightarrow j}(t)}\right)\right]\prod_{l\in\partial j\setminus a}\left(1 - \frac{\alpha_B\phi_B^{l\rightarrow j}(t)}{\theta_B^{l\rightarrow j}(t)}\right)1[t\neq T]$$

$$- \sum_{a\in\partial j\setminus i} \lambda_{ja}^{\hat{B}}(t+1) P_S^{j\to a}(t) \left[\frac{\alpha_A \phi_A^{i\to j}}{(\theta_A^{i\to j}(t))^2} \prod_{l\in\partial j\setminus a,i} \left(1 - \frac{\alpha_A \phi_A^{l\to j}(t)}{\theta_A^{l\to j}(t)}\right)\right] \left(1 - \prod_{l\in\partial j\setminus a}\left(1 - \frac{\alpha_B \phi_B^{l\to j}(t)}{\theta_B^{l\to j}(t)}\right)\right) 1[t\neq T]$$

$$- \lambda_j^{\hat{A}}(t+1) P_S^j(t) \left[1 - \frac{\alpha_A \phi_A^{i\to j}}{(\theta_A^{i\to j}(t))^2} \prod_{l\in\partial j\setminus i} \left(1 - \frac{\alpha_A \phi_A^{l\to j}(t)}{\theta_A^{l\to j}(t)}\right)\right] \prod_{l\in\partial j}\left(1 - \frac{\alpha_B \phi_B^{l\to j}(t)}{\theta_B^{l\to j}(t)}\right) 1[t\neq T]$$

$$- \lambda_j^{\hat{B}}(t+1) P_S^j(t) \left[\frac{\alpha_A \phi_A^{i\to j}}{(\theta_A^{i\to j}(t))^2} \prod_{l\in\partial j\setminus i} \left(1 - \frac{\alpha_A \phi_A^{l\to j}(t)}{\theta_A^{l\to j}(t)}\right)\right] \left(1 - \prod_{l\in\partial j}\left(1 - \frac{\alpha_B \phi_B^{l\to j}(t)}{\theta_B^{l\to j}(t)}\right)\right) 1[t\neq T], \tag{F5}$$

$$\frac{\partial \mathcal{L}}{\partial \phi_A^{i\to j}(t)} = \lambda_{ij}^{\phi_A}(t) + [p\lambda_{ij}^{\theta_A}(t+1) + (\alpha_A - 1)\lambda_{ij}^{\phi_A}(t+1)]1[t\neq T]$$

$$- \sum_{a\in\partial j\setminus i} \lambda_{ja}^{\hat{A}}(t+1) P_S^{j\to a}(t) \left[\prod_{l\in\partial j\setminus a}\left(1 - \frac{\alpha_B \phi_B^{l\to j}(t)}{\theta_B^{l\to j}(t)}\right)\right] \left(1 + \frac{\alpha_A}{\theta_A^{i\to j}(t)} \prod_{l\in\partial j\setminus a,i}\left(1 - \frac{\alpha_A \phi_A^{l\to j}(t)}{\theta_A^{l\to j}(t)}\right)\right) 1[t\neq T]$$

$$+ \sum_{a\in\partial j\setminus i} \lambda_{ja}^{\hat{B}}(t+1) P_S^{j\to a}(t) \frac{\alpha_A}{\theta_A^{i\to j}(t)} \left[\prod_{l\in\partial j\setminus a,i}\left(1 - \frac{\alpha_A \phi_A^{l\to j}(t)}{\theta_A^{l\to j}(t)}\right)\right] \left(1 - \prod_{l\in\partial j\setminus a}\left(1 - \frac{\alpha_B \phi_B^{l\to j}(t)}{\theta_B^{l\to j}(t)}\right)\right) 1[t\neq T]$$

$$- \lambda_j^{\hat{A}}(t+1) P_S^j(t) \left[\prod_{l\in\partial j}\left(1 - \frac{\alpha_B \phi_B^{l\to j}(t)}{\theta_B^{l\to j}(t)}\right)\right] \left(1 + \frac{\alpha_A}{\theta_A^{i\to j}(t)} \prod_{l\in\partial j\setminus i}\left(1 - \frac{\alpha_A \phi_A^{l\to j}(t)}{\theta_A^{l\to j}(t)}\right)\right) 1[t\neq T]$$

$$+ \lambda_j^{\hat{B}}(t+1) P_S^j(t) \frac{\alpha_A}{\theta_A^{i\to j}(t)} \left[\prod_{l\in\partial j\setminus i}\left(1 - \frac{\alpha_A \phi_A^{l\to j}(t)}{\theta_A^{l\to j}(t)}\right)\right] \left(1 - \prod_{l\in\partial j}\left(1 - \frac{\alpha_B \phi_B^{l\to j}(t)}{\theta_B^{l\to j}(t)}\right)\right) 1[t\neq T] \tag{F6}$$

$$\frac{\partial \mathcal{L}}{\partial \nu^i(0)} = -\lambda^{\mathrm{bu}}(0) - \lambda_i^{\hat{A}}(0) - \lambda_i^{A}(0) - \sum_j \lambda_{ij}^{\hat{A}}(0) - \sum_j \lambda_{ij}^{A}(0) - \sum_j \lambda_{ij}^{\phi_A}(0)$$

$$+ \lambda_i^{\hat{S}}(0) + \lambda_i^{S}(0) + \sum_j \lambda_{ij}^{\hat{S}}(0) + \sum_j \lambda_{ij}^{S}(0) + \epsilon\left(\frac{1}{\nu^i(0)} - \frac{1}{1 - \nu^i(0)}\right) = 0. \tag{F7}$$

The equations for $\partial \mathcal{L}/\partial P_B^{i\to j}(t)$, $\partial \mathcal{L}/\partial P_B^i(t)$, $\partial \mathcal{L}/\partial \hat{P}_B^{i\to j}(t)$, $\partial \mathcal{L}/\partial \hat{P}_B^i(t)$, $\partial \mathcal{L}/\partial \theta_B^{i\to j}(t)$, and $\partial \mathcal{L}/\partial \phi_B^{i\to j}(t)$ are similar to their A process counterparts, with an exchange of variables $A \leftrightarrow B$.

For simplicity, one can write Eq. (F7) as

$$\frac{\partial \mathcal{L}}{\partial \nu^i(0)} = -\lambda^{\mathrm{bu}}(0) + \psi_i + \epsilon\left(\frac{1}{\nu^i(0)} - \frac{1}{1 - \nu^i(0)}\right) = 0. \tag{F8}$$

The same method as in Ref. [76] is used here to solve the quadratic equation (F8), writing $\nu^i(0)$ as a function of ψ_i:

$$\nu^i(0) = \frac{\lambda^{\mathrm{bu}}(0) + \psi_i - 2\epsilon \pm \sqrt{(-\lambda^{\mathrm{bu}}(0) + \psi_i)^2 + 4\epsilon^2}}{-2\lambda^{\mathrm{bu}}(0) + 2\psi_i}. \tag{F9}$$

In this scenario, the positive square root solution satisfies restriction (F2). Given the budget restriction, one can obtain $\nu^i(0)$ numerically through

$$\sum_i \nu^i(0) = B_\nu. \tag{F10}$$

One can use the following update procedure to obtain the optimal solution iteratively:

Step 1: Carry out the forward iteration using the given initial conditions, Eqs. (C7)–(C10).

Step 2: Compute all Lagrangian multipliers at $t = T$.

Step 3: Compute $\lambda_{ij}^S(t-1), \lambda_i^S(t-1), \lambda_i^A(t-1), \lambda_i^B(t-1), \lambda_{ij}^{\phi_A}(t-1), \lambda_{ij}^{\phi_B}(t-1)$ using the obtained Lagrangian multipliers at time t from Eqs. 216 and (F6).

Step 4: Compute $\lambda_{ij}^A(t-1), \lambda_{ij}^B(t-1), \lambda_i^{\hat{A}}(t-1), \lambda_i^{\hat{B}}(t-1), \lambda_i^{\hat{S}}(t-1)$ using the obtained Lagrangian multipliers and Eqs. 216 and (F6).

Step 5: Compute $\lambda_{ij}^{\hat{A}}(t-1), \lambda_{ij}^{\hat{B}}(t-1), \lambda_{ij}^{\hat{S}}(t-1)$ using the obtained Lagrangian multipliers and Eqs. 216 and (F6).

Step 6: Compute $\lambda_{ij}^{\theta_A}(t-1), \lambda_{ij}^{\theta_B}(t-1)$ using the obtained Lagrangian multipliers.

Step 7: Repeat steps 1–5 back in time until all Lagrangian multipliers have been obtained for the range $t = 0, \ldots, T$.

Step 8: Solve Eqs. (F9) and (F10) numerically and update ν^i.

Step 9: Repeat steps 1–8 until convergence.

Step 10: Calculate the posterior marginals on the basis of the messages using Eq. (C10).

A similar approach is used in collaborative processes using the corresponding equations for the dynamics and the Lagrange multipliers' dynamics.

APPENDIX G: OPTIMIZATION OF COMPETITIVE PROCESSES—EXAMPLE

We also demonstrate the efficacy of the DMP-optimal algorithm on the denser network of the 1994 world metal trade network [95]] shown in Fig. 10. The network consists of 80 countries; we ignore their respective trade volumes and use the dense topology for the current example. The infection probabilities used for the two processes are $\alpha_A = \alpha_B = 0.5$; the budget for B is 1, allocated at node 1 (Argentina) at $t = 0$, and a budget of 1 per time step is assigned using the DMP-optimal strategy for process A within the time window of $T = 3$. The results are shown in Fig. 11, where the subfigures
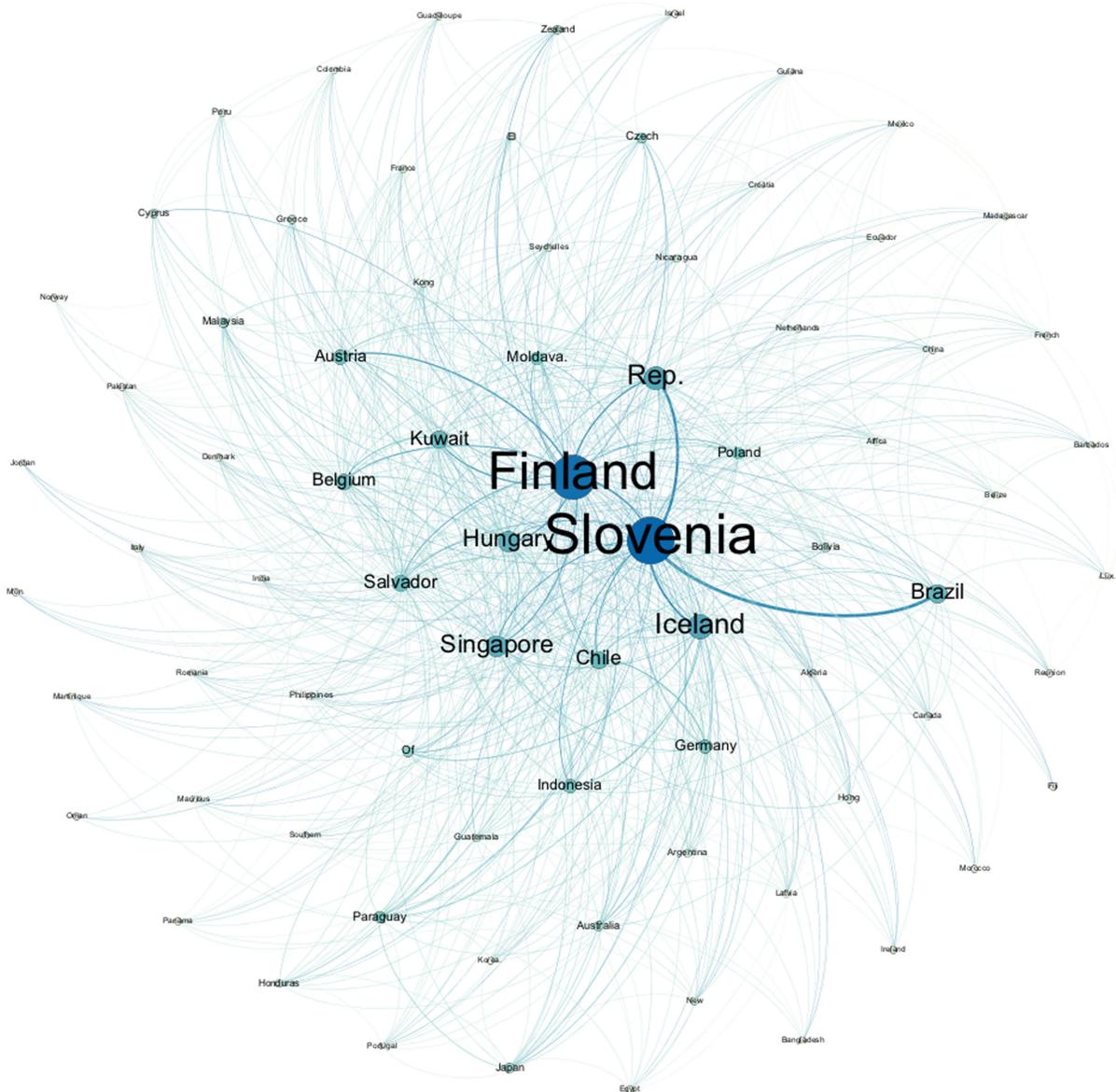

FIG. 10.   The 1994 network of world trade in metal. The size of the nodes represents their degree [94].
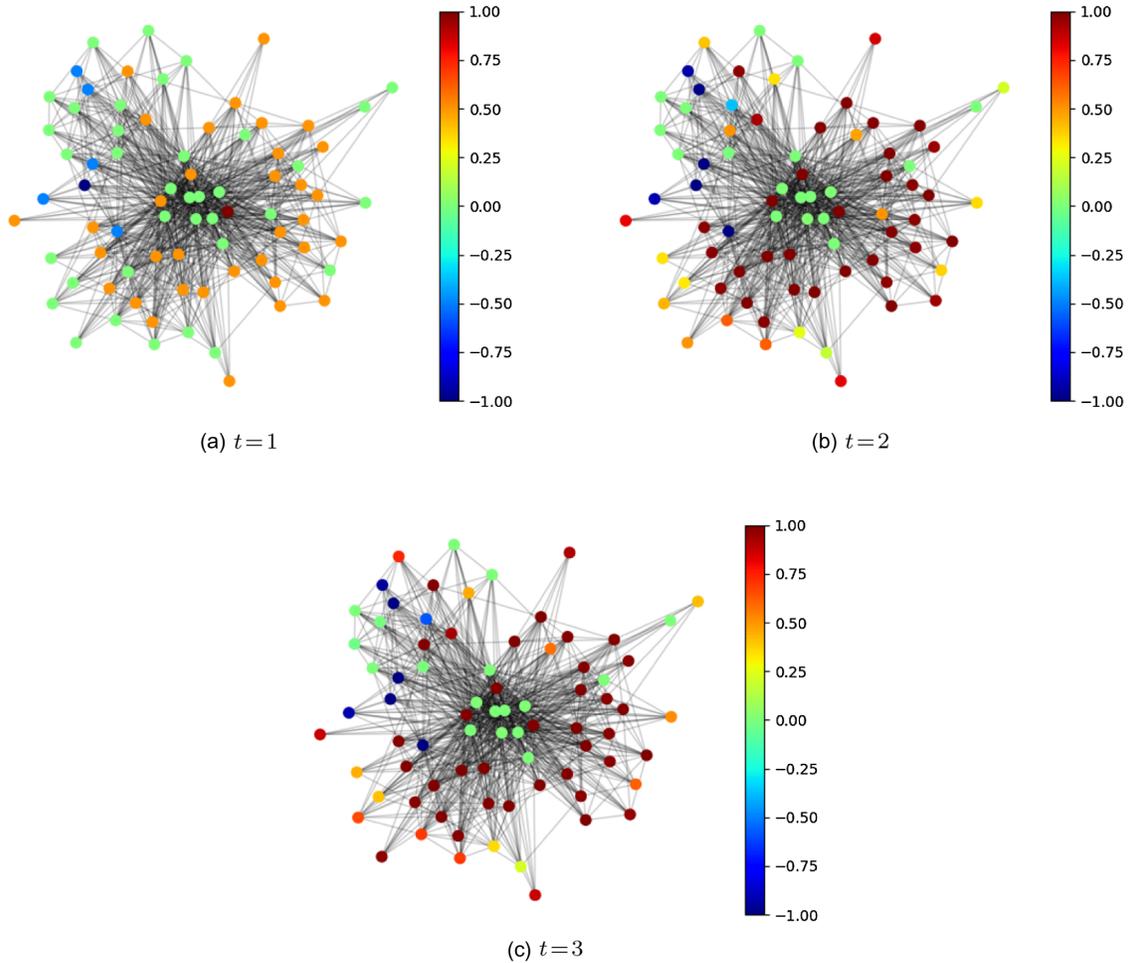
FIG. 11. Example of optimization of competitive processes on the 1994 network of world trade in metal. The infection parameters used for the two processes are $\alpha_A = \alpha_B = 0.5$; the budget for B is 1, allocated at node 1 (Argentina) at $t = 0$. A budget of 1 per time step is deployed using the DMP-optimal strategy for process A for a time window of $T = 3$. The subfigures represent the containment of process B at the different times: (a) $t = 1$; (b) $t = 2$; and (c) $t = 3$. The heat bar represents the dominating process per node through the value $P_A^i(t) - P_B^i(t)$. Red and blue represent dominating processes A and B, respectively.

represent the containment of process B at different times. The heat bar represents the dominating process per node through the value $P_A^i(t) - P_B^i(t)$. Red and blue represent dominating processes A and B, respectively.

APPENDIX H: OPTIMIZATION OF COLLABORATIVE PROCESSES

Optimization in collaborative processes is highly relevant in many health-related problems such as the coepidemic of HIV and tuberculosis, and in the marketing of correlated products or opinions. In the coepidemic case, identifying the most catastrophic health-risk scenario is an important aspect of evaluating the risk of a coepidemic event; however, in the marketing area, the aim is the best joint advertising campaign.

We consider two main scenarios in our model: (a) the multiagent seeding problem, i.e., computing the optimal allocation of one agent, which minimizes the number of susceptible vertices at a certain time T for a given spread of the second agent; (b) optimal allocation of vaccines against one of the processes for maximizing the impact on the spread of both processes. Assume we do not have any vaccines for HIV, but vaccines for tuberculosis are available (such as Bacillus Calmette-Gurin [96]). The optimization problem would be articulated as follows: Given a certain budget of vaccines for tuberculosis, what is the best vaccination policy to minimize the spreading of both?

The Lagrangian used is similar to the one of the competitive case but with different dynamics and initial conditions. The dynamics equations are enforced through a set of Lagrange multipliers λ as before:

$$\mathcal{D} = \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{\theta_A}(t+1)[\theta_A^{i\to j}(t+1) - \theta_A^{i\to j}(t) + \alpha_A\phi_A^{i\to j}(t)]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{\theta_B}(t+1)[\theta_B^{i\to j}(t+1) - \theta_B^{i\to j}(t) + \alpha_B\phi_B^{i\to j}(t)]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{S}(t+1)\left[P_S^{i\to j}(t+1) - P_S^i(0)\prod_{k\in\partial i\backslash j}\theta_A^{k\to i}(t+1)\theta_B^{k\to i}(t+1)\right]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{\phi_A}(t+1)[\phi_A^{i\to j}(t+1) - \phi_A^{i\to j}(t) + \alpha_A\phi_A^{i\to j}(t) - P_A^{i\to j}(t+1) + P_A^{i\to j}(t)]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{\phi_B}(t+1)[\phi_B^{i\to j}(t+1) - \phi_B^{i\to j}(t) + \alpha_B\phi_B^{i\to j}(t) - P_B^{i\to j}(t+1) + P_B^{i\to j}(t)]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{A}(t+1)\left[P_A^{i\to j}(t+1) - P_A^{i\to j}(t) - P_S^{i\to j}(t)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_A\phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\right]\right.$$

$$\left. - P_{B_*}^{i\to j}(t)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_{AB}\phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\right]\right]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{B}(t+1)\left[P_B^{i\to j}(t+1) - P_B^{i\to j}(t) - P_S^{i\to j}(t)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_B\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right]\right.$$

$$\left. - P_{A_*}^{i\to j}(t)\left[1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{\alpha_{BA}\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right]\right]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{AB}(t+1)[P_{AB}^{i\to j}(t+1) - P_A^{i\to j}(t+1) - P_B^{i\to j}(t+1) - P_S^{i\to j}(t+1) + 1]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{A_*}(t+1)[P_{A_*}^{i\to j}(t+1) - P_A^{i\to j}(t+1) + P_{AB}^{i\to j}(t+1)]$$

$$+ \sum_{ij}\sum_{t=0}^{T-1}\lambda_{ij}^{B_*}(t+1)[P_{B_*}^{i\to j}(t+1) - P_B^{i\to j}(t+1) + P_{AB}^{i\to j}(t+1)]$$

$$+ \sum_{i}\sum_{t=0}^{T-1}\lambda_{i}^{S}(t+1)\left[P_S^i(t+1) - P_S^i(0)\prod_{k\in\partial i}\theta_A^{k\to i}(t+1)\theta_B^{k\to i}(t+1)\right]$$

$$+ \sum_{i}\sum_{t=0}^{T-1}\lambda_{i}^{AB}(t+1)[P_{AB}^i(t+1) - P_A^i(t+1) - P_B^i(t+1) - P_S^i(t+1) + 1]$$

$$+ \sum_{i}\sum_{t=0}^{T-1}\lambda_{i}^{A}(t+1)\left[P_A^i(t+1) - P_A^i(t) - P_S^i(t)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_A\phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\right) - P_{B_*}^i(t)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_{AB}\phi_A^{k\to i}(t)}{\theta_A^{k\to i}(t)}\right)\right)\right]$$

$$+ \sum_{i}\sum_{t=0}^{T-1}\lambda_{i}^{B}(t+1)\left[P_B^i(t+1) - P_B^i(t) - P_S^i(t)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_B\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right) - P_{A_*}^i(t)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{\alpha_{BA}\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right)\right]$$

$$+ \sum_{i}\sum_{t=0}^{T-1}\lambda_{i}^{A_*}(t+1)[P_{A_*}^i(t+1) - P_A^i(t+1) + P_{AB}^i(t+1)] + \sum_{i}\sum_{t=0}^{T-1}\lambda_{i}^{B_*}(t+1)[P_{B_*}^i(t+1) - P_B^i(t+1) + P_{AB}^i(t+1)].$$

$$(H1)$$

The initial conditions in this case are of the form

$$
\begin{aligned}
\mathcal{I} =& \sum_{ij} \lambda_{ij}^{\theta_A}(0)(\theta_A^{i\to j}(0)-1) + \sum_{ij} \lambda_{ij}^{\theta_B}(0)(\theta_B^{i\to j}(0)-1) \\
&+ \sum_{ij} \lambda_{ij}^{\phi_A}(0)(\phi_A^{i\to j}(0)-\nu^i(0)(1-\delta_{\sigma_i(0),B})) + \sum_{ij} \lambda_{ij}^{\phi_B}(0)(\phi_B^{i\to j}(0)-\delta_{\sigma_i(0),B}) \\
&+ \sum_{ij} \lambda_{ij}^{S}(0)(P_S^{i\to j}(0)-1+\nu^i(0)(1-\delta_{\sigma_i(0),B})+\delta_{\sigma_i(0),B}) \\
&+ \sum_{ij} \lambda_{ij}^{A}(0)(P_A^{i\to j}(0)-\nu^i(0)(1-\delta_{\sigma_i(0),B})) + \sum_{ij} \lambda_{ij}^{B}(0)(P_B^{i\to j}(0)-\delta_{\sigma_i(0),B}) + \sum_{ij} \lambda_{ij}^{AB}(0)P_{AB}^{i\to j}(0) \\
&+ \sum_{ij} \lambda_{ij}^{A_*}(0)(P_{A_*}^{i\to j}(0)-\nu^i(0)(1-\delta_{\sigma_i(0),B})) + \sum_{ij} \lambda_{ij}^{B_*}(0)(P_{B_*}^{i\to j}(0)-\delta_{\sigma_i(0),B}) \\
&+ \sum_{i} \lambda_i^{S}(0)(P_S^i(0)-1+\nu^i(0)(1-\delta_{\sigma_i(0),B})+\delta_{\sigma_i(0),B}) \\
&+ \sum_{i} \lambda_i^{A_*}(0)(P_{A_*}^i(0)-\nu^i(0)(1-\delta_{\sigma_i(0),B})) + \sum_{i} \lambda_i^{B_*}(0)(P_{B_*}^i(0)-\delta_{\sigma_i(0),B}) \\
&+ \sum_{i} \lambda_i^{A}(0)(P_A^i(0)-\nu^i(0)(1-\delta_{\sigma_i(0),B})) + \sum_{i} \lambda_i^{B}(0)(P_B^i(0)-\delta_{\sigma_i(0),B}) + \sum_{i} \lambda_i^{AB}(0)P_{AB}^i(0).
\end{aligned}
\tag{H2}
$$

APPENDIX I: OPTIMIZATION OF VACCINE ALLOCATION

The multiprocess seeding problem in the collaborative case has a similar structure to that of the containment task in competitive processes. The vaccine allocation problem is of a slightly different nature. There are many ways in which it can be formulated; the model illustrated below is the one we have chosen to use.

If a node receives a unit of vaccine before being exposed to infected neighbors, it will be immune to that process (denoted as B in our model), which means the infection message from its neighbor will decrease to 0. However, sometimes a node receives part of a unit of vaccine; specifically, if one allocates a certain amount of vaccine, say, b to a node, the probability of it being infected will decrease to $\alpha_B - b$ (bounded by 0 from below), where α_B is the initial infection parameter. Meanwhile, the parameter α_{BA} will also decrease to $\alpha_{BA} - b$. Therefore, the budget for vaccines can be expressed by the budget for decreasing α_B and α_{BA} in Eqs. (E31)–(E37), which now take the following form:

$$
\begin{aligned}
\theta_B^{i\to j}(t) - \theta_B^{i\to j}(t-1) &= -(\alpha_B - b(j))\phi_B^{i\to j}(t-1) \\
\phi_B^{i\to j}(t) - \phi_B^{i\to j}(t-1) &= -(\alpha_B - b(j))\phi_B^{i\to j}(t-1) + P_B^{i\to j}(t) - P_B^{i\to j}(t-1) \\
P_B^i(t) &= P_B^i(t-1) + P_S^i(t-1)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{(\alpha_B - b(i))\phi_B^{k\to i}(t-1)}{\theta_B^{k\to i}(t-1)}\right)\right) \\
&\quad + P_{A_*}^i(t-1)\left(1 - \prod_{k\in\partial i}\left(1 - \frac{(\alpha_{BA} - b(i))\phi_B^{k\to i}(t-1)}{\theta_B^{k\to i}(t-1)}\right)\right), \\
P_B^{i\to j}(t) &= P_B^{i\to j}(t-1) + P_S^{i\to j}(t-1)\left(1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{(\alpha_B - b(i))\phi_B^{k\to i}(t-1)}{\theta_B^{k\to i}(t-1)}\right)\right) \\
&\quad + P_{A_*}^{i\to j}(t-1)\left(1 - \prod_{k\in\partial i\backslash j}\left(1 - \frac{(\alpha_{BA} - b(i))\phi_B^{k\to i}(t-1)}{\theta_B^{k\to i}(t-1)}\right)\right).
\end{aligned}
\tag{I1}
$$

The budget restriction in this case is

$$
\mathcal{B} = \lambda^{\mathrm{bu}}\sum_i (b(i) - Bu),
\tag{I2}
$$

where Bu is the total vaccination budget. Another restriction for $b(i)$ allocated on a node (not exceeding the infection probability) is expressed similarly to before as

$$\mathcal{P} = \epsilon[\log\left(b(i)\right) + \log\left(\alpha_B - b(i)\right)]. \tag{I3}$$

Finally, the derivative of the part of the Lagrangian that enforces the dynamics $\mathcal{D}$ is differentiated with respect to $b(i)$, which is required to complete the set of equations

$$
\begin{aligned}
\frac{\partial \mathcal{D}}{\partial b(i)} = \sum_{t=0}^{T-1} \Bigg[&-\sum_{k\in\partial i}\lambda_{ki}^{\theta_B}(t+1)\phi_B^{k\to i}(t) - \sum_{k\in\partial i}\lambda_{ki}^{\phi_B}(t+1)\phi_B^{k\to i}(t) \\
&+\sum_j \lambda_{ij}^B(t+1)P_S^{i\to j}(t)\left(\sum_{l\in\partial i\setminus j}\frac{\phi_B^{l\to i}(t)}{\theta_B^{l\to i}(t)}\prod_{k\in\partial i\setminus j,l}\left(1-\frac{(\alpha_B-b(i))\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right) \\
&+\sum_j \lambda_{ij}^B(t+1)P_{A_*}^{i\to j}(t)\left(\sum_{l\in\partial i\setminus j}\frac{\phi_B^{l\to i}(t)}{\theta_B^{l\to i}(t)}\prod_{k\in\partial i\setminus j,l}\left(1-\frac{(\alpha_{BA}-b(i))\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right) \\
&+\lambda_i^B(t+1)P_S^i(t)\left(\sum_{l\in\partial i}\frac{\phi_B^{l\to i}(t)}{\theta_B^{l\to i}(t)}\prod_{k\in\partial i\setminus l}\left(1-\frac{(\alpha_B-b(i))\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right) \\
&+\lambda_i^B(t+1)P_{A_*}^i(t)\left(\sum_{l\in\partial i}\frac{\phi_B^{l\to i}(t)}{\theta_B^{l\to i}(t)}\prod_{k\in\partial i\setminus l}\left(1-\frac{(\alpha_{BA}-b(i))\phi_B^{k\to i}(t)}{\theta_B^{k\to i}(t)}\right)\right) \Bigg].
\end{aligned}
\tag{I4}
$$

The same procedure [following Eq. (F8)] can be implemented for the optimization in this case.

[1] F. Wu, S. Zhao, B. Yu, Y.-M. Chen, W. Wang, Z.-G. Song, Y. Hu, Z.-W. Tao, J.-H. Tian, Y.-Y. Pei *et al.*, *A New Coronavirus Associated with Human Respiratory Disease in China*, Nature (London) **579**, 265 (2020).

[2] P. Zhou, X.-L. Yang, X.-G. Wang, B. Hu, L. Zhang, W. Zhang, H.-R. Si, Y. Zhu, B. Li, C.-L. Huang *et al.*, *A Pneumonia Outbreak Associated with a New Coronavirus of Probable Bat Origin*, Nature (London) **579**, 270 (2020).

[3] X. Zhou, N. Li, Y. Luo, Y. Liu, F. Miao, T. Chen, S. Zhang, P. Cao, X. Li, K. Tian, H.-J. Qiu, and R. Hu, *Emergence of African Swine Fever in China, 2018*, Transbound. Emerg. Dis. **65**, 1482 (2018).

[4] S. Mohurle and M. Patil, *A Brief Study of Wannacry Threat: Ransomware Attack 2017*, Int. J. Adv. Res. Comput. Sci. **8**, 1938 (2017).

[5] V. Saraceni, B. S. King, S. C. Cavalcante, J. E. Golub, L. M. Lauria, L. H. Moulton, R. E. Chaison, and B. Durovni, *Tuberculosis as Primary Cause of Death Among AIDS Cases in Rio de Janeiro, Brazil*, Int. J. Tuberc. Lung Dis. **12**, 769 (2008), https://www.ncbi.nlm.nih.gov/entrez/eutils/elink.fcgi?dbfrom=pubmed&retmode=ref&cmd=prlinks&id=18544202.

[6] J. Bruchfeld, M. Correia-Neves, and G. Källenius, *Tuberculosis and HIV Coinfection*, Cold Spring Harb. Perspect. Med. **5**, a017871 (2015).

[7] F. E. Jegede, T. I. Oyeyi, S. A. Abdulrahman, H. A. Mbah, T. Badru, C. Agbakwuru, and O. Adedokun, *Effect of HIV and Malaria Parasites Co-Infection on Immune-Hematological Profiles Among Patients Attending Anti-Retroviral Treatment (ART) Clinic in Infectious Disease Hospital Kano*, Nigeria, PLOS One **12**, e0174233 (2017).

[8] A. Munawwar and S. Singh, *Human Herpesviruses as Copathogens of HIV Infection, their Role in HIV Transmission, and Disease Progression*, J. Lab. Physicians **8**, 5 (2016).

[9] A. H. Limper, A. Adenis, T. Le, and T. S. Harrison, *Fungal Infections in HIV/AIDS*, Lancet Infect. Dis. **17**, e334 (2017).

[10] B. J. Berger, F. Hussain, and K. Roistacher, *Bacterial Infections in HIV-Infected Patients*, Infect. Dis. Clin. North Am. **8**, 449465 (1994), https://europepmc.org/article/med/1955696.

[11] M. Dupont-Rouzeyrol, O. O'Connor, E. Calvez, M. Daurès, M. John, J.-P. Grangeon, and A.-C. Gourinat, *Co-Infection with Zika and Dengue Viruses in 2 Patients, New Caledonia, 2014*, Emerg. Infect. Dis. **21**, 381 (2015).

[12] P. J. Hotez, *Texas and its Measles Epidemics*, PLoS Med. **13**, e1002153 (2016).

[13] Director-General, *Draft Thirteenth General Programme of Work, 2019–2023*, World Health Organization, 2018.

[14] *The Guardian, Facebook under Pressure to Halt Rise of Anti-Vaccination Groups*, 2019, https://www.theguardian.com/technology/2019/feb/12/facebook-anti-vaxxer-vaccination-groups-pressure-misinformation.

[15] I. Z. Kiss, J. C. Miller, and P. Simon, *Mathematics of Epidemics on Networks: From Exact to Approximate Models* (Springer, New York, 2017).

[16] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, *Epidemic Processes in Complex Networks*, Rev. Mod. Phys. **87**, 925 (2015).

[17] M. Shapiro and E. Delgado-Eckert, *Finding the Probability of Infection in an SIR Network Is NP-Hard*, Math. Biosci. **240**, 77 (2012).

[18] R. M. Anderson, R. M. May, and B. Anderson, *Infectious Diseases of Humans: Dynamics and Control* (Wiley Online Library, New York, 1992), Vol. 28.

[19] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, *Complex Networks: Structure and Dynamics*, Phys. Rep. **424**, 175 (2006).

[20] J. P. Gleeson, *High-Accuracy Approximation of Binary-State Dynamics on Networks*, Phys. Rev. Lett. **107**, 068701 (2011).

[21] J. P. Gleeson, *Binary-State Dynamics on Complex Networks: Pair Approximation and Beyond*, Phys. Rev. X **3**, 021004 (2013).

[22] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, *Epidemic Processes in Complex Networks*, Rev. Mod. Phys. **87**, 925 (2015).

[23] E. M. Rogers, *Diffusion of Innovations* (Simon and Schuster, New York, 2010).

[24] B. Karrer and M. E. J. Newman, *Message Passing Approach for General Epidemic Models*, Phys. Rev. E **82**, 016101 (2010).

[25] A. Y. Lokhov, M. Mézard, H. Ohta, and L. Zdeborová, *Inferring the Origin of an Epidemic with a Dynamic Message-Passing Algorithm*, Phys. Rev. E **90**, 012801 (2014).

[26] M. Shrestha and C. Moore, *Message-Passing Approach for Threshold Models of Behavior in Networks*, Phys. Rev. E **89**, 022805 (2014).

[27] J. C. Miller, *A Note on a Paper by Erik Volz: SIR Dynamics in Random Networks*, J. Math. Biol. **62**, 349 (2011).

[28] J. C. Miller, C. S. Anja, and E. M. Volz, *Edge-Based Compartmental Modelling for Infectious Disease Spread*, J. R. Soc. Interface **9**, 890 (2012).

[29] N. Sherborne, J. C. Miller, K. B. Blyuss, and I. Z. Kiss, *Mean-Field Models for Non-Markovian Epidemics on Networks*, J. Math. Biol. **76**, 755 (2018).

[30] F. Altarelli, A. Braunstein, L. Dall'Asta, and R. Zecchina, *Optimizing Spread Dynamics on Graphs by Message Passing*, J. Stat. Mech. (2013) P09011.

[31] A. Guggiola and G. Semerjian, *Minimal Contagious Sets in Random Regular Graphs*, J. Stat. Phys. **158**, 300 (2015).

[32] A. Y. Lokhov, M. Mézard, and L. Zdeborová, *Dynamic Message-Passing Equations for Models with Unidirectional Dynamics*, Phys. Rev. E **91**, 012811 (2015).

[33] A. Y. Lokhov and D. Saad, *Scalable Influence Estimation without Sampling*, arXiv:1912.12749.

[34] W. Cai, L. Chen, F. Ghanbarnejad, and P. Grassberger, *Avalanche Outbreaks Emerging in Cooperative Contagions*, Nat. Phys. **11**, 936 (2015).

[35] J. C. Miller, *Cocirculation of Infectious Diseases on Networks*, Phys. Rev. E **87**, 060801(R) (2013).

[36] M. E. J. Newman and C. R. Ferrario, *Interacting Epidemics and Coinfection on Contact Networks*, PLoS One **8**, 1 (2013).

[37] J. P. Gleeson and D. J. Cahalane, *Seed Size Strongly Affects Cascades on Random Networks*, Phys. Rev. E **75**, 056103 (2007).

[38] J. C. Miller, *Complex Contagions and Hybrid Phase Transitions*, J. Complex Netw. **4**, 201 (2016).

[39] L. Hébert-Dufresne, S. V. Scarpino, and J.-G. Young, *Macroscopic Patterns of Interacting Contagions are Indistinguishable from Social Reinforcement*, Nat. Phys. **16**, 426 (2020).

[40] F. D. Sahneh and C. Scoglio, *Competitive Epidemic Spreading over Arbitrary Multilayer Networks*, Phys. Rev. E **89**, 062817 (2014).

[41] B. Karrer and M. E. J. Newman, *Competing Epidemics on Complex Networks*, Phys. Rev. E **84**, 036106 (2011).

[42] J. Sanz, C.-Y. Xia, S. Meloni, and Y. Moreno, *Dynamics of Interacting Diseases*, Phys. Rev. X **4**, 041005 (2014).

[43] V. V. Vasconcelos, S. A. Levin, and F. L. Pinheiro, *Consensus and Polarization in Competing Complex Contagion Processes*, J. R. Soc. Interface **16**, 20190196 (2019).

[44] P. C. Ventura, Y. Moreno, and F. A. Rodrigues, *The Role of Time Scale in the Spreading of Asymmetrically Interacting Diseases*, arXiv:2007.02774.

[45] L. Yang, X. Yang, and Y. Y. Tang, *A Bi-Virus Competing Spreading Model with Generic Infection Rates*, IEEE Trans. Netw. Sci. Eng. **5**, 2 (2018).

[46] N. Azimi-Tafreshi, *Cooperative Epidemics on Multiplex Networks*, Phys. Rev. E **93**, 042303 (2016).

[47] H.-C. H. Chang and F. Fu, *Co-Diffusion of Social Contagions*, New J. Phys. **20**, 095001 (2018).

[48] L. Chen, F. Ghanbarnejad, and D. Brockmann, *Fundamental Properties of Cooperative Contagion Processes*, New J. Phys. **19**, 103041 (2017).

[49] L. Chen, F. Ghanbarnejad, W. Cai, and P. Grassberger, *Outbreaks of Coinfections: The Critical Role of Cooperativity*, Europhys. Lett. **104**, 50001 (2013).

[50] P.-B. Cui, F. Colaiori, and C. Castellano, *Mutually Cooperative Epidemics on Power-Law Networks*, Phys. Rev. E **96**, 022301 (2017).

[51] P. Grassberger, L. Chen, F. Ghanbarnejad, and W. Cai, *Phase Transitions in Cooperative Coinfections: Simulation Results for Networks and Lattices*, Phys. Rev. E **93**, 042316 (2016).

[52] L. Hébert-Dufresne and B. M. Althouse, *Complex Dynamics of Synergistic Coinfections on Realistically Clustered Networks*, Proc. Natl. Acad. Sci. U.S.A. **112**, 10551 (2015).

[53] H.-K. Janssen and O. Stenull, *First-Order Phase Transitions in Outbreaks of Co-Infectious Diseases and the Extended General Epidemic Process*, Europhys. Lett. **113**, 26005 (2016).

[54] Q.-H. Liu, L.-F. Zhong, W. Wang, T. Zhou, and H. E. Stanley, *Interactive Social Contagions and Co-Infections on Complex Networks*, Chaos **28**, 013120 (2018).

[55] M. Martcheva and S. S. Pilyugin, *The Role of Coinfection in Multidisease Dynamics*, SIAM J. Appl. Math. **66**, 843 (2006).

[56] J. P. Rodrguez, Yu.-H. Liang, Yu.-J. Huang, and J. Juang, *Diversity of Hysteresis in a Fully Cooperative Coinfection Model*, Chaos **28**, 023107 (2018).

[57] S. Sajjadi, M. R. Ejtehadi, and F. Ghanbarnejad, *Impact of Temporal Correlations on High Risk Outbreaks of Independent and Cooperative SIR Dynamics*, arXiv:2003.01268.

[58] B. Min and C. Castellano, *Message-Passing Theory for Cooperative Epidemics*, Chaos **30**, 023131 (2020).

[59] R. Pastor-Satorras and A. Vespignani, *Immunization of Complex Networks*, Phys. Rev. E **65**, 036104 (2002).

[60] R. Cohen, S. Havlin, and D. ben-Avraham, *Efficient Immunization Strategies for Computer Networks and Populations*, Phys. Rev. Lett. **91**, 247901 (2003).

[61] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, *Attack Vulnerability of Complex Networks*, Phys. Rev. E **65**, 056109 (2002).

[62] P. Holme, *Efficient Local Strategies for Vaccination and Network Attack*, Europhys. Lett. **68**, 908 (2004).

[63] Y. Chen, G. Paul, S. Havlin, F. Liljeros, and H. E. Stanley, *Finding a Better Immunization Strategy*, Phys. Rev. Lett. **101**, 058701 (2008).

[64] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, *Identification of Influential Spreaders in Complex Networks*, Nat. Phys. **6**, 888 (2010).

[65] J. Borge-Holthoefer and Y. Moreno, *Absence of Influential Spreaders in Rumor Dynamics*, Phys. Rev. E **85**, 026116 (2012).

[66] L. Hébert-Dufresne, A. Allard, J.-G. Young, and L. J. Dubé, *Global Efficiency of Local Immunization on Complex Networks*, Sci. Rep. **3**, 2171 (2013).

[67] A. Braunstein, L. Dall'Asta, G. Semerjian, and L. Zdeborová, *Network Dismantling*, Proc. Natl. Acad. Sci. U.S.A. **113**, 12368, 2016.

[68] F. Morone and H. A. Makse, *Influence Maximization in Complex Networks through Optimal Percolation*, Nature (London) **524**, 65 (2015).

[69] S. Mugisha and H.-J. Zhou, *Identifying Optimal Targets of Network Attack by Belief Propagation*, Phys. Rev. E **94**, 012305 (2016).

[70] F. Altarelli, A. Braunstein, L. Dall'Asta, J. R. Wakeling, and R. Zecchina, *Containing Epidemic Outbreaks by Message-Passing Techniques*, Phys. Rev. X **4**, 021024 (2014).

[71] W. Chen, L. V. S. Lakshmanan, and C. Castillo, *Information and Influence Propagation in Social Networks*, Synth. Lect. Data Manage. **5**, 1 (2013).

[72] P. Domingos and M. Richardson, *Mining the Network Value of Customers*, in *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York, NY, 2001), pp. 57–66.

[73] D. Kempe, J. Kleinberg, and É. Tardos, *Maximizing the Spread of Influence through a Social Network*, in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York, NY, 2003), pp. 137–146.

[74] N. Du, L. Song, M. Gomez-Rodriguez, and H. Zha, *Scalable Influence Estimation in Continuous-Time Diffusion Networks*, Adv. Neural Inf. Process. Syst. **26**, 3147 (2013), https://pubmed.ncbi.nlm.nih.gov/26752940/.

[75] C. Nowzari, V. M. Preciado, and G. J. Pappas, *Analysis and Control of Epidemics: A Survey of Spreading Processes on Complex Networks*, IEEE Control Syst. Mag. **36**, 26 (2016).

[76] A. Y. Lokhov and D. Saad, *Optimal Deployment of Resources for Maximizing Impact in Spreading Processes*, Proc. Natl. Acad. Sci. U.S.A. **114**, E8138 (2017).

[77] F. Altarelli, A. Braunstein, L. Dall'Asta, and R. Zecchina, *Large Deviations of Cascade Processes on Graphs*, Phys. Rev. E **87**, 062115 (2013).

[78] Y. Kanoria and A. Montanari, *Majority Dynamics on Trees and the Dynamic Cavity Method*, Ann. Appl. Probab. **21**, 1694 (2011).

[79] M. Mézard and A. Montanari, *Information, Physics, and Computation* (Oxford University Press, New York, 2009).

[80] M. J. Wainwright and M. Irwin Jordan, *Graphical Models, Exponential Families, and Variational Inference* (Now Publishers, Hanover, MA, 2008).

[81] V. Krebs, *Social Network Analysis Software & Services for Organizations, Communities, and Their Consultants*, 2020, http://www.orgnet.com.

[82] M. Girvan and M. E. J. Newman, *Community Structure in Social and Biological Networks*, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002).

[83] R. Pastor-Satorras and A. Vespignani, *Immunization of Complex Networks*, Phys. Rev. E **65**, 036104 (2002).

[84] D. E. Knuth, *The Stanford GraphBase: A Platform for Combinatorial Computing* (ACM Press, New York, 1993).

[85] W. W. Zachary, *An Information Flow Model for Conflict and Fission in Small Groups*, J. Anthropol Res. **33**, 452 (1977), https://www.jstor.org/stable/3629752.

[86] D. J. Watts and S. H. Strogatz, *Collective Dynamics of "Small-World" Networks*, Nature (London) **393**, 440 (1998).

[87] W. H. Fleming and R. W. Rishel, *Deterministic and Stochastic Optimal Control* (Springer-Verlag, Berlin, 1975).

[88] S. Meyn and S. P. Meyn, *Control Techniques for Complex Networks* (Cambridge University Press, Cambridge, England, 2008).

[89] *Highways England, Highways Agency Network Journey Time, and Traffic Flow Data*, 2019, https://data.gov.uk/dataset/dc18f7d5-2669-490f-b2b5-77f27ec133ad/highways-agency-network-journey-time-and-traffic-flow-data.

[90] M. Wilinski and A. Y. Lokhov, *Scalable Learning of Independent Cascade Dynamics from Partial Observations*, arXiv:2007.06557.

[91] A. Lokhov, *Reconstructing Parameters of Spreading Models from Partial Observations, Advances in Neural Information Processing Systems 3467* (2016).

[92] A. Y. Lokhov and T. Misiakiewicz, *Efficient Reconstruction of Transmission Probabilities in a Spreading Process from Partial Observations*, arXiv:1509.06893.

[93] A. Y. Lokhov, M. Mézard, H. Ohta, and L. Zdeborová, *Inferring the Origin of an Epidemic with a Dynamic Message-Passing Algorithm*, Phys. Rev. E **90**, 012801 (2014).

[94] H. Zhang, April 2018, https://studentwork.prattsi.org/infovis/labs/9209/.

[95] W. De Nooy, A. Mrvar, and V. Batagelj, *Exploratory Social Network Analysis with Pajek* (Cambridge University Press, New York, NY, 2011).

[96] *BCG Vaccines: WHO Position Paper—February 2018*, Relevé Épidémiologique Hebdomadaire **93**, 73 (2018).