Optimal deployment of resources for maximizing impact in spreading processes

Andrey Y. Lokhov[a,1] **and David Saad**[b]

[a]Center for Nonlinear Studies and Theoretical Division T-4, Los Alamos National Laboratory, Los Alamos, NM 87545; and [b]The Nonlinearity and Complexity Research Group, Aston University, Birmingham B4 7ET, United Kingdom

Edited by Giorgio Parisi, University of Rome, Rome, Italy, and approved July 24, 2017 (received for review September 1, 2016)

The effective use of limited resources for controlling spreading processes on networks is of prime significance in diverse contexts, ranging from the identification of "influential spreaders" for maximizing information dissemination and targeted interventions in regulatory networks, to the development of mitigation policies for infectious diseases and financial contagion in economic systems. Solutions for these optimization tasks that are based purely on topological arguments are not fully satisfactory; in realistic settings, the problem is often characterized by heterogeneous interactions and requires interventions in a dynamic fashion over a finite time window via a restricted set of controllable nodes. The optimal distribution of available resources hence results from an interplay between network topology and spreading dynamics. We show how these problems can be addressed as particular instances of a universal analytical framework based on a scalable dynamic message-passing approach and demonstrate the efficacy of the method on a variety of real-world examples.

optimal control of spreading processes | dynamic resource allocation | message-passing algorithms | influence maximization | mitigation of epidemic outbreak

Spreading corresponds to omnipresent processes describing a vast number of phenomena in social, natural, and technological networks (1–4) whereby information, viruses, and failures propagate through their edges via the interactions between individual constituents. Spreading cascades have a huge impact on the modern world, be it negative or positive. An 11-min power grid disturbance in Arizona and California in 2011 led to cascading outages and left 2.7 million customers without power (5). As many as 579,000 people around the world could have been killed by the H1N1 influenza pandemic, characterized by a rapid spreading through the global transportation networks (6). The US economy losses from the 2008 financial crisis resulting from cascading bankruptcies of major financial institutions are estimated at $22 trillion (7). Therefore, it is not surprising that efficient prediction and control of these undesired spreading processes are regarded as fundamental questions of paramount importance in developing policies for optimal placement of cascade-preventing devices in power grid, real-time distribution of vaccines and antidotes to mitigate epidemic spread, regulatory measures in interbanking lending networks, and other modern world problems, such as protection of critical infrastructures against cyberattacks and computer viruses (8).

On the other hand, spreading processes can also be considered beneficial. The Ice Bucket Challenge campaign in social networks raised $115 million donations to the ALS Association fighting amyotrophic lateral sclerosis, in particular due to a significant involvement of celebrities acting as "influencers" (9). In the context of political campaigning, there are already winners (10, 11) and losers, and this division is likely to become more pronounced and critical in the future (12). Winners are those who use communication and social networks effectively to set the opinions of voters or consumers, maximizing the impact of scarce resources such as activists or advertisements by applying control to the most influential groups of nodes at the right time, while losers will spend their resources suboptimally, relying on intuition and serendipity. Additional examples of domains where optimal resource allocation plays a crucial role in enhancing the effect of spreading include viral marketing campaigns (13); targeted chemically induced control of dynamic biological processes (14); drug discovery (15); and even gaining military advantage through the propagation of disinformation (16). All of these applications share several important common properties such as restricted budget, finite time windows for dynamical control interventions, and the need for fast and scalable optimization algorithms which can be deployed in real time.

There exists a large body of work on optimal resource deployment in various spreading settings. A widely addressed formulation focuses on identifying influential spreaders (i.e., nodes that play an important role in the dynamical process). Identification is often done by using different centrality measures related to the topology of the underlying interaction network, including selection strategies based on high-degree nodes (17), neighbors of randomly selected vertices (18), betweenness centrality (19), random-walk (20), graph-partitioning (21), and k-shell decomposition (22), to name a few. It is quite natural that algorithms based exclusively on topological characteristics appear to have variable performance depending on particular network instances and dynamical models used (23, 24). Another line of work consists in studying the nondeterministic polynomial time (NP)-complete problem of network dismantling (25–27): The underlying reasoning is that removal of key nodes fragments the giant component and hence is likely to prevent a global percolation of the contagion. The localization of an optimal immunization set has been addressed by using a belief

Significance

Spreading processes play an increasingly important role in marketing, opinion setting, and epidemic modeling. Most existing algorithms for optimal resource allocation in spreading processes are based on topological characteristics of the underlying network and aim to maximize impact at infinite time. Clearly, realistic and efficient real-time allocation policies should consider both network properties and details of the dynamics; additionally, control may be applied only to a restricted set of accessible nodes, and impact should be maximized in a limited time window. We introduce a probabilistic targeting framework that incorporates the dynamics and encompasses previously considered optimization formulations. It is based on a scalable dynamic message-passing approach that allows for the solution of large real-world network instances.

Author contributions: A.Y.L. and D.S. designed research, performed research, contributed new reagents/analytic tools, analyzed data, and wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence should be addressed. Email: lokhov@lanl.gov.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1614694114/-/DCSupplemental.

propagation algorithm built on top of percolation-like equations for SIR (susceptible, infected, recovered) and SIS (susceptible, infected, susceptible) models (28), based on cavity method techniques developed previously for deterministic threshold models (29, 30). This formulation is close to the problem of finding optimal seeds [i.e., the smallest set of initial nodes which maximizes the spread asymptotically (13)]. It was rigorously analyzed (31, 32) for two simple diffusion models with a special submodularity property, independent cascade (IC) and linear threshold, and was shown to be NP-hard for both. A greedy algorithm based on a sampling subroutine has been explored for the IC model (33) in the setting of a finite time horizon. For other spreading models, the impact maximization problem at finite time and resources has been addressed in the setting of optimal control as reported in a recent survey (34). However, only deterministic mean-field dynamics have been considered so far; this approximation ignores the topology of the specific network considered and yields nondistributed solutions to the control problem.

All of these techniques consider the problem of static (open-loop) resource allocation, preplanned at some initial time. A less explored direction consists of developing an online policy of assigning a limited remedial budget dynamically based on real-time feedback, also known as a closed-loop control. The impact of vaccination of the largest degree nodes or of those with the largest number of infected neighbors was investigated in refs. 35 and 36, while an alternative strategy is focused on the largest reduction in infectious edges (37). Finally, an online policy based on solution of the minimal maxcut problem was introduced (38), where optimization is carried out with respect to the expected time to extinction of the SIS epidemic.

We introduce a general optimization formulation which accommodates both dynamical and topological aspects of the problem and which allows for a broad range of objectives. The framework facilitates the optimization of objective functions beyond the maximization or minimization of the spread, including: targeting specific nodes at specific times given a subset of accessible nodes; a limited global budget, possibly distributed over time; and an optimal dynamic vaccination strategy using feedback from the spreading process. The problem is stated in a dynamical control setting with finite-time horizon that requires an explicit solution of the dynamics, which is addressed via a distributed message-passing algorithm. We test the efficacy of the method on particular synthetic optimization problems as well as on a set of real-world instances.

Model

A large number of spreading models have been suggested in the literature to describe stochastic dynamical processes in epidemiology, information and rumor propagation, and cascades in biological and infrastructure networks (2–4). They all share the same common features: The nodes transition from inactive to active state due to spontaneous activation mechanism associated with the nodes themselves or due to interactions with active neighbors through the network edges. As an illustration of our approach, we have chosen a popular stochastic spreading process known as the SIR model, which is often used to describe propagation of infectious diseases or information spreading (2). More precisely, we consider a modified version of the discrete-time SIR model defined as follows. A node i in the interaction graph $G = (V, E)$, where V denotes the set of nodes and E is the set of pairwise edges, can be found at time step t in either of three states σ_i^t: "susceptible" $\sigma_i^t = S$, "infected" $\sigma_i^t = I$ or "recovered" $\sigma_i^t = R$. At each time step, an infected (or, depending on the application domain, informed or active) individual i can transmit the activation signal to one of its susceptible (respectively, uninformed or inactive) neighbors j with probability α_{ij}, associated with the edge connecting them. Independently of the interaction between nodes, a node i in state S at time t can turn active, assuming state I with probability $\nu_i(t)$, or spontaneously become recovered (uninterested, protected) with probability $\mu_i(t)$. The first mechanism corresponds to a node activation due to an external influence such as advertisement in the context of information spreading. In the case of the epidemic spreading, the second mechanism models the effect of vaccination: Once a node goes to the protected R state, it becomes immune to the infection at all times. These probabilistic transmission rules at each time step t can be summarized using the following schematic rules (depicted in Fig. S1):

$$S(i) + I(j) \xrightarrow{\alpha_{ji}} I(i) + I(j), \qquad [1]$$

$$S(i) \xrightarrow{\nu_i(t)} I(i), \qquad S(i) \xrightarrow{\mu_i(t)} R(i). \qquad [2]$$

In the definition of the dynamic rules [1] and [2], $\nu_i(t)$ and $\mu_i(t)$ represent control parameters we could manipulate with a certain degree of freedom defined by a particular instance of the problem. Notice that these control parameters act in opposite directions, expediting or hindering the propagation process. In all examples considered, we typically study either the susceptible, infected (SI) model with the ν-mechanism as a paradigm for the propagation of information, or the modified SIR dynamics with the vaccination μ-mechanism as a model of the epidemic spreading. In what follows, we assume that the spreading couplings α_{ij} are known (or can be estimated) and are fixed in time. In some applications, α_{ij} may vary in time (e.g., this is true for temporal networks) or may represent a set of control parameters themselves. We outline such scenarios in *Discussion*; the optimization scheme presented below can be straightforwardly generalized to include time-varying and edge-related control parameters. However, for simplicity, we will only present optimization involving node-related control parameters.

To quantify the success of the spreading process, one may look for instance at the expected spread (the total number of infected nodes) at final time horizon T, $\mathcal{S}(T)$, given by

$$\mathcal{S}(T) = \mathbb{E}\left[\sum_{i \in V} \mathbb{1}[\sigma_i^T = I]\right] = \sum_{i \in V} P_I^i(T), \qquad [3]$$

where the expectation is taken with respect to the realization of the stochastic dynamics and $P_I^i(T)$ denotes the marginal probability of node i to be found in state I at time T. The quantities $P_S^i(T)$ and $P_R^i(T)$ can be defined in a similar way for the susceptible and recovered states, respectively. Hence, it is important to understand how to compute approximately the marginal probabilities $P_\sigma^i(t)$ on a given network, with σ representing the corresponding state; note that in the general case, an exact evaluation of marginals in the SIR model is an NP-hard problem (39). We use the recently introduced dynamic message-passing (DMP) equations (40–42), which provide estimates of the probabilities $P_\sigma^i(t)$ with a linear computational complexity in the number of edges and time steps. These equations are derived under the assumption of a locally tree-like network and provide asymptotically exact estimates on sparse random graphs. When applied to real-world loopy networks, the DMP algorithm typically yields an accurate prediction of marginal probabilities as validated empirically (42) for a large class of spreading models on real-world networks. In *Methods*, we provide an intuitive derivation of the corresponding DMP equations for the considered modified SIR model. An example of the DMP performance on real-world networks is provided in Fig. 1, where the method predictions are compared with values obtained through extensive Monte Carlo simulations of the SIR dynamics on a network of flights between major US hubs (a detailed description of this dataset is provided below and in *SI Text*). The accuracy of marginals estimation supports the use of the DMP equations at the core of our optimization algorithm.

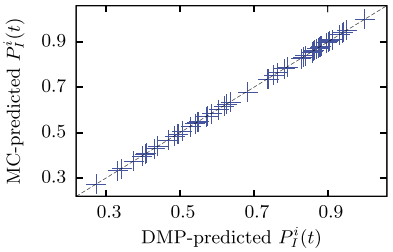

Fig. 1. Performance of DMP equations for the modified SIR model on a network of flights between major US airports. The network represents $M = 383$ flight routes between the $N = 61$ largest US hubs. The weights α_{ij} are proportional to the average number of transported passengers on each route and are distributed in the interval [0.05, 0.5]; ν_i and μ_i are generated at random in the range [0, 0.1]. The scatter plot represents marginal probabilities $P_I^i(T)$ obtained from the DMP equations and by averaging over 10^7 Monte Carlo (MC) simulations. There is one randomly selected active node at the initial time, and the dynamics is simulated for $t = 5$ time steps.

Optimization Framework

We formulate the dynamic allocation of resource as a general optimization problem with respect to an objective function $\mathcal{O}$ and a set of constraints associated with the budget of available resources $\mathcal{B}$, accessible values of control parameters $\mathcal{P}$, initial conditions $\mathcal{I}$, and the dynamical model equations $\mathcal{D}$. We use the Lagrangian formulation of the constrained optimization problem:

$$\mathcal{L} = \underbrace{\mathcal{O}}_{\text{objective}} + \underbrace{\mathcal{B} + \mathcal{P} + \mathcal{I} + \mathcal{D}}_{\text{constraints}}. \quad [4]$$

Let us discuss the form of each of the terms in the expression [**4**].

$\mathcal{O}$: Many objective functions of interest relate to the delivered information *at particular times* defined for each node. So, for the general case we define:

$$\mathcal{O} = \mathbb{E}\left[\sum_{i \in U} \mathbb{1}[\sigma_i^{t_i} = I]\right] = \sum_{i \in U} P_I^i(t_i), \quad [5]$$

where t_i is the required activation time for node i and the sum is over the subset of nodes $U \subset V$ that is required to be activated. We refer to this general formulation as the targeting problem. The popular problem of maximizing the total spread $\mathcal{S}(T)$ is a special case whereby $U = V$ and $t_i = T$ for all $i \in V$.

$\mathcal{B}$: In many relevant situations, resources are not fully available at a given time, but rather become available on the fly, and their amount may vary across the time steps. For example, it takes some time to develop and produce the vaccines, or the advertisement budget can be allocated in stages depending on the success of the campaign. Hence, we define the budget constraints in the following form:

$$\sum_{i \in V} \nu_i(t) = B_\nu(t), \quad \sum_{i \in V} \mu_i(t) = B_\mu(t), \quad [6]$$

where $B_\nu(t)$ and $B_\mu(t)$ denote the available total budget for the control parameters $\nu_i(t)$ (spontaneous infection) and $\mu_i(t)$ (recovery through vaccination) at time t. The constraint $\mathcal{B}$ reads

$$\mathcal{B} = \sum_{t=0}^{T-1} \lambda_B^\nu(t) \left[\sum_{i \in V} \nu_i(t) - B_\nu(t)\right], \quad [7]$$

with a similar expression for the parameters $\mu_i(t)$, where $\lambda_B^\nu(t)$ and $\lambda_B^\mu(t)$ are the associated Lagrange multipliers, respectively. Allocation of budget only at the initial time corresponds to the optimal seeding problem.

$\mathcal{P}$: In an unrestricted scenario, where all nodes are accessible, control parameters associated with node i, $\nu_i(t)$ and $\mu_i(t)$ may take arbitrary values from zero to one depending on total budget. However, in realistic situations access level to different nodes may differ: For example, only a subset $W \subseteq V$ of nodes may be controllable together with additional restrictions on parameter values. The parameter block $\mathcal{P}$ is introduced to enforce parameters $\nu_i(t)$ to take values in the range $[\underline{\nu_i^t}, \overline{\nu_i^t}]$ at each time step. This can be accomplished with the help of barrier functions, widely used in constrained optimization, assuming the form

$$\mathcal{P} = \epsilon \sum_{t=0}^{T-1} \sum_{i \in V} \left(\log\left[\nu_i(t) - \underline{\nu_i^t}\right] + \log\left[\overline{\nu_i^t} - \nu_i(t)\right]\right), \quad [8]$$

where ϵ is a small regularization parameter chosen to minimize the impact on the objective $\mathcal{O}$ in the regime of allowed $\nu_i(t)$ values, away from the borders. An equivalent expression can be written for the constraints on the $\mu_i(t)$ values.

$\mathcal{I}$ and $\mathcal{D}$: Finally, the constraints $\mathcal{I}$ and $\mathcal{D}$ enforce the given initial conditions and dynamics of the system via the associated Lagrange multipliers. For example, if no active individuals are present at the initial time, then we set $P_I^i(0) = 0$ for all nodes using the constraint set $\mathcal{I}$; if some infected or recovered nodes are present, they assume an initial value 1 for the respective marginal probabilities. The set $\mathcal{D}$ encodes the evolution of the marginal probabilities with the DMP equations, as explained in *Methods*.

The extremization of the Lagrangian [**4**] is done 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. We solve the coupled systems of equations by forward–backward propagation, a widely used method in control, as well as for learning and optimization in artificial neural networks (43), detailed in *Methods* and schematically illustrated in Fig. 2. This method has a number of advantages compared with other localized optimization procedures such as gradient descent and its variants. In particular, it is simple to implement, is of modest computational complexity due to the gradient-free nature of the optimization, does not require any adjustable parameters, and is less prone to being trapped in local minima since the optimization is performed globally (44).

In what follows, we illustrate this general optimization framework on three practical case studies: driving dynamic trajectory of a spreading process in time through a set of targets (targeting problem), selecting an optimal set of initial seeds for a maximum dissemination of influence or information (influence maximization problem), and online closed-loop distribution of vaccines for stabilizing the spread of an infection (online mitigation of epidemic spreading). In each of these examples, we show how our algorithm can be used to solve particular problems from social or biological sciences, compare the performance to existing techniques (where competing algorithms exist), and indicate a number of other prospective applications.

Case Study: Guiding Spreading Through Desired Targets

Targeting is quite a general task and can provide a useful problem formulation in many application domains where the underlying dynamics is governed by a set of nonlinear differential or difference equations. The nature of these applications can be very diverse: They range from targeting biochemical cascades to treat cancer (45, 46) and controlling the trajectories of brain dynamics among states characterized by the activation of various cognitive systems (47) to maximizing the species abundance by targeted interventions in food webs (48) and ecological mutualistic networks (49). In the context of spreading processes, targeting tasks appear in several problems of social importance: online

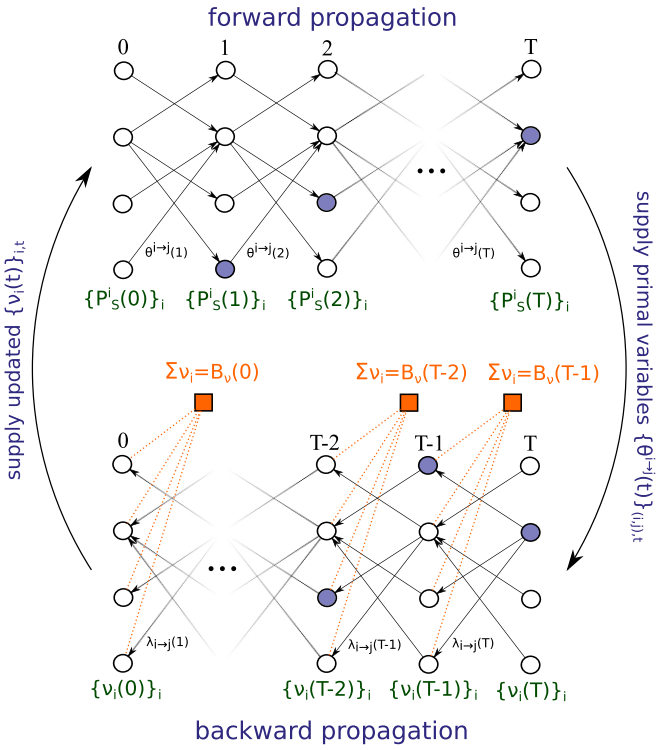

Fig. 2. Schematic representation of the forward–backward propagation algorithm. The optimization scheme is presented in the case of the SI model with spontaneous activations due to the ν-mechanism. The state of the network is presented at each time step; nodes to be targeted at particular time steps are colored in blue. Given the current values of the control parameters $\{\nu_i(t)\}_{i\in V}$, the marginal probabilities $\{P^i(t)\}_{i\in V}$ are computed in the forward propagation stage through the exchange of messages $\{\theta^{i\to j}(t)\}_{(ij)\in E}$ along the edges of the graph between neighboring nodes, according to the update rules of the DMP equations. In the backward propagation phase, the nodes exchange the dual messages, represented by the Lagrange multipliers $\{\lambda_{i\to j}(t)\}_{(ij)\in E}$ associated with the primal variables. At each time step, the parameters $\{\nu_i(t)\}_{i\in V}$ are updated according to the backward dynamic equations, subject to the budget constraints, depicted by orange squares, and the targeting requirements. The two stages are iterated until convergence of the algorithm to a fixed point, or for a predefined number of steps.

optimal distribution of the mitigation budget prioritizing the "too big to fail" financial institutions due to the financial contagion (50) modeled as a spreading process (51, 52); strategy for the active cyber defense dynamics (53) based on spreading "benign" worms (54) for targeting the infected computers and servers; and development of the optimal policy (55) for the acceleration of the diffusion of innovations (3). Let us also mention that targeting can provide algorithms to solve a number of related problems. For instance, identifying the origin of the spreading process from measurements at sparsely located sensors at different times (56) is a difficult problem that has been addressed by other approaches (57, 58), but can be equally viewed as optimally allocating a budget at time 0 to target the sensor nodes at specific times that correspond to the times when measurements were taken.

Despite a wide applicability of the targeting task, until now, no general algorithm is known to drive efficiently the activation process through desired states. In this section, we illustrate the performance of the DMP approach using the general targeting formulation, one of the features of the suggested framework. As a toy example, we consider disinformation spreading on a small network extracted from the 9/11 case study of terrorist associations, representing the established trusted contacts between the hijackers (59). A number of studies suggested methods for destabilizing covert networks; see refs. 60 and 61 for a literature survey. Our rationale in using this example is the ability to demonstrate the targeted activation of nodes at given times, which corresponds to the intentional exposure of the respective individual to misinformation, and considered as one of the protective measures undertaken by the counterterrorism intelligence (62). The spontaneous activation parameters have the interpretation of an aggregated influence [e.g., through counternarratives diffused through the social media by external operatives and special agencies such as the Center for Strategic Counterterrorism Communications (63)]; the resources for such interventions are limited by a certain budget per time step. In the original study (59), the networks of terrorist contacts have been analyzed from the leadership identification perspective: A removal of just several nodes is sufficient to break the network. In our example, the targets specified at each time step may reflect the order of priority in which the nodes should be influenced, in particular those having unique skills for the planned operation (e.g., pilots in the 9/11 example); similar argument has been put forward in the study of the criminal networks (64).

More specifically, we assume that the spreading dynamics follows a particular case of the dynamical model with $\mu_i(t) = 0 \,\forall\, t$ and $i \in V$, corresponding to the SI model with controlled spontaneous transition to the informed state I due to external influence via the control parameters $\nu_i(t)$. The activation of nodes is required in a predefined priority order, targeting selected nodes at specific times. The DMP-based optimization scheme converges to a unique optimal solution within a few forward–backward iterations. The resources are allocated dynamically over time such that the activation path meets the targeting requirements, as reported in Fig. 3: $P_I^i(t_i) > 0.95$ is achieved at all nodes, with the majority of nodes targeted with probability one. Our algorithm is computationally efficient and can be applied to very large network instances, as we show below.

Case Study: Influence Maximization via Seed Selection

The seeding problem, which deals with the optimization over the initial condition only, can be viewed as a particular instance of the targeting formulation. A classical formulation of the seeding task consists of finding the best K nodes which, when activated at initial time, would lead to the maximum spread at time T (31). With the DMP approach being inherently probabilistic,

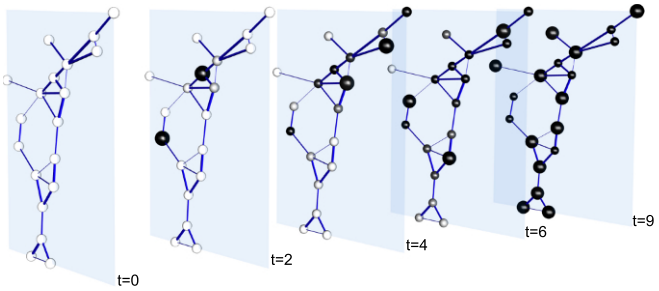

Fig. 3. Optimal targeting with the DMP algorithm on a small network of terrorist associations. Edge thickness indicates the strength of the corresponding pairwise transmission probability α_{ij}, generated uniformly at random in the interval [0, 1]. The size of nodes relates to the time activation requirements: Large nodes should be activated by the corresponding time. In this example, two chosen nodes should be activated at time $t = 2$, another two nodes by time $t = 4$, three particular nodes by time $t = 6$, and all remaining nodes by time $t = 9$; available budget for each time step has been fixed to $B_\nu(t) = 0.1\,N$. Color intensity (gradually from white to black) indicates the value of the marginal probabilities $P_I^i(t)$ which result from the dynamics using the optimal distribution of resources provided by the DMP algorithm. The visualization has been created by using the MuxViz software (65).

we consider a slightly more general (and arguably more realistic in applications) space of initial conditions. Assume that the initial condition is generated probabilistically, so that each node i at the initial time is infected independently with probability p_i; the conventional formulation is recovered when p_i take only binary values, zero or one, and $\sum_{i\in V} p_i = K$. In the considered formulation involving control parameters $\nu_i(t)$, we optimize over the set of arbitrary probabilities $p_i \in [0, 1]$: Setting the initial conditions for such p_i at time $t = 1$ is equivalent to fixing the values of the parameters $\nu_i(0)$ at the auxiliary time $t = 0$ in the system where all nodes are in state S. Therefore, an optimal distribution of the budget $B_\nu(0)$ (not necessarily integer) at time $t = 0$ would thus lead through spontaneous infection to the maximum spread $\mathcal{S}(T + 1)$. In *SI Text*, we present an example of the influence maximization problem in a network of relations between political parties illustrated in Fig. S2. As in the previous example for the targeting problem, the forward–backward optimization scheme quickly converges to a unique ground-truth optimal solution, which in this small test case can be established by a direct maximization of the explicit symbolic form of the objective function $\mathcal{O}$; see Fig. S3 for additional details. This small-scale example hence serves as a validation of our optimization procedure.

A large number of topology-based algorithms have been designed to address the seeding problem, mostly in the case of the homogeneous transmission probabilities (17–22, 25, 28). To test the efficacy of the DMP-optimization approach on large-scale instances, we compare its performance to that of popular heuristics for the restricted setting of near-deterministic spreading. The choice for this setting is motivated by the fact that one can devise a simple algorithm providing a good approximation to the ground-truth solution, which can serve as a benchmark for comparing different algorithms, and a number of centrality techniques (17, 22, 25) selecting combinations of high-degree nodes should perform well in this case; see *SI Text* for a detailed discussion of methods used for comparison, implementations, and additional remarks. The results of comparisons on a number of real-world and synthetic networks of different sizes and topologies are summarized in Table 1 and Fig. 4. In *SI Text*, we also discuss additional numerical results for the case of heterogeneous couplings, assessing the performance of the DMP method compared with a natural generalization of centrality algorithms to the heterogeneous setting. The main message emerging from these tests can be formulated as follows: Although the DMP method has not been optimized for the seeding problem and does not rely explicitly on topological features such as targeting high-degree nodes, we find that it is close to the best-performing heuristics in all cases, showing a consistently good performance. This suggests that the DMP algorithm performs well also for more general dynamic resource allocation problems for which other principled methods do not exist, such as targeting problems described in the previous section and global-time closed-loop control policies discussed further.

An interesting observation is that in the case of large network instances, the forward–backward iteration scheme no longer converges to a unique optimum as in the case of small networks considered previously. Instead, the algorithm makes large "jumps" on the manifold representing different control-parameter distributions that obey the budget constraints [**6**]: This is a manifestation of the NP-hardness of the problem with a more complex optimization landscape and a multitude of local optima. The presence of many solutions with comparable costs is an indication that it is arguably more appropriate to view the different seeding sets as a collective phenomenon, rather than assigning "influence" measure to individual nodes. In terms of computational complexity, solving the dynamics with DMP is linear in T and $|E|$; the number of forward–backward iterations is typically small and can be controlled, as explained in *Methods*. Let us also point out that the DMP-estimated marginals provide a natural and convenient measure for comparing the performance of different algorithms in the finite time horizon setting, especially on large graphs with millions of nodes where running extensive Monte Carlo simulations is computationally prohibitive.

Case Study: Online Mitigation of Epidemic Spreading

To illustrate the suitability of the DMP algorithm to online deployment of resources in a dynamic setting with feedback, we use a prototypical example: developing an effective mitigation policy for confining an infectious disease—a practical and challenging question of public concern. A modified SIR model with vaccination is an appropriate dynamic model in this case, where the $\nu_i(t)$ variables are set to zero, and the parameters $\mu_i(t)$ play the role of vaccination control, allowing the susceptible nodes to assume a protected state R. The vaccination mechanism modeled via an S to R transition has been studied in the context of the SIR-type spreading models in refs. 70 and 71. Note that extension to other spreading models with different vaccination mechanisms is straightforward, as pointed out in *Discussion*. In contrast to the targeting and seeding problems, the initial conditions (origin of the epidemic) are specified in this setting, and the vaccination budget has to be allocated dynamically according to the current state of the spreading process (monitored at each time step) to suppress the epidemic. The goal is to deploy

Table 1. Comparison of the DMP algorithm for the seeding problem in the setting of near-deterministic dynamics with popular heuristics on various real-world and artificial networks

Network	N	M	Random	HDA	k-shell	CI_2	CI_4	Uniform	DMP	Covering
Road EU	1174	1417	0.305	0.480	0.163	0.500	0.468	0.324	0.513	0.565
Protein	2361	6646	0.736	0.863	0.772	0.861	0.838	0.752	0.856	0.903
US power grid	4941	6594	0.367	0.602	0.206	0.605	0.565	0.397	0.601	0.684
GR collaborations	5242	14,484	0.565	0.644	0.291	0.660	0.658	0.634	0.710	0.796
Internet	22,963	48,436	0.880	0.998	0.987	0.996	0.994	0.891	0.972	0.995
Web-sk	121,422	334,419	0.645	0.833	0.242	0.751	0.734	0.699	0.837	0.937
Scale-free	500,000	397,848	0.214	0.398	0.220	0.372	0.323	0.215	0.321	0.427
Erdős-Rényi	500,000	750,000	0.447	0.681	0.494	0.677	0.679	0.446	0.704	0.719

First three columns on the left of the table provide topological information on the networks considered (66–69). In the remaining columns on the right are presented values of the normalized total spread $\mathcal{S}(T)/N$ at time $T = 3$, given homogeneous transmission probabilities $\alpha_{ij} = \alpha = 0.99$ and the total available budget $B_\nu(0) = 0.05\,N$, for the different algorithms: assignment to randomly-selected nodes, an adaptive version of the high-degree strategy of ref. 17 (HDA) and of the k-shell decomposition (22), collective influence $(CI)_l$ (25) (with $l = 2$ and $l = 4$), uniform assignment, the DMP algorithm, and the Covering algorithm which has a near-optimal performance in this case and serves as a benchmark. Description of these algorithms along with the implementation details is provided in *SI Text*. For different test cases, solutions obtained by DMP span the range between delocalized and node-centric assignments and are on par with the best-performing centrality heuristics. The results presented in this table are graphically summarized in Fig. 4. See *SI Text* and Table S1 for analogous comparisons in the case of heterogeneous couplings.

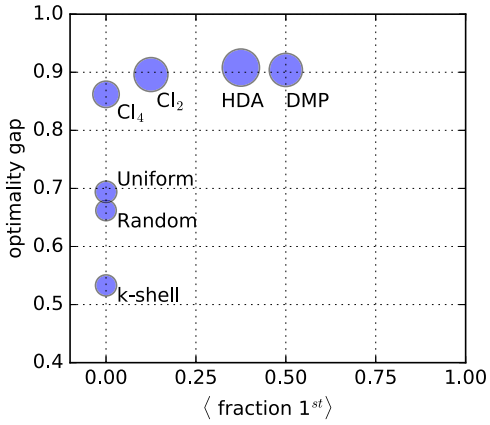

Fig. 4. Comparison of different algorithms for the seeding problem. This figure summarizes in a graphical form the results of comparisons between several seeding algorithms for the near-deterministic spreading on a number of real and synthetic network instances; details and raw numbers are provided in Table 1. Each algorithm is represented by a circle, centered at the point with coordinates (x, y), with x being the fraction of test cases for which the algorithm had the best performance, and y defined as the average "optimality gap" [i.e., average result normalized by the value output by the specially designed Covering strategy (near-optimal in this setting, as explained in *SI Text* and in Fig. S4) for each of the networks studied]. The ideal algorithm for this problem should lie in the right upper corner with coordinates (1, 1). The size of each circle is inversely proportional to the average rank of the corresponding algorithm. The high degree adaptive (HDA) policy shows a slightly lower optimality gap compared with DMP; nevertheless, overall, the DMP approach (which is not specifically optimized for the seeding problem) demonstrates a consistently good performance; this is a premise for its proper performance in general large-scale general dynamic resource allocation problems.

the resources optimally so that the total number of infected nodes $\mathcal{S}(T)$ at the final time is minimized. The assumption of a time-distributed budget $B_\mu(t)$ is highly reasonable due to the restricted vaccine availability.

Previously developed real-time strategies for mitigating contagion on a given network (35, 37, 38) explored policies that were based on topological characteristics of the graph under the assumption of homogeneous transmission probabilities. The common denominator of existing approaches consists in local interventions, which ensure the islanding of infected nodes. We generalize the methods (35, 38) to the case of heterogeneous transmission probabilities using a "high-risk" (36) ranking of susceptible nodes at time t according to their probability of getting infected at the next time step. This measure is defined in our case as

$$P_i^t(S \to I) = 1 - \prod_{j \in \partial i} (1 - \alpha_{ji} \mathbb{1}[\sigma_j^t = I]), \qquad [9]$$

where ∂i denotes the set of neighbors of a susceptible node i. A reasonable local intervention strategy for benchmarking consists of distributing the vaccination budget to priority nodes with a high-risk measure [**9**]. This algorithm will be referred to as the greedy strategy.

Several policies can be conceived by using the DMP optimization framework. As a reference, we consider the planned deployment of resources which does not take into account feedback from an actual realization of the process, but merely follows the solution of the dynamic resource allocation problem with a specified initial condition. Two other closed-loop strategies take into account the real-time information on the spreading process, using the seeding formulation as a subroutine: (*i*) The first, termed "DMP-greedy," is close in spirit (but differs in the algorithmic implementation, based here on the DMP optimization framework) to the greedy algorithm and uses the current state of the epidemic as the initial condition, aiming to minimize the spread at the next time step only. (*ii*) The second uses the full power of the DMP framework by exploiting the up-to-date information available to reinitialize the dynamics at each time step t for allocating the resources at the next time step $t+1$, by running the optimization procedure for the remaining $T-t$ time steps. This "DMP-optimal" policy is similar to the planned strategy, but takes advantage of the new information available from the realization of the process.

We compare these strategies for the case of infection spreading mediated by air traffic, which has been recognized as an important factor facilitating the spread of infectious diseases (72) and thus plays a major role in recent world's pandemics (73). As a particular example, we study the real-world transportation network of busiest flight routes between major US airports, extracted from the Bureau of Transportation Statistics (BTS) data (74) and depicted in Fig. 5*A*. The use of the modified SIR model in this case is justified by the fact that this type of spreading models has been widely used for modeling a traffic-mediated epidemic (75, 76). We use a plausible assumption that the infection transmission probability associated with a link between airports is proportional to the number of passengers carried along this route (see *SI Text* for a detailed description of the network and data used). The "vaccination" interventions on this network can be interpreted as quarantine measures taken in different airports using the updates on the newly infected cases. Indeed, containment measures and travel restrictions have been pointed out as important factors limiting the spread of an epidemic (77). In the simulations, we assume that the epidemic starts at the largest airport hub of Atlanta.

The comparison of different mitigation algorithms is given in Fig. 5*B*, showing the average number of infected sites as a function of time under different mitigation strategies. As expected, the DMP-optimal scheme represents the best performing policy, which leads to stabilization of the expected number of infected nodes by $t = 6$, at a lower level compared with the greedy algorithm that optimizes the spread at the next time step only. Notice that on a short time scale, the greedy algorithm has a slightly better performance, which represents a typical situation when localized and immediate optimal decisions lead ultimately to suboptimal global optimization results; an illustration is provided in Fig. 5*C*.

Discussion

We introduce an efficient and versatile optimization framework for solving dynamic resource allocation problems for spreading processes, which allows for the synthesis of previously studied settings within a general targeting formulation. This probabilistic framework allows one to study problems that involve a finite-time horizon, which requires an explicit solution of the dynamics, the targeting of specific nodes at given times in both open- and closed-loop setting, as well as scenarios where only a subset of the nodes is accessible. This is done in our scheme by using the DMP equations for spreading processes. Although in this work we focus on examples involving the discrete-time modified SIR model, the approach can be straightforwardly applied to the case of continuous dynamics (the continuous formulation is expounded in *SI Text*) and to other cascading models, including but not limited to threshold models (29, 41) and rumor dynamics (42). Another possible application area of the present framework relates to systems defined on temporal graphs, where network dynamics can be encoded into the time-dependent coefficients $\alpha_{ij}(t)$ within the existing framework.

Although we show that the method can be used in the case where transmission probabilities are uniform and only the detailed topology of the network is known, its major advantage consists of the ability to incorporate efficiently detailed

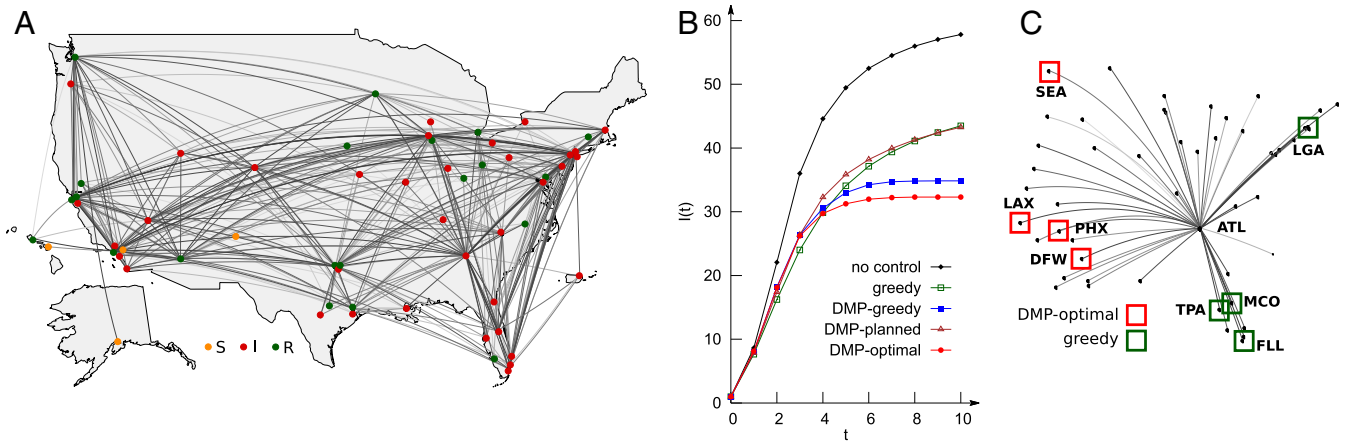

Fig. 5. Online mitigation of air-traffic mediated epidemic on the network of flights between major US hubs. (*A*) A geographical layout of the air transportation subnetwork extracted from the BTS data (74). The transmission probabilities are indicated by the thickness of the corresponding edges, which are proportional to the aggregated traffic between airports. Different colors of airports (yellow, red, and green) represent an outcome of a single realization of the spreading dynamics (nodes in the susceptible, infected, and recovered states, respectively) under the DMP-optimal policy. (*B*) Comparisons of mitigation strategies show the average number of infected sites as a function of time, averaged over 100 random realizations of the dynamics, for the different policies. In the simulations, the epidemic starts at the largest airport hub of Atlanta; a budget of $B_\mu(t) = 0.05\ N$ is available at each time step, and the objective is to suppress the epidemic by $T = 10$. The DMP-optimal algorithm demonstrates the best performance in the number of infected nodes at time T. (*C*) An illustration of a radically different decisions taken by the DMP-optimal and greedy algorithms already at the first step of the optimization: The greedy policy chooses to vaccinate nodes which are most "in danger" at the next time step, while the decision done by the DMP-optimal scheme takes into account the forecasted evolution of the dynamics.

information on transmission probabilities when such prior information is available, or can be either estimated (as in the examples of flight transportation networks, given above, and Slovene political parties, treated in *SI Text*) or learned from past observations of the dynamics (78, 79). Despite the global budget constraints involving all network nodes, the resulting message-passing scheme is fast and distributed, requiring a number of operations which grows linearly in time and with respect to the number of edges in the network. An attractive property of the suggested framework is its versatility: Instead of optimizing the spread given a fixed budget, one can minimize the budget needed to meet certain requirements on the spread, imposed as a constraint in the Lagrangian formulation. Another interesting scenario is the optimization over the spreading parameters α_{ij}: This formulation is useful in the design of technological networks or for mitigation of an epidemic by removing and adding links in the graph. It would be interesting to apply the presented optimization scheme to the percolation-type equations describing the asymptotic $T \to \infty$ limit of the spreading dynamics with heterogeneous couplings.

The optimization method used is based on changes to the entire trajectory, instead of taking incremental improvement steps in the direction of the gradient; thus, the suggested algorithm results in large steps and arguably explores more effectively the parameter space. The fact the optimization is gradient-free represents an additional advantage from the point of view of the computational complexity in problems where the gradient is hard to compute; for instance, in the case of the DMP equations presented in this work, computation of the gradient requires $O(|E|NT)$ operations for the node-related control parameters, and $O(|E|^2 T)$ operations for the edge-related parameters, which would make the algorithm impractical for large networks. This property of the optimization scheme makes it an attractive option for the DMP-based learning algorithms (78), where the gradient computation represents a scalability bottleneck. The solution of the learning problem in the presence of hidden nodes together with the introduced targeting formulation would make it possible to construct the DMP-based artificial learning architectures.

Notice that, in principle, the forward–backward algorithm is not tailored to the DMP equations paradigm and can be used in the same context for a broader class of dynamical systems governed by nonlinear differential equations; however, implementing forward and backward steps through simulation of the dynamics may significantly increase the computational complexity of the overall algorithm. Another open problem is dealing with uncertainties within the presented framework. In realistic applications, the spreading parameters are never known with an absolute accuracy, but in a certain range, defined by the estimation error. Obviously, the optimization algorithm should be able to take these uncertainties into account. It would be useful to develop the robust version of our formulation, in the spirit of the setting known as robust influence maximization (80, 81).

Methods

DMP Equations. DMP belongs to the family of algorithms derived by using the cavity method of statistical physics and may be given an interpretation of passing messages along the graph edges. The obtained marginals are exact on tree graphs and asymptotically exact on sparse random networks. We provide an intuitive derivation of the DMP equations for the adopted modified SIR model, defined by Eqs. **1** and **2**. On a given instance of a network, these equations allow one to compute the marginal probability distributions $P^i_\sigma(t)$, where $\sigma \in \{S, I, R\}$ denotes the node state. The first key equation reads:

$$P^i_S(t) = P^i_S(0) \left(\prod_{t'=0}^{t-1} (1 - \nu_i(t'))(1 - \mu_i(t')) \right) \prod_{k \in \partial i} \theta^{k \to i}(t). \qquad [10]$$

It states the probability of node i to be susceptible at time t and is equal to the probability that i was in the S state at initial time $P^i_S(0)$ and remained so until time t. It neither changed states by following the ν and μ mechanisms (in brackets), nor by being infected by a neighbor (final term on right); the dynamic message $\theta^{k \to i}(t)$ has a meaning of the probability that node k did not pass an activation message to node i until time t. Strictly speaking, Eq. **10** is only valid on a tree graph; only in this case $\theta^{k \to i}(t)$ are independent for all $k \in \partial i$, so that the corresponding probability is factorized as in Eq. **10**. However, in practice, the decorrelation assumption holds to a good precision on general networks, even with small loops (see ref. 42 for in-depth discussions and supporting numerical experiments). The quantities $\theta^{k \to i}(t)$ are updated as follows:

$$\theta^{k \to i}(t) = \theta^{k \to i}(t-1) - \alpha_{ki} \phi^{k \to i}(t-1), \qquad [11]$$

which corresponds to the fact that $\theta^{k\to i}(t)$ can only decrease if an activation signal is passed along the directed link (ki); the corresponding probability equals the product of α_{ki} and the dynamic variable $\phi^{k\to i}(t-1)$, which has a meaning of the probability that node k is in the state I at time $t-1$, but has not infected node i until time $t-1$. To simplify further explanations, we introduce the dynamic messages $P_S^{k\to i}(t)$, $P_I^{k\to i}(t)$ and $P_R^{k\to i}(t)$, which denote the probabilities that node k is found at time t in the states S, I, or R, respectively, conditioned on node i remaining in state S. Alternatively, these variables can be thought of as the probabilities of k being susceptible, infected or recovered on a cavity graph, on which node i has been removed. Formally,

$$P_S^{k\to i}(t) = P_S^k(0)\left(\prod_{t'=0}^{t-1}(1-\nu_k(t'))(1-\mu_k(t'))\right)\prod_{l\in\partial k\setminus i}\theta^{l\to k}(t), \quad [12]$$

which coincides with the expression [**10**], except that $\theta^{i\to k}(t)$ is not included in the product on the right ($\partial k\setminus i$ denotes the set of neighbors of k without i). We also have

$$P_R^{k\to i}(t) = P_R^{k\to i}(t-1) + \mu_k(t-1)P_S^{k\to i}(t-1), \quad [13]$$

which expresses the monotonic increase of $P_R^{k\to i}(t)$ at each time step with the probability $\mu_k(t-1)P_S^{k\to i}(t-1)$, and

$$P_I^{k\to i}(t) = 1 - P_S^{k\to i}(t) - P_R^{k\to i}(t) \quad [14]$$

due to the normalization of probabilities. We are now ready to formulate the last relation which leads to the closure of the system of message-passing equations. The evolution of the message $\phi^{k\to i}(t)$ reads:

$$\phi^{k\to i}(t) = (1-\alpha_{ki})\phi^{k\to i}(t-1) + \Delta P_I^{k\to i}(t-1) \quad [15]$$

where $\Delta P_I^{k\to i}(t-1) \equiv P_I^{k\to i}(t) - P_I^{k\to i}(t-1)$. The physical meaning of Eq. **15** is as follows: $\phi^{k\to i}(t)$ decreases if the activation signal is actually transmitted (first term) and increases if node k transitions to the state I at the current time step. Eqs. **10**–**15** can be iterated in time starting from the given initial conditions $\{P_S^i(0), P_I^i(0), P_R^i(0)\}_{i\in V}$, with

$$\theta^{i\to j}(0) = 1, \quad \phi^{i\to j}(0) = \delta_{\sigma_i^0, I} = P_I^i(0). \quad [16]$$

The marginals $P_S^i(t)$ used throughout the text are obtained by using Eq. **10**, while $P_I^i(t)$ and $P_R^i(t)$ are computed via

$$P_R^i(t) = P_R^i(t-1) + \mu_i(t-1)P_S^i(t-1), \quad [17]$$

$$P_I^i(t) = 1 - P_S^i(t) - P_R^i(t). \quad [18]$$

The computational complexity of the DMP equations for solving the dynamics up to time T is given by $O(|E|T)$, where $|E|$ is the number of edges in the graph, which makes them scalable to sparse networks with millions of nodes. For spreading models other than SIR, DMP equations can be systematically derived from the initial dynamic transition rules, as shown in ref. 42.

Enforcing Dynamical Constraints and Backward Equations. The dynamics $\mathcal{D}$ and initial conditions $\mathcal{I}$ constraints are enforced in a similar way to that of $\mathcal{P}$ and the budget $\mathcal{B}$ constraints in Eqs. **7** and **8**. To each generic dynamic variable $\xi^i(t)$ and message $\chi^{k\to i}(t)$, we associate the corresponding Lagrange multipliers $\lambda_i^\xi(t)$ and $\lambda_{k\to i}^\chi(t)$, which enforce the relation between dynamic variables at subsequent times. For instance, the evolution of the quantities $\{P_R^i(t)\}_{i\in V}$ in the Lagrangian $\mathcal{L}$ is enforced via the term

$$\sum_{i\in V}\sum_{t=0}^{T-1}\lambda_i^R(t+1)\left[P_R^i(t+1) - P_R^i(t) - \mu_i(t)P_S^i(t)\right].$$

Variation with respect to the dual variables $\lambda_i^\xi(t)$ and $\lambda_{k\to i}^\chi(t)$ returns the forward DMP Eqs. **10**–**18**, while setting to zero the derivative of $\mathcal{L}$ with respect to the primal dynamic variables yields the relations between the Lagrange multipliers at subsequent times, which we interpret as the backward dynamic equations in our scheme. Similarly to Eqs. **10**–**15**, the backward equations have a distributed message-passing structure with linear computational complexity $O(|E|T)$ and are used to update the values of control parameters $\nu_i(t)$ and $\mu_i(t)$ at each iteration, taking into account the budget requirements [**6**]. Specifically, initializing the control parameters $\nu_i(t)$ and $\mu_i(t)$ to some arbitrary values (e.g., uniform over all nodes and times), we first propagate the DMP equations forward in time, up to the horizon T; then, using the existing primal parameter values, we fix end-point conditions for the dual parameters and propagate the equations for the dual parameters backward in time, updating the control parameters respecting the budget and variation constraints. These two steps are iterated for a predefined number of times or until global convergence of the process.

In the large-scale problems, where the algorithm explores the space of parameters by hopping from one solution to another, we choose a simple strategy: We run the forward–backward algorithm for several iterations for a range of values of the regularization parameter ϵ, which appears in the $\mathcal{P}$ block, and keep track of the best local optimum which provides the solution to the optimization problem after a maximum number of iterations (kept below the desired threshold which determines the computational complexity) is reached. The choice of ϵ impacts the type of solution obtained: Larger values of ϵ correspond to solutions where the budget is disseminated more uniformly across nodes, while smaller values lead to weight concentration on particular nodes. Depending on the application and the level of control over nodes, one type of solution can be preferred to another; this flexibility represents an attractive feature of the DMP algorithm. An explicit form of the Lagrangian for the problems considered in this work together with additional details is given in *SI Text*.

ACKNOWLEDGMENTS. We thank M. Chertkov, S. Misra, and M. Vuffray for fruitful discussions and valuable comments. A.Y.L. was supported by Laboratory Directed Research and Development Program at Los Alamos National Laboratory by the National Nuclear Security Administration of the US Department of Energy under Contract DE-AC52-06NA25396. D.S. was supported by Leverhulme Trust Grant RPG-2013-48.

1. Anderson RM, May RM, Anderson B (1992) *Infectious Diseases of Humans: Dynamics and Control* (Oxford Univ Press, Oxford), Vol 28.
2. Boccaletti S, Latora V, Moreno Y, Chavez M, Hwang DU (2006) Complex networks: Structure and dynamics. *Phys Rep* 424:175–308.
3. Rogers EM (2010) *Diffusion of Innovations* (Simon and Schuster, New York).
4. Pastor-Satorras R, Castellano C, Van Mieghem P, Vespignani A (2015) Epidemic processes in complex networks. *Rev Mod Phys* 87:925–979.
5. Federal Energy Regulatory Commission and North American Electric Reliability Corporation (2012) *Arizona - Southern California Outages on September 8, 2011: Causes and Recommendations* (Federal Energy Regulatory Commission and North American Electric Reliability Corporation, Washington, DC).
6. Dawood FS, et al. (2012) Estimated global mortality associated with the first 12 months of 2009 pandemic influenza a h1n1 virus circulation: A modelling study. *Lancet Infect Dis* 12:687–695.
7. US Government Accountability Office (2012) *Financial Regulatory Reform: Financial Crisis Losses and Potential Impacts of the Dodd-Frank Act* (Government Accountability Office, Washington, DC).
8. Lokhov AY, Lemons N, McAndrew TC, Hagberg A, Backhaus S (2016) Detection of cyber-physical faults and intrusions from physical correlations. *Proceedings of the 2016 IEEE 16th International Conference on Data Mining Workshops* (IEEE, New York), pp 303–310.
9. ALS Association (2016) ALS Ice Bucket Challenge. Available at www.alsa.org/fight-als/ice-bucket-challenge.html. Accessed August 31, 2016.
10. Rutledge P (2013) How Obama won the social media battle in the 2012 presidential campaign, The Media Psychology Blog. Available at mprcenter.org/blog/2013/01/how-obama-won-the-social-media-battle-in-the-2012-presidential-campaign/. Accessed August 31, 2016.
11. Epstein R, Robertson RE (2015) The search engine manipulation effect (seme) and its possible impact on the outcomes of elections. *Proc Natl Acad Sci USA* 112:E4512–E4521.
12. Margetts H, John P, Hale S, Yasseri T (2015) *Political Turbulence: How Social Media Shape Collective Action* (Princeton Univ Press, Princeton).
13. Domingos P, Richardson M (2001) Mining the network value of customers. *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York), pp 57–66.
14. Martin KR, et al. (2013) Computational model for autophagic vesicle dynamics in single cells. *Autophagy* 9:74–92.
15. Csermely P, Korcsmáros T, Kiss HJ, London G, Nussinov R (2013) Structure and dynamics of molecular networks: A novel paradigm of drug discovery: A comprehensive review. *Pharmacol Therpeut* 138:333–408.
16. Jones S (2015) Army revives chindits as 'Facebook warriors' for smart battle. *Financ Times*. Available at https://www.ft.com/content/537c7436-a892-11e4-ad01-00144feab7de.
17. Pastor-Satorras R, Vespignani A (2002) Immunization of complex networks. *Phys Rev E* 65:036104.
18. Cohen R, Havlin S, Ben-Avraham D (2003) Efficient immunization strategies for computer networks and populations. *Phys Rev Lett* 91:247901.
19. Holme P, Kim BJ, Yoon CN, Han SK (2002) Attack vulnerability of complex networks. *Phys Rev E* 65:056109.
20. Holme P (2004) Efficient local strategies for vaccination and network attack. *Europhys Lett* 68:908–914.
21. Chen Y, Paul G, Havlin S, Liljeros F, Stanley HE (2008) Finding a better immunization strategy. *Phys Rev Lett* 101:058701.
22. Kitsak M, et al. (2010) Identification of influential spreaders in complex networks. *Nat Phys* 6:888–893.

23. Borge-Holthoefer J, Moreno Y (2012) Absence of influential spreaders in rumor dynamics. *Phys Rev E* 85:026116.
24. Hébert-Dufresne L, Allard A, Young JG, Dubé LJ (2013) Global efficiency of local immunization on complex networks. *Sci Rep* 3:2171.
25. Morone F, Makse HA (2015) Influence maximization in complex networks through optimal percolation. *Nature* 524:65–68.
26. Mugisha S, Zhou HJ (2016) Identifying optimal targets of network attack by belief propagation. *Phys Rev E* 94:012305.
27. Braunstein A, Dall'Asta L, Semerjian G, Zdeborová L (2016) Network dismantling. *Proc Natl Acad Sci USA* 113:12368–12373.
28. Altarelli F, Braunstein A, Dall'Asta L, Wakeling JR, Zecchina R (2014) Containing epidemic outbreaks by message-passing techniques. *Phys Rev X* 4:021024.
29. Altarelli F, Braunstein A, Dall'Asta L, Zecchina R (2013) Optimizing spread dynamics on graphs by message passing. *J Stat Mech Theor Exp* 2013:P09011.
30. Guggiola A, Semerjian G (2015) Minimal contagious sets in random regular graphs. *J Stat Phys* 158:300–358.
31. Kempe D, Kleinberg J, Tardos É (2003) Maximizing the spread of influence through a social network. *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York), pp 137–146.
32. Chen W, Lakshmanan LV, Castillo C (2013) Information and Influence Propagation in Social Networks, *Synthesis Lectures on Data Management* (Morgan & Claypool, Williston, VT), Vol 5.
33. Du N, Song L, Gomez-Rodriguez M, Zha H (2013) Scalable influence estimation in continuous-time diffusion networks. *Adv Neural Inf Process Syst* 26:3147–3155.
34. Nowzari C, Preciado VM, Pappas GJ (2016) Analysis and control of epidemics: A survey of spreading processes on complex networks. *IEEE Control Syst* 36:26–46.
35. Borgs C, Chayes J, Ganesh A, Saberi A (2010) How to distribute antidote to control epidemics. *Random Struct Algorithm* 37(2):204–222.
36. Nian F, Wang X (2010) Efficient immunization strategies on complex networks. *J Theor Biol* 264(1):77–83.
37. Scaman K, Kalogeratos A, Vayatis N (2015) A Greedy Approach for Dynamic Control of Diffusion Processes in Networks. *Proceedings of the IEEE 27th International Conference on Tools with Artificial Intelligence (IEEE, New York),* pp 652–659.
38. Drakopoulos K, Ozdaglar A, Tsitsiklis JN (2014) An efficient curing policy for epidemics on graphs. *IEEE Trans Netw Sci Eng* 1:67–75.
39. Shapiro M, Delgado-Eckert E (2012) Finding the probability of infection in an SIR network is NP-hard. *Math biosciences* 240:77–84.
40. Karrer B, Newman MEJ (2010) Message passing approach for general epidemic models. *Phys Rev E* 82:016101.
41. Shrestha M, Moore C (2014) Message-passing approach for threshold models of behavior in networks. *Phys Rev E* 89:022805.
42. Lokhov AY, Mézard M, Zdeborová L (2015) Dynamic message-passing equations for models with unidirectional dynamics. *Phys Rev E* 91:012811.
43. le Cun Y, (1988) A theoretical framework for back-propagation. *Proceedings of the 1988 Connectionist Models Summer School*, eds Touresky D, Hinton G, Sejnowski T (Morgan Kaufman, San Mateo, CA), Vol 1, pp 21–28.
44. Saad D, Rattray M (1997) Globally optimal parameters for on-line learning in multilayer neural networks. *Phys Rev Lett* 79:2578.
45. Sebolt-Leopold JS, Herrera R (2004) Targeting the mitogen-activated protein kinase cascade to treat cancer. *Nat Rev Cancer* 4:937–947.
46. Roberts PJ, Der CJ (2007) Targeting the raf-mek-erk mitogen-activated protein kinase cascade for the treatment of cancer. *Oncogene* 26:3291–3310.
47. Gu S, et al. (2017) Optimal trajectories of brain state transitions. *NeuroImage* 148: 305–317.
48. Sahasrabudhe S, Motter AE (2011) Rescuing ecosystems from extinction cascades through compensatory perturbations. *Nat Commun* 2:170.
49. Suweis S, Simini F, Banavar JR, Maritan A (2013) Emergence of structural and dynamical properties of ecological mutualistic networks. *Nature* 500:449–452.
50. Allen F, Gale D (2000) Financial contagion. *J Polit Econ* 108:1–33.
51. Paga P, Kühn R (2015) Contagion in an interacting economy. *J Stat Mech Theor Exp* 2015:P03008.
52. Caccioli F, Shrestha M, Moore C, Farmer JD (2014) Stability analysis of financial contagion due to overlapping portfolios. *J Banking Finance* 46:233–245.
53. Lu W, Xu S, Yi X (2013) *Optimizing Active Cyber Defense in International Conference on Decision and Game Theory for Security* (Springer, New York), pp 206–225.
54. Kephart JO, White SR (1991) *Directed-graph epidemiological models of computer viruses. Proceedings of the IEEE Computer Society Symposium on in Research in Security and Privacy IEEE* (IEEE, New York), pp 343–359.
55. Maienhofer D, Finholt T (2002) Finding optimal targets for change agents: A computer simulation of innovation diffusion. *Comput Math Organ Theor* 8:259–280.
56. Pinto PC, Thiran P, Vetterli M (2012) Locating the source of diffusion in large-scale networks. *Phys Rev Lett* 109:068702.
57. Lokhov AY, Mézard M, Ohta H, Zdeborová L (2014) Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Phys Rev E* 90:012801.
58. Altarelli F, Braunstein A, Dall'Asta L, Lage-Castellanos A, Zecchina R (2014) Bayesian inference of epidemics on networks via belief propagation. *Phys Rev Lett* 112: 118701.
59. Krebs VE (2002) Mapping networks of terrorist cells. *Connections* 24:43–52.
60. Choudhary P, Singh U (2015) A survey on social network analysis for counter-terrorism. *Int J Computer Appl* 112:14–29.
61. Knoke D (2015) *Emerging Trends in Social Network Analysis of Terrorism and Counterterrorism, Emerging Trends in the Social and Behavioral Sciences* (Wiley Online Library, New York).
62. Ushakov IA (2013) *Optimal Resource Allocation: With Practical Statistical Applications and Theory* (John Wiley & Sons, New York).
63. United Nations Office on Drugs and Crime (2012) *The Use of the Internet for Terrorist Purposes (United Nations, Vienna)*.
64. Klerks P (2001) The network paradigm applied to criminal organizations: Theoretical nitpicking or a relevant doctrine for investigators? Recent developments in The Netherlands. *Connections* 24:53–65.
65. De Domenico M, Porter MA, Arenas A (2015) Muxviz: A tool for multilayer analysis and visualization of networks. *J Complex Networks* 3:159–176.
66. Šubelj L, Bajec M (2011) Robust network community detection using balanced propagation. *The Eur Phys J B* 81:353–362.
67. Bu D, et al. (2003) Topological structure analysis of the protein–protein interaction network in budding yeast. *Nucleic Acids Res* 31:2443–2450.
68. Leskovec J, Kleinberg J, Faloutsos C (2007) Graph evolution: Densification and shrinking diameters. *ACM Trans Knowledge Discov Data* 1:2.
69. Boldi P, Codenotti B, Santini M, Vigna S (2004) UbiCrawler: A scalable fully distributed web crawler. *Software Pract Ex* 34:711–726.
70. Kribs-Zaleta CM, Velasco-Hernandez JX (2000) A simple vaccination model with multiple endemic states. *Math biosciences* 164:183–201.
71. Reluga TC, Medlock J (2007) Resistance mechanisms matter in SIR models. *Math Biosciences Eng* 4:553–563.
72. Hollingsworth TD, Ferguson NM, Anderson RM (2007) Frequent travelers and rate of spread of epidemics. *Emerging Infect Dis* 13:1288–1294.
73. Tatem AJ, Rogers DJ, Hay SI (2006) Global transport networks and infectious disease spread. *Adv Parasitol* 62:293–343.
74. Bureau of Transportation Statistics (2016) Bureau of Transportation Statistics. Available at https://www.transtats.bts.gov/DataIndex.asp. Accessed August 31, 2016.
75. Colizza V, Barrat A, Barthélemy M, Vespignani A (2006) The role of the airline transportation network in the prediction and predictability of global epidemics. *Proc Natl Acad Sci USA* 103:2015–2020.
76. Brockmann D, Helbing D (2013) The hidden geometry of complex, network-driven contagion phenomena. *Science* 342:1337–1342.
77. Epstein JM, et al. (2007) Controlling pandemic flu: The value of international air travel restrictions. *PLoS One* 2:e401.
78. Lokhov A (2016) Reconstructing parameters of spreading models from partial observations. *Adv Neural Inf Process Syst* 29:3467–3475.
79. Lokhov AY, Misiakiewicz T (2015) Efficient reconstruction of transmission probabilities in a spreading process from partial observations. arXiv:1509.06893.
80. He X, Kempe D (2016) Robust influence maximization. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York), pp 885–894.
81. Chen W, Lin T, Tan Z, Zhao M, Zhou X (2016) Robust influence maximization. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York), pp 795–804.
82. Doreian P, Mrvar A (1996) A partitioning approach to structural balance. *Social networks* 18:149–168.
83. Newman ME (2002) Spread of epidemic disease on networks. *Phys Rev E* 66:016128.
84. Granovetter M (1978) Threshold models of collective behavior. *Am J Sociol* 83: 1420–1443.
85. Morone F, Min B, Bo L, Mari R, Makse HA (2016) Collective influence algorithm to find influencers via optimal percolation in massively large social media. *Sci Rep* 6: 30062.
86. Hildebrand FB (1992) *Methods of Applied Mathematics* (Dover Publications, New York).

Supporting Information

Lokhov and Saad 10.1073/pnas.1614694114

SI Text

Model and DMP Equations

All models considered in the main text represent variants of a SIR model with possible spontaneous infection and vaccination transitions. We refer to this model as the modified SIR model, because the spontaneous $S \to I$ and $S \to R$ transitions are not considered in the standard SIR dynamics; instead, the conventional SIR model contains a spontaneous recovery transition $I \to R$ which was not relevant for the examples considered in the work and therefore was ignored. However, the inclusion of this transition in the DMP equations is very easy and has been done in ref. 42. Hence, the model we consider here is defined as follows: At each time step t, the transitions from the state S to I and R occur with the probabilities summarized in Fig. S1 for individual nodes and edges. In the discrete time setting considered throughout the work, it may occur that both transitions to I and R states are realized at the same time; in this case, we assume that the transition to the R state effectively takes place. Note that in the continuous time setting (*Case of Continuous Dynamics*), this effect is of a second order in the discretization step dt, and hence this tie-breaking rule is not required for sufficiently small dt.

The DMP equations associated with this model, as well as the intuition behind them, are described in *Methods*. We repeat them here for consistency:

$$P_S^{k\to i}(t) = P_S^k(0)\left(\prod_{t'=0}^{t-1}(1-\nu_k(t'))\right)\left(\prod_{t'=0}^{t-1}(1-\mu_k(t'))\right)\prod_{l\in\partial k\setminus i}\theta^{l\to k}(t), \quad [S1]$$

$$P_R^{k\to i}(t) = P_R^{k\to i}(t-1) + \mu_k(t-1)P_S^{k\to i}(t-1), \quad [S2]$$

$$\theta^{k\to i}(t) = \theta^{k\to i}(t-1) - \alpha_{ki}\phi^{k\to i}(t-1), \quad [S3]$$

$$\phi^{k\to i}(t) = (1-\alpha_{ki})\phi^{k\to i}(t-1) + \left[\left(P_S^{k\to i}(t-1) - P_S^{k\to i}(t)\right) - \left(P_R^{k\to i}(t) - P_R^{k\to i}(t-1)\right)\right]. \quad [S4]$$

The initial conditions are

$$\theta^{i\to j}(0) = 1, \quad [S5]$$

$$\phi^{i\to j}(0) = \delta_{\sigma_i^0,I} = P_I^i(0) = 1 - P_S^i(0). \quad [S6]$$

The marginal probabilities for nodes to be in the states S or I at time t are computed via

$$P_S^i(t) = P_S^i(0)\left(\prod_{t'=0}^{t-1}(1-\nu_i(t'))\right)\left(\prod_{t'=0}^{t-1}(1-\mu_i(t'))\right)\prod_{k\in\partial i}\theta^{k\to i}(t), \quad [S7]$$

$$P_R^i(t) = P_R^i(t-1) + \mu_i(t-1)P_S^i(t-1), \quad [S8]$$

$$P_I^i(t) = 1 - P_S^i(t) - P_R^i(t). \quad [S9]$$

As explained in the main text, the optimization problem is stated in the form of a Lagrangian to be extremized:

$$\mathcal{L} = \underbrace{\mathcal{O}}_{\text{objective}} + \underbrace{\mathcal{B}+\mathcal{D}+\mathcal{I}+\mathcal{P}}_{\text{constraints}}, \quad [S10]$$

where $\mathcal{O}$ is the objective function one would like to maximize, and $\mathcal{B}$, $\mathcal{D}$, $\mathcal{I}$ and $\mathcal{P}$ correspond to the constraints representing budget, dynamics, initial conditions, and limits on the parameters, respectively. Below, we discuss in detail how to obtain an approximate solution to the optimal control problem using this framework, for the examples of the spread maximization or minimization under different constraints. In all examples, we assume that budget constraints for spontaneous infections or vaccination take a global form over a subset $W \subseteq V$ of nodes in the network and are specified at each time step:

$$\sum_{i\in W}\nu_i(t) = B_\nu(t), \quad [S11]$$

$$\sum_{i\in W}\mu_i(t) = B_\mu(t). \quad [S12]$$

Without loss of generality and for the sake of simplicity, in the equations below we assume that all nodes in the network are controllable, $W = V$; this case is the hardest in terms of the optimization procedure. Extending the derivation to the general case of any subset W is straightforward (an illustration for the case $W \neq V$ is given in *Tests on Real-World Networks and Comparisons with Popular Heuristics*).

Maximizing Information Spread Under Special Targeting Policy

In this section, we write the detailed form of the Lagrangian for the targeting problem, explained in the main text: We assume that for each node $i \in V$, the activation is required at a predefined time t_i. We derive the corresponding forward and backward equations

for the particular case of the modified SIR model without a vaccination transition [i.e., assuming $\mu_i(t) = 0$ for all $i \in V$ and $t \in [0, T-1]$].

Lagrangian Formulation. The Lagrangian in this case takes the following form:

$$\begin{aligned}
\mathcal{L} = &\underbrace{\sum_{i\in V}\Big(1 - P_S^i(t_i)\Big)}_{\mathcal{O}} + \underbrace{\sum_{t=0}^{T-1}\lambda_B^\nu(t)\left[\sum_{i\in V}\nu_i(t) - B_\nu(t)\right]}_{\mathcal{B}} + \underbrace{\epsilon\sum_{t=0}^{T-1}\sum_{i\in V}\Big(\log\left[\nu_i(t) - \underline{\nu_i^t}\right] + \log\left[\overline{\nu_i^t} - \nu_i(t)\right]\Big)}_{\mathcal{P}} \\
&\left.\begin{aligned}
&+ \sum_{i\in V}\sum_{t=0}^{T-1}\lambda_i^S(t+1)\left[P_S^i(t+1) - P_S^i(t)(1-\nu_i(t))\prod_{k\in\partial i}\frac{\theta^{k\to i}(t+1)}{\theta^{k\to i}(t)}\right] \\
&+ \sum_{(ki)\in E}\sum_{t=0}^{T-1}\lambda_{k\to i}^S(t+1)\left[P_S^{k\to i}(t+1) - P_S^{k\to i}(t)(1-\nu_k(t))\prod_{l\in\partial k\setminus i}\frac{\theta^{l\to k}(t+1)}{\theta^{l\to k}(t)}\right] \\
&+ \sum_{(ki)\in E}\sum_{t=0}^{T-1}\lambda_{k\to i}^\theta(t+1)\left[\theta^{k\to i}(t+1) - \theta^{k\to i}(t) + \alpha_{ki}\phi^{k\to i}(t)\right] \\
&+ \sum_{(ki)\in E}\sum_{t=0}^{T-1}\lambda_{k\to i}^\phi(t+1)\Big[\phi^{k\to i}(t+1) - (1-\alpha_{ki})\phi^{k\to i}(t) - P_S^{k\to i}(t) + P_S^{k\to i}(t+1)\Big]
\end{aligned}\right\}\mathcal{D} \\
&\left.\begin{aligned}
&+ \sum_{i\in V}\lambda_i^S(0)\left[P_S^i(0) - 1 + \delta_{\sigma_i^0,I}\right] + \sum_{(ki)\in E}\lambda_{k\to i}^S(0)\left[P_S^{k\to i}(0) - 1 + \delta_{\sigma_k^0,I}\right] \\
&+ \sum_{(ki)\in E}\lambda_{k\to i}^\theta(0)\left[\theta^{k\to i}(0) - 1\right] + \sum_{(ki)\in E}\lambda_{k\to i}^\phi(0)\left[\phi^{k\to i}(0) - \delta_{\sigma_k^0,I}\right].
\end{aligned}\right\}\mathcal{I}
\end{aligned}$$

In this expression, the dual variables $\lambda_i^S(t)$, $\lambda_{k\to i}^S(t)$, $\lambda_{k\to i}^\theta(t)$, and $\lambda_{k\to i}^\phi(t)$ in $\mathcal{D}$ and $\mathcal{I}$ enforce the dynamics given by the DMP equations as well as the initial conditions at time 0, while $\lambda_B^\nu(t)$ is the corresponding Lagrange multiplier for the budget constraint in $\mathcal{B}$. The parameter constraint $\mathcal{P}$ has a form of the logarithmic barrier function, often used in constrained optimization, forcing each of the parameters $\nu_i(t)$ to take values inside the interval $[\underline{\nu_i^t}, \overline{\nu_i^t}]$; for sufficiently small value of a positive coefficient ϵ, this regularization has a negligible impact on the objective function away from the extreme values.

Note that in the formulation above, we have made an implicit assumption that $\alpha_{ki} < 1$ for all $(ki) \in E$, so that the ratios $\frac{\theta^{k\to i}(t+1)}{\theta^{k\to i}(t)}$ are correctly defined. This assumption can be easily avoided by introducing other auxiliary variables to decouple $\theta^{k\to i}(t)$ at different times, which results in a slightly more complicated formulation.

Forward and Backward Equations. Once the form of the Lagrangian is established, we use a standard derivation with respect to primal and dual variables. The variation with respect to the multipliers $\lambda_i^S(t)$, $\lambda_{k\to i}^S(t)$, $\lambda_{k\to i}^\theta(t)$, and $\lambda_{k\to i}^\phi(t)$ give us back the direct DMP Eqs. **S1–S9**, while the variation with respect to $\lambda_B^\nu(t)$ yields the cost constraint **[S11]**. Setting the derivatives with respect to the primal variables to zero leads to the following set of dual equations:

$$\partial\mathcal{L}/\partial\phi^{k\to i}(t) = \Big[\alpha_{ki}\lambda_{k\to i}^\theta(t+1) - (1-\alpha_{ki})\lambda_{k\to i}^\phi(t+1)\Big]\mathbb{1}[t \neq T] + \lambda_{k\to i}^\phi(t) = 0, \qquad \textbf{[S13]}$$

$$\begin{aligned}
\partial\mathcal{L}/\partial P_S^{k\to i}(t) = &\lambda_{k\to i}^S(t) + \lambda_{k\to i}^\phi(t) \\
&- \left[\lambda_{k\to i}^S(t+1)(1-\nu_k(t))\prod_{l\in\partial k\setminus i}\frac{\theta^{l\to k}(t+1)}{\theta^{l\to k}(t)} + \lambda_{k\to i}^\phi(t+1)\right]\mathbb{1}[t \neq T] = 0,
\end{aligned} \qquad \textbf{[S14]}$$

$$\begin{aligned}
\partial\mathcal{L}/\partial\theta^{k\to i}(t) = &\lambda_{k\to i}^\theta(t) - \lambda_{k\to i}^\theta(t+1)\mathbb{1}[t \neq T] \\
&- \sum_{l\in\partial i\setminus k}\lambda_{i\to l}^S(t)P_S^{i\to l}(t-1)(1-\nu_i(t-1))\frac{1}{\theta^{k\to i}(t-1)}\prod_{m\in\partial i\setminus\{k,l\}}\frac{\theta^{m\to i}(t)}{\theta^{m\to i}(t-1)}\mathbb{1}[t \neq 0] \\
&+ \sum_{l\in\partial i\setminus k}\lambda_{i\to l}^S(t+1)P_S^{i\to l}(t)(1-\nu_i(t))\frac{1}{\theta^{k\to i}(t)}\prod_{m\in\partial i\setminus l}\frac{\theta^{m\to i}(t+1)}{\theta^{m\to i}(t)}\mathbb{1}[t \neq T] \\
&- \lambda_i^S(t)P_S^i(t-1)(1-\nu_i(t-1))\frac{1}{\theta^{k\to i}(t-1)}\prod_{m\in\partial i\setminus k}\frac{\theta^{m\to i}(t)}{\theta^{m\to i}(t-1)}\mathbb{1}[t \neq 0] \\
&+ \lambda_i^S(t+1)P_S^i(t)(1-\nu_i(t))\frac{1}{\theta^{k\to i}(t)}\prod_{m\in\partial i}\frac{\theta^{m\to i}(t+1)}{\theta^{m\to i}(t)}\mathbb{1}[t \neq T] = 0.
\end{aligned} \qquad \textbf{[S15]}$$

$$\partial\mathcal{L}/\partial P_S^i(t) = -\mathbb{1}[t = t_i] + \lambda_i^S(t) - \lambda_i^S(t+1)(1-\nu_i(t)) \prod_{k\in\partial i} \frac{\theta^{k\to i}(t+1)}{\theta^{k\to i}(t)} \mathbb{1}[t \neq T] = 0. \quad \textbf{[S16]}$$

Finally, the variation with respect to the parameters $\nu_i(t)$ for $t < T$ gives

$$\partial\mathcal{L}/\partial\nu_i(t) = \lambda_B^\nu(t) + \sum_{l\in\partial i} \lambda_{i\to l}^S(t+1) P_S^{i\to l}(t) \prod_{m\in\partial i\setminus l} \frac{\theta^{m\to i}(t+1)}{\theta^{m\to i}(t)} + \lambda_i^S(t+1) P_S^i(t) \prod_{m\in\partial i} \frac{\theta^{m\to i}(t+1)}{\theta^{m\to i}(t)}$$
$$+ \frac{\epsilon}{\left(\nu_i(t) - \underline{\nu_i^t}\right)} - \frac{\epsilon}{\left(\overline{\nu_i^t} - \nu_i(t)\right)} = 0. \quad \textbf{[S17]}$$

This set of forward and backward equations is sufficient for determining $\{\nu_i\}_{i\in V}$, by using the following optimization procedure, analogous to refs. 43 and 44. First, initialize $\nu_i(t)$ to some initial values, for example, using the uniform assignment $\nu_i(t) = 1/NT$ for all $i \in V$ and $t \in [0, T-1]$. Then, repeat the following steps for a fixed number of iterations or until convergence:

1. Starting from the initial values for the dynamics variables **[S5]** and **[S6]**, propagate the DMP Eqs. **S1–S9** forward, storing the values of dynamic messages at different times.
2. Use Eqs. **S13–S17** for fixing the boundary values of Lagrange multipliers at time T:

   - Eq. **S13** assigns $\lambda_{k\to i}^\phi(T) = 0$;
   - Eq. **S14** gives $\lambda_{k\to i}^S(T) = 0$;
   - Eq. **S16** gives $\lambda_i^S(T) = \delta_{t_i,T}$;
   - Eq. **S17** sets $\lambda_B^\nu(T-1)$ and $\nu_i(T-1)$ through $\lambda_i^S(T)$ and $\lambda_{k\to i}^S(T)$. Writing Eq. **S17** as

   $$\lambda_B^\nu(t) + \psi_i(t) + \frac{\epsilon}{\left(\nu_i(t) - \underline{\nu_i^t}\right)} - \frac{\epsilon}{\left(\overline{\nu_i^t} - \nu_i(t)\right)} = 0, \quad \textbf{[S18]}$$

   we first express $\nu_i(T-1)$ for each $i \in V$ as a function of $\lambda_B^\nu(T-1)$ as a solution of the second order equation. There are two a priori possible solutions:

   $$\nu_i^\pm(t) = \frac{(\lambda_B^\nu(t) + \psi_i(t))(\underline{\nu_i^t} + \overline{\nu_i^t}) - 2\epsilon \pm \sqrt{\left[(\lambda_B^\nu(t) + \psi_i(t))(\underline{\nu_i^t} + \overline{\nu_i^t})\right]^2 + 4\epsilon^2}}{2\lambda_B^\nu(t) + 2\psi_i(t)}. \quad \textbf{[S19]}$$

   Note that in the special case when control parameters can take all possible values between 0 and 1, meaning that $\underline{\nu_i^t} = 0$ and $\overline{\nu_i^t} = 1$ for all $i \in V$ and $t \in [0, T-1]$, the factor $\underline{\nu_i^t} + \overline{\nu_i^t} = 1$, which leads to simplified expressions. Assuming that ϵ is positive, we choose the solution with the positive sign in front of the square root, which always leads to $\underline{\nu_i^t} < \nu_i^+(t) < \overline{\nu_i^t}$, thus expressing $\{\nu_i(T-1)\}_{i\in V}$ as a function of $\lambda_B^\nu(T-1)$. We then determine $\lambda_B^\nu(T-1)$ [and hence $\{\nu_i(T-1)\}_{i\in V}$] from the numerical solution of the budget constraint Eqs. **S11** and **S15** gives $\lambda_{k\to i}^\theta(T)$ through $\lambda_i^S(T)$, $\lambda_{k\to i}^S(T)$, and $\nu_i(T-1)$.

3. Using the computed values for dynamic messages at different times, propagate the Eqs. **S13–S17** backward to compute the values of the Lagrange multipliers and parameters at all times (compute values at time t given values at times $t+1$):

   - Eq. **S13** gives $\lambda_{k\to i}^\phi(t)$;
   - Solve Eqs. **S14** and **S16** for evaluating $\lambda_{k\to i}^S(t)$ and $\lambda_i^S(t)$, correspondingly.
   - Compute $\lambda_B^\nu(t-1)$ and $\nu_i(t-1)$ using Eqs. **S17** and **S11** in the same way as it has been done for $t = T-1$; see Eq. **S19**;
   - Eq. **S15** gives $\lambda_{k\to i}^\theta(t)$ through $\lambda_{k\to i}^\theta(t+1)$, $\lambda_i^S(t+1)$, $\lambda_{k\to i}^S(t+1)$, $\lambda_i^S(t)$, $\lambda_{k\to i}^S(t)$, $\nu_i(t)$ and $\nu_i(t-1)$.

4. Using the updated values of $\nu_i(t)$ in the step 1.

At each iteration, this procedure leads to "jumps" between points of the objective $\mathcal{O}$ located on the manifold verifying the budget constraint [**S11**], where we keep track of the best solution which provides the final answer to the optimization problem after a maximum number of iterations is reached. Additionally, several different initializations for $\{\nu_i(t)\}_{i,t}$ and ϵ can be used in this scheme to achieve the best solution. Note that the initial distribution of $\{\nu_i(t)\}_{i,t}$ does not have to satisfy the budget condition [**S11**], but the solutions obey the constraint already after the first forward–backward iteration. In the case of a small number of nodes in the network N which corresponds to a simpler optimization landscape, we observed that the scheme quickly converges to a unique optimum starting from an arbitrary initial condition. In the next subsection, we describe tests of this scheme on a number of real-world networks.

Tests on Real-World Networks and Comparisons with Popular Heuristics. For validating the optimization scheme and its performance against existing approaches, we test the algorithm on a particular case of the targeting setting: the optimal seeding problem which corresponds to optimizing the initial conditions. Indeed, as explained in the main text, the initial conditions at time 1 can be defined as an optimization over $\{\nu_i(0)\}_{i\in V}$ at time $t = 0$, at which nodes will have a spontaneous probability of switching to the infected state. This leads to a more general space of possible initial conditions, without restrictions to a selection of an integer number of infected seeds. A classical setting corresponding to the selection of a group of "influential spreaders" at initial time $t = 1$ can be recovered in

our framework by imposing additional constraints to the domain of variation of $\{\nu_i(0)\}_{i\in V}$, restricting them to take only values close to zero or one.

Comparison with an explicit symbolic resolution of the DMP equations. For demonstration and validation purposes, we first consider the small network of relations between Slovene parliamentary parties in 1994, depicted in Fig. S2. Links in this network represent estimated similarity relations based on a sociological survey of the Parliament members who were asked to estimate the distance between each pair of parties in the political space (82). We have renormalized the weights of edges in such a way that the maximum pairwise mutual influence receives the value $\alpha_{ij}^{\max} = 0.5$, and the other weights are distributed proportionally to the survey data. We use this real-world network for the first tests in the seeding problem, which was defined as follows. We assumed that three nodes in this network belong to a controllable set $W = 1, 3, 5$, selected at random and corresponding to the parties SKD, ZS-ESS, and SDSS, correspondingly. Given the total campaigning budget $B_\nu(0) = 1.5$, the goal is to maximize the total informational influence at time $T = 3$ by optimally distributing the "lobbying" budget at initial time [i.e., finding the optimal distribution of the associated control parameters $\nu_1(0)$, $\nu_5(0)$, and $\nu_3(0) = 1.5 - \nu_1(0) - \nu_5(0)$].

As previously suggested, the forward–backward optimization scheme in this case converges to a unique optimal solution in about 7 iterations with $\epsilon = 5 \cdot 10^{-4}$ starting from an arbitrary initial values of the control parameters, outputting the values $\nu_1^*(0) = 0.993217$ and $\nu_5^*(0) = 0.505758$. One advantage of this small test case is that we can check of the validity of the scheme via an explicit symbolic resolution of DMP equations, obtaining a closed-form expression of the objective function $\mathcal{O} = \sum_{i\in V} P_I^i(T)$, representing the final spread at time T as a function of the independent control parameters $\nu_1(0)$ and $\nu_5(0)$:

$$\begin{aligned}\mathcal{O} = &-0.0213192\nu_5(0)^3\nu_1(0) - 0.0113728\nu_5(0)^4 - 0.00149759\nu_5(0)^3 + 0.125857\nu_5(0)^2\nu_1(0)\\
&+ 1.19019\nu_5(0)^2 + 1.05873\nu_5(0)\nu_1(0) - 1.5701\nu_5(0) - 0.00130489\nu_1(0)^4 + 0.0456274\nu_1(0)^3\\
&- 0.754842\nu_1(0)^2 + 1.74962\nu_1(0) - 0.00124248\nu_1(0)^3\nu_5(0)^2 - 0.00274613\nu_1(0)^2\nu_5(0)^3\\
&- 0.00470875\nu_1(0)^2\nu_5(0)^2 + 0.00523158\nu_1(0)^3\nu_5(0) + 0.121085\nu_1(0)^2\nu_5(0) - 0.0017293\nu_1(0)\nu_5(0)^4\\
&- 2.9969 \cdot 10^{-6}\nu_1(0)^4\nu_5(0)^2 + 7.3384 \cdot 10^{-5}\nu_1(0)^3\nu_5(0)^3 + 8.88018 \cdot 10^{-6}\nu_1(0)^4\nu_5(0)\\
&+ 0.000238133\nu_1(0)^2\nu_5(0)^4 + 0.000244127\nu_1(0)\nu_5(0)^5 - 0.000234529\nu_5(0)^5 + 8.23747 \cdot 10^{-5}\nu_5(0)^6\\
&+ 2.10571 \cdot 10^{-7}\nu_5(0)^3\nu_1(0)^4 + 8.42285 \cdot 10^{-7}\nu_5(0)^4\nu_1(0)^3 + 1.26343 \cdot 10^{-6}\nu_5(0)^5\nu_1(0)^2\\
&+ 8.42285 \cdot 10^{-7}\nu_5(0)^6\nu_1(0) + 2.10571 \cdot 10^{-7}\nu_5(0)^7 + 3.63985.\end{aligned}$$

The ground-truth optimal parameter values can be obtained by a direct maximization of $\mathcal{O}(\nu_1(0), \nu_5(0))$ plotted in Fig. S3. The optimal parameter values are given in this case by $\nu_1^*(0) = 1.0$ and $\nu_5^*(0) = 0.5$ (Fig. S2), in full agreement with the solution of the forward–backward iteration scheme up to insignificant domain border perturbations due to the finite value of ϵ, the regularization parameter that keeps values away from border values. This example validates our optimization procedure on this small-scale problem.

Comparison with heuristic centrality measures on large real-world networks. To test the efficacy of the DMP-optimization approach on large-scale instances, we compare its performance to that of popular heuristics in a particular setting of homogeneous transmission probabilities $\alpha_{ij} = \alpha$ for all edges (ij) and in the problem of spread maximization, for which many heuristic algorithms exist. Note that studying the minimization of the spread on networks is less interesting, because the optimal solution corresponds to activating nodes in clusters which are disconnected from the majority of nodes belonging to the giant component. Table 1 presents the normalized total spread for some of the best-performing centrality measures (17, 22, 25), the DMP algorithm and the special "Covering" algorithm for this setting after $T = 3$ time steps of the dynamics on different benchmark networks of various topologies and sizes. The transmission probabilities have been set to a uniform value $\alpha = 0.99$, and the total available seeding budget is equal to $B_\nu(0) = 0.05\ N$. The DMP-estimated marginals provide a natural and convenient measure for comparing the performance of different algorithms in the finite time horizon setting, especially on large graphs where running extensive Monte Carlo simulations is computationally prohibitive. Note that this measure is different for various dynamical models used. Before discussing the results of the comparison, let us briefly describe the algorithms and implementation details used for comparisons.

1. Naive strategies. As a reference, we present results obtained by a naive allocation of budget to a set of $K = \lceil B_\nu(0) \rceil$ (ceiling of $B_\nu(0)$) randomly selected nodes $\mathcal{R}$ (random), so that $\nu_i(0) = 1$ for $K - 1$ nodes i in $\mathcal{R}$, and $\nu_j(0) = \{B_\nu(0)\}$ for the last node j in the set, where $\{B_\nu(0)\}$ denotes the fractional part of $B_\nu(0)$. Another naive reference method that we consider is a uniform allocation of the budget in a probabilistic way (uniform), meaning that we set $\nu_i(0) = B_\nu(0)/N$ for all $i \in V$.
2. High degree adaptive (HDA). The algorithm is based on the idea that the best spreaders correspond to nodes with the highest degree (17), which is a valid and reasonable assumption for a large number of models, including the SI model considered here. The budget is iteratively attributed to $K = \lceil B_\nu(0) \rceil$ nodes with the current largest degree. Note the adaptive nature of the algorithm: Once the node is selected, it is effectively removed from the network, and the degrees of all nodes are recomputed accordingly; this strategy is much more efficient than the allocation of the budget to the high-degree nodes computed once and for all nodes in the original graph.
3. k-shell decomposition. In this algorithm, the influential spreaders are ranked according to their belonging to the k-core with maximum k. However, it has been noted in ref. 22 that, while this strategy is successful in identifying a single influential node, it performs badly when a group of nodes is selected, which is confirmed by our findings. We have considered both the original algorithm of ref. 22 and the adaptive version, where the cores of the graph are recomputed after each budget allocation to a particular node which is then removed from the graph. We found that overall, the results are very similar, with a slightly better performance of the adaptive version; therefore, we only present results for the latter.
4. Collective influence. A relation between the destruction of the giant component and "optimal percolation" has been suggested in ref. 25, where the authors have put forward another topological centrality measure called collective influence, defined as

$$\mathrm{CI}_l(i) = (d_i - 1) \sum_{j \in \partial B(i,l)} (d_j - 1), \qquad \textbf{[S20]}$$

where $\partial B(i,l)$ denotes the set of nodes at a distance l from node i. This topological characteristic results from mapping the spreading process asymptotically onto percolation (83) [in a particular instance of the linear threshold model (84)]. The intuition behind this measure is that the "collective influence" of node i is not only given by its degree, but also by the contribution of the degrees of nodes at a certain distance from it. However, on many graphs, the number of nodes in the set $\partial B(i,l)$ grows exponentially with l, which makes the computation of CI_l rather involved even for small values of l. Similarly to HDA, the algorithm is adaptive: After an allocation of resources to the node with the highest score, the measure is recomputed for all nodes. It is intuitive that in the case of a finite-time horizon objective, the best performance should be attained for $l \leq T$; indeed, we find that $l = T - 1 = 2$ realizes the best choice for the algorithm, with CI_4 leading to suboptimal results in this near-deterministic spreading case. A naive implementation of the algorithm, when the CI_l is recomputed for all vertices after the removal of the node with the highest score, leads to an algorithm with complexity $O(N^2)$ for $B_\nu(0) = O(N)$. Since we carried out comparisons on large networks with up to half a million nodes, we used a more efficient implementation of the algorithm which makes use of max-heap data structures in the spirit of ref. 85. This leads to a quasilinear algorithm in the number of nodes N, with complexity $O(N \log N)$ (multiplied by an important constant factor corresponding to the computations in the ball of radius l, which grows with T in the case $l \propto T$).

5. Benchmark: Covering algorithm. One advantage of the setting of a near-deterministic dynamics (when susceptible neighbors of each infected node with high probability become infected at the next time step) is that one can devise a natural algorithm with a near-optimal performance for this particular problem. For each vertex, we compute the associated score given by the size of its neighborhood up to distance T. Then, we proceed adaptively, adding the node with the highest score to the set of seeds, and removing the node and its whole neighborhood (in any case activated by time T with high probability) from the graph. In some sense, the method tries to optimally "cover" the graph with the balls of radius T. The algorithm does not always lead to the optimal solution; one counterexample is presented in Fig. S4. Nevertheless, it is intuitive that by design in most cases the Covering algorithm should lead to the best solution for the near-deterministic dynamics. Therefore, we consider the corresponding results as benchmark-optimal, which allows us to normalize all other algorithms (including DMP) to the values output by the Covering procedure. Similarly to collective influence, this algorithm should be implemented by using an appropriate data structure which facilitates storing and sorting of the scores. The computational complexity of the Covering algorithm is close to the one of collective influence, and scales as $O(N \log N)$ in the number of nodes in the graph N. All algorithms presented above (at the exception of naive strategies) should be completed with a tie-breaking rule in the case where several nodes show the same best current score at a given step of the adaptive resource allocation; in such cases, we break ties uniformly at random.

Results of comparison presented in the Table 1 show that the DMP algorithm is close to the best-performing heuristics in all cases, showing a consistently good performance. In principle, several different initializations for $\nu_i(t)$ can be used in the DMP algorithm to achieve the best solution; the results reported here correspond to the uniform starting values of the control parameters. A graphical summary of the Table 1 is given in Fig. 4, where we compare different methods considered in terms of algorithms' ranks and average "optimality gap" to the Covering algorithm, demonstrating the best performance in all cases except for the "Internet" network, where almost the entire graph is activated by time T, while Covering exhibits a nonoptimality similar to the example depicted in Fig. S4. Let us make a remark about the results of comparison of different algorithms on artificially-generated sparse Erdős–Rényi (ER) and scale-free (SF) graphs. Given a small value of $T = 3$ considered, the predictions of the DMP equations are exact on such large graphs. However, given that the optimization using the forward–backward swaps is agnostic to the accuracy of the DMP equations, there is no reason to expect that these networks would be more beneficial to our algorithm compared with the other techniques. This is indeed what happens in reality: Although the DMP method shows good results on the ER graph (as in the loopy cases of real graphs, where the optimal seeding set is not obvious), it is outperformed by other algorithms on the SF networks, where it is clear that the high-degree nodes should be targeted. Notice that our method does not rely explicitly on topological features such as targeting high-degree nodes, but instead explores a large space of parameters with impact on the full dynamic trajectory, still producing consistently good results in this comparison setting. This suggests that the DMP algorithm performs well also for more general dynamic resource allocation problems, for which other principled methods do not exist.

Additional comparisons in the case of heterogeneous couplings. Aiming at a better understanding of the effect of heterogeneity on the performance of the algorithms, we have run several tests in the case of a seeding problem with heterogeneous transmission probabilities. To carry out a fair comparison with the DMP method, we provide a natural generalization of the centrality-based measures presented in the previous section, except for the naive strategies that remain unchanged and serve as a benchmark. In particular, let us define an "effective degree" of node i as $\tilde{d}_i = \sum_{j \in i} \alpha_{ij}$. Then, in the HDA algorithm, we adaptively remove nodes with the highest current $\tilde{d}_i$. We modify the adaptive k-shell procedure in such a way that the nodes are greedily removed according to their current values of the effective degrees. Finally, we consider the following modification of the collective influence measure:

$$\mathrm{CI}_l(i) = \mathbb{1}[i \text{ not a leaf}]\, \tilde{d}_i \sum_{j \in \partial B(i,l)} \mathbb{1}[j \text{ not a leaf}]\, \tilde{d}_j,$$

where $\mathbb{1}[i \text{ not a leaf}]$ is an indicator function equal to one only when node i is not a leaf of the graph, which is an important feature of the original measure (25), and we use $\tilde{d}_i$ instead of $\tilde{d}_i - 1$ since the latter quantity can be negative. In our comparisons, we do not include the Covering algorithm since several generalizations that we tried have a very variable performance as a function of concrete graph and distribution of the couplings. At the same time, the generalizations for other algorithms make sense and should still lead to good results, especially for low T. Note, however, that such generalizations are not unique; a detailed study of other possible schemes in the heterogeneous case is beyond the scope of the present work.

Given the discussion on the performance of DMP on large networks with high-degree nodes in the previous section, we chose for our tests a distribution of heterogeneous couplings that is dependent on the topology of the underlying graph. In particular, we check the performance of the algorithms in the case where the symmetric transmission probabilities are chosen in such a way that they slightly reduce the role of the high-degree nodes: $\alpha_{ij} \propto 2d_i d_j/(d_i + d_j)$ (i.e., proportional to the ratio of the geometric and arithmetic mean). This ratio is always less than or equal to one, with equality realized only for the same-degree nodes, and hence favors the edges that connect nodes with equal degrees.

The corresponding numerical results are presented in the Table S1. We see that heterogeneity has a huge impact on the success of the algorithms: For instance, the relative performance of DMP is drastically improved on large networks (including the scale-free instance) where the resulting solution now represents a combination of high- and medium-degree nodes. On the other hand, DMP has a slightly worse performance on smaller-size networks in this case, at the same time outperforming other algorithms on some medium-size graphs by a very large margin (compare, for instance, the "web-sk" test case). These examples illustrate that it is crucial to take into account the heterogeneity of couplings when such information is available. The main message resulting from these results remains unchanged with respect to the conclusions reached in the previous section: The DMP algorithm shows consistently good results, and hence should be an appropriate optimization tool for more general problems of resource allocation in the dynamic fashion, as, for instance, illustrated in the next section. Finally, let us note that the performance of the DMP method for the seeding problem can be significantly improved by combining the scheme with other techniques: for example, initializing the algorithm at the solution output by one of the high-degree solutions or considering resource allocation on the restricted graph given by the k-core of the original network. However, our goal in the presented comparisons is to gain an intuition on the relative performance of the vanilla version of the DMP approach, aiming at previously unexplored applications to dynamic resource allocation problems.

Remarks on existing message-passing algorithms. Let us briefly mention the relation between the introduced DMP approach and existing message-passing algorithms (for the seeding problem). DMP equations can be derived from the belief propagation (BP) method applied to the time trajectories of the nodes; see the corresponding derivation in ref. 42, which also proves that the DMP equations are exact on tree graphs (since BP is exact on graphs without loops). For unconstrained problems (i.e., for a simple inference task of computing marginal probabilities), the two formulations give exactly the same results. However, in unconstrained problems, DMP equations are easier to implement and do not require any iterations until convergence (a usual method for solving BP fixed-point equations). One advantage of the BP formulation is that in the optimization problems (constrained by the cost function or the data), the constraints can be directly incorporated into the problem, and the BP algorithm then runs on factor-graph, which differs from the original network topology; however, at the risk of algorithmic nonconvergence for problems with a large density of constraints.

The BP-based formulations (of a slightly different type) have been previously used for solving the seeding problem (in the infinite time setting), leading to the max-sum algorithm for the threshold (29) and epidemic-type spreading (28) models. Besides the fact that the DMP-based approach presented here has a different scope and applies to a variety of problems in a dynamic setting, this framework has several other attractive features. For example, the budget of resources $B_\nu(0)$ to be allocated can take an arbitrary value, in particular, both $O(1)$ and $O(N)$ regimes are possible. The BP algorithm for the spreading models operates only in the latter regime, and the complexity diverges as $O(b^{-3})$ when $b = B_\nu(0)/N$ (the budget density) tends to zero because of the need for the discretization of the BP equations. Since the max-sum algorithm is based on the percolation-like equations for the epidemic spreading models (which can be derived as the $T \to \infty$ limit of the corresponding DMP equations), it does not directly apply to the considered finite-time setting. As noted in *Discussion*, it would be interesting to compare this approach with the forward–backward optimization algorithm applied to percolation equations. This comparison is left for future work.

Offline and Online Mitigation of Epidemic via Optimal Vaccination

Here, we consider the problem of spread minimization, for example, in the case of an undesired spreading process such as the propagation of epidemic. The minimization of the spread at time T is achieved by an effective distribution of vaccines to vulnerable susceptible nodes in a dynamical fashion. We use the variant of the modified SIR model with a vaccination transition: The optimization parameters in this case are given by $\{\mu_i(t)\}_{i\in V}$ for $t \in [0, T-1]$, and we assume that the spontaneous transition to the I state is absent, meaning that $\nu_i(t) = 0$ for all i and t. Modifications with respect to the case considered previously in *Maximizing Information Spread Under Special Targeting Policy* include:

1. We are interested in maximizing the objective $\sum_{i\in V}\left(1 - P_I^i(T)\right) = \sum_{i\in V}\left(P_S^i(T) + P_R^i(T)\right)$ at this time, so the objective has a different sign with respect to the previous case of the spread maximization;
2. Since now nodes can transit to the R state, we additionally need to keep track of the evolution of the quantities $P_R^i(t)$ and $P_R^{i\to j}(t)$;
3. Similarly to the formulation in the spread maximization, it is possible to include additional information on a list of target nodes which require priority protection, thus forcing the optimization to minimize the probability of infection on "key" nodes first;
4. In the setting of "online" mitigation of the spreading process, one should reinitialize the optimization process at each time step once the new state of the network is available, getting a more accurate and updated account of the state of the system. One then solves the optimization problem of spread minimization at the horizon $t = T$ for the allocation of resources at the next time step only (seeding problem) using new data as initial conditions for the DMP equations. Running optimization for a few more time steps results in a vaccine distribution plan with forecasted estimates, which might be an important feature in realistic settings.

Lagrangian and Forward–Backward Equations. The corresponding Lagrangian in the case of spreading minimization under vaccination constraints takes the following form:

$$
\begin{aligned}
\mathcal{L} = & \underbrace{\sum_{i\in V}\Big(P_S^i(T)+P_R^i(T)\Big)}_{\mathcal{O}} + \underbrace{\sum_{t=0}^{T-1}\lambda_B^{\mu}(t)\left[\sum_{i\in V}\mu_i(t)-B_\mu(t)\right]}_{\mathcal{B}} + \underbrace{\epsilon\sum_{t=0}^{T-1}\sum_{i\in V}\Big(\log\left[\mu_i(t)-\underline{\mu_i^t}\right]+\log\left[\overline{\mu_i^t}-\mu_i(t)\right]\Big)}_{\mathcal{P}} \\
& + \sum_{i\in V}\sum_{t=0}^{T-1}\lambda_i^S(t+1)\left[P_S^i(t+1)-P_S^i(t)(1-\mu_i(t))\prod_{k\in\partial i}\frac{\theta^{k\to i}(t+1)}{\theta^{k\to i}(t)}\right] \\
& + \sum_{i\in V}\sum_{t=0}^{T-1}\lambda_i^R(t+1)\left[P_R^i(t+1)-P_R^i(t)-\mu_i(t)P_S^i(t)\right] \\
& + \sum_{(ki)\in E}\sum_{t=0}^{T-1}\lambda_{k\to i}^S(t+1)\left[P_S^{k\to i}(t+1)-P_S^{k\to i}(t)(1-\mu_k(t))\prod_{l\in\partial k\setminus i}\frac{\theta^{l\to k}(t+1)}{\theta^{l\to k}(t)}\right] \\
& + \sum_{(ki)\in E}\sum_{t=0}^{T-1}\lambda_{k\to i}^R(t+1)\left[P_R^{k\to i}(t+1)-P_R^{k\to i}(t)-\mu_k(t)P_S^{k\to i}(t)\right] \\
& + \sum_{(ki)\in E}\sum_{t=0}^{T-1}\lambda_{k\to i}^{\theta}(t+1)\left[\theta^{k\to i}(t+1)-\theta^{k\to i}(t)+\alpha_{ki}\phi^{k\to i}(t)\right] \\
& + \sum_{(ki)\in E}\sum_{t=0}^{T-1}\lambda_{k\to i}^{\phi}(t+1)\Big[\phi^{k\to i}(t+1)-(1-\alpha_{ki})\phi^{k\to i}(t)-P_S^{k\to i}(t)+P_S^{k\to i}(t+1)-P_R^{k\to i}(t)+P_R^{k\to i}(t+1)\Big] \Big\}\mathcal{D} \\
& + \sum_{i\in V}\lambda_i^S(0)\left[P_S^i(0)-\delta_{\sigma_i^0,S}\right]+\sum_{i\in V}\lambda_i^R(0)\left[P_R^i(0)-\delta_{\sigma_i^0,R}\right]+\sum_{(ki)\in E}\lambda_{k\to i}^S(0)\left[P_S^{k\to i}(0)-\delta_{\sigma_k^0,S}\right] \\
& + \sum_{(ki)\in E}\lambda_{k\to i}^R(0)\left[P_R^{k\to i}(0)-\delta_{\sigma_k^0,R}\right]+\sum_{(ki)\in E}\lambda_{k\to i}^{\theta}(0)\left[\theta^{k\to i}(0)-1\right]+\sum_{(ki)\in E}\lambda_{k\to i}^{\phi}(0)\left[\phi^{k\to i}(0)-\delta_{\sigma_k^0,I}\right]. \Big\}\mathcal{I}
\end{aligned}
$$

The variation of the Lagrangian with respect to the dual variables yield the forward DMP Eqs. **S1**–**S9**. The derivation of the backward equations follow the same principles as in the case of the SI model, with obvious modifications: change of the control parameters $\nu_i(t)\to\mu_i(t)$, additional equations

$$\partial\mathcal{L}/\partial P_R^i(t) = \mathbb{1}[t=T]+\lambda_i^R(t)-\lambda_i^R(t+1)\mathbb{1}[t\neq T]=0, \qquad \textbf{[S21]}$$

$$\partial\mathcal{L}/\partial P_R^{k\to i}(t) = \lambda_{k\to i}^R(t)-\lambda_{k\to i}^R(t+1)\mathbb{1}[t\neq T]+\lambda_{k\to i}^{\phi}(t)-\lambda_{k\to i}^{\phi}(t+1)\mathbb{1}[t\neq T]=0, \qquad \textbf{[S22]}$$

and additions to the previous backward equations due to the new terms

$$\Delta\partial\mathcal{L}/\partial P_S^i(t) = -\lambda_i^R(t+1)\mu_i(t)\mathbb{1}[t\neq T], \qquad \textbf{[S23]}$$

$$\Delta\partial\mathcal{L}/\partial P_S^{k\to i}(t) = -\lambda_{k\to i}^R(t+1)\mu_k(t)\mathbb{1}[t\neq T], \qquad \textbf{[S24]}$$

$$\Delta\partial\mathcal{L}/\partial\mu_i(t) = -\lambda_i^R(t+1)P_S^i(t)-\sum_{k\in\partial i}\lambda_{i\to k}^R(t+1)P_S^{i\to k}(t). \qquad \textbf{[S25]}$$

Description of the Dataset Used for Tests. As a test example for the vaccination problem, we construct a transportation network of busiest flight routes between major US airports, extracted from the publicly available BTS data (74). We use the data table providing the number of transported passengers by different companies between US airports over the past several years. First, we extract the subtable of flights between 61 biggest airports in terms of the total number of emplaned passengers per year, including 30 "major" and 31 "largest" hubs according to the BTS classification. Then, multiple entries corresponding to the same route are aggregated, and the routes carrying $<10\%$ of the passengers transported by the busiest route are pruned as less significant ones, primarily for rending the network reasonably sparse for a clear visualization. This results in a network with $M=|E|=383$ edges. Finally, we assign the spreading couplings $\alpha_{ij\,(ij)\in E}$ proportionally to the number of carried passengers in such a way that the lightest route has the value $\alpha_{ij}^{\min}=0.05$, and hence the busiest route receives the value $\alpha_{ij}^{\max}\simeq 0.495$. This choice is based on a reasonable assumption that the probability of infection transmission along each link is proportional to the number of carried passengers on this route.

Case of Continuous Dynamics

The optimization procedure described in this work can be directly applied to the case of continuous dynamics, which might be more relevant in other applications. In the continuous case, the backward equations are obtained through the variation of the Lagrangian, resulting in the continuous Euler–Lagrange equations. In this section, we illustrate the approach in the setting of maximizing the spread in the SI model using the continuous version of the DMP equations.

Maximization of Spread at the Time Horizon *T*. In the continuous case, the marginal probability for node i to be in the state S at time t reads

$$P_S^i(t) = P_S^i(0)\exp\left\{-\int_0^t dt'\nu_i(t')\right\}\prod_{k\in\partial i}\theta^{k\to i}(t). \qquad \textbf{[S26]}$$

In the case of constant rates α_{ij}, we define the transmission function as $f_{ij}(t) = \alpha_{ij}\, e^{-\alpha_{ij} t}$. Then, the functions $\theta^{i\to j}(t)$ are computed as follows (40):

$$\begin{aligned}\theta^{i\to j}(t) &= 1 - \int_0^t d\tau\, f_{ij}(\tau) \left[1 - P_S^i(0) \exp\left\{-\int_0^{t-\tau} dt' \nu_i(t')\right\} \prod_{k\in\partial i\setminus j} \theta^{k\to i}(t-\tau)\right] \\ &= e^{-\alpha_{ij} t} + P_S^i(0)\alpha_{ij}\, e^{-\alpha_{ij} t} \int_0^t d\tau\; e^{\alpha_{ij}\tau} \exp\left\{-\int_0^{\tau} dt' \nu_i(t')\right\} \prod_{k\in\partial i\setminus j} \theta^{k\to i}(\tau). \end{aligned} \quad \textbf{[S27]}$$

To compute the dynamic messages $\theta^{i\to j}(t)$, we can either integrate the Eq. **S27** numerically, or transform it into an ordinary differential equation by differentiating with respect to t:

$$\dot\theta^{i\to j}(t) = -\alpha_{ij}\theta^{i\to j}(t) + \alpha_{ij} P_S^i(0) \exp\left\{-\int_0^t dt' \nu_i(t')\right\} \prod_{k\in\partial i\setminus j} \theta^{k\to i}(t), \quad \textbf{[S28]}$$

which represents the dynamics of $\theta^{i\to j}(t)$ (the dot notation represents $\frac{d}{dt}$) and can be solved numerically starting from initial conditions $\theta^{i\to j}(0) = 1$. To derive the dynamics of $P_S^i(t)$, one can differentiate Eq. **S26** to obtain:

$$\dot P_S^i(t) = P_S^i(0) e^{-\nu_i(t)t} \prod_{k\in\partial i} \theta^{k\to i}(t) + P_S^i(t) \sum_{k\in\partial i} \frac{\dot\theta^{k\to i}(t)}{\theta^{k\to i}(t)}. \quad \textbf{[S29]}$$

Maximizing $\mathcal{O} = \sum_{i\in V} P_I^i(T)$ is equivalent to minimizing

$$\hat{\mathcal{O}}(T) = \sum_{i\in V} P_S^i(T) = \int_0^T \sum_{i\in V} \dot P_S^i(t) dt \quad \textbf{[S30]}$$

under the constraints imposed by Eqs. **S28** and **S29** and the budget constraint **[S11]**

$$\sum_{i\in V} \nu_i(t) = B_\nu(t). \quad \textbf{[S31]}$$

All constraints will be imposed by the corresponding Lagrange multipliers. The Lagrangian to be extremized takes the form

$$\begin{aligned}\mathcal{L} = \int_0^T \Bigg\{ &\sum_{i\in V} \dot P_S^i(t) + \lambda_B^\nu(t) \left[\sum_{i\in V} \nu_i(t) - B_\nu(t)\right] \\ &+ \sum_{(i,j)\in E} \lambda_{i\to j}^\theta(t) \left[\dot\theta^{i\to j}(t) + \alpha_{ij}\theta^{i\to j}(t) - \alpha_{ij} P_S^i(0) \exp\left\{-\int_0^t dt' \nu_i(t')\right\} \prod_{k\in\partial i\setminus j} \theta^{k\to i}(t)\right] \\ &+ \sum_{i\in V} \lambda_i^S(t) \left[\dot P_S^i(t) - P_S^i(0) e^{-\nu_i(t)t} \prod_{k\in\partial i} \theta^{k\to i}(t) - P_S^i(t) \sum_{k\in\partial i} \frac{\dot\theta^{k\to i}(t)}{\theta^{k\to i}(t)}\right]\Bigg\} dt \\ = \int_0^T \Bigg\{ &\sum_{i\in V} \dot P_S^i(t) + \lambda_B^\nu(t) \left[\sum_{i\in V} \nu_i(t) - B_\nu(t)\right] \\ &+ \sum_{(i,j)\in E} \lambda_{i\to j}^\theta(t) \left[\dot\theta^{i\to j}(t) + \alpha_{ij}\theta^{i\to j}(t) - \alpha_{ij} \frac{P_S^i(t)}{\theta^{j\to i}(t)}\right] \\ &+ \sum_{i\in V} \lambda_i^S(t) \left[\dot P_S^i(t) - P_S^i(0) e^{-\nu_i(t)t} \prod_{k\in\partial i} \theta^{k\to i}(t) - P_S^i(t) \sum_{k\in\partial i} \frac{\dot\theta^{k\to i}(t)}{\theta^{k\to i}(t)}\right]\Bigg\} dt \end{aligned} \quad \textbf{[S32]}$$

Variational maximization of a Lagrangian

$$\mathcal{L} = \int_0^T F(t, y, \dot y)\; dt$$

where $F(t, y, \dot y)$ is a function of some variable t, y, and $\dot y$ its derivative with respect to t, is carried out by varying y with respect to a small perturbation $\epsilon \to 0$, throughout the t interval (86), to $y + \epsilon\delta y$. Extremizing $\mathcal{L}$ by setting $\frac{d\mathcal{L}}{d\epsilon} = 0$ and using integration by parts one obtains the optimization condition

$$\left.\frac{d\mathcal{L}}{d\epsilon}\right|_{\epsilon=0} = \int_0^T \left[\frac{\partial F}{\partial y} - \frac{d}{dt}\frac{\partial F}{\partial \dot y}\right] \delta y\; dt + \left[\frac{\partial F}{\partial \dot y}\right]_0^T = 0.$$

Both the Euler–Lagrange equation $\left[\frac{\partial F}{\partial y} - \frac{d}{dt}\frac{\partial F}{\partial \dot y}\right]$ and the boundary conditions $\left[\frac{\partial F}{\partial \dot y}\right]_0^T$ should be zero.

Varying Eq. **S32** with respect to $\delta P_S^i(t)$ and $\delta\theta^{i\to j}(t)$ results in the following differential equations:

$$\dot\lambda_i^S(t) = -\sum_{j\in\partial i}\frac{\lambda_{i\to j}^\theta\alpha_{ij}}{\theta^{j\to i}(t)} - \lambda_i^S(t)\sum_{k\in\partial i}\frac{\dot\theta^{k\to i}(t)}{\theta^{k\to i}(t)} - \delta_{t,0}\lambda_i^S(0)\prod_{k\in\partial i}\theta^{k\to i}(0), \quad \textbf{[S33]}$$

$$\begin{aligned}\dot\lambda_{i\to j}^\theta(t) &= \lambda_{i\to j}^\theta(t)\alpha_{ij} + \lambda_{j\to i}^\theta(t)\alpha_{ji}\frac{P_S^j(t)}{[\theta^{i\to j}(t)]^2} \\ &\quad - \lambda_j^S(t)\left\{P_S^j(0)e^{-\nu_j(t)t}\prod_{k\in\partial j/i}\theta^{k\to j}(t) - \frac{\dot P_S^j(t)}{\theta^{i\to j}(t)}\right\} + \dot\lambda_j^S(t)\frac{P_S^j(t)}{\theta^{i\to j}(t)} \\ &= \lambda_{i\to j}^\theta(t)\alpha_{ij} - \sum_{k\in\partial j/i}\frac{\lambda_{j\to k}^\theta(t)\alpha_{jk}P_S^j(t)}{\theta^{k\to j}(t)\theta^{i\to j}(t)} - \delta_{t,0}\lambda_j^S(0)P_S^j(t)\prod_{k\in\partial j/i}\theta^{k\to i}(0),\end{aligned} \quad \textbf{[S34]}$$

where the expression for $\dot\theta^{k\to i}(t)$ in both Eqs. **S33** and **S34** is calculated on the basis of Eq. **S28** and the expressions for $\dot\lambda_j^S(t)$ and $\dot P_S^j(t)$ in Eq. **S34)** are taken from Eqs. **S33** and **S29**, respectively, and are used in the simplifications of Eq. **S34**. These equations will be integrated back on the basis of endpoints.

Varying Eq. **S32** with respect to $\delta\nu_i(t)$ one obtains:

$$\lambda_B^\nu(t) = -t\lambda_i^S(t)P_S^i(0)e^{-\nu_i(t)t}\prod_{k\in\partial i}\theta^{k\to i}(t), \quad \textbf{[S35]}$$

which can be rewritten as

$$\nu_i(t) = -\frac{1}{t}\ln\left[-\frac{\lambda_B^\nu(t)}{t\lambda_i^S(t)P_S^i(0)\prod_{k\in\partial i}\theta^{k\to i}(t)}\right]. \quad \textbf{[S36]}$$

Using the condition $\sum_{i\in V}\nu_i(t) = B_\nu(t)$, we obtain

$$B_\nu(t) = -\frac{N}{t}\ln\lambda_B^\nu(t) + \frac{1}{t}\sum_{i\in V}\ln\left[-t\lambda_i^S(t)P_S^i(0)\prod_{k\in\partial i}\theta^{k\to i}(t)\right]. \quad \textbf{[S37]}$$

That leads to a straightforward solution for $\lambda_B^\nu(t)$

$$\lambda_B^\nu(t) = \exp\left\{\frac{1}{N}\sum_{i\in V}\ln\left[-t\lambda_i^S(t)P_S^i(0)\prod_{k\in\partial i}\theta^{k\to i}(t)\right] - \frac{t}{N}B_\nu(t)\right\}, \quad \textbf{[S38]}$$

from which the $\nu_i(t)$ values can be calculated.

Remark. An alternative path to the derivation of Eqs. **S36** and **S38** consists in directly using the normalized representation.

$$\nu_i(t) = \frac{e^{-\beta_i(t)}}{\sum_{j\in V}e^{-\beta_j(t)}}B_\nu(t)\,.$$

instead of enforcing the budget constraint with the Lagrange multiplier $\lambda_B^\nu(t)$ in Eq. **S32**. For the optimization with respect to $\beta_i(t)$, in this case it is then possible to use

$$\frac{\partial}{\partial\beta_k} = -\sum_{j\in V}\left[\delta_{jk}\nu_k(t) - \nu_k(t)\nu_j(t)\right]\frac{\partial}{\partial\nu_j}.$$

Optimization with respect to $\beta_i(t)$ gives

$$\nu_i(t) = \frac{\lambda_i^S(t)P_S^i(0)e^{-\nu_i(t)t}\prod_{k\in\partial i}\theta^{k\to i}(t)}{\sum_{j\in V}\lambda_j^S(t)P_S^j(0)e^{-\nu_j(t)t}\prod_{k\in\partial j}\theta^{k\to j}(t)}B_\nu(t), \quad \textbf{[S39]}$$

which is equivalent to Eqs. **S36** and **S38**. ■

Additionally, one should enforce the boundary conditions in the two sets of equations:

$$0 = \delta P_S^i(T)\left[1+\lambda_i^S(T)\right] - \delta P_S^i(0)\left[1+\lambda_i^S(0)\right] \quad \textbf{[S40]}$$

$$0 = \delta\theta^{i\to j}(T)\left[\lambda_{i\to j}^\theta(T) - \frac{P_S^j(T)\lambda_j^S(T)}{\theta^{j\to i}(T)}\right] - \delta\theta^{i\to j}(0)\left[\lambda_{i\to j}^\theta(0) - \frac{P_S^j(0)\lambda_j^S(0)}{\theta^{j\to i}(0)}\right]. \quad \textbf{[S41]}$$

Since $\delta P_S^i(0) = \delta\theta^{i\to j}(0) = 0\ \forall i,j\in V$ it provides the end conditions:

$$\lambda_i^S(T) = -1 \quad \textbf{[S42]}$$

$$\lambda_{i\to j}^\theta(T) = \frac{P_S^j(T)}{\theta^{j\to i}(T)}. \quad \textbf{[S43]}$$

Note that $\lambda_i^S(T) = -1$ is a result of the minimization of $\hat{\mathcal{O}}(T)$; a maximization of $\mathcal{O}$ would provide a boundary condition of $\lambda_i^S(T) = 1$.

The optimization process should be carried out as follows:

1. Using the initial conditions $\theta^{i\to j}(0) = 1\ \forall i$ and $P_S^i(0)$ according to the case studied, and some valid initial set of $\nu_i(t)$ (can be uniform), one can solve forward the Eqs. **S28** and **S29** to obtain (and register) values throughout the dynamics $t = 0 \to T$.
2. Using the boundary conditions [**S44**] and [**S43**] one solves backward [**S33**] and [**S34**] and updates the values of $\nu_i(t)$ according to Eqs. **S38** and **S36**.
3. The process is repeated until it converges and the final $\nu_i(t)$ values represent the solution.

Continuous Dynamics with Targeted and Accessible Nodes. In this variant of the problem, one targets specific nodes $i \in U$ where $U \subseteq V$ is the subset of all nodes V, aiming to maximize the impact at predefined times t_i, which may be different for each of the nodes. We will define $T \equiv \max_{i\in U} t_i$. We also assume one has access to a subset on the nodes $i \in W$ where $W \subseteq V$ and $W \cap U = \emptyset$. Again, maximizing $\mathcal{O}$ is equivalent to minimizing

$$\hat{\mathcal{O}} = \sum_{i\in U} P_S^i(t_i) = \sum_{i\in U} \int_0^{t_i} \dot{P}_S^i(t)\ dt. \qquad \textbf{[S44]}$$

Since the budget for nodes $i \notin W$ is zero by definition $\nu_i(t) = 0, \forall i \notin W, \forall t$, and the cost constraint corresponding to Eq. **S11** becomes

$$\sum_{i\in W} \nu_i(t) = B_\nu(t), \qquad \textbf{[S45]}$$

where we can use the normalized representation where $\beta_i(t) = -\infty, \forall i \notin W, \forall t$,

$$\nu_i(t) = \frac{e^{-\beta_i(t)}}{\sum_{j\in V} e^{-\beta_j(t)}} B_\nu(t).$$

The Lagrangian to be extremized takes the form (the budget constraint on accessible nodes [**S45**] is embedded in the corresponding ν variables through β, as remarked in *Maximization of Spread at the Time Horizon* T

$$\begin{aligned}
\mathcal{L} = &\sum_{i\in U} \int_0^{t_i} \dot{P}_S^i(t)\ dt \\
&+ \int_0^T \left\{ \sum_{(i,j)\in E} \lambda_{i\to j}^\theta(t) \left[\dot{\theta}^{i\to j}(t) + \alpha_{ij}\theta^{i\to j}(t) - \alpha_{ij} P_S^i(0) \exp\left\{ -\int_0^t dt' \nu_i(t') \right\} \prod_{k\in\partial i\setminus j} \theta^{k\to i}(t) \right] \right. \\
&\left. + \sum_{i\in V} \lambda_i^S(t) \left[\dot{P}_S^i(t) - P_S^i(0) e^{-\nu_i(t)t} \prod_{k\in\partial i} \theta^{k\to i}(t) - P_S^i(t) \sum_{k\in\partial i} \frac{\dot{\theta}^{k\to i}(t)}{\theta^{k\to i}(t)} \right] \right\} dt \\
= &\sum_{i\in U} \int_0^{t_i} \dot{P}_S^i(t)\ dt \\
&+ \int_0^T \left\{ \sum_{(i,j)\in E} \lambda_{i\to j}^\theta(t) \left[\dot{\theta}^{i\to j}(t) + \alpha_{ij}\theta^{i\to j}(t) - \alpha_{ij} \frac{P_S^i(t)}{\theta^{j\to i}(t)} \right] \right. \\
&\left. + \sum_{i\in V} \lambda_i^S(t) \left[\dot{P}_S^i(t) - P_S^i(0) e^{-\nu_i(t)t} \prod_{k\in\partial i} \theta^{k\to i}(t) - P_S^i(t) \sum_{k\in\partial i} \frac{\dot{\theta}^{k\to i}(t)}{\theta^{k\to i}(t)} \right] \right\} dt
\end{aligned} \qquad \textbf{[S46]}$$

Alternatively, the first term can be written as

$$\int_0^T \sum_{i\in U} \dot{P}_S^i(t)\Theta(t_i - t)\ dt.$$

Since the first term does not contribute to the Euler–Lagrange equation, varying Eq. **S47** with respect to $\delta P_S^i(t)$, $\delta\theta^{i\to j}(t)$ and $\delta\nu_i(t)$ results in the same differential equations for $\dot{\lambda}_i^S(t)$ [**S33**] and $\dot{\lambda}_{ij}(t)$ [**S34**].

Optimization with respect to $\beta_i(t)$ or $\nu_i(t)$ gives a similar expression for $i \in W$ [**S39**], but $\nu_i(t) = 0, \forall i \notin W$ and $\forall t$. Additionally, one should enforce the boundary conditions in the two sets of Eq. **S40** and **S42**. For $j \in U$, [**S40**] and [**S42**] are the same as in the nontargeted case, but for $j \notin U$, there is no constant (of value 1) in Eq. **S40**, leading to:

$$\lambda_i^S(T) = \begin{cases} -1 & \forall\ i \notin U \\ 0 & \forall\ i \notin U \end{cases} \qquad \textbf{[S47]}$$

$$\lambda_{i\to j}^\theta(t_j) = \begin{cases} 0 & \forall\ i \notin U \\ \dfrac{P_S^j(t_j)}{\theta^{j\to i}(t_j)} & \forall\ j \in U \end{cases}. \qquad \textbf{[S48]}$$

The optimization process should be carried out as before.

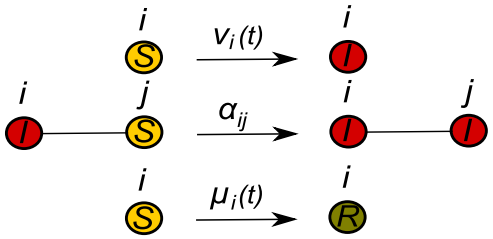

Fig. S1. Transition diagram summarizing dynamic rules in the modified SIR model considered in this work. In the case of a simultaneous transition of a susceptible node (marked in yellow) to the states *I* (infected; red) and *R* (recovered; green) at time *t*, it assumes a recovered state at the next time step.

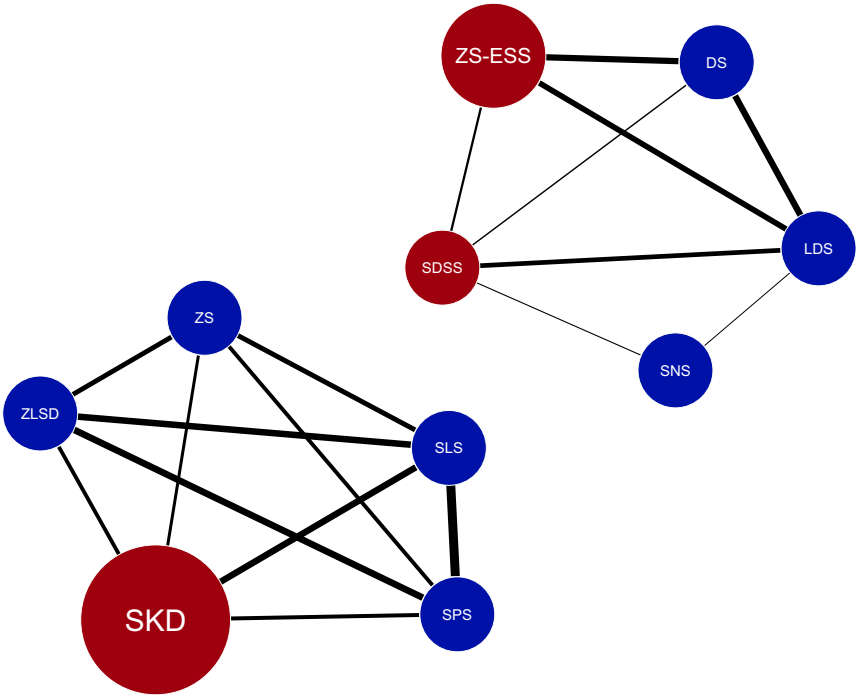

Fig. S2. A small network of relations between Slovene parliamentary political parties from the social study of ref. 82. The edge thickness is proportional to the estimated mutual influence α_{ij} between a pair of parties. Only red nodes are assumed to be controllable in this network. The sizes of the nodes reflect the optimal distribution of the seeding budget $B_\nu(0) = 1.5$ for maximizing the global impact: Small-size nodes do not require any control, and the spontaneous activation probabilities associated with the medium- and large-size nodes are equal to 0.5 and 1.0, respectively.

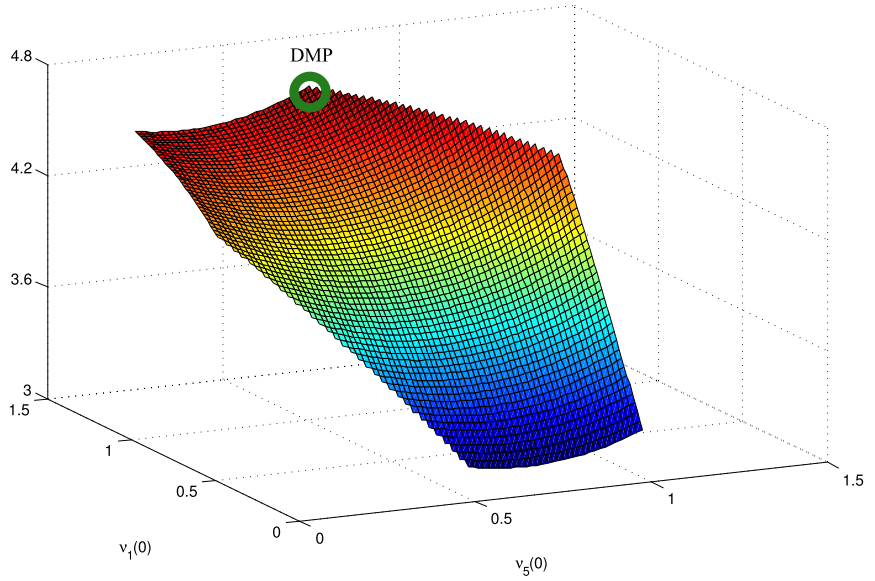

Fig. S3. Validation of the DMP algorithm on a small network of relations between Slovene parliamentary parties with an explicit evaluation of the objective function. Edge weights have been renormalized such that the maximum pairwise mutual influence receives the value $\alpha_{ij}^{\max} = 0.5$, and other weights are distributed proportionally to the survey data. We assume that only three nodes in this network belong to a controllable set $W = \{1, 3, 5\}$, and the total campaigning budget $B_\nu(0) = 1.5$ so that $\nu_3(0) = 1.5 - \nu_1(0) - \nu_5(0)$. The plotted surface represents the total informational influence $\mathcal{O}(\nu_1(0), \nu_5(0))$ as a function of two independent seeding control parameters $\nu_1(0)$ and $\nu_5(0)$, computed via a symbolic solution of the DMP equations. Color variations reflect the change of the value of the objective, changing gradually from the minimum (blue) to the maximum (red). The DMP algorithm correctly recovers the parameter values that maximize the total spread at the final time (the corresponding solution is marked by a green circle).

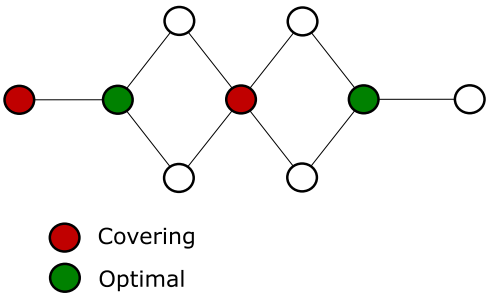

Fig. S4. A counterexample demonstrating the nonoptimality of the Covering algorithm. Consider the spread maximization problem on the depicted graph, with the budget $B_\nu(0) = 2$ and the time-horizon $T = 1$. The central node will be chosen by the Covering algorithm as a first seed, because this choice leads to the highest number of activated nodes by time $T = 1$. Removal of the entire central neighborhood would break the network in two equal pieces, each containing a pair nodes. Therefore, the choice of seeds according to the Covering strategy (one possibility is highlighted in red) would lead to a suboptimal spread ($\mathcal{S} = 7$) compared with the optimal choice of seeds in this case, shown in green ($\mathcal{S} = 8$). This suboptimality effect of the Covering method may occur on "finite-size" networks with diameter of order T, or for high values of the budget $B_\nu(0)$.

Table S1. Comparison of the DMP algorithm for the seeding problem in the setting of heterogeneous couplings with popular heuristics on various real-world and artificial networks

Network	*N*	*M*	Random	HDA	k-shell	CI_2	CI_4	Uniform	DMP
Road EU	1,174	1,417	0.307	0.473	0.304	0.495	0.448	0.310	0.444
Protein	2,361	6,646	0.555	0.714	0.630	0.715	0.663	0.596	0.703
US power grid	4,941	6,594	0.350	0.559	0.264	0.570	0.525	0.360	0.557
GR collaborations	5,242	14,484	0.487	0.568	0.246	0.585	0.588	0.557	0.626
Internet	22,963	48,436	0.363	0.636	0.540	0.617	0.609	0.368	0.637
Web-sk	121,422	334,419	0.450	0.561	0.309	0.479	0.469	0.506	0.690
Scale-free	500,000	397,848	0.184	0.367	0.209	0.332	0.280	0.184	0.367
Erdős–Rényi	500,000	750,000	0.424	0.654	0.477	0.650	0.648	0.425	0.660

See description of the networks and algorithms tested in the legend of Table 1. The centrality-based algorithms have been generalized to the case of heterogeneous couplings, as explained in *Additional Comparisons in the Case of Heterogeneous Couplings*. Solutions obtained through the vanilla DMP method are in general on par with the best-performing heuristics and are better for larger networks.