Artificial Intelligence 215 (2014) 79–119

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

On influence, stable behavior, and the most influential individuals in networks: A game-theoretic approach

Mohammad T. Irfan [a,1], Luis E. Ortiz [b,*]

[a] Department of Computer Science, Bowdoin College, Brunswick, ME 04011, United States
[b] Department of Computer Science, Stony Brook University, Stony Brook, NY 11794, United States

ARTICLE INFO

Article history:
Received 13 September 2012
Received in revised form 31 May 2014
Accepted 17 June 2014
Available online 25 June 2014

Keywords:
Computational game theory
Social network analysis
Influence in social networks
Nash equilibrium
Computational complexity

ABSTRACT

We introduce a new approach to the study of influence in strategic settings where the action of an individual depends on that of others in a network-structured way. We propose *network influence games* as a *game-theoretic* model of the behavior of a large but finite networked population. In particular, we study an instance we call *linear-influence games* that allows *both* positive and negative *influence factors*, permitting reversals in behavioral choices. We embrace *pure-strategy Nash equilibrium*, an important solution concept in non-cooperative game theory, to formally define the *stable outcomes* of a network influence game and to predict potential outcomes without explicitly considering intricate dynamics. We address an important problem in network influence, the identification of the *most influential individuals*, and approach it algorithmically using pure-strategy Nash-equilibria computation. *Computationally*, we provide (a) complexity characterizations of various problems on linear-influence games; (b) efficient algorithms for several special cases and heuristics for hard cases; and (c) approximation algorithms, with provable guarantees, for the problem of identifying the most influential individuals. *Experimentally*, we evaluate our approach using both synthetic network influence games and real-world settings of general interest, each corresponding to a separate branch of the U.S. Government. *Mathematically*, we connect linear-influence games to important models in game theory: *potential and polymatrix games*.

1. Introduction

The influence of an entity on its peers is a commonly noted phenomenon in both online and real-life social networks. In fact, there is growing scientific evidence that suggests that influence can induce behavioral changes among the entities in a network. For example, recent work in medical social sciences posits the intriguing hypothesis that much of our behavior such as smoking [16], obesity [15], and even happiness [24] is contagious within a social network.

Regardless of the specific problem addressed, the underlying system studied by Christakis and Fowler exhibits several core features. First, it is often very *large and complex*, with the entities exhibiting different behaviors and interactions. Second, the *network structure of complex interactions* is central to the emergence of collective (global) behavior from individual (local) behavior. For example, in their work on obesity, individuals locally interact with their friends and relatives within their

[*] Corresponding author. Tel.: +1 631 632 1805; fax: +1 631 632 8334.
E-mail addresses: mirfan@bowdoin.edu (M.T. Irfan), leortiz@cs.stonybrook.edu (L.E. Ortiz).
[1] Parts of this work were done while the author was a PhD student at Stony Brook University.

http://dx.doi.org/10.1016/j.artint.2014.06.004
0004-3702/© 2014 Published by Elsevier B.V.

social network. These local interactions appear to give rise to a global phenomenon, namely, the clustering of medically obese individuals [15]. Third, the *directions and strengths of local influences* are highlighted as very relevant to the global behavior of the system as a whole. Fourth, given that one's behavioral choice depends on others, the individuals potentially act in a *strategic* way.

The prevalence of systems and problems like the ones just described, combined with the obvious issue of often-limited control over individuals, raises immediate, broad, difficult, and longstanding policy questions: e.g., *Can we achieve a desired goal, such as reducing the level of smoking or controlling obesity via targeted, minimal interventions in a system? How do we optimally allocate our often limited resources to achieve the largest impact in such systems?*

Clearly, these issues are not exclusive to obesity, smoking or happiness; similar issues arise in a large variety of settings: drug use, vaccination, crime networks, security, marketing, markets, the economy, public policy-making and regulations, and even congressional voting![2] The work reported in this paper is in large part motivated by such questions/settings and their broader implication.

We begin by providing a brief and informal description of our approach to influence in networks. In the next section, we place our approach within the context of the existing literature.

1.1. Overview of our model of influence

Consider a social network where each *individual* has a *binary* choice of *action or behavior*, denoted by -1 and 1. Let us represent this network as a *directed* graph, where each node represents an individual. Each node of this graph has a *threshold level*, which can be positive, negative, or zero; and the threshold levels of all the nodes are not required to be the same. Each arc of this graph is weighted by an *influence factor*, which signifies the level of *direct* influence the tail node of that arc has on the head node. Again, the influence factors can be positive, negative, or zero and are not required to be the same (i.e., symmetric) between two nodes.

Given such a network, our model specifies the *best response* of a node (i.e., what action it should choose) with respect to the actions chosen by the other nodes. The best response of a node is to adopt the action 1 if the *total influence* on it exceeds its threshold and -1 if the opposite happens. In case of a tie, the node is *indifferent* between choosing 1 and -1; i.e., either would be its best response. Here, we calculate the total influence on a node as follows. First, sum up the incoming influence factors on the node from the ones who have adopted the action 1. Second, sum up those influence factors that are coming in from the ones who have adopted -1. Finally, subtract the second sum from the first to get the total influence on that node.

Clearly, in a network with n nodes, there are 2^n possible *joint actions*, resulting from the action choice of each individual node. Among all these joint actions, we call the ones where every node has chosen its best response to everyone else a *pure-strategy Nash equilibria (PSNE)*. We use PSNE to mathematically model the *stable outcomes* that such a networked system could support.

1.2. Overview of the most-influential-nodes problem

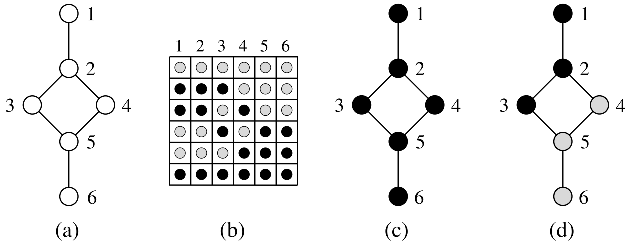
We formulate the most-influential-nodes problem *with respect to a goal of interest*. The goal of interest indirectly determines what we call the *desired stable outcome(s)*. Unlike the mainstream literature on the most-influential-nodes problem [49], maximizing the *spread* of a particular behavior is *not* our objective. Rather, *the desired stable outcome(s) resulting from the goal of interest is what determines our computational objective*. In addition, our solution concept abstracts away the dynamics and does not rely on the "diffusion" process by which such a "spread of behavior" happens.

Roughly speaking, in our approach, we consider a set of individuals S in a network to be a *most influential* set, with respect to a particular goal of interest, if S is the *most preferred* subset among all those that satisfy the following condition: were the individuals in S to choose the behavior $\mathbf{x}_S$ prescribed to them by a desired stable outcome $\mathbf{x} \equiv (\mathbf{x}_S, \mathbf{x}_{-S})$ which achieves the goal of interest, then the *only* stable outcome of the system that remains consistent with their choices $\mathbf{x}_S$ is $\mathbf{x}$ itself.

Said more intuitively, once the nodes in the most influential set S follow the behavior $\mathbf{x}_S$ prescribed to them by a desired stable outcome $\mathbf{x}$ achieving the goal of interest, they become collectively "so influential" that their behavior "forces" every other individual to a unique choice of behavior! Our proposed concept of the most influential individuals is illustrated in Fig. 1 with a very simple example.

Now, there could be many different sets S that satisfy the above condition. For example, S could consist of all the individuals, which might not be what we want. To account for this, we also specify a preference function over all subsets of individuals. While this preference function could in principle be arbitrary, a natural example would be the one that prefers a set S of minimum cardinality.

[2] The headline-grabbing U.S. "debt-ceiling crisis" in 2011, especially the last-minute deal to increase the debt ceiling, is evidence of influence among senators in a strategic setting. We can also view the bipartisan "gang-of-six" senators, specifically chosen to work out a solution, as an *intervention* as such a group would not naturally arise otherwise.



1	2	3	4	5	6
○	○	○	○	○	○
●	●	●	○	○	○
●	●	○	●	○	○
○	○	●	○	●	●
○	○	○	●	●	●
●	●	●	●	●	●

(a) (b) (c) (d)

Fig. 1. Illustration of our approach to influence in networks. Each node has a binary choice of behavior, $\{-1, +1\}$, and, *in this instance*, wants to behave like the majority of its neighbors (and is indifferent if there is a tie). We adopt *pure-strategy Nash equilibrium* (formally defined later), abbreviated as PSNE, as the notion of stable outcome. Here, (a) shows the network, and (b) shows all the PSNE, one in each row, where black denotes behavior 1, gray -1. The goal of interest or the desired outcome is to have everyone choose 1. Selecting the set of nodes $\{1, 2, 3\}$ and assigning these nodes the behavior prescribed by the desired outcome (i.e., 1 for each) lead to two consistent stable outcomes of the system, shown in (c) and (d). Thus, $\{1, 2, 3\}$ cannot be a most-influential set of nodes in our setting. On the other hand, selecting $\{1, 6\}$ and assigning these nodes the behavior 1 lead to the desired outcome as the only possible PSNE. Therefore, $\{1, 6\}$ is a most-influential set, even though these two nodes are at the fringes of the network. Furthermore, note that $\{1, 6\}$ is not most influential in the diffusion setting, since it does not maximize the spread of behavior 1. Also note that $\{3, 4\}$ is another most-influential set in our setting. (Of course, we study a much richer class of games in this paper than this particular instance.)

1.3. Our contributions

Our major contributions include

1. a new approach, grounded in non-cooperative game theory, to the study of influence in networks where individuals exhibit *strategic behavior*;
2. *general-influence games* as a new class of games to model the behavior of individuals, which we call *network influence games* when the individuals are embedded in a (possibly fully-connected, directed) network and the resulting games are (possibly fully-connected, directed) graphical games;
3. *linear-influence games* as a special class of network influence games, including establishing connections to potential games and polymatrix games;
4. a theoretical and empirical study of various computational aspects of linear-influence games, including an algorithm for identifying the most influential individuals; and
5. the application of our approach to two real-world settings: the *U.S. Supreme Court* and the *U.S. Senate*.

The next section introduces the necessary background material to put our work and contributions in a broader context. After that, in Section 3, we define our model of network influence games and the related problems on it. In Sections 4 and 5, we present our computational results. We apply these results to several real-world as well as synthetic datasets in Section 6. We conclude this paper and suggest several future directions in Section 7.

2. Background

"Influence" in social networks, however defined, has been a subject of both theoretical and empirical studies for decades (see, e.g., Wasserman and Faust [86] and the references therein). Although our focus is primarily on computation, the roots of our model go back to the early literature in sociology on collective behavior as well as the more recent literature on collective action. In this section, we will place our model in the context of the relevant literature from sociology, economics, and computer science.[3]

2.1. Connection to collective action in sociology

Although our approach may seem close to the rational calculus models of collective action (see Appendix A for a detailed description), particularly to Granovetter's [31] threshold models, our objective is very much different from that of collective action theory. The focus of the collective-action theory in sociology is to explain *how* individual behavior in a group leads to a collective outcome. For example, Schelling's [77,76] models explain how different distributions of the level of tolerances of individuals lead to residential segregations of different properties. Berk [9] explains how a compromise (such as placing a barricade) evolves within a mixture of rational individuals of different predispositions (militants vs. moderates). Granovetter [31] shows how a little perturbation in the distribution of thresholds can possibly lead a system to a completely different collective outcome. In short, *explaining* collective social phenomena is at the heart of all these studies.

While such an explanation is a scientific pursuit of utmost importance, our focus is rather on an engineering approach to *predicting* stable behavior in a networked population setting. Our approach is not to go through the fine-grained details of

[3] To avoid a long aside and to keep the flow of our presentation, we refer the interested reader to Appendix B for a more detailed exposition on the connection of our model to the century-old study of collective behavior and collective action in sociology.

a process, such as forward recursion [31, p. 1426], which is often plagued with problems when the sociomatrix [31, p. 1429] contains negative elements. Instead, we adopt the notion of PSNE to define stable outcomes. Said differently, the *path* to an equilibrium is not what we focus on; rather, it is the prediction of the equilibrium itself that we focus on.

We next justify our approach in the context of rational calculus models.

2.1.1. PSNE as the solution concept

Nash equilibrium is the most central solution concept in non-cooperative game theory. The fundamental aspect of a Nash equilibrium is stability–no "player" has any incentive to unilaterally deviate from a Nash equilibrium. As a result, Nash equilibrium is very often a natural choice to mathematically model stable outcomes of a complex system.

In this work, we adopt PSNE, one particular form of Nash equilibrium, to model the stable behavioral outcomes of a networked system of influence.[4]

2.1.2. Abstraction of the fine-grained details

Sociologists have recorded minute details of various collective action scenarios in order to substantiate their theories with empirical accounts [57, Ch. 5, 6]. However, in the application scenarios that we are interested in, such as the strategic interaction in the U.S. Congress and the U.S. Supreme Court, very little details can be obtained about *how* a collective outcome emerges.

For example, the Budget Control Act of 2011 was passed by 74–26 votes in the U.S. Senate on August 2, 2011, ending a much debated debt-ceiling crisis. Despite intense media coverage, it would be difficult, if not impossible, to give an accurate account of how this agreement on debt-ceiling was reached. Even if there were an exact account of every conversation and every negotiation that had taken place, it would be extremely challenging to translate such a subjective account into a mathematically defined process, let alone learning the parameters and computing stable outcomes of such a complex model. Therefore, one of our goals is to abstract the fine-grained details using the notion of PSNE.

2.1.3. Network influence games as a less-restrictive model

Typical models of dynamics used in the existing literature almost always impose restrictions or assumptions to keep the model simple enough to permit analytical solutions or to facilitate algorithmic analysis. For example, as mentioned above, the forward recursion process implicitly assumes that the sociomatrix does not have negative elements. The hope is that the essence of the general phenomena that one wants to capture with the model remains even after imposing such restrictions.

In our case, by using the concept of PSNE to abstract dynamical processes, we can deal with rich models without having to impose some of the same restrictions, and at the same time, we can capture equilibria beyond the ones captured by a simple model of dynamics. In particular, our model captures any equilibrium that the process of forward recursion converges to (with any initial configuration); but in addition, our model can also capture equilibria that the forward recursion process cannot.

2.1.4. Focus on practical applications

Although our model of network influence games is grounded in non-cooperative game theory, the way we apply it to real-world settings such as the U.S. Congress is deeply rooted in modern AI [75]. One of the distinctive features of the field of AI is that it is able to build useful tools, often without gaining the full scientific knowledge of *how* a system works. For example, even though we do not have a model of exactly how humans perform inferences while doing tasks as simple as playing different types of parlor games, modern AI has been able to devise systems that perform better than humans on the same tasks, often by a considerable margin [75].

The scientific question of *how* humans reason or perform inferences is of course very important, but, in our view, which may be the prevailing view, the modern focus of AI in general is to *engineer solutions* that would serve our purpose without necessarily having to explain the specific and intricate details of the complex physical phenomena often found in the real world [75]. Of course, under the right conditions, AI does sometimes help experts understand physical phenomena too, although not necessarily purposefully, by suggesting effective insights and potentially useful directions.

In short, we propose an AI-based approach, including AI-inspired models and algorithms, to build a computational tool for predicting the behavior of large, networked populations. Our approach does not model the complex behavioral dynamics in the network, but abstracts it via PSNE. This allows us to deal with a rich set of models and concentrate our efforts on the prediction of stable outcomes.

2.2. Connection to literature on most-influential nodes

To date, the study of influence in a network, by both economists [66,17] and computer scientists [49,22], has been rooted in rational calculus models of behavior. Their approach to connecting individual behavior to collective outcome is mostly by adopting the process of forward recursion [31, p. 1426], which is often employed in studying the diffusion of innovations

[4] Interested readers can find more on the interpretation of Nash equilibrium, including its underlying concept of stability, in Luce and Raiffa [55, Ch. 7] and Osborne [71, Ch. 2–4]. Although we purposefully avoid the question of *how* a PSNE is reached, there is a large body of literature on it [25].

[32, p. 168]. As a result, the term "contagion" in these settings has a rational connotation contrary to the early sociology literature on collective behavior, where "contagion" or "social contagion" alludes to irrational and often hysteric nature of the individuals in a crowd [72,10]. The computational question of identifying the most influential nodes in a network [49], originally posed by Domingos and Richardson [20], has also been studied using forward recursion within rational calculus models.

In the traditional setting of cascade or diffusion models, as described by Kleinberg [49], each node behaves in one of these two ways—it either adopts a new behavior or it does not. Given a number k, their formulation of the most-influential-nodes problem within the diffusion setting asks one to select a set of k nodes such that the *spread* of the new behavior is maximized by the selected nodes being the initial adopters of the new behavior.[5] The most-influential-nodes question in the cascade or diffusion settings typically concerns *infinite* graphs [49, p. 615], such as Morris's local interaction games [66, p. 59]. In contrast, we concern ourselves with large but *finite* graphs here.

The notion of "most-influential nodes" used in this paper is very much different from the one traditionally used within the diffusion setting, mainly because ours seeks to address problems in a different setting (i.e., fully/strictly strategic) and achieves generally different objectives (i.e., desired stable outcomes relative to whatever the goal of interest happens to be, instead of maximizing the number of adopters of the new behavior). If anything, our approach, by taking a strictly game-theoretic perspective, may *complement* the traditional line of work based on diffusion, although even that is not really our main intent. In our view, these are clearly disparate, non-competing approaches. Despite these fundamental differences in objectives, settings, models, problem formulations, and solution concepts, one cannot escape the high level of interest in diffusion models within the computer science community. Therefore, in the remaining of this subsection, we still attempt to briefly mention some possible points of contrast between the typical approach to identifying the most influential nodes in the diffusion setting, as described by Kleinberg [49], and ours.

2.2.1. Stability of outcomes

A subtle aspect of diffusion models is that each node in the network behaves as an independent agent. Any observed influence that a node's neighbors impose on the node is the result of the same node's "rational" or "natural" response to the neighbors' behavior. Thus, in many cases, it would be desirable that the solution to the most-influential-nodes problem leads us to a stable outcome of the system, in which each node's behavior is a best response to the neighbors' behavior. However, if we select a set of nodes with the goal of maximizing the spread of the new behavior, then some of the selected nodes may end up "unhappy" being the initial adopters of the new behavior, with respect to their neighbors' final behavior at the end of forward recursion. For example, a selected node's best behavioral response could be *not* adopting the new behavior after all.

We believe that in some cases it is more natural to require that the desired final state of the system, e.g., the state in which the maximum *possible* number of individuals adopt a particular behavior, be stable (i.e., everyone must be "happy" with their behavioral response).

2.2.2. Arbitrary influence factors: positive and negative

In general, to address the question of finding the most influential nodes, the forward recursion process has been modeled as "monotonic." (Here, a monotonic process refers to the setting where once an agent adopts the new behavior, it cannot go back.) Even in more recent work on influence maximization and minimization [12,14,34], the influence factors (or weights on the edges) are defined to be non-negative.[6] If we think of an application such as reducing the incidence of smoking or obesity, then a model that allows a "change of mind" based on the response of the immediate neighborhood may make more sense. Thus, a notable contrast between the traditional treatment of the most-influential-nodes problem and ours is that we do not restrict the influence among the nodes of the network to non-negative numbers.

In fact, in many applications, both positive and negative influence factors may exist in the same problem instance. Take the U.S. Congress as an example: senators belonging to the same party may have non-negative influence factors on each other (as usually perceived from voting instances on legislation issues), but one senator may (and often does) have a negative influence on another belonging to a different party. While generalized versions of threshold models that allow "reversals" have been derived in the social science literature, to the best of our knowledge, there is no substantive work on the most-influential-nodes problem in that context.[7]

2.2.3. Abstraction of intricate dynamics

The traditional approach to the most-influential-nodes problem emphasizes modeling the complex dynamics of interactions among nodes as a way to a final answer: a set of the most influential nodes. In fact, our model is inspired by the

[5] Note that in Kleinberg's [49] setting, the set of initial adopters, some of whom may have thresholds greater than 0, are *externally* selected in order to set off the forward recursion process, whereas in Granovetter's [31] setting, the initial adopters must have a threshold of 0.

[6] From the description of He et al.'s [34] model, it may at first seem that they are allowing both positive and negative influence weights, which is not the case. Their terminology of "positive" and "negative" weights on the edges refers to the positive and negative cascades that are defined in their context. The weights are non-negative.

[7] Note that this is different from the recent work in diffusion settings on the notion of positive and negative *opinions* [14,12], which in our case would correspond to differing choices of behavior.

same threshold models used in the traditional approach. However, as mentioned earlier, *our emphasis is not on the dynamics of interactions, but on the stable outcomes in a game-theoretic setting.* By doing this, we seek to capture significant, basic, and core *strategic* aspects of complex interactions in networks that naturally appear in many real-world problems (e.g., identifying the most influential senators in the U.S. Congress). Of course, we recognize the importance of the dynamics of interactions for problems requiring a fine level of detail. Yet, we believe that our approach can still capture significant aspects of the problem even at the coarser level of "steady-state" or stable outcomes.

2.2.4. A brief note on submodularity

Let Ω be a set. A set function $f : 2^{\Omega} \to \mathbb{R}$ is *submodular* if for all pairs of subsets $S, T \subset \Omega$ with $S \subset T$ and for any element $x \in \Omega \setminus T$, we have $f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$. Submodular functions show the well-known diminishing marginal return property. It means that if we add one more element x to the input set S of a submodular function, the increase in the function's output is at least the increase in its output when we add the same element to a superset T of S. See Schrijver [78] for details.

The submodularity of the "influence spread function" plays a central role in the algorithmic analysis of the traditional diffusion models. Given the initial adopters as the input, the influence spread function outputs the number of nodes that would ultimately adopt the new behavior in a diffusion setting. See, e.g., Kleinberg [49] and the references therein for more details.

As we mentioned earlier, we allow negative influence factors in our model. If we allow the same in the traditional diffusion models, the influence spread function will no longer remain submodular. This would void the highly heralded theoretical guarantee of a simple, greedy approximation algorithm commonly used for identifying the most influential nodes in a diffusion setting.

In general, *it is not evident what role submodularity plays in our approach, if any*. In fact, it is unclear what the equivalent or analogous concept of the "influence spread function" is in our setting, given that we do not explicitly consider the dynamics by which stable outcomes may arise. We present a more substantive discussion on this matter in Appendix D.

2.2.5. Cooperative vs. non-cooperative games

Departing from the setting of non-cooperative games, there is a completely different approach to computing the most influential nodes [67]. In that approach, a cooperative game [71, Ch. 8] is defined on the underlying social network, and the most influential nodes are computed based on the nodes' (approximate) Shapley values. The Shapley value of a node quantifies the node's marginal importance to a coalition [79]. The underlying model of Narayanam and Narahari [67] is very similar to the ones described by Kleinberg [49]. For example, the normalized influence weights are assumed to be non-negative [67, pp. 2, 15]. It should be noted here that *the computation of the exact Shapley value is intractable in general,* and it is estimated using a sampling-based method [67, pp. 2, 15]. The efficient computation of Shapley value has also received some attention recently [59].

In contrast to the above studies, the behavior of the nodes in our setting is governed by a non-cooperative game. See, e.g., [71, Ch. 1] for the difference between cooperative and non-cooperative games.

2.3. Related work in game theory

Other researchers have used similar game-theoretic notions of "influential individuals" in specific contexts. Particularly close to ours is the work of Heal and Kunreuther [35–38], Kunreuther and Heal [50], Kunreuther and Michel-Kerjan [51], Ballester et al. [5,6], and Kearns and Ortiz [47].

Also, our interest is on identifying an "optimal" set of influential nodes for a variety of optimality criteria, depending on the particular context of interest. For instance, we may prefer the set of influential individuals of minimal size. Such a preference is similar to the concept of "minimal critical coalitions" in the work of Heal and Kunreuther [35–38] and Kunreuther and Michel-Kerjan [51].

Mechanism design is a core area in game theory whose main focus is to "engineer" games, by changing the existing underlaying game or by creating a new one, whose stable outcomes (i.e., equilibria) achieve a desired objective [69]. Although our notion of the most influential individuals is also defined with respect to a desired objective, our approach is conceptually very different. We are not interested in changing, defining, or engineering a new system—*the system is what it is*. Rather, our interest is in altering the *behavior* within the *same system* so as to "lead" or "tip" the system to achieve a desired stable outcome.

In the next section, we will formally define our model and our notion of "most influential nodes" in a network. We will also establish connections to several well-studied classes of game models in game theory: polymatrix and potential games.

3. Network influence games

Inspired by threshold models [31], we first introduce *general-influence games* and *network influence games* as our models of influence in a large, networked population. In Section 3.3, we will introduce *linear-influence games (LIGs)* as a special case of network influence games and will eventually restrict our attention to LIGs.[8] The distinction among these models will be clear over the next few sections.

3.1. Our game-theoretic model of behavior

We first formalize *general-influence games* as a model of behavior.[9] We will use the following notation throughout this paper.

Let n be the number of individuals in a population. For simplicity, we restrict our attention to the case of binary behavior, a common assumption in most of the work in this area. Thus, $x_i \in \{-1, 1\}$ denotes the *behavior* of individual i, where $x_i = 1$ indicates that i "adopts" a particular behavior and $x_i = -1$ indicates i "does not adopt" the behavior. Some examples of behavior of this kind are supporting a particular political measure, candidate or party; holding a particular view or belief; vaccinating against a particular disease; installing and maintaining antivirus software;acquiring fire/home insurance; eating healthy; taking up smoking; participating in criminal activities; among many others.

We denote by $f_i : \{-1, 1\}^{n-1} \to \mathbb{R}$ the function that quantifies the *influence* of other individuals on i.

Definition 3.1 *(Payoff Function).* In general-influence games, we define the *payoff* function $u_i : \{-1, 1\}^n \to \mathbb{R}$ quantifying the preferences of each player i as $u_i(x_i, \mathbf{x}_{-i}) \equiv x_i f_i(\mathbf{x}_{-i})$, where $\mathbf{x}_{-i}$ denotes the vector of a joint-action of all players except i.

Using the above definition and notation, we define a general-influence game as follows.

Definition 3.2 *(General-Influence Game).* A *general-influence game* $\mathcal{G}$ is defined by a set of n players and for each player i, a set of actions $\{-1, 1\}$ and a payoff function u_i given in Definition 3.1.

Next, we characterize the stable outcomes of general-influence games. We start with the definition of the best-response correspondence.

Definition 3.3 *(Best-Response Correspondence).* Given $\mathbf{x}_{-i} \in \{-1, 1\}^{n-1}$, the *best-response correspondence* $\mathcal{BR}_i^{\mathcal{G}} : \{-1, 1\}^{n-1} \to 2^{\{-1,1\}}$ of a player i of a general-influence game $\mathcal{G}$ is defined as follows.

$$\mathcal{BR}_i^{\mathcal{G}}(\mathbf{x}_{-i}) \equiv \arg\max_{x_i \in \{-1,1\}} u_i(x_i, \mathbf{x}_{-i}).$$

Therefore, for all individuals i and any possible behavior $\mathbf{x}_{-i} \in \{-1, 1\}^{n-1}$ of the other individuals in the population, the *best-response behavior* x_i^* of individual i to the behavior $\mathbf{x}_{-i}^*$ of others satisfies

$$\begin{aligned} f_i(\mathbf{x}_{-i}^*) > 0 \quad &\Longrightarrow \quad x_i^* = 1, \\ f_i(\mathbf{x}_{-i}^*) < 0 \quad &\Longrightarrow \quad x_i^* = -1, \quad \text{and} \\ f_i(\mathbf{x}_{-i}^*) = 0 \quad &\Longrightarrow \quad x_i^* \in \{-1, 1\}. \end{aligned}$$

Informally, "positive influences" lead an individual to adopt the behavior, while "negative influences" lead the individual to "reject" the behavior; the individual is indifferent if there is "no influence." We formally characterize the *stable outcomes* of the system by the following notion of *pure-strategy Nash equilibria (PSNE)* of the corresponding general-influence game.

Definition 3.4 *(Pure-Strategy Nash Equilibrium).* A *pure-strategy Nash equilibrium (PSNE)* of a general-influence game $\mathcal{G}$ is a behavior assignment $\mathbf{x}^* \in \{-1, 1\}^n$ that satisfies the following condition. Each player i's behavior x_i^* is a (simultaneous) best-response to the behavior $\mathbf{x}_{-i}^*$ of the rest.

We denote the set of all PSNE of the game $\mathcal{G}$ by

$$\mathcal{NE}(\mathcal{G}) \equiv \left\{\mathbf{x}^* \in \{-1, 1\}^n \;\middle|\; x_i^* \in \mathcal{BR}_i^{\mathcal{G}}(\mathbf{x}_{-i}^*) \text{ for all } i\right\}.$$

[8] As pointed out by a reviewer, the term *influence games* has been used in various contexts. For example, there exist *cooperative influence games* [62], *political influence games* [18,8], *judicial influence games* [7], and *dynamic influence games* [53], to name a few. Our notion is different from the above and does not seek to generalize any of them.

[9] We use the term "general-influence games" to indicate that the game need not be defined on a network formed by individual players (i.e., it does not have to be a graphical game with a not-fully-connected graph), or have a specific "influence form" (i.e., our model of the "influence" interactions need not have a specific form).

3.2. Most influential individuals/nodes: problem formulation

We adopt general-influence games as the model of strategic behavior among the individuals. If, in particular, the individuals are nodes in a network, then we define a graphical-game version of the general-influence game as follows.

Definition 3.5 *(Network Influence Games).* A *network influence game* with directed graph $G = (V, E)$ is a general-influence game with players represented by the nodes V and the influence function f_i of each player/node $i \in V$ represented by a *local potential function* $f_i^{\text{local}} : \{-1, +1\}^{|Pa(i)|} \to \mathbb{R}$, such that $f_i(\mathbf{x}_{-i}) \equiv f_i^{\text{local}}(\mathbf{x}_{Pa(i)})$, where $Pa(i) \equiv \{j \in V \mid (j, i) \in E\}$ are the parents of i in G, and $\mathbf{x}_{Pa(i)}$ is the joint-action of the parents of i consistent with $\mathbf{x}_{-i}$.

If we are dealing with a network influence game, we refer to the most-influential-individuals problem as the *most-influential-nodes* problem. Note that, in this case, by adopting network influence games as the model of strategic behavior among the individuals/nodes in the network, we are departing from the traditional model of diffusion in networks.

We introduce two functions in our definition that we discuss in more detail immediately after the problem formulation. One is what we call the “goal” or “objective function,” denoted by g, which we introduce as a way to formally express that, in our approach, the notion of “most influential” is *relative to a specific goal or objective of interest*. The other, which we call the “set-preference function” and denote by h, is a way to choose among all sets of nodes that achieve our goal of interest.

Definition 3.6 *(Most Influential Nodes).* Let $\mathcal{G}$ be a general-influence game, $g : \{-1, 1\}^n \times 2^{[n]} \to \mathbb{R}$ be the *goal or objective function* mapping a joint-action and a subset of the individuals/nodes/players in $\mathcal{G}$ to a real number quantifying the general preferences over the space of joint-actions and players' subsets, and $h : 2^{[n]} \to \mathbb{R}$ be the *set-preference function* mapping a subset of the players to a real number quantifying the *a priori* preference over the space of players' subsets. Denote by $\mathcal{X}_g^*(S) \equiv \arg\max_{\mathbf{x} \in \mathcal{NE}(\mathcal{G})} g(\mathbf{x}, S)$ the *optimal set of PSNE* of $\mathcal{G}$, with respect to g and a fixed subset of players $S \subset [n]$. We say that a set of players $S^* \subset [n]$ in $\mathcal{G}$ is *most influential* with respect to g and h, if

$$S^* \in \arg\max_{S \subset [n]} h(S), \quad \text{s.t.,} \quad \left|\left\{\mathbf{x} \in \mathcal{NE}(\mathcal{G}) \mid \mathbf{x}_S = \mathbf{x}_S^*, \mathbf{x}^* \in \mathcal{X}_g^*(S)\right\}\right| = 1.$$

As mentioned earlier, we can interpret the players in S^* to be collectively so influential that they are able to restrict every other player's choice of action to a unique one: the action prescribed by some desired stable outcome $\mathbf{x}^*$.

An example of a goal function g that captures the objective of achieving a specific stable outcome $\mathbf{x}^* \in \mathcal{NE}(\mathcal{G})$ is $g(\mathbf{x}, S) \equiv \mathbb{1}[\mathbf{x} = \mathbf{x}^*]$. Another example that captures the objective of achieving a stable outcome with the largest number of individuals adopting the behavior 1 is $g(\mathbf{x}, S) \equiv \sum_{i=1}^n \frac{x_i+1}{2}$, or equivalently, $g(\mathbf{x}, S) \equiv \sum_{i=1}^n x_i$. Note that both of the functions just presented ignore the set S. One alternative that does not is $g(\mathbf{x}, S) \equiv \sum_{i \in S} t_i x_i - \sum_{i \notin S} t_i x_i$, where the t_i's reflect a weighted preference over individual nodes, thus capturing our interest in some “weighted maximum set of adopters.”

A common example of the set-preference function h that captures the preference for sets of small cardinality is to simply define h such that $h(S) > h(S')$ iff $|S| < |S'|$. Similar to the last objective function described in the previous paragraph, an alternative is $h(S) \equiv \sum_{i \in S} v_i - \sum_{i \notin S} v_i$, where the v_i's reflect a weighted preference over individual players in any set S that achieves the objective of interest.

3.3. Linear-influence games

A simple instantiation of the general-influence-game model just described is the case of linear influences. We call this a linear-influence game (LIG). Even though this model falls within the general class of graphical games [46], a distinctive feature of LIGs is a very compact, parametric representation. In addition, it is important to recall that our emphasis here is on the problem of computing stable outcomes of systems of influence and identifying influential agents relative to a particular objective.

Definition 3.7 *(Linear-Influence Game).* In a *linear-influence game (LIG)*, the influence function of each individual i is defined as $f_i(\mathbf{x}_{-i}) \equiv \sum_{j \neq i} w_{ji} x_j - b_i$ where for any other individual j, $w_{ji} \in \mathbb{R}$ is a *weight* parameter quantifying the “influence factor” that j has on i, and $b_i \in \mathbb{R}$ is a *threshold* parameter for i's level of “tolerance” for negative effects.

It follows from Definition 3.1 that although the influence function of an LIG is linear, its payoff function is quadratic. Furthermore, the following argument shows that an LIG is a special type of graphical game in *parametric* form. In general, the influence factors w_{ji} induce a (potentially completely-connected) directed graph, where nodes represent individuals/players, and therefore, we obtain a graphical game having a *linear* (in the number of edges) representation size, as opposed to the *exponential* (in the maximum degree of a node) representation size of general graphical games in normal form [46]. In particular, there is a directed edge (or arc) from individual j to i iff $w_{ji} \neq 0$. Viewed from this perspective, LIGs form a subclass of network influence games.

Example. Fig. 1 shows an example of an LIG with binary behavior. Here, for each edge (i, j), $w_{ji} = 1$ and $w_{ij} = 1$. That is, the game is a special type of LIG with symmetric influence factors. Furthermore, for each node i, its threshold b_i is defined to be 0. Therefore, at any PSNE of this game, each node wants to adopt the behavior of the majority of its neighbors and it is indifferent in the case of a tie.

3.3.1. Connection to polymatrix games

Polymatrix games [42] are n-player non-cooperative games where a player's total payoff is the sum of *partial, individual payoffs* defined relative to each other individual player's action. Formally, for any joint action $\mathbf{x}$, player i's payoff is given by $M_i(x_i, \mathbf{x}_{-i}) \equiv \sum_{j \neq i} \alpha_{ji}(x_j, x_i)$, where $\alpha_{ji}(x_j, x_i)$ is the partial payoff that i receives when i plays x_i and j plays x_j. Note that this partial payoff is local in nature and is not affected by any other node's action. We will consider polymatrix games with only binary actions $\{1, -1\}$ here.

The following property shows an equivalence between LIGs and 2-action polymatrix games. Thus, our computational study of LIGs directly carries over to 2-action polymatrix games.

Proposition 3.8. *LIGs are equivalent to* 2*-action polymatrix games, modulo the set of PSNE.*[10]

Proof. Assume that the number of players $n > 1$; otherwise, the statement holds trivially. We first show that given any instance of an LIG, we can design a polymatrix game that has the same set of PSNE. In an LIG instance, player i's payoff is given by

$$\begin{aligned} u_i(x_i, \mathbf{x}_{-i}) &= x_i \left(\sum_{j \neq i} w_{ji} x_j - b_i \right) \\ &= x_i \sum_{j \neq i} \left(w_{ji} x_j - \frac{b_i}{n-1} \right) \\ &= \sum_{j \neq i} \left(x_i w_{ji} x_j - \frac{x_i b_i}{n-1} \right). \end{aligned}$$

Thus, constructing a polymatrix game instance by defining $\alpha_{ji}(x_j, x_i) \equiv x_i w_{ji} x_j - \frac{x_i b_i}{n-1}$, we have the same set of PSNE in both instances.

Next, we show the reverse direction. Player i's payoff in a 2-action polymatrix game is given by

$$\begin{aligned} M_i(x_i, \mathbf{x}_{-i}) &= \sum_{j \neq i} \alpha_{ji}(x_j, x_i) \\ &= \sum_{j \neq i} \big(\mathbb{1}[x_i = 1] \alpha_{ji}(x_j, 1) + \mathbb{1}[x_i = -1] \alpha_{ji}(x_j, -1) \big) \\ &= \sum_{j \neq i} \left(\frac{1 + x_i}{2} \alpha_{ji}(x_j, 1) + \frac{1 - x_i}{2} \alpha_{ji}(x_j, -1) \right) \\ &= \frac{x_i}{2} \sum_{j \neq i} \big(\alpha_{ji}(x_j, 1) - \alpha_{ji}(x_j, -1) \big) + \frac{1}{2} \sum_{j \neq i} \big(\alpha_{ji}(x_j, 1) + \alpha_{ji}(x_j, -1) \big). \end{aligned}$$

Note that the second term above does not have any effect on i's choice of action. Thus, we can re-define the payoff of player i, without making any change to the set of PSNE of the original polymatrix game, as follows.

$$\begin{aligned} M'_i(x_i, \mathbf{x}_{-i}) &= \frac{x_i}{2} \sum_{j \neq i} \big(\alpha_{ji}(x_j, 1) - \alpha_{ji}(x_j, -1) \big) \\ &= \frac{x_i}{2} \left(\sum_{j \neq i} \big(\mathbb{1}[x_j = 1] \alpha_{ji}(1, 1) + \mathbb{1}[x_j = -1] \alpha_{ji}(-1, 1) \big) \right. \\ &\qquad \left. - \sum_{j \neq i} \big(\mathbb{1}[x_j = 1] \alpha_{ji}(1, -1) + \mathbb{1}[x_j = -1] \alpha_{ji}(-1, -1) \big) \right) \\ &= \frac{x_i}{2} \left(\sum_{j \neq i} \left(\frac{1 + x_j}{2} \alpha_{ji}(1, 1) + \frac{1 - x_j}{2} \alpha_{ji}(-1, 1) \right) \right. \end{aligned}$$

[10] We present this result in the context of PSNE, because this is the solution concept we use throughout the paper. However, this result easily extends to the more general notion of *correlated equilibria* (CE) [2,3], which in turn generalizes the notion of *mixed-strategy Nash equilibria* (MSNE) [68]. (See Fudenberg and Tirole [26] for textbook definitions of CE and MSNE. See Appendix C.4 for additional information.)

$$
\begin{aligned}
&- \sum_{j\neq i}\left(\frac{1+x_j}{2}\alpha_{ji}(1,-1) + \frac{1-x_j}{2}\alpha_{ji}(-1,-1)\right)\Bigg) \\
&= \frac{x_i}{4}\Bigg(\sum_{j\neq i} x_j\big(\alpha_{ji}(1,1) - \alpha_{ji}(-1,1) - \alpha_{ji}(1,-1) + \alpha_{ji}(-1,-1)\big) \\
&\quad + \sum_{j\neq i}\big(\alpha_{ji}(1,1) + \alpha_{ji}(-1,1) - \alpha_{ji}(1,-1) - \alpha_{ji}(-1,-1)\big)\Bigg).
\end{aligned}
$$

Therefore, we can construct an LIG that has exactly the same set of PSNE as the polymatrix game, in the following way. For any player i, define $b_i \equiv -\sum_{j\neq i}\frac{1}{4}(\alpha_{ji}(1,1) + \alpha_{ji}(-1,1) - \alpha_{ji}(1,-1) - \alpha_{ji}(-1,-1))$, and for any player i and any other player j, define $w_{ji} \equiv \frac{1}{4}(\alpha_{ji}(1,1) - \alpha_{ji}(-1,1) - \alpha_{ji}(1,-1) + \alpha_{ji}(-1,-1))$. (Note that the factor 1/4 is not necessary to define the LIG parameters because it does not affect the best response of the players.) □

4. Equilibria computation in linear-influence games

We first study the problem of computing and counting PSNE in LIGs. We show that several special cases of LIGs present us with attractive computational advantages, while the general problem is intractable unless P = NP. We present several heuristics to compute PSNE in LIGs.

4.1. Nonnegative influence factors

When all the influence factors are non-negative, an LIG is supermodular [60,82]. In particular, the game exhibits what is called *strategic complementarity* (see Appendix C.1 for a definition and brief discussion) [13,60,82,81]. Hence, the best-response dynamics converges in at most n rounds. From this, we obtain the following result.

Proposition 4.1. *The problem of computing a PSNE is in P for LIGs on general graphs with only non-negative influence factors.*

This property implies certain monotonicity of the best-response correspondences. More specifically, for each player i, if any subset of the other players "increases his/her strategy" by adopting the behavior 1, then player i's best-response cannot be to abandon adoption (i.e., move from 1 to -1). In other words, once a player adopts the behavior 1, it has no incentive to go back. This monotonicity property also follows directly from the linear threshold model. Strategic complementarity implies other interesting characterizations of the structure of PSNE in LIGs and the behavior of best-response dynamics. For example, such games always have a PSNE: If we start with the complete assignment in which either everyone is playing 1, or everyone is playing -1, parallel/synchronous best-response dynamics converges after at most n rounds [60]. If both best-response processes starting with all -1's and all 1's converge to the same PSNE, then that PSNE is unique. Otherwise, any other PSNE of the game must be "contained" between the two different PSNE. We can also view this from the perspective of constraint propagation with monotonic constraints [74].

4.2. Special influence structures and potential games

Several special subclasses of LIGs are potential games [63]. (See Appendix C.3 for more information.) This connection guarantees the existence of PSNE in such games.

Proposition 4.2. *If the influence factors of an LIG $\mathcal{G}$ are symmetric (i.e., $w_{ji} = w_{ij}$, for all i, j), then $\mathcal{G}$ is an (exact) potential game.*

Proof. We show that the game has an exact potential function,

$$\Phi(\mathbf{x}) = \sum_{t=1}^{n} x_t\left(\sum_{i\neq t}\frac{x_i w_{it}}{2} - b_t\right). \tag{1}$$

Consider any player j. The difference in j's payoff for $x_j = 1$ and $x_j = -1$ (assuming all other players play $\mathbf{x}_{-j}$ in both cases) is

$$
\begin{aligned}
u_j(1,\mathbf{x}_{-j}) - u_j(-1,\mathbf{x}_{-j}) &= 1 \times \left(\sum_{i\neq j} x_i w_{ij} - b_j\right) - (-1) \times \left(\sum_{i\neq j} x_i w_{ij} - b_j\right) \\
&= 2 \times \left(\sum_{i\neq j} x_i w_{ij} - b_j\right).
\end{aligned} \tag{2}
$$

Next, the difference in the potential function when j plays 1 and -1 is

$$\begin{aligned}
&\Phi(1,\mathbf{x}_{-j})-\Phi(-1,\mathbf{x}_{-j})\\
&\quad=1\times\left(\sum_{i\neq j}\frac{x_i w_{ij}}{2}-b_j\right)+\sum_{t\neq j}x_t\left(\sum_{i\neq t}\mathbb{1}[i\neq j]\frac{x_i w_{it}}{2}-b_t\right)+\sum_{t\neq j}x_t\left(\sum_{i\neq t}\mathbb{1}[i=j]\frac{1\times w_{it}}{2}-b_t\right)\\
&\qquad-(-1)\times\left(\sum_{i\neq j}\frac{x_i w_{ij}}{2}-b_j\right)-\sum_{t\neq j}x_t\left(\sum_{i\neq t}\mathbb{1}[i\neq j]\frac{x_i w_{it}}{2}-b_t\right)-\sum_{t\neq j}x_t\left(\sum_{i\neq t}\mathbb{1}[i=j]\frac{(-1)\times w_{it}}{2}-b_t\right)\\
&\quad=2\times\left(\sum_{i\neq j}\frac{x_i w_{ij}}{2}-b_j\right)+2\times\left(\sum_{t\neq j}\frac{x_t w_{jt}}{2}\right)\\
&\quad=2\times\left(\sum_{i\neq j}x_i w_{ij}-b_j\right).
\end{aligned}$$

The last line follows by the symmetry of the weights (i.e., $w_{ij}=w_{ji}$). □

If, in addition, the threshold $b_i=0$ for all i, the game is a *party-affiliation game*, and computing a PSNE in such games is PLS-complete [23].

The following result is on a large class of games that we call *indiscriminate* LIGs, where for every player i, the influence weight, $w_{ij}\equiv\delta_i\neq 0$, that i imposes on every other player j is the same. The interesting aspect of this result is that these LIGs are potential games despite being possibly *asymmetric* and exhibiting strategic substitutability, due to negative influence factors (see Appendix C.2).

Proposition 4.3. *Let $\mathcal{G}$ be an indiscriminate LIG in which all δ_i for all i, have the same sign, denoted by $\rho\in\{-1,+1\}$. Then $\mathcal{G}$ is an ordinal potential game with potential function $\Phi(\mathbf{x})=\rho[(\sum_{i=1}^n\delta_i x_i)^2-2\sum_{i=1}^n b_i\delta_i x_i]$.*

Proof. It is sufficient to show that the sign of the difference in the individual utilities of any player due to changing her action unilaterally, is the same as the sign of the difference in the corresponding potential functions. For any player j, the first difference is

$$1\times\left(\sum_{i\neq j}\delta_i x_i-b_j\right)-(-1)\times\left(\sum_{i\neq j}\delta_i x_i-b_j\right)=2\left(\sum_{i\neq j}\delta_i x_i-b_j\right). \tag{3}$$

The potential function when j plays 1,

$$\begin{aligned}
\Phi(1,\mathbf{x}_{-j})&=\rho\left[\left(\sum_{i\neq j}\delta_i x_i+\delta_j\times 1\right)^2-2\sum_{i\neq j}b_i\delta_i x_i-2b_j\delta_j\times 1\right]\\
&=\rho\left[\left(\sum_{i\neq j}\delta_i x_i\right)^2+\delta_j{}^2+2\left(\sum_{i\neq j}\delta_i x_i\right)\delta_j-2\sum_{i\neq j}b_i\delta_i x_i-2b_j\delta_j\right].
\end{aligned}$$

The potential function when j plays -1,

$$\begin{aligned}
\Phi(-1,\mathbf{x}_{-j})&=\rho\left[\left(\sum_{i\neq j}\delta_i x_i+\delta_j\times(-1)\right)^2-2\sum_{i\neq j}b_i\delta_i x_i-2b_j\delta_j\times(-1)\right]\\
&=\rho\left[\left(\sum_{i\neq j}\delta_i x_i\right)^2+\delta_j{}^2-2\left(\sum_{i\neq j}\delta_i x_i\right)\delta_j-2\sum_{i\neq j}b_i\delta_i x_i+2b_j\delta_j\right].
\end{aligned}$$

Thus, the difference in the potential functions,

$$\Phi(1,\mathbf{x}_{-j})-\Phi(-1,\mathbf{x}_{-j})=4\rho\delta_j\left(\sum_{i\neq j}\delta_i x_i-b_j\right). \tag{4}$$

Since $\rho\delta_j>0$, the quantities given in (3) and (4) have the same sign. □

4.3. Tree-structured influence graphs

The following result follows from a careful, non-trivial modification of the **TreeNash** algorithm [46]. Note that the running time of the **TreeNash** algorithm is exponential in the degree of a node and thus also exponential in the representation size of an LIG! In contrast, our algorithm is linear in the maximum degree and thereby linear in the representation size of an LIG. The complete proof follows a proof sketch.

Theorem 4.4. *There exists an $O(nd)$ time algorithm to find a PSNE, or to decide that there exists none, in LIGs with tree structures, where d is the maximum degree of a node.*

Proof sketch. We borrow the notation of Kearns et al. [46]. The modification of the **TreeNash** algorithm involves efficiently (in $O(d)$ time, not $O(2^d)$) determining the existence of a witness vector and constructing one, if it exists, at each node during the downstream pass, in the following way.

Suppose that an internal node i receives tables $T_{ki}(x_k, x_i)$ from its parents k, and that i wants to send a table $T_{ij}(x_i, x_j)$ to its unique child j. If for some parent k of i, $T_{ki}(-1, x_i) = 0$ and $T_{ki}(1, x_i) = 0$, then i sends the following table entries to j: $T_{ij}(x_i, -1) = 0$ and $T_{ij}(x_i, 1) = 0$. Otherwise, we first partition i's set of parents into two sets in $O(d)$ time: $Pa_1(i, x_i)$ consisting of the parents k of i that have a unique best response $\hat{x}_k$ to i's playing x_i and $Pa_2(i, x_i)$ consisting of the remaining parents of i. We show that $T_{ij}(x_i, x_j) = 1$ iff

$$x_i\left(x_j w_{ji} + \sum_{k \in Pa_1(i,x_i)} \hat{x}_k w_{ki} + \sum_{t \in Pa_2(i,x_i)} \underbrace{\left(2 \times \mathbb{1}[x_i w_{ti} > 0] - 1\right)}_{t\text{'s action in witness vector}} w_{ti} - b_i\right) \geq 0,$$

from which we get a witness vector, if it exists. □

Following is the complete proof of Theorem 4.4.

Proof. We denote any node i's action by $x_i \in \{-1, 1\}$, its threshold by b_i, and the influence of any node i on another node j by w_{ij}. Furthermore, denote the set of parents of a node i by $Pa(i)$. We now describe the two phases of the modified **TreeNash** algorithm.

1. **Downstream phase.** In this phase each node sends a table to its unique child. We denote the table that node i sends to its child j as $T_{ij}(x_i, x_j)$, indexed by the actions of i and j, and define the set of conditional best-responses of a node i to a neighboring node j's action x_j as $BR_i(j, x_j) \equiv \{x_i \mid T_{ij}(x_i, x_j) = 1\}$. If $|BR_i(j, x_j)| = 1$ then we will abuse this notation by letting $BR_i(j, x_j)$ be the unique best-response of i to j's action x_j.
   The downstream phase starts at the leaf nodes. Each leaf node l sends a table $T_{lk}(x_l, x_k)$ to its child k, where $T_{lk}(x_l, x_k) = 1$ if and only if x_l is a conditional best-response of l to k's choice of action x_k. Suppose that an internal node i obtains tables $T_{ki}(x_k, x_i)$ from its parents $k \in Pa(i)$, and that i needs to send a table to its child j. Once i receives the tables from its parents, it first computes (in $O(d)$ time) the following three sets that partition the parents of i based on the size of their conditional best-response sets when i plays x_i.

   $$Pa_r(i, x_i) \equiv \left\{k \in Pa(i) \mid \left|BR_k(i, x_i)\right| = r\right\}, \quad \text{for } r = 0, 1, 2.$$

   This is how i computes the table $T_{ij}(x_i, x_j)$ sent to j: $T_{ij}(x_i, x_j) = 1$ if and only if there exists a *witness vector* $(x_k)_{k \in Pa(i)}$ that satisfies the following two conditions:
   Condition 1. $T_{ki}(x_k, x_i) = 1$ for all $k \in Pa(i)$.
   Condition 2. The action x_i is a best-response of node i when every node $k \in Pa(i)$ plays x_k and j plays x_j.
   There are two cases.
   Case I: $Pa_0(i, x_i) \neq \emptyset$**.** In this case, there exists some parent k of i for which both $T_{ki}(-1, x_i) = 0$ and $T_{ki}(1, x_i) = 0$. Therefore, there exists no witness vector that satisfies Condition 1, and i sends the following table entries to j: $T_{ij}(x_i, x_j) = 0$, for $x_j = -1, 1$.
   Case II: $Pa_0(i, x_i) = \emptyset$**.** In this case, we will show that there exists a witness vector for $T_{ij}(x_i, x_j) = 1$ satisfying Conditions 1 and 2 if and only if the following inequality holds (which can be verified in $O(d)$ time). Below, we will use the *sign* function σ: $\sigma(x) = 1$ if $x > 0$, and $\sigma(x) = -1$ otherwise.

   $$x_i\left(w_{ji} x_j + \sum_{k \in Pa_1(i,x_i)} w_{ki} BR_k(i, x_i) + \sum_{k \in Pa_2(i,x_i)} w_{ki} \sigma(x_i w_{ki}) - b_i\right) \geq 0. \tag{5}$$

   In fact, if Inequality (5) holds then we can construct a witness vector in the following way: If $k \in Pa_1(i, x_i)$, then let $x_k = BR_k(i, x_i)$, otherwise, let $x_k = \sigma(x_i w_{ki})$. Since each parent k of i is playing its conditional best-response x_k to i's choice of action x_i, we obtain, $T_{ki}(x_k, x_i) = 1$ for all $k \in Pa(i)$. Furthermore, Inequality (5) says that i is playing its best-response x_i to each of its parents k playing x_k and its child j playing x_j.

To prove the reverse direction, we start with a witness vector $(x_k)_{k \in Pa(i)}$ such that Conditions 1 and 2 specified above hold. In particular, we can write Condition 2 as

$$x_i \left(w_{ji} x_j + \sum_{k \in Pa(i)} w_{ki} x_k - b_i \right) \geq 0. \tag{6}$$

The following line of arguments shows that Inequality (5) holds.

$$\begin{aligned}
&x_i w_{ki} \sigma(x_i w_{ki}) \geq x_i w_{ki} x_k, \quad \text{for any } k \in Pa_2(i, x_i) \\
&\quad \Rightarrow \quad x_i \sum_{k \in Pa_2(i,x_i)} w_{ki} \sigma(x_i w_{ki}) \geq x_i \sum_{k \in Pa_2(i,x_i)} w_{ki} x_k \\
&\quad \Rightarrow \quad x_i \left(w_{ji} x_j + \sum_{k \in Pa_1(i,x_i)} w_{ki} BR_k(i, x_i) + \sum_{k \in Pa_2(i,x_i)} w_{ki} \sigma(x_i w_{ki}) - b_i \right) \geq x_i \left(w_{ji} x_j + \sum_{k \in Pa(i)} w_{ki} x_k - b_i \right) \\
&\quad \Rightarrow \quad x_i \left(w_{ji} x_j + \sum_{k \in Pa_1(i,x_i)} w_{ki} BR_k(i, x_i) + \sum_{k \in Pa_2(i,x_i)} w_{ki} \sigma(x_i w_{ki}) - b_i \right) \geq 0, \quad \text{using Inequality (6).}
\end{aligned}$$

In addition to computing the table T_{ij}, node i stores the following witness vector $(x_k)_{k \in Pa(i)}$ for each table entry $T_{ij}(x_i, x_j)$ that is 1: if $k \in Pa_1(i, x_i)$, then $x_k = BR_k(i, x_i)$, otherwise, $x_k = \sigma(x_i w_{ki})$. The downstream phase ends at the root node z, and z computes a unary table $T_z(x_z)$ such that $T_z(x_z) = 1$ if and only if there exists a witness vector $(x_k)_{k \in Pa(z)}$ such that $T_{kz}(x_k, x_z) = 1$ for all $k \in Pa(z)$ and x_z is a best-response of z to $(x_k)_{k \in Pa(z)}$.

The computation of the table at each node, which runs in $O(d)$ time, dominates the time complexity of the downstream phase. We visit every node exactly once. So, the total running time of the downstream phase is $O(nd)$. Note that if there does not exist any PSNE in the game then all the table entries computed by some node will be 0.

2. **Upstream phase.** In the upstream phase, each node sends instructions to its parents about which actions to play, along with the action that the node itself is playing. The upstream phase begins at the root node z. For any table entry $T_z(x_z) = 1$, z decides to play x_z itself and instructs each of its parents to play the action in the witness vector associated with $T_z(x_z) = 1$. At an intermediate node i, suppose that its child j is playing x_j and instruct i to play x_i. The node i looks up the witness vector $(x_k)_{k \in Pa(i)}$ associated with $T_{ij}(x_i, x_j) = 1$ and instructs its parents to play according to that witness vector. This process propagates upward, and when we reach all the leaf nodes, we obtain a PSNE for the game. Note that we can find a PSNE in this phase if and only if there exists one.

In the upstream phase, each node sends $O(d)$ instructions to its parents. Thus, the upstream phase takes $O(nd)$ time, and the whole algorithm takes $O(nd)$ time. □

4.4. Hardness results

Computational problems are often classified into complexity classes according to their hardness. Some of the hardest classes of problems are NP-complete problems, co-NP-complete problems, and #P-complete problems [27]. In this section, we show that many of the computational problems related to LIGs belong to these hard classes.

First, computing a PSNE in a general graphical game is known to be computationally hard [29]. However, that result does not imply intractability of our problem, nor do the proofs seem easily adaptable to our case. LIGs are a special type of graphical game with quadratic payoffs, or in other words a graphical, *parametric* polymatrix game [42], and thus have a more succinct representation than general graphical games ($O(nd)$ in contrast to $O(n2^d)$, where d is the maximum degree of a node). Next, we show that various interesting computational questions on LIGs are intractable, unless P = NP.

We settle the central hardness question on LIGs (and also on 2-action polymatrix games) in 1(a) below. Related to the most-influential-nodes problem formulation, 1(b) states that given a subset of players, it is NP-complete to decide whether there exists a PSNE in which these players adopt the new behavior. We present a similar result in 1(c).

A prime feature of our formulation of the most-influential-nodes problem is the uniqueness of the desired stable outcome when the most influential nodes adopt their behavior according to the desired stable outcome. We show in (2) that deciding whether a given set of players fulfills this criterion is co-NP-complete.

As we will see later, in order to compute a set of the most influential nodes, it suffices to be able to count the number of PSNE of an LIG (to be more specific, it suffices to count the number of PSNE extensions for a given partial assignment to the players' actions). We show in (3) that this problem is #P-complete. Note that the #P-completeness result for LIGs even with star structure is in contrast to the polynomial-time counterpart for general graphical games with tree graphs, for which not only deciding the existence of a PSNE is in P, but also counting PSNE on general graphical games with tree graphs is in P [46,85]. To better appreciate this result, consider the representation sizes of LIGs and tree-structured graphical games, which are linear and exponential in the maximum degree, respectively.

Below, we first summarize the hardness results, followed by an informal outline of the proofs. We then present each individual statement as a separate theorem (or corollary of a more general theorem) and provide the complete formal proofs.

1. *It is NP-complete to decide the following questions in LIGs.*
   (a) *Does there exist a PSNE?*
   (b) *Given a designated non-empty set of players, does there exist a PSNE consistent with those players playing* 1*?*
   (c) *Given a number* $k \geq 1$, *does there exist a PSNE with at least* k *players playing* 1*?*
2. *Given an LIG and a designated non-empty set of players, it is co-NP-complete to decide if there exists a unique PSNE with those players playing* 1.
3. *It is #P-complete to count the number of PSNE, even for special classes of the underlying graph, such as a bipartite or a star graph.*

Proof sketch. The complete proofs appear immediately following this proof sketch. The proof of 1(a) reduces the 3-SAT problem to an LIG that consists of a player for each clause and each variable of the 3-SAT instance. The influence factors among these players are designed such that the LIG instance possesses a PSNE if and only if the 3-SAT instance has a satisfying assignment. Since the underlying graph of the LIG instance is always bipartite, we obtain as a corollary that the NP-completeness of that existence problem holds even for LIGs on bipartite graphs.

The proofs of 1(b), 1(c), and 2 use reductions from the *monotone one-in-three SAT* problem. For 1(b), given a monotone one-in-three SAT instance I, we construct an LIG instance J having a player for each clause and each variable of I. Again, we design the influence factors in such a way that I is satisfiable if and only if J has a PSNE. The reduction for 1(c) builds upon that of 1(b) with specifically designed *extra players* and additional connectivity in the LIG instance. Again, the gadgets used in the proof of 1(c) are extended for the proof of 2.

The proof of 3 uses reductions from the 3-SAT and the #KNAPSACK problem. The reduction from the 3-SAT problem is the same as that used in 1(a), and the proof of the #P-hardness of the bipartite case is by showing that the number of solutions to the 3-SAT instance is the same as the number of PSNE of the LIG instance. On the other hand, to prove the claim of #P-completeness of counting PSNEs of LIGs having star graphs, we give a reduction from the #KNAPSACK problem. Given a #KNAPSACK instance, we create an LIG instance with a star structure among the players and with specifically designed influence factors such that the number of PSNE of the LIG instance is the same as the number of solutions to the #KNAPSACK instance. □

Complete proofs of hardness results

To enhance the clarity of the proofs, we reduced existing NP-complete problems to LIGs with binary actions $\{0, 1\}$, instead of $\{-1, 1\}$. We next show, via a linear transformation, that one can reduce any LIG with actions $\{0, 1\}$ to an LIG with the same underlying graph, but with actions $\{-1, 1\}$.

Reduction from $\{0, 1\}$-action LIG to $\{-1, 1\}$-action LIG. Consider any $\{0, 1\}$-action LIG instance I, where w and b denote the influence factors and the thresholds, respectively (see Definition 3.7). We next construct a $\{-1, 1\}$-action LIG instance J with the same players that are in I and with influence factors $w'_{ji} \equiv \frac{w_{ji}}{2}$ (for any i and any $j \neq i$), thresholds $b'_i \equiv b_i - \sum_{j\neq i} \frac{w_{ji}}{2}$ (for any i). We show that $\mathbf{x}$ is a PSNE of I if and only if $\mathbf{x}'$ is a PSNE of J, where $x'_i = 2x_i - 1$ for any i.

By definition, $\mathbf{x}$ is a PSNE of I if and only if for any player i,

$$\begin{aligned}
& x_i\Big(\sum_{j\neq i} x_j w_{ji} - b_i\Big) \geq (1 - x_i)\Big(\sum_{j\neq i} x_j w_{ji} - b_i\Big) \\
\Leftrightarrow \quad & (2x_i - 1)\Big(\sum_{j\neq i} x_j w_{ji} - b_i\Big) \geq 0 \\
\Leftrightarrow \quad & x'_i\Big(\sum_{j\neq i} \frac{x'_j + 1}{2} w_{ji} - b_i\Big) \geq 0 \\
\Leftrightarrow \quad & x'_i\Big(\sum_{j\neq i} x'_j \frac{w_{ji}}{2} - \Big(b_i - \sum_{j\neq i} \frac{w_{ji}}{2}\Big)\Big) \geq 0 \\
\Leftrightarrow \quad & x'_i\Big(\sum_{j\neq i} x'_j w'_{ji} - b'_i\Big) \geq 0,
\end{aligned}$$

which is the equivalent statement of $\mathbf{x}'$ being a PSNE of J. □

Theorem 4.5. *It is NP-complete to decide if there exists a PSNE in an LIG.*

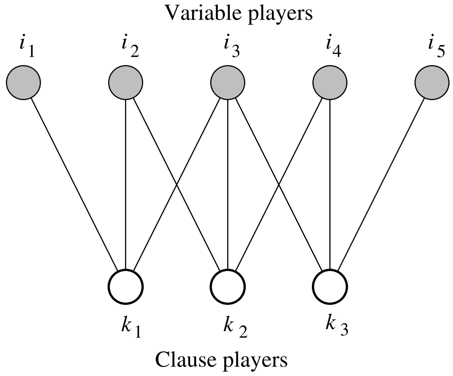


Fig. 2. Illustration of the structure of an LIG instance from a 3-SAT instance (each undirected edge represents two arcs of opposite directions between the same two nodes). In this example, the 3-SAT instance is $(i_1 \vee i_2 \vee i_3) \wedge (\neg i_2 \vee i_3 \vee i_4) \wedge (\neg i_3 \vee i_4 \vee \neg i_5)$.

Proof. Since we can verify whether a joint action is a PSNE or not in polynomial time, the problem is in NP. We use a reduction from the 3-SAT problem to show that the problem is NP-hard.

Let I be an instance of the 3-SAT problem. Suppose that I has m clauses and n variables. For any variable i, we define C_i to be the set of clauses in which i appears, and for any clause k, we define V_k to be the set of variables appearing in clause k. For any clause k and any variable $i \in V_k$, let $l_{k,i}$ be 1 if i appears in k in non-negated form and 0 otherwise. We now build an LIG instance J from I. In this game, every clause as well as every variable is a player. Each clause k has arcs to variables in V_k, and each variable i has arcs to clauses in C_i. The structure of the graph is illustrated in Fig. 2. We next define the thresholds of the players and the influence factors on the arcs. For any clause k, let its threshold be $1 - \epsilon - \sum_{i \in V_k}(1 - l_{k,i})$. Here, ϵ is a constant, and $0 < \epsilon < 1$. For any variable i let its threshold be $\sum_{k \in C_i}(1 - 2l_{k,i})$. The weight on the arc from any clause k to any variable $i \in V_k$ is defined to be $1 - 2l_{k,i}$, and that from any variable i to any clause $k \in C_i$ is $2l_{k,i} - 1$. We denote the action of any clause k by $z_k \in \{0, 1\}$ and that of any variable i by $x_i \in \{0, 1\}$.

First, we prove that if there exists a satisfying truth assignment in I then there exists a PSNE in J. Consider any satisfying truth assignment S in I. Let the players in J choose their actions according to their truth values in S, that is, 1 for *true* and 0 for *false*. Clearly, every clause player is playing 1. Next, we show that every player in J is playing its best response under this choice of actions.

We now show that no clause has incentive to play 0, given that the other players do not change their actions. In the solution S to I, every clause has a literal that is *true*. Therefore, in J every clause k has some variable $i \in V_k$ such that $x_i = l_{k,i}$. We have to show that the total influence on k is at least the threshold of k:

$$\begin{aligned}
&\sum_{i \in V_k} x_i(2l_{k,i} - 1) \geq 1 - \epsilon - \sum_{i \in V_k}(1 - l_{k,i}) \\
&\quad \Leftrightarrow \quad \sum_{i \in V_k}\big(x_i(2l_{k,i} - 1) + (1 - l_{k,i})\big) \geq 1 - \epsilon \\
&\quad \Leftrightarrow \quad \sum_{i \in V_k}\big(x_i l_{k,i} + (1 - x_i)(1 - l_{k,i})\big) \geq 1 - \epsilon.
\end{aligned}$$

Since for some $i \in V_k$, $x_i = l_{k,i}$, the above inequality holds strictly, that is,

$$\sum_{i \in V_k}\big(x_i l_{k,i} + (1 - x_i)(1 - l_{k,i})\big) > 1 - \epsilon.$$

Therefore, every clause k must play 1.

We need to show that no variable player has incentive to deviate, given that the other players do not change their actions. The total influence on any variable player i is $\sum_{k \in C_i} z_k(1 - 2l_{k,i}) = \sum_{k \in C_i}(1 - 2l_{k,i})$ (since $z_r = 1$ for every clause r). The threshold of i is $\sum_{k \in C_i}(1 - 2l_{k,i})$. Thus, every variable player i is indifferent between choosing actions 1 and 0 and has no incentive to deviate.

We now consider the reverse direction, that is, given a PSNE in J we show that there exists a satisfying assignment in I. We first show that at any PSNE, every clause must play 1. If this is not the case, suppose, for a contradiction, that for some clause r, $z_r = 0$. Since r's best response is 0 (this is a PSNE), we obtain

$$\sum_{i \in V_r} x_i(2l_{r,i} - 1) \leq 1 - \epsilon - \sum_{i \in V_r}(1 - l_{r,i}) \quad \Leftrightarrow \quad \sum_{i \in V_r}\big(x_i l_{r,i} + (1 - x_i)(1 - l_{r,i})\big) \leq 1 - \epsilon.$$

Therefore, for every variable player $j \in V_r$, $x_j \neq l_{r,j}$. Furthermore, for any $j \in V_r$, j does not have any incentive to deviate. Using these properties of a PSNE we will arrive at a contradiction, and thereby prove that z_r must be 1.

Consider any variable player $j \in V_r$, and let the difference between j's total incoming influence and its threshold be U_j. We get

$$\begin{aligned}
&U_j = \sum_{k \in C_j} z_k(1-2l_{k,j}) - \sum_{k \in C_j}(1-2l_{k,j}) = \sum_{k \in C_j}\big((1-z_k)(2l_{k,j}-1)\big) \\
\Leftrightarrow\quad & U_j = \sum_{k \in C_j}\big((1-z_k)(2l_{k,j}-1)\mathbb{1}[l_{k,j}=1]\big) + \sum_{k \in C_j}\big((1-z_k)(2l_{k,j}-1)\mathbb{1}[l_{k,j}=0]\big) \\
\Leftrightarrow\quad & U_j = \sum_{k \in C_j}\big((1-z_k)\mathbb{1}[l_{k,j}=1]\big) - \sum_{k \in C_j}\big((1-z_k)\mathbb{1}[l_{k,j}=0]\big).
\end{aligned}$$

At any PSNE, if $x_j = 1$ then $U_j \geq 0$; otherwise, $U_j \leq 0$. Thus, the best response condition for variable j gives us

$$\begin{aligned}
&\sum_{k \in C_j}\big((1-z_k)\mathbb{1}[l_{k,j}=x_j]\big) \geq \sum_{k \in C_j}\big((1-z_k)\mathbb{1}[l_{k,j}\neq x_j]\big) \\
\Leftrightarrow\quad & \sum_{k \in C_j-\{r\}}\big((1-z_k)\mathbb{1}[l_{k,j}=x_j]\big) + (1-z_r)\mathbb{1}[l_{r,j}=x_j] \geq \sum_{k \in C_j-\{r\}}\big((1-z_k)\mathbb{1}[l_{k,j}\neq x_j]\big) + (1-z_r)\mathbb{1}[l_{r,j}\neq x_j] \\
\Leftrightarrow\quad & \sum_{k \in C_j-\{r\}}\big((1-z_k)\mathbb{1}[l_{k,j}=x_j]\big) \geq \sum_{k \in C_j-\{r\}}\big((1-z_k)\mathbb{1}[l_{k,j}\neq x_j]\big) + 1, \quad \text{since } l_{r,j} \neq x_j.
\end{aligned}$$

The above inequality cannot be true, because the left hand side is always 0 (if $l_{k,j} = x_j$ then z_k must be 1 at any PSNE), and the right hand side is ≥ 1. Thus, we obtained a contradiction, and z_r cannot be 0.

So far, we showed that at any PSNE $z_k = 1$ for any clause player k. To complete the proof, we now show that for every clause player k, there exists a variable player $i \in V_k$ such that $x_i = l_{k,i}$. If we can show this then we can translate the semantics of the actions in J to the truth values in I and thereby obtain a satisfying truth assignment for I.

Suppose, for the sake of a contradiction, that for some clause k and for all variable $i \in V_k$, $x_i \neq l_{k,i}$. Since $z_k = 1$, we find that

$$\begin{aligned}
&\sum_{i \in V_k} x_i(2l_{k,i}-1) \geq 1-\epsilon-\sum_{i \in V_k}(1-l_{k,i}) \\
\Leftrightarrow\quad & \sum_{i \in V_k}\big(x_i l_{k,i} + (1-x_i)(1-l_{k,i})\big) \geq 1-\epsilon \\
\Leftrightarrow\quad & 0 \geq 1-\epsilon, \quad \text{which gives us the desired contradiction.} \quad \square
\end{aligned}$$

The proof of Theorem 4.5 reduces the 3-SAT problem to an LIG where the underlying graph is bipartite. Thus, we obtain the following corollary.

Corollary 4.6. *It is NP-complete to decide if there exists a PSNE in an LIG on a bipartite graph.*

The proof of Theorem 4.5 directly leads us to the following result that the counting version of the problem is #P-complete.

Corollary 4.7. *It is #P-complete to count the number of PSNE of an LIG.*

Proof. The proof follows from the proof of Theorem 4.5. Membership of this counting problem in #P is easy to see. Using the same reduction as in the proof of Theorem 4.5, we find that each satisfying truth assignment (among the 2^n possibilities) to the variables of the 3-SAT instance I can be mapped to a distinct PSNE of the LIG instance J. Furthermore, we saw that at each PSNE in J, every clause player must play 1. Thus, for each of the 2^n joint strategies of the variable players (while having the clause players play 1), if the joint strategy is a PSNE then we can map it to a distinct satisfying assignment in I. Moreover, each of these two mappings are the inverse of the other. Therefore, the number of satisfying assignments of I is the same as the number of PSNE in J. Since counting the number of satisfying assignments of a 3-SAT instance is #P-complete, counting the number of PSNE of an LIG, even on a bipartite graph, is also #P-complete. □

While Corollary 4.7 shows the hardness of counting the number of PSNE of an LIG on a general graph, we can show the same hardness result even on special classes of graphs, such as star graphs:

Theorem 4.8. *Counting the number of PSNE of an LIG on a star graph is #P-complete.*

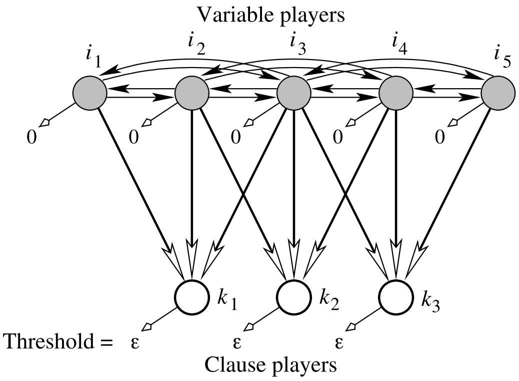


Fig. 3. Illustration of the NP-hardness reduction of Theorem 4.9. The monotone one-in-three SAT instance is $(i_1 \vee i_2 \vee i_3) \wedge (i_2 \vee i_3 \vee i_4) \wedge (i_3 \vee i_4 \vee i_5)$. The threshold of each variable player is 0, and that of each clause player is ϵ.

Proof. Since we can verify whether a joint strategy is a PSNE in polynomial time, the problem is in #P. We will show the #P-hardness using a reduction from #KNAPSACK, which is the problem of counting the number of feasible solutions in a 0-1 Knapsack problem: Given n items, the weight $a_i \in \mathcal{Z}^+$ of each item i, and the maximum capacity of the sack $W \in \mathcal{Z}^+$, #KNAPSACK asks how many ways we can pick the items to satisfy $\sum_{i=1}^{n} a_i x_i \leq W$, where $x_i = 1$ if the i-th item has been picked, and $x_i = 0$ otherwise. Given an instance I of the #KNAPSACK problem with n items, we construct an LIG instance J on a star graph with $n+1$ nodes. Let us label the nodes $v_0, \ldots, v_n$, where v_0 is connected to all other nodes. We define the influence factors among the nodes as follows: the influence of v_0 to any other node v_i, $w_{v_0 v_i} = 1$, and the influence in the reverse direction, $w_{v_i v_0} = -a_i$. The threshold of v_0 is defined as $b_{v_0} = -W$, and the threshold of every other node v_i, $b_{v_i} = 1$. We denote the action of any node v_i by $x_i \in \{0, 1\}$. Note that at any PSNE of J, v_0 must play 1. Otherwise, if v_0 plays 0 then all other nodes must also play 0, and this implies that v_0 must play 1, giving us a contradiction.

We prove that the number of feasible solutions in I is the same as the number of PSNE in J. For any $(x_1, \ldots, x_n) \in \{0, 1\}^n$ in I, we map each x_i to the action selected by v_i in J, for $1 \leq i \leq n$. As proved earlier, the action of v_0 must be 1 at any PSNE. Furthermore, when v_0 plays 1, all other nodes become indifferent between playing 0 and 1. Thus, the number of PSNE in J is the number of ways of satisfying the inequality $\sum_{i=1}^{n} w_{v_i v_0} x_i \geq b_{v_0}$, which is equivalent to $\sum_{i=1}^{n} a_i x_i \leq W$. Thus the number of PSNE in J is equal to the number of feasible solutions in I. □

The following three theorems show the hardness of several other variants of the problem of computing a PSNE of an LIG.

Theorem 4.9. *Given an LIG, along with a designated subset of k players in it, it is NP-complete to decide if there exists a PSNE consistent with those k players playing the action 1.*

Proof. It is easy to see that the problem is in NP, since a succinct yes certificate can be specified by a joint action of the players, where the designated players play 1, and it can be verified in polynomial time whether this is a PSNE or not.

We show a reduction from the monotone one-in-three SAT problem, a known NP-complete problem, to prove that the problem is NP-hard. An instance of the monotone one-in-three SAT problem consists of a set of m clauses and a set of n variables, where each clause has exactly three variables. The problem asks whether there exists a truth assignment to the variables such that each clause has exactly one variable with the truth value of *true*. Given an instance of the monotone one-in-three SAT problem, we construct an instance of LIG as follows (please refer to Fig. 3 for an illustration). For each variable we have a *variable player* in the game, and for each clause we have a *clause player*. Each variable player has a threshold of 0, and each clause player has a threshold of ϵ, where $0 < \epsilon < 1$. We now define the connectivity among the players of the game. There is an arc with weight (or influence) -1 from a variable player u to another variable player v if and only if, in the monotone one-in-three SAT instance, both of the corresponding variables appear together in at least one clause. Also, for each clause t and each variable w appearing in t, there is an arc from the variable player (corresponding to w) to the clause player (corresponding to t) with weight 1. Furthermore, we assign $k = m$, and assume that the designated set of players is the set of clause players. We also assume that the action 1 in the LIG corresponds to the truth value of *true* in the monotone one-in-three SAT problem and 0 to *false*.

Note that the way we constructed the LIG, at most one variable player per clause can play the action 1 at any PSNE. To see this, assume, for contradiction, that at some PSNE two variable players u and v, both connected to the same clause t, are playing the action 1. Then the influence on either of these two variable players is ≤ -1, which is less than its threshold 0, and this contradicts the PSNE assumption. Also, note that at any PSNE, each clause player will play the action 1 if and only if at least one of the variable players connected to it plays 1.

First, we show that if there exists a solution to the monotone one-in-three SAT instance then there exists a PSNE in the LIG where the set of clause players play 1. A solution to the monotone one-in-three SAT problem implies that each clause has the truth value of *true* with exactly one of its variables having the truth value of *true*. We claim that in the LIG, every

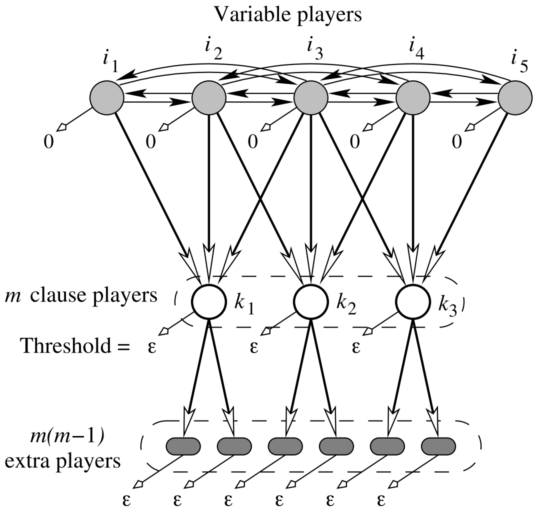


Fig. 4. Illustration of the NP-hardness reduction of Theorem 4.10.

player playing according to its truth assignment, is a PSNE. First, observe that the variable players do not have any incentive to change their actions, since the ones playing 1 are indifferent between playing 0 and 1 (because the total influence $= 0 =$ threshold) and the remaining must play 0 (because the total influence is $\leq -1 <$ threshold). Since each clause has one of its variables playing 1, each clause player must play 1 (because $1 > \epsilon$). This concludes the first part of the proof.

We next show that if there exists a PSNE with the clause players playing 1 then there exists a solution to the monotone one-in-three SAT instance. Consider any PSNE where the clause players are playing 1. Since each clause player is playing 1, *at least* one of the three variable players connected to the clause player is playing 1. Furthermore, as we showed earlier, no two variables belonging to the same clause can play 1 at any PSNE. Thus, for each clause player, *at most* one variable player connected to it is playing 1. Therefore, for every clause player, exactly one variable player connected to it is playing 1. Translating the semantics of the actions to the truth values of the variables and the clauses, we obtain a solution to the monotone one-in-three SAT instance. □

Theorem 4.10. *Given an LIG and a number $k \geq 1$, it is NP-complete to decide if there exists a PSNE with at least k players playing the action 1.*

Proof. Clearly, the problem is in NP, since we can verify whether a joint action is a PSNE or not in polynomial time.

For the proof of NP-hardness, once again we show a reduction from the monotone one-in-three SAT problem. Please see Fig. 4 for an illustration. Given an instance I of the monotone one-in-three SAT problem, we first build an LIG as shown in the proof of Theorem 4.9. We then add $m(m-1)$ additional players, named *extra players*, to the game, where m is the number of clauses in I. Each of these extra players is assigned a threshold of ϵ, where $0 < \epsilon < 1$. The way we connect the extra players to the other players is as follows: From each clause player we introduce $m-1$ arcs, each weighted by 1, to $m-1$ distinct extra players. That is, no two clause players have arcs to the same extra player. Finally, we set $k = m^2$. We denote this instance of LIG by J.

We prove that for any solution to I there exists a PSNE with k players playing 1 in J. Suppose that each of the variable and clause players is playing according to their corresponding truth value in the solution to I. None of the variable players has any incentive to change its action, because exactly one variable player connected to each clause player is playing 1. For the same reason, the clause players, each playing 1, also do not have any incentive to deviate. Considering the extra players, each of these players must play 1, because each of the clause players is playing 1. The total number of clause and extra players is k. Therefore, we have a Nash equilibrium where at least k players are playing 1.

On the other direction, consider any PSNE in J with at least k players playing 1. We claim that all the clause and extra players are playing 1 at this PSNE. If this is not true then at least one of these players is playing 0. This implies that at least one clause player is playing 0, because conditioned on a PSNE, whenever a clause player plays 1, all the extra players connected to it also play 1. Furthermore, by our construction at most one of the variable players connected to each clause player can play 1. So, the total number of players playing 1 is $\leq (m-1)(m+1) < m^2$ (at most $m-1$ clause players are playing 1, and for each of these clause players, $m-1$ extra players, 1 variable player, and the clause player itself are playing 1), which contradicts our assumption that m^2 players are playing 1. Thus, at any PSNE with k players playing 1, it must be the case that every clause player is playing 1. This leads us to a solution for I. □

Theorem 4.11. *Given an LIG and a designated set of $k \geq 1$ players, it is co-NP-complete to decide if there exists a unique PSNE with those players playing the action* 1.

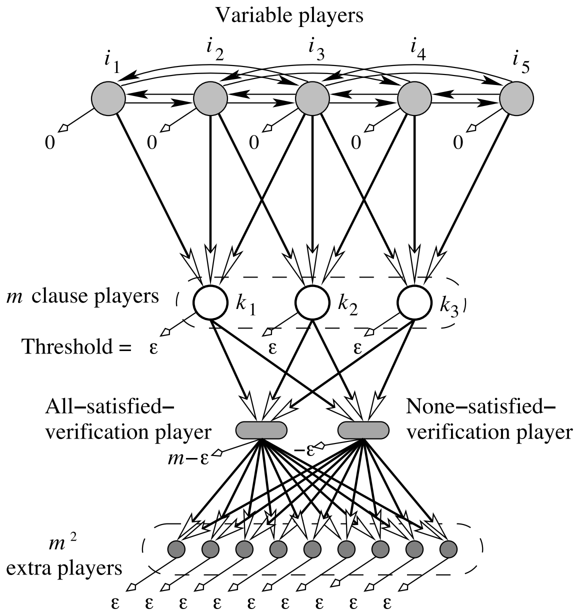


Fig. 5. Illustration of the NP-hardness reduction (Theorem 4.11). For the monotone one-in-three SAT instance of Fig. 3, we first obtain the same construction as in Theorem 4.9. We add two extra players, the all-satisfied-verification player and the none-satisfied-verification player, whose tasks are to verify if all clauses are satisfied and if no clause is satisfied, respectively. These two players are connected to m^2 extra players.

Proof. Two distinct joint actions (PSNE), each having the same k players playing 1, can serve as a succinct no certificate, and we can check in polynomial time if these two joint actions are indeed PSNE or not.

Suppose that I is an instance of the monotone one-in-three SAT problem. We reduce I to an instance J of our problem in polynomial time and show that J has a "no" answer if and only if I has a "yes" answer.

Given I, we start constructing an LIG in the same way as in Theorem 4.9 (see Figs. 5 and 3). Assign $k = m^2$. Now, add two new players, named the *all-satisfied-verification player* and the *none-satisfied-verification player*, which have threshold values of $m - \epsilon$ and $-\epsilon$, respectively. We add arcs from every clause player to these two new players, and the arcs to the all-satisfied-verification player are weighted by 1, and the ones to the none-satisfied-verification player are weighted by -1.

In addition, add $k = m^2$ new players, named *extra players*, and let these players constitute the set of designated players. Assign a threshold value of ϵ to each of these extra players, and introduce new arcs, each with weight 1, from the all-satisfied-verification player and the none-satisfied-verification player to every extra player. The resulting LIG is the instance J of the problem in question.

Note that at any PSNE the all-satisfied-verification player plays 1 if and only if every clause player plays 1, and the none-satisfied-verification player plays 1 if and only if no clause player plays 1. Furthermore, at any PSNE, each extra player plays 1 if and only if either every clause player plays 1 or no clause player plays 1. Therefore, we find that every extra player playing 1, the none-satisfied-verification player playing 1, and every other player playing 0 is a PSNE, and we denote this equilibrium by E_0. We claim that there exists a different PSNE where every extra player plays 1 if and only if I has a solution.

Suppose that there exists a solution S_I to I. It can be verified that making the all-satisfied-verification player play 1, none-satisfied-verification player play 0, every extra player play 1, and choosing the actions of the clause and the variable players according to the corresponding truth values in S_I gives us a PSNE that we call E_1. Thus J has two PSNE E_0 and E_1, where the k extra players play 1 in both cases.

Considering the reverse direction, suppose that there exists no solution to I. This implies that at any PSNE in J all clause players can never play 1, otherwise we could have translated the PSNE to a satisfying truth assignment for I. This further implies that the all-satisfied-verification player always plays 0. The none-satisfied-verification player plays 1 if and only if none of the clause players plays 1. Thus, every extra player plays 1 if and only if no clause player plays 1, if and only if no variable player plays 1. Therefore, E_0 is the only PSNE in J with the k extra players playing 1. □

4.5. Heuristics for computing and counting equilibria

The fundamental computational problem at hand is that of computing PSNE in LIGs. We just saw that various computational questions pertaining to LIGs on general graphs, sometimes even on bipartite graphs, are NP-hard. We now present a heuristic to compute the PSNE of an LIG on a general graph.

A natural approach to finding all the PSNE in an LIG would be to perform a backtracking search. However, a standard instantiation of the backtracking search method [74, Ch. 5] that ignores the structure of the graph would be destined to failure in practice. Thus, we need to order the node selections in a way that would facilitate pruning the search space.

An outline of the backtracking search procedure that we used is given below. Here, the two main additions to the standard backtracking search method are exploiting the graph, including the influence factors, for node selection and implementing constraint propagation by adapting the NashProp algorithm [70] to run in polynomial time.

The first node selected by the procedure is a node with the maximum outdegree. Afterwards, we do not select nodes by their degrees. We rather select a node i that will most likely show that the current partial joint action cannot lead to a PSNE and explore the two actions of i, $x_i \in \{-1, 1\}$ in a suitable order. A good node selection heuristic that has worked well in our experiments is to select the one that has the maximum influence on any of the already selected nodes.

Suppose that the nodes are selected in the order $1, 2, ..., n$ (wlog). After selecting node $i+1$ and assigning it an action x_{i+1}, we determine if the partial joint action $\mathbf{x}_{1:(i+1)} \equiv (x_1, \ldots, x_{i+1})$ can possibly lead to a PSNE and prune the corresponding search space if not. Note that a "no" answer to this requires a proof that one of the players j, $1 \leq j \leq i+1$, can never play x_j according to the partial joint action $\mathbf{x}_{1:(i+1)}$. A straightforward way of doing this is to consider each player j, $1 \leq j \leq i+1$, and compute the quantities $\gamma_j^+ \equiv \sum_{k=1,k\neq j}^{i+1} x_k w_{kj} + \sum_{k=i+2}^{n} |x_k w_{kj}|$ and $\gamma_j^- \equiv \sum_{k=1,k\neq j}^{i+1} x_k w_{kj} - \sum_{k=i+2}^{n} |x_k w_{kj}|$, and then test if the logical expression $((\gamma_j^- > b_j) \wedge x_j = -1) \vee ((\gamma_j^+ < b_j) \wedge x_j = 1)$ holds, in which case we can discard the partial joint action $\mathbf{x}_{1:(i+1)}$ and prune the corresponding search space. Furthermore, it may happen that due to $\mathbf{x}_{1:(i+1)}$, the choices of some of the not-yet-selected players became restricted. To this end, we apply **NashProp** [70] with $\mathbf{x}_{1:(i+1)}$ as the starting configuration, and see if the choices of the other players became restricted because of $\mathbf{x}_{1:(i+1)}$. Although each round of updating the table messages in **NashProp** takes exponential time in the maximum degree in general graphical games, we can show in a way similar to Theorem 4.4 that we can adapt the table updates to the case of LIGs so that it takes polynomial time.

A divide-and-conquer approach

To further exploit the structure of the graph in computing the PSNE, we propose a divide-and-conquer approach that relies on the following separation property of LIGs.

Property 4.12. *Let $G = (V, E)$ be the underlying graph of an LIG and S be a vertex separator of G such that removing S from G results in $k \geq 2$ disconnected components: $G_1 = (V_1, E_1)$, ..., $G_k = (V_k, E_k)$. Let G'_i be the subgraph of G induced by $V_i \cup S$, for $1 \leq i \leq k$. Consider the LIGs on these (smaller) graphs G'_i's, where we retain all the weights of the original graph, except that we treat the nodes in S to be indifferent (that is, we remove all the incoming arcs to these nodes and set their thresholds to 0). Computing the set of PSNE on G'_i's and then merging the PSNE (by performing outer-joins of joint actions and testing for PSNE in the original LIG), we obtain the set of all PSNE of the original game.*

Proof sketch. First, since the joint actions are tested for PSNE in the original LIG, the output will never contain a joint action that is not a PSNE. Second, since the nodes in S are made indifferent in the LIGs on G'_i, $1 \leq i \leq k$, no PSNE of the original LIG can get omitted from the result of the outer-join operation. □

To obtain a vertex separator, we first find an edge separator (using well-known tools such as METIS [45]), and then convert the edge separator to a vertex separator (by computing a maximum matching on the bipartite graph spanned by the edge separator). We then use this vertex separator to compute PSNE of the game in the way outlined in Property 4.12. The benefits of this approach are two-fold: (1) for graphs that have good separation properties (such as preferential-attachment graphs), we found this approach to be computationally effective in practice; and (2) this approach leads to an *anytime algorithm* for enumerating or counting PSNE: Observe that ignoring some edges from the edge separator may result in a smaller vertex separator, which greatly reduces the computation time of the divide-and-conquer algorithm at the expense of producing only a subset of all PSNE. (The reason we obtain a subset of all PSNE is that the edges that are ignored from the edge separator are not permanently removed from the original graph, and that after merging, every resulting joint action is tested for PSNE in the *original* game, not in the game where some of the edges were temporarily removed. As a result, some of the original PSNE may not be included in the final output. At the same time, we can never have a joint action in the final output that is not a PSNE.) We can obtain progressively better results as we ignore less number of edges from the edge separator.

5. Computing the most influential nodes

We now focus on the problem of computing the most influential set of nodes with respect to a specified desired PSNE and a preference for sets of minimal size. In the discussion below, we also assume, *only* for the purpose of establishing and describing the equivalence to the *minimum hitting set problem* [44], that we are given the set of all PSNE. (As we will see, a counting routine is all that our algorithm requires, not a complete list of PSNE.) We give a hypergraph representation of this problem that would lead us to a logarithmic-factor approximation by a natural greedy algorithm.

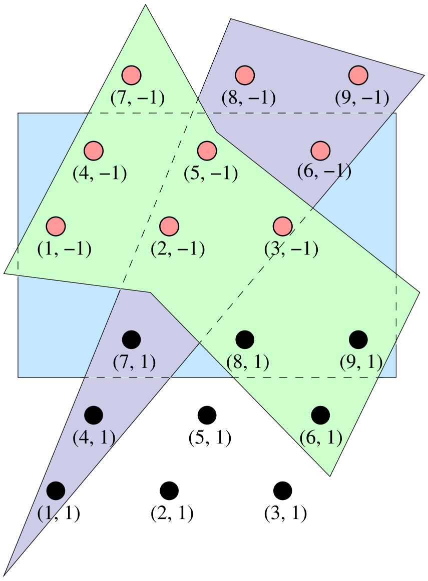


Fig. 6. A hypergraph representation of three PSNE in a 9-player game with binary actions. The PSNE shown here are the followings: $(1,-1,-1,1,-1,-1,1,-1,-1)$ (triangle), $(-1,-1,-1,-1,-1,-1,1,1,1)$ (rectangle), and $(-1,-1,-1,-1,-1,1,-1,1,1)$ (6-gon). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Let us start by building a hypergraph that can represent the PSNE of a binary-action game. The nodes of this hypergraph are the player-action tuples of the game. Thus, for an n-player, binary-action game, we have $2n$ nodes in the hypergraph. That is, for each player i of the game, there are two nodes in the hypergraph: one in which i plays -1 (tuple $(i,-1)$, colored red in Fig. 6) and the other in which i plays 1 (tuple $(i,1)$, colored black). For every PSNE $\mathbf{x}$ we construct a hyperedge $\{(i,x_i) \mid 1 \le i \le n\}$. Let us call the resulting hypergraph the *game hypergraph*. By construction, a set of players S play the same joint-action $\mathbf{a}_S \in \{-1,1\}^{|S|}$ in two distinct PSNE $\mathbf{x}$ and $\mathbf{y}$ of the LIG if and only if both of the corresponding hyperedges $e_{\mathbf{x}}$ and $e_{\mathbf{y}}$ (resp.) of the game hypergraph contains $T = \{(i,a_i) \mid i \in S\}$.

We can use the above property to translate the most-influential-nodes selection problem, given all PSNE, to an equivalent combinatorial problem on the corresponding game hypergraph H. Let $e_{\mathbf{x}^*}$ be the hyperedge in H corresponding to the desired PSNE $\mathbf{x}^*$. Let us call $e_{\mathbf{x}^*}$ the *goal hyperedge*. Then the most-influential-nodes selection problem is the problem of selecting a minimum-cardinality set of nodes $T \subseteq e_{\mathbf{x}^*}$ such that T is contained in no other hyperedge of H (recall that we are dealing with a set-preference function that captures the preference for sets of minimal cardinality). Let us call the latter problem the *unique hyperedge problem*. Using the notation above, the equivalence relationship between the influential nodes selection problem (given the set of all PSNE) and the unique hyperedge problem can be stated as follows. The set $S \subseteq \{1,\dots,n\}$ is a (feasible) solution to the most-influential-nodes selection problem if and only if $T = \{(i, {x_i}^*) \mid i \in S\}$ is a (feasible) solution to the unique hyperedge problem.

We now show that the unique hyperedge problem is equivalent to the minimum hitting set problem. Immediate consequences of this result are that the unique hyperedge problem is not approximable within a factor of $c \log h$ for some constant $c > 0$, and that it admits a $(1 + \log h)$-factor approximation [73,43], where h is the total number of PSNE.

Theorem 5.1. *The unique hyperedge problem having* $2n$ *players and* h *hyperedges is equivalent to the minimum hitting set problem having* n *nodes and* h *hyperedges.*

Proof. Let us consider an instance I of the unique hyperedge problem, given by a game hypergraph $G = (V, E)$, where V is the set of $2n$ nodes and E is the set of h hyperedges, along with a specification of the goal hyperedge $e_{\mathbf{x}^*}$. Given I, we now construct an instance J of the minimum hitting set problem, specified by the hypergraph $G' = (e_{\mathbf{x}^*}, \{e_{\mathbf{x}^*}\} \cup \{\bar{e} \cap e_{\mathbf{x}^*} \mid e \in E \text{ and } e \ne e_{\mathbf{x}^*}\})$, where $\bar{e}$ indicates the complement set of the hyperedge e. Thus, the nodes of G' are exactly the n nodes of $e_{\mathbf{x}^*}$ and the hyperedges of it are constructed from the complement hyperedges of G except $e_{\mathbf{x}^*}$, which is present in both G and G'. We show that a set S of nodes is a feasible solution to I if and only if it is a feasible solution to J.

If S is a feasible solution to I then $S \subseteq e_{\mathbf{x}^*}$ (because in the unique hyperedge problem, we are only allowed to select nodes from the goal hyperedge) and $S \nsubseteq e$ for any hyperedge $e \ne e_{\mathbf{x}^*}$ of G (otherwise, the uniqueness property is violated). This implies that for any hyperedge $e \ne e_{\mathbf{x}^*}$ of G, there exists a node $v \in S$ such that $v \notin e$, which further implies that $v \in \bar{e} \cap e_{\mathbf{x}^*}$. Thus, every hyperedge of G', including $e_{\mathbf{x}^*}$, of course, has at least one of its nodes selected in S, and therefore, S is a feasible solution to J. On the other hand, if S is a feasible solution to J then for any hyperedge of G', at least one of its nodes has been selected in S. That is, for any hyperedge $e \ne e_{\mathbf{x}^*}$ of G, we have $e' \equiv \bar{e} \cap e_{\mathbf{x}^*}$ as the corresponding complementary hyperedge in G', and there exists a node $v \in S$ such that $v \in e'$, which implies that $v \notin e$. Thus, $S \nsubseteq e$

for any hyperedge $e \neq e_{\mathbf{x}^*}$ of G. Furthermore, we have selected all the nodes of S from $e_{\mathbf{x}^*}$ of G. Thus, $e_{\mathbf{x}^*}$ is the unique hyperedge of G containing the nodes of S.

To prove the reverse direction, we start with an instance J of the minimum hitting set problem, specified by a hypergraph $G' = (V, E)$, where V is a set of n nodes and E is a set of h hyperedges. Without the loss of generality, we assume that E contains the hyperedge e^* consisting of all the nodes of V. We now construct an instance I of the unique hyperedge problem that has a hypergraph G with $2n$ nodes and h hyperedges. The node set of G literally consists of two copies of the nodes of V, denoted by $V \times \{1, -1\}$. We now construct the hyperedges of G. For each hyperedge $e \neq e^*$ of the minimum hitting set instance, we include a hyperedge $e' \equiv \bar{e} \times \{1\} \cup e \times \{-1\}$ in G, and for the hyperedge e^* of J, we include the hyperedge $e^* \times \{1\}$ in G. Thus, the game hypergraph can be defined as $G = (V \times \{1, -1\}, \{e^* \times \{1\}\} \cup \{\bar{e} \times \{1\} \cup e \times \{-1\} \mid e \in E \text{ and } e \neq e^*\})$. Finally, we designate $e^* \times \{1\}$ as the goal hyperedge of I. We will show that $S \subseteq V$ is a feasible solution to J if and only if $S \times \{1\}$ is a feasible solution to the unique hyperedge problem instance I. The set S is a feasible solution to J if and only if for every hyperedge $e \neq e^*$ of G', there exists a node $v \in S$ such that $v \in e$ (note that $S \subseteq e^*$). This is equivalent to saying that for every hyperedge $e \times \{1\} \neq e^* \times \{1\}$ of G, there exists a node $v \in S \times \{1\}$ such that $v \notin e \times \{1\}$. Using the fact that $S \times \{1\} \subseteq e^* \times \{1\}$, $S \times \{1\}$ is a feasible solution to I. □

The adaptation of the well-known hitting set approximation algorithm [4] for our problem can be outlined as follows: At each step, select the least-degree node v of the goal hyperedge, remove the hyperedges that do not contain v, remove v from the game hypergraph, and include v in the solution set, until the goal hyperedge becomes the last remaining hyperedge in the hypergraph. In the context of the original LIG, at every round, this algorithm is essentially picking the node whose assignment would reduce the set of PSNE consistent with the current partial assignment the most. Hence, the algorithm only requires a subroutine to *count* the PSNE extensions for some given partial assignment to the players' actions, not an *a priori* full list or enumeration of all the PSNE. Of course, it may require a complete list of PSNE in the worst case.

6. Experimental results

We performed empirical studies on several types of LIGs, namely, (1) random LIGs (Erdös–Rényi and uniformly random), (2) preferential-attachment LIGs, and (3) LIGs created to model potential interactions in two different real-world scenarios: interactions among U.S. Supreme Court justices and those among U.S. senators. While the first two types of LIGs are synthetic/artificial, the latter two are the result of inferring the LIGs from real-world data using machine learning techniques [39].

The reason for experimenting with synthetic LIGs using Erdös–Rényi [21] and uniformly random graphs is that those types of network-generation techniques are the most basic available. They serve as precursors to the more sophisticated preferential-attachment graphs, which capture the heavy-tailed degree distribution often observed in real-world networks [1].

Here is our overall plan for this section. For the synthetic games (random LIGs and preferential-attachment LIGs), we generate game instances by varying the appropriate parameters, such as the size of the game, and evaluate both the number of PSNE of these games and the computation time of our algorithm. We also compute the most influential nodes in these games using our approximation algorithm and compare it to the optimal (i.e., minimum-cardinality) set of most-influential nodes. For the real-world games on Supreme Court rulings and congressional voting, we discuss how we learn these games from data [39], and how we compute the set of PSNE and identify the most influential nodes. For the congressional voting case, we also *adapt* the simple, greedy selection approximation algorithm to identify most-influential individuals using the cascade model in the diffusion setting, as described by Kleinberg [49], to use as a *heuristic* to solve *specific instances* of the most-influential-nodes problem formulation in our context. We compare and contrast the output of our approximation algorithm and that of the diffusion-based heuristic resulting from the adaptation.

6.1. Random influence games

We began our experiments by generating instances of random graphs using the Erdös–Rényi model [21]. We varied the number of nodes, from 10 to 30, and the probability of including an edge. Assuming binary actions, 1 and −1, we chose the threshold b_i and the influence factors w_{ji} of the incoming arcs of each node i uniformly at random from a unit hyperball. That is, for each node i, $b_i^2 + \sum_{j \in N(i)} w_{ji}^2 = 1$, where $N(i)$ is the set of nodes having arcs toward i. Then, we chose the sign of each threshold, as well as each weight, to be either + or − with 0.5 probability. We applied the heuristic given earlier to find the set of all PSNE in these random graphs. Our experimental results show that in all of these random LIGs, the number of PSNE is almost always very small—usually one or two, and sometimes none.

We also studied LIGs on uniformly random directed graphs. While constructing the random graphs, we have independently chosen each arc with probability 0.50, and assigned it a weight of −1 with probability p (named *flip probability*) and 1 with probability $1 - p$. Several interesting findings emerged from our study of this parameterized family of LIGs on uniformly random graphs. Appendix E summarizes the results in tabular form. For various flip probabilities, we independently generated 100 uniformly random graphs of 25 nodes each. For each of these random graphs, we first computed all PSNE using our heuristic. We then applied the greedy approximation algorithm to obtain a set of the most influential nodes in each graph and compared the approximation results to the optimal set.

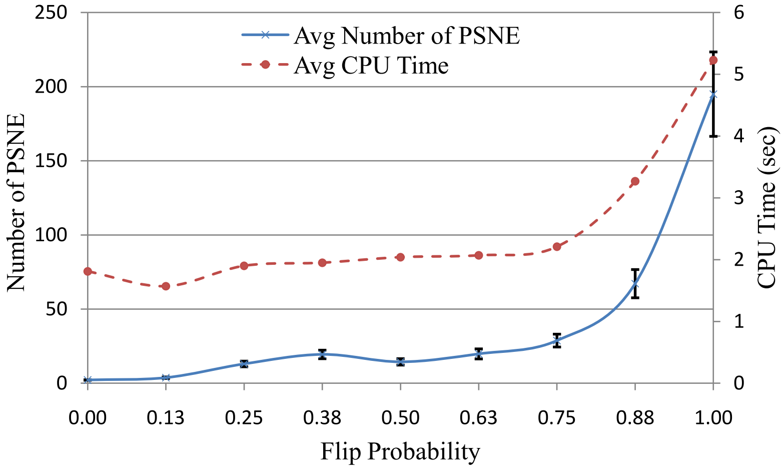


Fig. 7. PSNE computation on random LIGs. The vertical bars denote 95% confidence intervals.

Unless p is either 0 or 1, one cannot guarantee the existence of a PSNE. In our experiments, we found that, in fact, for $p = 0.50$, the probability of not having a PSNE is highest (around 5%), and as we go toward the two extremes of p, the probability of not having a PSNE decreases. We report only the games with at least one equilibrium in this experimental study, because it is these games we care about for computing the most influential nodes. Another interesting finding about the number of PSNE is that it is very small when $p = 0$, that is when all the arcs have weight 1, and it is large when $p = 1$, although quite small (on average, a fraction $5.81 \times 10^{-6} \approx 2^{-17.29}$) relative to the total number of 2^{25} possible joint actions. Also, the average number of nodes of the search tree that the backtracking method visits per equilibrium computation is relatively small on the two extremes of p, compared with p around 0.5. Note that the backtracking method does a very good job with respect to the number of search-tree nodes visited in searching the 2^{25} space. In fact, our experiments show that the addition of the **NashProp**-based heuristic on top of the node selection heuristic considerably speeds up the search. Finally, we found that although the approximation algorithm has a logarithmic factor worst-case bound, the results of the approximation algorithm are most often very close to the optimal solution.

Fig. 7 shows that the number of PSNE usually increases if we have more negative-weighted arcs than positive ones; although the number of PSNE is still very small relative to the maximum potential number, as remarked earlier. We also found once again that although the approximation algorithm for the influential-nodes selection problem has a logarithmic factor worst-case bound, the result of the approximation algorithm is most often very close to the optimal solution. For example, for the random games having all negative influence factors, in 87% of the trials the approximate solution size $\leq$ optimal size $+1$, and in 99% of the trials the approximate solution size $\leq$ optimal size $+2$ (see Appendix E for more details in a tabular form).

6.2. *Preferential-attachment LIGs*

We also experimented with LIGs based on preferential-attachment graphs primarily because of its power to explain the structure of many real-world social networks in a generative fashion [1]. In order to construct these graphs, we started with three nodes in a triangle and then progressively added each node to the graph, connecting it with three existing nodes with probabilities proportionate to the degrees. We made each connection bidirectional and imposed the same weighting scheme as above: with flip probability p, the weight of an arc is -1 and with probability $1 - p$ it is 1. We set the threshold of each node to 0. We observe that for $0 < p < 1$, these games have very few PSNE, while for $p = 0$ and $p = 1$ the number of PSNE is considerably larger than that. Furthermore, these games show very good separation properties, making the computation amenable to the divide-and-conquer approach. We show the average number of PSNE and the average computation time for graphs of sizes 20 to 50 nodes in Fig. 8 for $p = 1$ (each average is over 20 trials). Note that in contrast to uniformly random LIGs, preferential-attachment graphs show an exponential increase in the number of PSNE as the number of nodes increase, although the number of PSNE is still a very small fraction of the maximum potential number.

6.3. *Illustration: supreme court rulings*

We used our model to analyze the influence among the justices of the U.S. Supreme Court. The Supreme Court of the United States (SCOTUS),[11] is the highest federal court of the judicial branch of the government. (The other branches of government are the executive branch, lead by the President, and the legislative branch, represented by Congress.) The

[11] http://www.supremecourtus.gov.

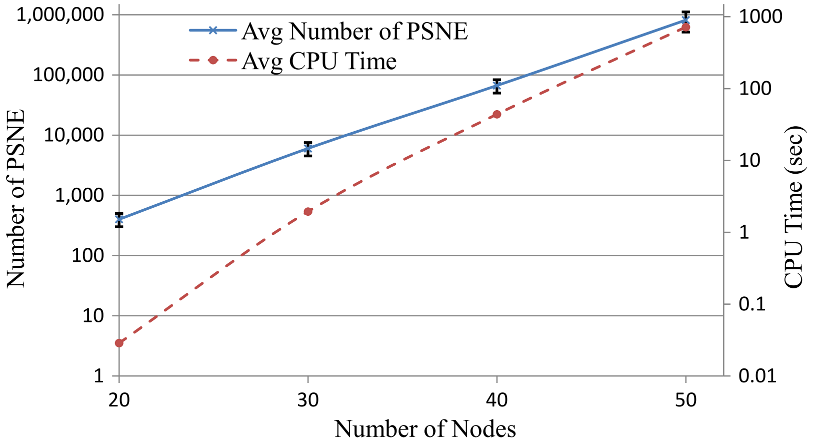


Fig. 8. PSNE computation on preferential-attachment LIGs (y-axis is in log scale). The vertical bars denote 95% confidence intervals.

Table 1
Interpretation of votes.

varVote	Original meaning	Our interpretation
1	Voted with majority	Yes
2	Dissent	No
3	Regular concurrence	Yes
4	Special concurrence	Yes
5	Judgment of the Court	Yes
6	Dissent from a denial or dismissal of certiorari, or dissent from summary affirmation of an appeal (Interpreted as absent from voting in final outcome)	Majority
7	Jurisdictional dissent (Interpreted as absent from voting in final outcome)	Majority
8	Justice participated in an equally divided vote	–

SCOTUS is the main interpreter of the Constitution and has final say on the constitutionality of any federal law created by the legislative branch or any action taken by the executive branch. It consists of nine justices—a chief justice and eight associate justices. We chose to study the SCOTUS in the context of influence games because this is an application domain where the strategic aspects of influence seem of prime importance.

There are two distinctive features that make our approach particularly suitable to the SCOTUS domain. First, we can model the individual outcomes (in this case, the decisions of the justices on each case) as outcomes of a one-shot non-cooperative game (an LIG in our case). Second, the physical interpretation of the diffusion process is not as clear in this setting as it is in applications like viral marketing.

6.3.1. Data

We obtained data from the Supreme Court Database.[12] Although the database captures fine-grained details of the cases, we only focused on the variable varVote. Again, the votes of the justices are not simple yes/no instances. Instead, each vote can have eight distinct values. However, for practical purposes, we can attach a simple yes/no interpretation to the values of the votes, as shown in Table 1.

In Table 1, “majority” in the third column signifies that we interpreted the corresponding justice’s vote as yes or no, whichever occurs most among the other justices. Also, among the natural courts we studied, we did not encounter voting instances where varVote has a value of 8.

We now present our study of the natural court (with timeline 1994–2004) comprising of Justices WH Rehnquist, JP Stevens, SD O’Connor, A Scalia, AM Kennedy, DH Souter, C Thomas, RB Ginsburg, and SG Breyer.

6.3.2. Learning LIG

The data for the above natural court consists of 971 voting instances (each voting instance consists of the votes of all nine justices). Many instances (i.e., voting patterns) appear repeatedly in the data set, of course. For example, the most repeated instance consists of all the justices voting yes, occurring 438 times. The second most repeated instance, occurring 85 times, consists of five of the justices, namely, Justices Scalia, Thomas, Rehnquist, O’Connor, and Kennedy voting yes, and

[12] http://scdb.wustl.edu/.

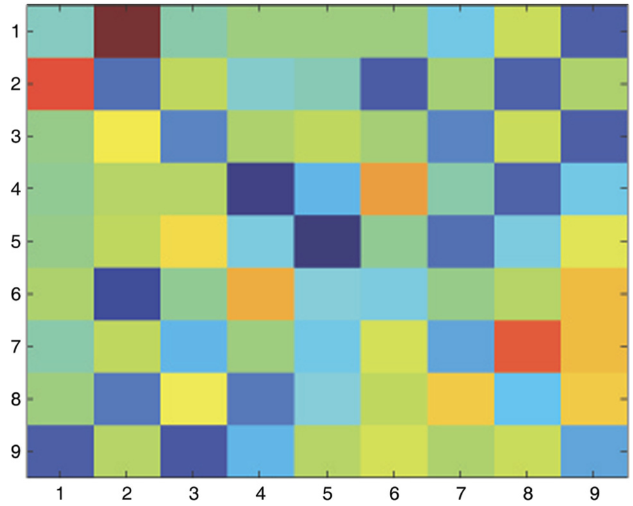


Fig. 9. Pictorial representation of LIG learned from data—the non-diagonal elements represent influence factors and the diagonal elements biases. The numbering of the players (from 1 to 9) corresponds to the justices in this order: Justices A Scalia, C Thomas, WH Rehnquist, SD O'Connor, AM Kennedy, SG Breyer, DH Souter, RB Ginsburg, and JP Stevens. The darker the color of a cell, the more negative the corresponding number. For example, the most negative number (-0.2634) occurs in cell $(5, 5)$ (i.e., the bias of Justice Kennedy). The most positive number (0.4282) occurs in cell $(1, 2)$ (i.e., the influence factor from Justice Scalia to Justice Thomas) and the number closest to zero is 0.001 in cell $(2, 4)$ (i.e., the influence factor from Justice Thomas to Justice Kennedy).

the remaining justices voting no. We used L_2-regularized logistic regression (simultaneous classification) to learn an LIG from this data. The influence factors and the biases of the learned LIG appear in tabular form in Appendix E. Fig. 9 shows a pictorial representation of the same LIG.

The learned LIG represents 589 of the 971 voting instances as PSNE. As expected, it represents the frequently repeated voting instances (such as the ones mentioned above). Fig. 10 shows a graphical representation of the LIG. We clustered the nodes based on the traditional perception that Justices Scalia, Thomas, Rehnquist, and O'Connor are "conservative;" Justices Breyer, Souter, Ginsburg, and Stevens are "liberal;" and Justice Kennedy is a "moderate." As illustrated in Fig. 10, negative influence factors occur only between players of two different clusters.

6.3.3. *Most influential nodes*

Analysis of the PSNE of this LIG shows that there is a set of two nodes that is "most influential" with respect to achieving the objective of every justice voting yes. This most-influential set consists of one node from the set {Scalia, Thomas} and another one from the set {Breyer, Souter, Ginsburg, Stevens}. Furthermore, any one node from the set {Breyer, Souter, Ginsburg, Stevens} is alone most-influential with respect to achieving the objective of a 5-4 vote mentioned above (i.e., the second most repeated instance in the data).

6.4. *Illustration: congressional voting*

We further illustrate our computational scheme in another real-world scenario—the U.S. Congress—where the strategic aspects of the agents' behavior are also of prime importance. We particularly focus on the U.S. Senate, which consists of 100 senators; two senators for each of the 50 U.S. states. Together, they form the most important unit of the legislative branch of the U.S. Federal Government.

We first learned the LIGs among the senators of the 101st and the 110th U.S. Congress [39]. The 101st Congress LIG consists of 100 nodes, each representing a senator, and 936 weighted arcs among these nodes. On the other hand, the 110th Congress LIG has the same number of nodes, but it is a little sparser than the 101st one, having 762 arcs. In these LIGs, each node can play one of the two actions: 1 (yes vote) and -1 (no vote). Fig. 11 shows a bird's eye view of the 110th Congress LIG, while Fig. 12 shows a magnified part of it.

First, we applied the divide-and-conquer algorithm that exploits the nice separation properties of these LIGs, to find the set of all PSNE (we precompute the whole set of PSNE for convenience; as discussed earlier, counting alone is sufficient). We obtained a total of 143,601 PSNE for the 101st Congress graph and 310,608 PSNE for the 110th. Note that the number of PSNE in these games is extremely small (e.g., a fraction $2.45 \times 10^{-25} \approx 2^{-81.76}$ for the 110th Congress) relative to the maximum possible 2^{100} joint actions. Regarding computation time, solving the 110th Congress using the divide-and-conquer approach takes about seven hours, whereas solving the same without this approach, simply relying on the backtracking search, takes about 15 hours on a modern quad-core desktop computer.

Next, we computed the most influential senators using the approximation algorithm outlined at the end of Section 5. We obtained a solution of size five for the 101st Congress graph, which we verified as an optimal solution. This solution consists

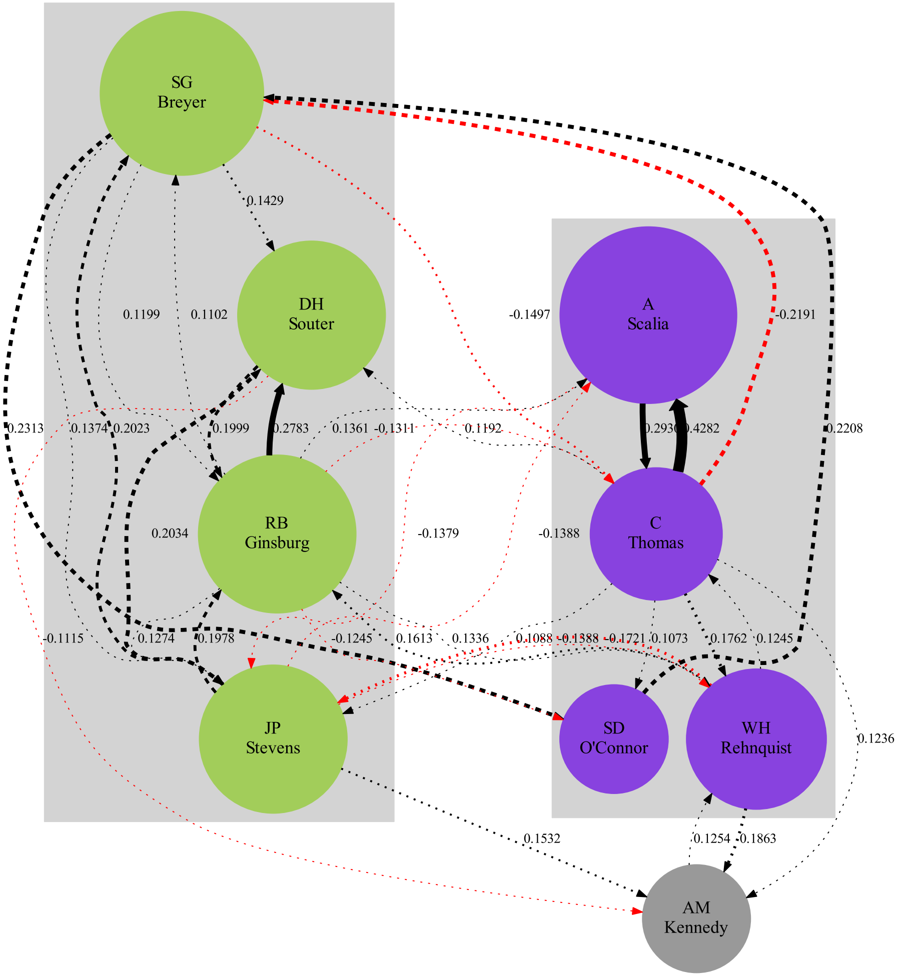


Fig. 10. Graphical representation of LIG learned from data. Larger node sizes indicate higher thresholds (more stubborn). Black and red arcs indicate positive or negative influence factors, respectively. While the learned LIG is a complete graph, we are only showing approximately half of the arcs (i.e., we are not showing the "weakest" arcs in this graph). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

of Senators Rockefeller (Democrat, WV), Sarbanes (Democrat, MD), Thurmond (Republican, SC), Symms (Republican, ID), and Dole (Republican, KS). Interestingly, none of the maximum-degree nodes were selected. Similarly, the six most influential senators of the more recent 110th Congress (January 2007–January 2009) are Kerry (Democrat, MA), Bennett (Republican, UT), Sessions (Republican, AL), Enzi (Republican, WY), Rockefeller (Democrat, WV), and Lautenberg (Democrat, NJ).

We also applied our technique to the more recent 112th Congress, using voting data from May 9, 2011 to August 23, 2012. A set of the most influential senators with respect to the outcome of everyone voting "yes" consists of Senators Reid (Democrat, NV), Inouye (Democrat, HI), Johnson (Republican, WI), Sanders (Independent, VT), Hagan (Democrat, NC), Collins (Republican, ME), Crapo (Republican, ID), DeMint (Republican, SC), Reed (Democrat, RI), and Barrasso (Republican, WY). Note that the set of most-influential senators in the 112th Congress consists of 10 senators, whereas it consists of only six senators in the earlier Congresses that we studied. This implies that, according to our model, for the 112th Congress, we now need a broader group of "influencing" senators to lead everyone to a consensus. This is consistent with the contemporary perception of polarization in Congress, which has been highlighted both in the mainstream media and formal research studies in recent times [65].

Besides identifying the most influential senators with respect to passing a bill (e.g., the outcome of everyone voting "yes"), we can also apply our model to study the other extreme of *not* passing a bill (e.g., the outcome of everyone voting "no"). According to our model, we can achieve the latter outcome in the 112th Congress if the following 10 senators choose to vote "no": Senators Nelson (Democrat, FL), Cardin (Democrat, MD), Klobuchar (Democrat, MN), Reed (Democrat, RI), Murkowski (Republican, AK), Moran (Republican, KS), Vitter (Republican, LA), Enzi (Republican, WY), Crapo (Republican, ID), and DeMint (Republican, SC).

Later on, we will show that besides the desired outcomes of everyone voting "yes" or everyone voting "no," our model and approach extend to studying even more general outcomes, such as breaking filibusters or preventing clotures.

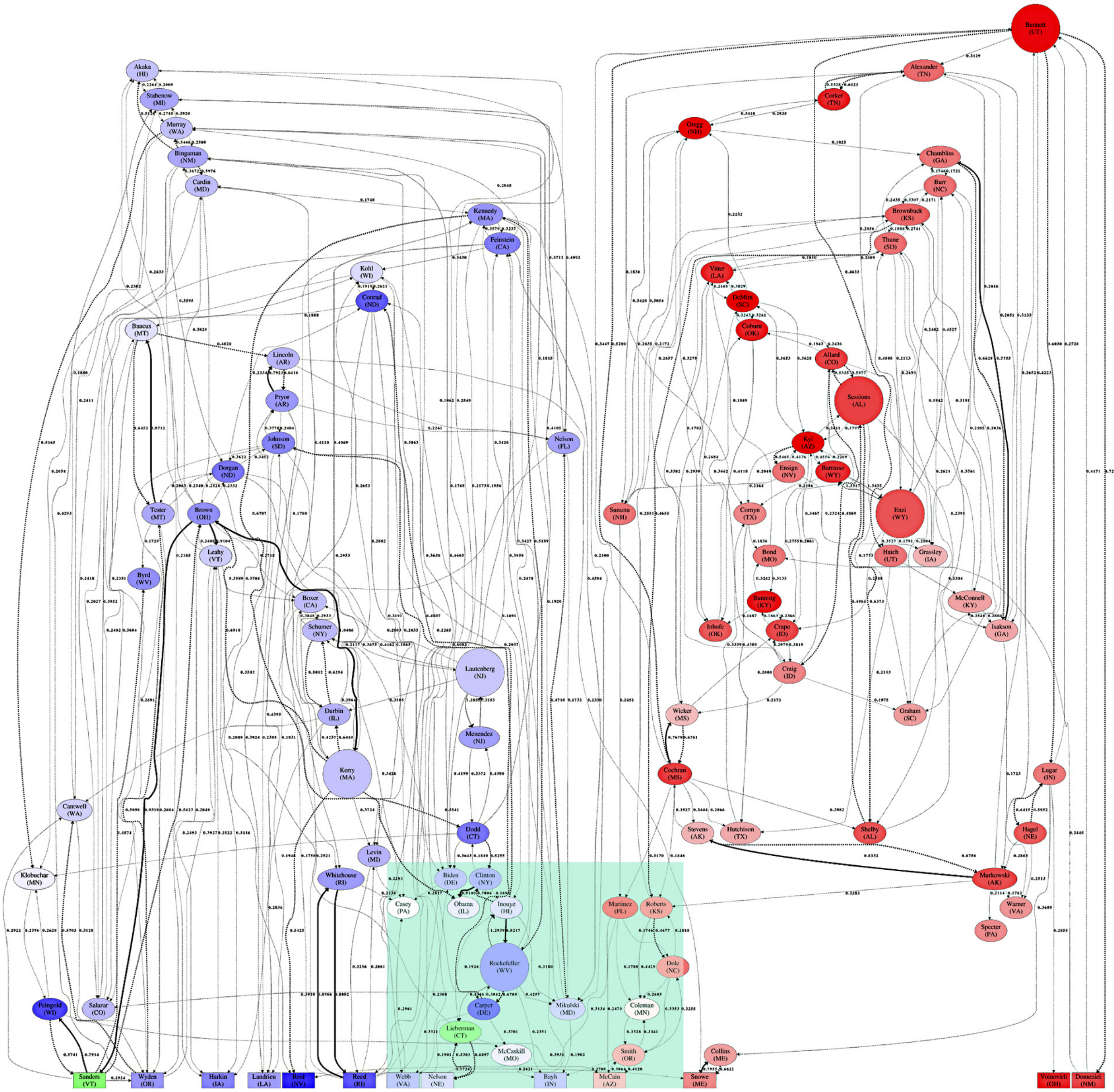


Fig. 11. LIG for the 110th U.S. Congress: darker color of nodes represent higher threshold (more stubborn); thicker arcs denote influence factors of higher magnitude (only half of the original arcs with the highest magnitude of influence factors are shown here); circles denote most-influential senators; rectangles denote cut nodes used in the divide-and-conquer algorithm. Fig. 12 shows the shaded part for better visualization.

6.4.1. *Adapting a popular diffusion-based algorithm as a heuristic for the most-influential-nodes problem instance in LIGs*

Our one-shot non-cooperative game-theoretic approach is fundamentally different from the diffusion approach, both syntactically (i.e., mathematical foundations) and semantically (e.g., interpretation of objectives, nature of problem formulations, and applicable domains and contexts). Of course, the most obvious key difference is that our *solution concept* does not consider dynamics. We concentrate on "end-state" behavior and characterize *stable* outcomes using the notion of PSNE, which is "static" by nature.[13] As we reviewed in Section 2.2, many of the diffusion-based approaches lead to the outcomes that are not stable.[14]

[13] Note that PSNE may, or may not, be reachable by (possibly networked) best-response dynamics.

[14] The reader should bear in mind that the intention behind our statements contrasting the approaches/models is simple: to highlight some of the differences. We do not mean to imply that one approach or model is "better" than the other: they are just different and largely incomparable, each with its own "pros and cons" depending on the problem, context, or domain of interest. In other words, each has its own place.

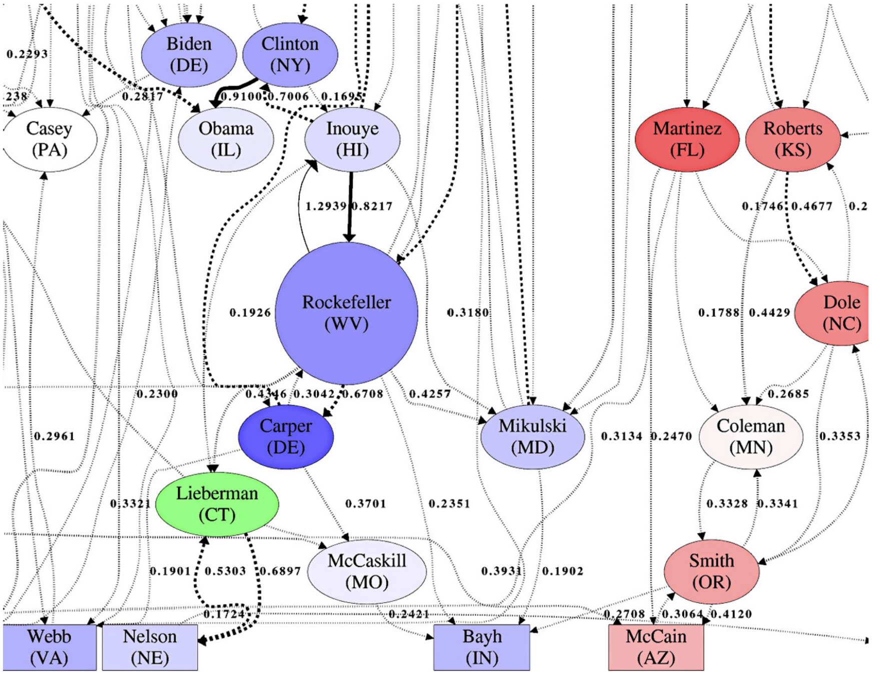


Fig. 12. A part of the LIG for the 110th U.S. Congress: blue nodes represent Democrat senators, red Republican, and white independent; darker color of nodes represent higher threshold (more stubborn); thicker arcs denote influence factors of higher magnitude; circled node (Senator Rockefeller) denotes one of the most influential senators; rectangles at the bottom denote cut nodes used in the divide-and-conquer algorithm. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Yet, out of purely scientific curiosity and suggestions/feedback about our work, we present preliminary results based on a *heuristic* for the problem of most influential nodes in our context (i.e., LIGs) that we designed by *adapting* what is arguably the most popular and simple greedy-selection algorithm for identifying "most influential" nodes developed specifically for the cascade model of diffusion [49]. The simplicity of the original algorithm in the diffusion context facilitates the adaptation to our context. Note however that this adaptation is exclusive to a *very specific instance of our general problem formulation*: identifying "most influential nodes" *with respect to the goal being maximizing the number of* $+1$*'s in the PSNE* (see Definition 3.6 in Section 3.2 for a definition and discussion of our problem formulation and the role that the "objective function" $g(.)$ plays in the formulation).

We realize that, inspired mostly by the *cascade model* described in Kleinberg [49], the literature on diffusion models for and approaches to problems related to "influence maximization" and "minimization" has increased considerably since the early groundbreaking work of Kempe et al. [48] and particularly over the last few years.[15] We reemphasize that all that work uses a diffusion-based approach and thus fundamentally differs from the work presented here, both in terms of the problem definition and the solution approach. As a result, comparing the result of such disparate approaches is not scientifically meaningful, in our view. Furthermore, it would be nontrivial and out-of-scope for this paper to *adapt* the techniques used or proposed to solve their specific variations of the "influence maximization" or "minimization" problem in a diffusion-based setting, to employ as *heuristics* to solve our problem; just as we would not expect adaptations of our techniques as heuristics to solve *their* problems.

Therefore, *in this paper, we only adapt the popular, greedy algorithm used to find the "most influential individuals" in the cascade model, as described in Kleinberg [49], to use as a heuristic to solve what would be the equivalent/analogous instance of that problem in the more general problem formulation in our setting* presented in Section 3.2. We remind the reader that that greedy algorithm, originally meant for the particular cascade model of diffusion, is arguably the most simple and fundamental of the work in that area. The resulting heuristic corresponds to a very "controlled" version of (possibly networked) *best-response dynamics* on the influence game of interest starting from specific initial conditions.

6.4.1.1. Diffusion-based heuristic We first note that we use the same influence factors and thresholds of the previously learned LIGs to perform both of these analyses. In particular, for the diffusion-based heuristic, at each iteration, we select a node u that achieves the "maximum spread" of action 1, by which we mean the node u that leads to the largest (marginal difference in the) number of other players in the network with action 1 after performing best-response dynamics

[15] See, for example, Budak et al. [12], Chen et al. [14], He et al. [34] and the references therein, for variations of the basic model, problem and techniques. A survey by Guille et al. [33] also contains latest developments.

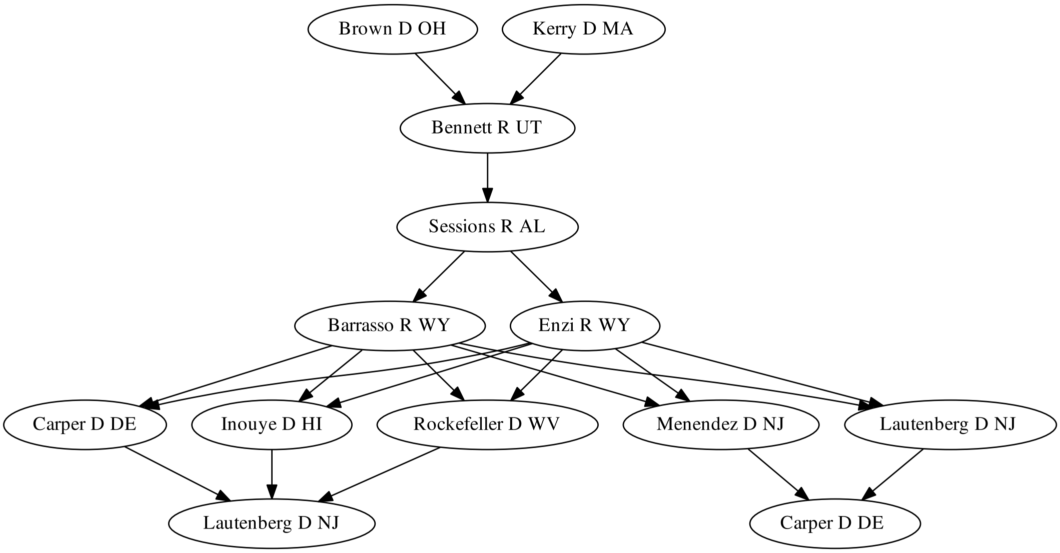


Fig. 13. Most-influential nodes in our setting, computed using our approximation algorithm. This directed acyclic graph (dag) illustrates all possible options for node selection that our approximation algorithm considers. A source node represents a node selected in the first iteration and a sink node represents a node selected in the last step. Any directed path from a source to a sink represents a sequence of nodes selected in successive iterations by our algorithm. All nodes in the same level and having the same parent, are tied in an iteration of the algorithm. Also note that the same node can appear in different paths of the dag at different levels.

in the LIG; force u to adopt action 1; and let all but the previously selected nodes modify their actions as best responses to u's adoption of action 1. We repeat this process until every node adopts action 1.[16]

Note that because of negative influence factors, cycling may occur and this procedure may never come to a stop. However, in our case, even in the presence of negative influence factors, we did not encounter such cycling.

Furthermore, it is well known that the general greedy-selection recipe just described produces a provable approximation algorithm in the cascade model with submodular spread function [49]. But given the presence of negative influence factors in our LIG and, more importantly, the fact that we do not even know what the corresponding “submodular spread function” is or means in our setting, *this claim of approximation guarantee essentially vanishes or is irrelevant in our setting.*

6.4.1.2. Results using diffusion-based heuristic We can visualize all possible choices of the most influential nodes that an algorithm can make as a directed acyclic graph, as shown in Figs. 13 and 14.

Although Fig. 13 looks more complicated than Fig. 14 (due to the appearance of the same node in different source-sink paths of the dag at different levels), comparing them we find that, not only a set of six nodes are most-influential in both cases, but also most of the nodes are common between these two distinct algorithms. More remarkably, some of these common nodes are selected at the same iteration in both algorithms. The obvious question arises, is a set of most-influential nodes in the LIG setting, as output by our approximation algorithm, also a possible output of the diffusion-based heuristic? We have exhaustively tested all possible sets of the most influential nodes (Fig. 13) and settled the answer in the negative for each set. Interestingly, if we add the “Alexander R TN” node to *any* of the most influential sets in the LIG setting, the resulting set can be the output of the diffusion-based heuristic. The apparent similarity in results between the output of our approximation algorithm and the diffusion-based heuristic gives rise to an intriguing open question as to the characterization of the exact connection between these two seemingly different algorithms for identifying this particular instance (i.e., all players play 1) of the most-influential problem in our setting. This open question is beyond the scope of this paper.

6.5. Filibuster

Beyond predicting stable behavior and identifying the most influential nodes in a network, we can also use our model to study other interesting aspects of a networked population. One example is the filibuster phenomenon in the U.S. Congress, where a senator uses his or her right to hold floor for an indefinite time in an effort to delay the passing of a bill. The procedure of “cloture,” which refers to gathering a majority of at least 60 votes among the current 100 senators, can break a filibuster. However, not every possible cloture scenario of 60 or more “yes” votes may be a *stable* outcome due to influence among the senators. We call the set of such outcomes that are indeed stable in the sense of PSNE a *stable cloture set*.

[16] At the end, we also perform a post-processing step, where we try to remove one of the selected nodes to test if the remaining nodes are still most-influential.

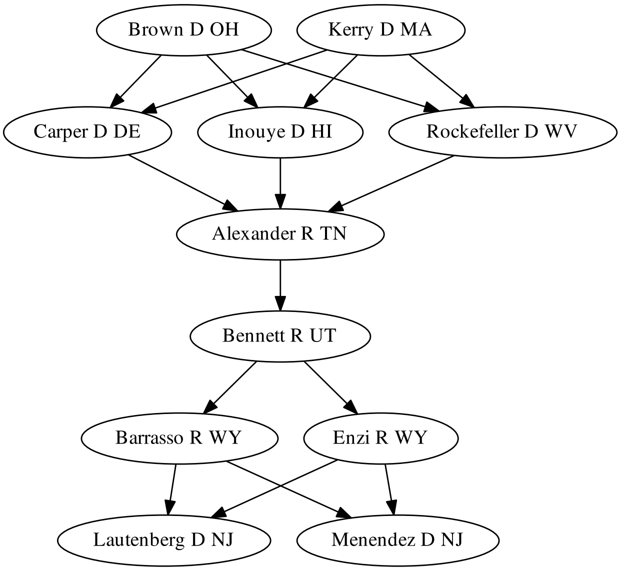


Fig. 14. Most-influential nodes as output by the diffusion-based heuristic. Each directed path from a source to a sink represents a sequence of nodes that are most-influential, as determined by applying the variant of the standard greedy algorithm used for traditional diffusion models to identify the most influential nodes (described in the main body).

An interesting general question is whether there exists a small coalition of senators that can break filibusters. We can think of preventing a filibuster from the democratic or the republican perspective (i.e., favoring the respective party). Similarly, we can also ask what in some sense is the opposite question: is there a small coalition of senators that can *prevent clotures* by voting "no?" First, let us formally define the problem for breaking filibusters; the formulation of the problem for preventing clotures is similar.

6.5.1. Problem formulation

Given the set $\mathcal{S}$ of all stable outcomes (i.e., PSNE) and a subset of $\mathcal{C}$ of these stable outcomes, find a minimal set T of players such that

$$T \in \operatorname*{arg\,max}_{V \subset \{1,\dots,n\}} \bigl\{ \bigl| P_{\mathcal{S}}(V) \bigr| \bigm| P_{\mathcal{S}}(V) \subseteq \mathcal{C} \bigr\},$$

where $P_{\mathcal{S}}(V)$ is the set of PSNE-extensions of the nodes in V playing action 1, i.e.,

$$P_{\mathcal{S}}(V) = \{\mathbf{x} \mid \mathbf{x} \in \mathcal{S},\ x_i = 1\ \forall i \in V\}.$$

In words, $\mathcal{C}$ is the stable cloture set, consisting of stable outcomes that can prevent a filibuster (i.e., every PSNE in $\mathcal{C}$ contains at least 60 "yes" votes and thus, can induce a cloture). When we consider the notion of preventing a filibuster in favor of a specific party, we define $\mathcal{C}$ consisting of exactly those PSNE that contain 60 or more "yes" votes (thereby representing cloture scenarios) and in addition, are supported (through "yes" votes) by the majority of the senators affiliated with that party. Other definitions are possible, as long as the stable cloture set $\mathcal{C}$ is well-defined.

Now, we would like to select a *minimal* set of senators such that $\mathcal{C}$ contains the set $P_{\mathcal{S}}(V)$ of the PSNE-extensions of these senators' voting "yes" (i.e., their voting "yes" can only lead to a stable cloture scenario, thereby preventing a filibuster). In addition, we would also like to achieve a maximum *stable-cloture cover*; that is, we wish to achieve the maximum possible set $P_{\mathcal{S}}(V)$ so that we are able to capture as many of the stable cloture scenarios as possible. In this formulation, we set up as the objective to select a minimal, not minimum, set of senators; this keeps the formulation simple by avoiding bicriteria optimization (minimum set of senators vs. maximum stable-cloture cover). Further note that adding an extra senator to the set of selected senators can only reduce the stable-cloture cover because of additional constraints.

The problem formulation above guarantees a nonempty solution T if there exists some PSNE in $\mathcal{C}$ that is not "dominated" by any PSNE in $\mathcal{S} \setminus \mathcal{C}$. Here, a PSNE $\mathbf{x}$ dominates another PSNE $\mathbf{y}$ if for every i, $y_i = 1 \Longrightarrow x_i = 1$.

6.5.2. A heuristic

We can modify the approximation algorithm for identifying the most influential nodes, presented at the end of Section 5, to design a heuristic for the problem formulated above in the following way. At each iteration, we select a node such that adding it to the set of already selected nodes minimizes the number of PSNE-extensions of the selected nodes playing 1 that are in $\mathcal{S} \setminus \mathcal{C}$. If there is a tie among several nodes in this step, then we can store these nodes in order to explore all solutions that this heuristic can produce. We stop when the above number of PSNE-extensions within $\mathcal{S} \setminus \mathcal{C}$ goes to 0. We then perform a minimality test by excluding nodes from the selected set of nodes and testing whether the resulting set can be a solution. Note that although we can select the "best" solution (in terms of the coverage of $\mathcal{C}$) among the ones found due to ties, this heuristic does not guarantee an approximation of the maximum coverage of $\mathcal{C}$.

6.5.3. Experimental results on the 110th Congress

For the 110th Congress, $\mathcal{C}$ consists of 15,288 and 10,029 stable cloture scenarios (i.e., PSNE) with respect to the democratic and republican parties, respectively. Overall, the total number of stable cloture scenarios is 15,595, and most of these are common in both democratic and republican cases.

Regarding breaking filibusters with respect to the democratic party, the best solutions found by the above heuristic are Senators {Brown (D, OH), Roberts (R, KS), and Graham (R, SC)} and {Kerry (D, MA), Roberts (R, KS), and Graham (R, SC)}, both of which cover 1500 of the 15,288 stable cloture scenarios. The optimal solutions found by a brute-force procedure are Senators {Brown (D, OH), Craig (R, ID), and Dole (R, NC)} and {Kerry (D, MA), Craig (R, ID), and Dole (R, NC)}, both covering 1728 stable cloture scenarios. With respect to the republican party, the heuristic gives the following two solutions as the best, each covering 40 of 10,029 stable cloture scenarios: Senators {Brown (D, OH), Bennett (R, UT), and Gregg (R, NH)} and Senators {Kerry (D, MA), Bennett (R, UT), and Gregg (R, NH)}. The optimal solution for this case is Senators {Bennett (R, UT), Conrad (D, ND), and Sessions (R, AL)}, which covers 138 stable cloture scenarios.

We now consider the case of preventing cloture scenarios with respect to the democratic party; that is, the majority of the senators belonging to the democratic party want to pass a bill, but cannot gather 60 votes due to some senators voting "no." To find a small set of senators whose voting choice of "no" prevents cloture scenarios and potentially leads to filibusters, we adapt the above heuristic. With respect to the democratic party, there are 295,320 non-cloture stable outcomes (where a majority of the democratic senators voted "yes," yet there are fewer than 60 "yes" votes in total). The output of our heuristic matches the optimal set in this case, which covers 9681 non-cloture scenarios. If Senators McCain (R, AZ), McConnell (R, KY), Coburn (R, OK), and Hutchison (R, TX) vote "no" then the set of PSNE is exactly the set of these 9681 non-cloture scenarios. In that case, according to our model, a cloture can never take place in the event of a filibuster.

6.5.4. Application of another diffusion-based heuristic

We can once again try to adapt the popular greedy-selection algorithm for identifying most-influential nodes in the cascade model in the diffusion setting to work as another *heuristic* for the filibuster problem in our setting. In doing so, we encounter two notable obstacles that highlight another difference between our approach and that based on diffusion.

First, the notion of stable-cloture cover is not well-defined in the diffusion setting. The forward recursion mechanism central to diffusion models begins with a set of initial adopters (those senators selected to vote "yes" in our case) and propagates the effects of behavioral changes throughout the network until it reaches a steady state (i.e., no change occurs). However, this mechanism focuses on how the dynamics of behavioral changes evolves, not on the count of steady states that are consistent with a given set of players being among the adopters (not necessary early adopters), which is required for stable-cloture covers. In contrast, the stable-cloture cover is well-defined in our approach.

Second, and most important, even if we allow reversals of actions due to negative influence factors, forward recursion may produce an *unstable* outcome (i.e., not a PSNE). Although Granovetter's original model precludes this by requiring the initial adopters to have a threshold of 0 [31], subsequent development allows forward recursion to start with a set of initial adopters whose thresholds are not necessarily 0 [49]. Next, we illustrate this point using our experimental results.

Per the discussion in the last two paragraphs, in our experimental setting regarding the diffusion-based heuristic, which, once again, result from *our adaptation* of the popular greedy-selection approximation algorithm used for the cascade model in diffusion settings, we omit the notion of maximum stable-cloture cover and thereby forgo the measure of goodness of a solution. We only concentrate on finding a set of initial adopters that can drive the forward recursion process to *some* stable cloture scenario (i.e., a PSNE in $\mathcal{C}$).

In the following paragraph, we outline our *diffusion-based heuristic specifically adapted for the filibuster problem.*

For $k = 1, 2, \ldots,$ do the following. For all possible sets of k senators, start forward recursion with these k senators forced to play 1 all the time and other senators initially playing -1 (but are permitted to switch between 1 and -1 later on). When a steady state is reached, verify if there are at least 60 senators who are playing 1 in this state. If this is the case, then further verify if the k senators who are forced to play 1 are indeed playing their best response with respect to others' actions, which is the condition for the cloture scenario being stable. Stop iterating over k once you find stable cloture scenarios.

In our problem instances, which contain both positive and negative influence factors, it is very much possible that forward recursion oscillates indefinitely. However, that did not happen in our experiments. We tried all possible sets of $k \leq 3$ initial adopters, but failed to reach any cloture scenario (stable or unstable). We then tried all possible quadruplets of initial adopters. With respect to the democratic party, 1189 different quadruplets led the forward recursion process to a cloture scenario, but nearly half of these quadruplets (536 to be exact) led to unstable outcomes. Essentially, those unstable outcomes were due to some of the initial adopters not playing their best response in voting "yes"—all other nodes were indeed playing their best response (otherwise, the process would not terminate).

Therefore, beyond just emphasizing the stability of an outcome, the approximation algorithm based on our approach also captures certain phenomena that the heuristic based on the traditional approach seems unable to do. As stated earlier, of course, we would need more research to better understand this discrepancy, and the degree to which it could be reduced, as well as the effectiveness and potential for improvement of the diffusion-based heuristic. Such research is beyond the scope of this paper and remains open for future work.

7. Conclusion

In this paper, we studied influence and stable behavior from a new game-theoretic perspective. To that end, we introduced a rich class of games, named network influence games, to capture the core strategic component of complex interactions in a network. We characterized the computational complexity of computing and counting PSNE in LIGs. We proposed practical, effective heuristics to compute PSNE in such games and demonstrated their effectiveness empirically. Besides predicting stable behavior, we gave a framework for computing the most influential nodes and its variants (e.g., identifying a small coalition of senators that can prevent filibuster). We also gave a provable approximation algorithm for the most-influential-nodes problem.

Although our models are inspired by decades of research by sociologists, at the heart of our whole approach is abstracting the complex dynamics of interactions using the solution concept of PSNE. This allowed us to deal with richer problem instances (e.g., the ones with negative influence factors) as well as to tread into new problem settings beyond identifying the most influential nodes. We conclude this paper by outlining several interesting lines of future work.

First, we leave several computational problems open. We showed that counting the number of PSNE even in a star-type LIG is #P-complete, but does there exist an FPRAS for the counting problem? The computational complexity of indiscriminant LIGs, which we conjecture to be PLS-complete, is unresolved. Also, computing mixed-strategy Nash equilibria of LIGs, even for special types such as trees, remains an open question.

Second, we can apply our models to the general setting of “strategic interventions,” where we study the effects of changes in node thresholds, connectivity, or influence factors, usually without the possibility of having corresponding behavioral data. The following is an illustrative example of it in the context of the 111th U.S. Congress. After the death of Ted Kennedy, who was a democratic senator from the state of Massachusetts, a republican named Scott Brown was elected in his place. Not only was it Senator Brown’s first appointment in the Senate, he was also the first republican from Massachusetts to be elected to Senate for a long time. With no behavioral data available at that point of time, we can perform interventions using our model under various assumptions of thresholds, connectivity, and influence factors regarding Senator Brown, with the general goal of predicting stable outcomes and investigating the effects of the above interventions in various settings, such as filibuster scenarios or the setting of the most influential senators.

Another example of intervention, in the context of the Framingham heart study alluded in Section 1, is the following. Suppose that we would like to implement a policy of targeted interventions in order to reduce smoking by some margin. Using our model, we can modify the thresholds of the selected targets and predict how it could affect the overall level of smoking.

Besides interventions, we can also use our model to analyze past happenings, such as the role of the bipartisan “gang-of-six” senators in leading the members of the two major parties to an agreement during the U.S. debt ceiling crisis.[17] We can use our model to find how influential the gang-of-six senators were as a group. One approach to this problem would be to first find the set of stable outcomes consistent with the gang-of-six senators voting “yes” and then to analyze what fraction of these stable outcomes has 60 or more “yes” votes (signifying the passing of the corresponding bill without any possibility of a filibuster).

Our model of influence game can be considered as a step in the direction of modeling competitive contagion in strategic settings (see, for example, Budak et al. [12], Goyal and Kearns [30]). Here, one of the main challenges would be to formulate the competitive aspects of multiple “campaigns” without having to go through the usual network dynamics. In this general setting, we can ask questions such as, who are the most influential individuals with respect to achieving a certain objective that favors one “campaign” over the other? Such questions, and many others, shape the long-term goal of this research.

Acknowledgements

We are grateful to the reviewers for their comments, suggestions, and constructive criticisms. These not only improved this paper significantly but also contributed to the PhD Dissertation of M.T. Irfan [40]. His dissertation contains parts of the work presented in this paper.

This article is a significantly enhanced version of an earlier conference paper [41]. Besides a more detailed exposition that places this work in the context of existing ones in sociology, economics, and computer science, we extended the technical content in several ways. First, we made the connection between linear influence games and polymatrix games. As a result, the computational complexity results carry over to 2-action polymatrix games. Second, we gave the complete proofs of the theoretical results here. In addition to enhancing the theoretical part, we also extended the experimental part substantially. First and foremost, we illustrated our approach to influence in a new setting—the U.S. Supreme Court rulings. Given the U.S. Supreme Court dataset, we first illustrated how we can learn an LIG to model the potential strategic interactions among the Supreme Court justices [39]. We then applied the schemes for PSNE computation and the identification of the most

[17] http://en.wikipedia.org/wiki/United_States_debt-ceiling_crisis.

influential individuals in this new setting. Second, we extended our empirical study of the U.S. congressional voting by contrasting the output of our approximation algorithm to that of a diffusion-based heuristic we created by adapting the simple, greedy-selection approximation algorithm used for the analogous problem in the cascade model. We also applied our approach to a new problem of preventing filibuster by a coalition of senators, which highlights the broad range of scenarios where our techniques can be applied.

This work was supported in part by NSF CAREER Award IIS-1054541.

Appendix A. On the connection to rational calculus models of collective action

The formal study of individual behavior in a collective setting originally began under the umbrella of "collective behavior" in sociology and social psychology. The classical treatment of collective behavior views individuals in a "crowd" as irrational beings with a lowered intellectual and reasoning ability. The proposition is that an increased level of suggestibility among the individuals facilitates the rapid spread of the homogeneous "mind of the crowd" [52,72,10]. Herbert Blumer's work, in particular, popularized the classical theory of collective behavior well beyond academia and into such domains as police and the armed forces [57, p. 9]. However, this theory was subjected to much criticism primarily because it did not study empirical accounts systematically.[18]

In response to that, Clark McPhail undertook a massive effort, spanning three decades, to record the behavior of individuals in collective settings that he calls "gatherings" in order to distinguish it from (homogeneous) "crowds" in collective behavior (see McPhail [57, Ch. 5, 6] for a summary of his two-decade study). His empirical accounts, stored in a range of media formats as technology improved, reveal one common thing—that a gathering consists of individuals with diverse objectives, who nevertheless behave rationally and purposefully. To distinguish this purposive nature of individuals from irrationality in the classical treatment, he calls his study "collective action" and broadly defines it as "any activity that two or more individuals take with or in relation to one another" [58, p. 881]. In short, collective action can be seen as the modern approach, as opposed to the "old" (but not unimportant) approach of collective behavior [61, pp. 14–15].[19]

Many of the rational calculus or economic choice models that were originally proposed for collective behavior, are now discussed under collective action due to the purposive nature of the individuals. Here, we conduct a very narrow and focused review of the relevant literature in order to place our model in its proper context. Our review focuses on Mark Granovetter's threshold models [31], one of the most influential models of collective action to date. Before that, we will briefly review two prominent precursors to Granovetter's models—Schelling's models of segregation and Berk's "gaming" approach.

A.1. Schelling's models of segregation

A notable precursor to Granovetter's threshold models is Nobel-laureate economist Thomas Schelling's models of segregation [77,76]. Schelling's models account for segregations that take place as a result of *discriminatory individual behavior* as opposed to organized processes (e.g., separation of on-campus residence between graduate and undergraduate students due to a university's housing policy) or economic reasons (e.g., segregation between the poor and the rich in many contexts). An example of a segregation due to individual choice, or "individually motivated segregation" as Schelling puts it [77, p. 145], is the residential segregation by color in the U.S. Although Schelling's models expressly focus on this case, these can be applied to many other scenarios as well.

In Schelling's *spatial proximity model*, if an individual's *level of tolerance* for population of the opposing type is exceeded in his neighborhood, he moves to another spatial location where he can be "happy." Schelling studied the dynamics of segregation in this model using a rule of movement for the "unhappy" individuals. The *bounded-neighborhood model* is concerned with one global neighborhood. An individual enters it if it satisfies its level of tolerance constraint and leaves it otherwise. Schelling studies the stability of equilibria and the *tipping* phenomenon in this model when the distribution of tolerances and the population ratio of the two types are varied. An important finding is that in the cases studied, the modal level of tolerance does not correspond to a tipping point.[20]

A.2. Berk's "gaming" approach

Another notable precursor to Granovetter's models is Berk's rational calculus approach [9]. Berk strongly criticizes the assumption of individual irrationality which became prevalent in collective behavior literature. He formulates his approach by first giving a detailed empirical account of an anti-war protest at Northwestern University that originated in a town-hall meeting addressing dormitory rent hike.[21] He explains individual decision making through Raiffa's decision theory principles.

[18] For instance, Blumer himself referred to this as a "miserable job" by sociologists [11].

[19] A brief review of collective behavior and collective action literature is included in Appendix B for interested readers.

[20] More on Schelling's models can be found in Appendix B.

[21] Berk's description gives accounts of both mundane and exciting happenings during the course of the protest and is recognized as "among the best in the literature" [57, p. 126].

To motivate his approach, he first notes that participating individuals in that protest were diverse in their disposition and that they exercised their reasoning power. He then broadly classifies the participants into two types—militants (with the desired action of trashing properties) and moderates (with the desired action of an anti-war activity, but not trashing). Each participant, militant or moderate, estimates the support in favor of his disposition, and with enough support, he will "act" (e.g., trash properties if he is militant).

Clearly, an individual's estimate of support directly affects his "payoff." If an individual estimates that there is not enough support to act in favor of his disposition, he can try to persuade others to support his disposition so that he can receive a higher payoff by being able to act. This can be translated as an attempt to change others' payoff matrices, which is facilitated by the *milling* phase when they communicate and negotiate with each other. The milling phase ends when a consensus or a compromise is reached and becomes common knowledge. In this way, a concerted action takes place according to Berk's model.

A.3. Granovetter's threshold models

Granovetter [31] presented his threshold models in the setting of a crowd, where each individual is deciding whether to riot or not. In the simplest setting, each individual has a threshold and his decision is influenced by the decisions of others—if the number (or the proportion) of individuals already rioting is below his threshold, then he remains inactive, otherwise he engages in rioting. The emphasis is on investigating equilibrium outcomes due to the process of forward recursion [31, p. 1426], given a distribution of the thresholds of the population. It may be mentioned here that forward recursion starts only if there is an individual with a threshold of 0.

Granovetter's models are inspired by Schelling's models of segregation. In fact, one can draw a parallel between Schelling's level of tolerance and Granovetter's threshold in the following way. In Schelling's models, an individual leaves a neighborhood if his level of tolerance is exceeded, whereas in Granovetter's models, an individual becomes active in rioting if his threshold is exceeded. Furthermore, in both models, dynamics is of utmost importance and serves the purpose of explaining *how* an equilibrium collective outcome emerges from individual behavior. However, apart from these similarities, these two models are semantically different and also focus on completely different outlooks. First, Granovetter ascribes a deeper meaning to the concept of threshold. Threshold of an individual is not just "a number that he carries with him" from one situation to another [31, p. 1436]. It rather depends on the situation in question and can even vary within the same situation due to changes occurring in it. Second, in Granovetter's models, a very small perturbation in the distribution of population threshold may lead to sharply different equilibrium outcomes. Granovetter highlights this property of his models as an explanation of seemingly paradoxical outcomes that goes against the predispositions of the individuals.

Two features of Granovetter's models make it stand out among the rational calculus models. First, the models are capable of capturing scenarios beyond the classical realm of collective behavior. Granovetter begins by setting up his model to complement the emergent norm theory (see Appendix B) by providing an explicit model of how "individual preferences interact and aggregate" to form a new norm [31, p. 1421]. Not only does such an explicit model eliminates the need for implicit assumptions (such as a new norm emerges when the majority of the population align themselves with that norm), it can also capture paradoxical outcomes alluded above that cannot be captured by the implicit assumption on the majority. Beyond the emergent norm theory, Granovetter's models can capture a wide range of phenomena that do not fall within the classical realm of collective behavior, such as diffusion of innovation, voting, public opinion, and residential segregation, to name a few. The second prominent feature of Granovetter's models is its ease of adaptation when dealing with a networked population. The same mechanism of forward recursion is applicable when the underlying influence structure is specified by a "sociomatrix," which accounts for how much an individual influences another [31, p. 1429]. This is particularly useful for studying collective action in the setting of a social network.

A.4. Criticism of rational calculus models

An implicit assumption regarding Granovetter's sociomatrix is that the elements of the matrix are non-negative. Otherwise, the process of forward recursion may never terminate, even on the simplest of examples. However, many real world scenarios do exhibit co-existence of both positive and negative influences. For the most part, democrat senators in the U.S. Congress influence their republicans colleagues negatively, while they influence colleagues of their own party positively. In residential segregation involving more than two types of individuals, an individual is negatively influenced, in different magnitudes, by individuals belonging to other types. Clearly, such a situation cannot be modeled using a non-negative sociomatrix. Furthermore, if we take a second look at Berk's account, militant individuals positively reinforce each other in their decision to engage in trashing properties, whereas their decision is negatively affected by the moderates (that is, the presence of too many moderates makes it risky for militants to engage in violent action).

Critiques of rational calculus models point out the lack of behavioral adjustment in a "negative feedback" fashion [58, p. 883]. Here, negative feedback is defined in the context of the perceptual control theory that lays the foundation of McPhail's *sociocybernetics theory* of collective action (see Appendix B). In a negative feedback system, an individual can adjust his behavior depending on the discrepancy between the input signal and the desired signal (the sign of this discrepancy has no correlation to negative feedback). In contrast, in a positive feedback system, such control of behavior is not possible. A typical example of a positive feedback system is a chemical chain reaction. An analogue to this is the "domino effect" cited often in rational calculus models [31, p. 1424]. It is true that rational calculus models neither account for "errors" as desired by the proponents of the sociocybernetics theory, nor is the concept of "errors" well-defined in the context of rational calculus. But it is not the case that "reversal" of behavior, which can be thought of as a crude form of behavioral adjustment, is precluded in rational calculus models. Such a form of behavioral adjustment can certainly be incorporated by allowing negative elements in the sociomatrix, but the challenge lies in the forward recursion process which may oscillate indefinitely because of those negative elements.

Appendix B. Brief review of collective behavior and collective action in sociology

In sociology, the umbrella of *collective behavior* is very broad and encompasses an incredibly rich set of models explaining various aspects of a wide range of social phenomena such as revolutions, movements, riots, strikes, disaster, panic, and diffusion of innovations (e.g., fashion, adopting contraceptives, electronic gadgets, or even religion), just to name a few.[22] Sociologists Marx and McAdam, in their concise introductory book on collective behavior [56], contend that unlike many other fields of sociology, the field of collective behavior is not easy to define, partly because of the varied opinion of scholars: from very narrow perspectives, to such wide, all-encompassing perspectives (e.g., Robert Park and Herbert Blumer's) that virtually eliminates the need to have collective behavior as an individual field in sociology.[23] Yet, Marx and McAdam point out the traditional disposition to categorize collective behavior as a "residual field" in sociology; that is, the study of collective behavior consists of those elements of behavior (e.g., fads, fashion, crazes), organization (e.g., social movement), group (e.g., crowd), individual (e.g., psychological states such as panic), etc., that do not readily fit into well-established and commonly observed social structures. Similarly, collective behavior is defined in Goode's textbook [28, p. 17] as the "relatively spontaneous, unstructured, extrainstitutional behavior of a fairly large number of individuals."

B.1. Classical treatment of collective behavior: mass hysteria

The classical treatment of collective behavior views individuals in a crowd as non-rational, transformed into hysteria by the collective environment. The central tenet of the early work of Gustave Le Bon's is that individuals in a crowd share a "mind of the crowd," and that the "psychological law of the mental unity of crowds" guides their psychological state and behavior in the collective setting [52, p. 5]. In Le Bon's account, an individual in a crowd may retain some of the ordinary characteristics he shows in isolation; but the emphasis is on the extraordinary characteristics that emerge only in a crowd because of "a sentiment of invincible power," contagion, and most importantly, the susceptibility of individuals to take suggestions as if they were hypnotic subjects. Examples of such extraordinary characteristics of a crowd are "impulsiveness," "incapacity to reason," and "the absence of judgment," to name a few [52, p. 16]. In sum, individuals in a crowd are depleted of their intellectual capacity and become uniform in their psychological state. This leads the crowd to an identical direction of collective behavior that may be heroic or criminal, depending on the type of "hypnotic suggestion" alluded above (although Le Bon gives examples of heroic crowds [52, p. 14], for the most part, he tends to give "crowds" a negative connotation).

Le Bon's work influenced around half-a-century of subsequent developments. Park and Burgess upheld his proposition on the transformation of individuals in a crowd. They put forward the concept of *circular reaction* [72], later refined by Herbert Blumer [10]. Circular reaction refers to a reciprocal process of social interaction that explains *how* crowd members become uniform in their behavior, something that Le Bon could not really explain. In this process, an individual's behavior stimulates another individual to behave alike; and when the latter individual does so, it reinforces the stimulation that the former individual acted upon. In circular reaction, individuals do not act rationally or intellectually; that is, they do not reason about the action of others, but instead, they only align themselves with the behavior of others. This is different from *interpretative interaction*, another mechanism that Blumer defined to explain routine group behavior (e.g., a group of individuals shopping in a mall), as opposed to collective behavior (e.g., social movement). In interpretative interaction, individuals react (perhaps differently) to their *interpretation* of others' action, not the action itself. Therefore, in interpretative interaction, one can treat individuals as rational beings.[24]

In Blumer's account, a crowd goes through several well-defined stages before a collective behavior finally emerges. The three underlying mechanisms that facilitate transitions among these stages are circular reaction, collective excitement, and

[22] In fact, the richness of just one subfield of collective behavior, termed *micro-level theories of collective behavior*, led Montgomery to comment in his book [64, p. 67], "The variety of theories focusing on the micro level is confusing, but is an indication of the complexity and variations in the process by which movements emerge or perhaps fail to emerge..."

[23] Quoting from their book, "The field of collective behavior is like the elephant in Kipling's fable of the blind persons and the elephant. Each person correctly identifies a separate part, but all fail to see the whole animal."

[24] For clarity of presentation, we differentiate between routine group behavior and collective behavior. To the contrary, Blumer, as well as Park in his earlier work, viewed collective behavior as encompassing a wide range of social phenomena, including routine group behavior. Within the continuum of collective behavior in Blumer's view, the presence or the absence of rationality of individuals earmarks two distinct mechanisms named interpretative interaction and circular reaction, respectively.

social contagion; one can roughly think of collective excitement as a more intense form of circular reaction, and of social contagion as an even more intense form of circular reaction [57, p. 11].[25]

B.2. Emergent norm theory

Although Blumer's account of collective behavior received wide-spread acceptance, even beyond academia [57, p. 9], others deemed many of the underlying assumptions in it, as well as in the general mass-hysteria theory, unrealistic. Arguably, individuals with *different* objectives in mind participate in a collective behavior, and one can observe *changes* in their individual behavior throughout the process of a collective behavior too.[26] Therefore, the assumption of complete uniformity behavior in the classical mass-hysteria treatment is very much a stretch. Furthermore, the assumption of hysteric crowd in the classical approach has also been called into question. One notable critique of the mass-hysteria theory comes from Ralph Turner and Lewis Killian [83]. They view individuals in a crowd as behaving under normative constraints and showing "differential expression." However, a new norm emerges when the established norms of the society cannot adequately guide a crowd facing an extraordinary situation. They call this the *emergent norm* and contend that it is the emergent norm that gives the "illusion of unanimity."

B.3. Collective action

The goal-oriented nature of collective behavior was further highlighted by sociologists studying social movements during the 1970s and 80s. In order to distinguish their approach from the traditional approach to collective behavior, dominated by the assumption of irrational and aimless nature of crowds, they used the term *collective action* to mean "people acting together in pursuit of common interests" [80]. Strikingly, based on a series of systematic observations, Clark McPhail's contends that the goal-oriented nature of crowds is not limited to social movements and revolutions alone, but is a feature of various other types of crowds. In his book *the Myth of the Madding Crowd*, he uses two decades of empirical observations pertaining to a multitude of crowd settings to formulate a theory of collective behavior now recognized as a significant paradigm shift [57, Ch. 5, 6]. To distance himself from the term "crowds," which has already gained several meanings depending on whose theory is being considered, he gives his formulation in the setting of "gatherings." But first, he places a justifiably strong emphasis on the definition of collective behavior. His "working definition of collective behavior" is the study of "two or more persons engaged in one or more behaviors (e.g., locomotion, ...) judged common or concerted on one or more dimensions (e.g., direction, velocity, ...)"[27] [57, p. 159].

The broad nature of McPhail's definition of collective behavior, although based on extensive empirical evidence, did not receive immediate acceptance. Even modern textbooks on collective behavior try to conserve the classical appeal of collective behavior.[28] Perhaps to further distance himself from the traditional viewpoint, McPhail later began to use the term *collective action* instead of collective behavior (for example, in a recent encyclopedia article, McPhail refers to the above mentioned definition as that of collective action [58]). According to David Miller, the modern view on the distinction between collective action and collective behavior is beyond simply terminological. Collective action is given the status of a "new" theory in sociology, while collective behavior is marked as "old," but not unimportant [61, pp. 14–15].[29]

McPhail's approach to collective action is known as the *social behavioral interactionist (SBI)* approach. As much as it agrees with the emergent-norm theory, in terms of the diversity of individual objectives in a collective setting, it does not agree with the concept of an emergent norm suppressing this diversity. The SBI approach studies gatherings in three phases of its life cycle: the assembling process, collective action within the assembled gathering, and the dispersal process [57, p. 153]. Although each of these three phases is rich and interdependent, the goal is to manage the complexity of collective action as a whole by focusing on the recognizable parts of it. Interestingly, the underlying mechanism to explain collective action is drawn from the *perceptual control theory* [57, Ch. 6]. McPhail adapts this theory to formulate his *sociocybernetics theory* of collective action. In brief, an individual receives sensory inputs, compares the input signal to its desired signal,[30] and

[25] The mechanism of "social contagion" as defined by Blumer or the "social contagion theory" [54, p. 11] in general should not be confused with the term "social contagion" that computer scientists use [84]. Although both have their roots in epidemics, in the former case, individuals are transformed into being more suggestible, which facilitates "rapid, unwitting, and non-rational dissemination" of behavior [10]; whereas in the latter case, individuals act rationally.

[26] We refer the reader to [61, pp. 26–27] for a beautiful example in the context of the 1967 anti-war demonstration in Washington, DC, which was participated by nearly 250,000 people. While many of the participants might have been there to genuinely voice their opinion against the war, some might have been looking for "excitement, drug, or sex." Yet again, individuals playing different roles, such as protest leaders, street vendors, and the police, behaved differently.

[27] Note that it is the "behavior," not a specific action, that needs to be common or concerted. For example, when a group of people are chatting together, their behavior is concerted, even though they are not speaking identical words.

[28] For example, in reference to McPhail's definition, Goode writes in his textbook, "In the view of most observers, myself included, many gatherings are *not* sites of collective behavior (most casual and conventional crowds, for example), and much collective behavior does not take place in gatherings of any size (the behavior of most masses and publics, for example)" [28, p. 17]. Ironically, this is the very viewpoint that McPhail seeks to portray as a myth.

[29] Miller also points out that sociology textbooks are likely to talk about collective behavior only, whereas recent journal articles talk about collective action only.

[30] In contrast to engineering control systems theory, the desired signal is not external, but set by individuals themselves (which is also highlighted by the term *cybernetics*).

adjusts its behavior in response to the discrepancy. The behavior of individuals affects the "environment," which in turn affects the input signal, thereby completing a loop. An important aspect of this theory is that various external factors (or "disturbances") may drive an individual to make different behavioral adjustments at different points in time even if the discrepancy between the input signal and the desired signal remains the same.

B.3.1. Additional notes on Schelling's models

Schelling's models assume that individuals behave in a discriminatory way. For example, individuals are aware, consciously or unconsciously, about the types of other individuals in their neighborhood and behave (i.e., stay in the neighborhood or leave) according to their preference. This is different from organized processes (e.g., separation of on-campus residence between graduate and undergraduate students due to a university's housing policy) or economic reasons (e.g., segregation between the poor and the rich in many contexts) [77,76]. An example of a segregation due to individual choice, or "individually motivated segregation" as Schelling puts it [77, p. 145], is the residential segregation by color in the United States. In fact, Schelling's models and their analyses expressly focus on this case. Yet, we can apply Schelling's theory to many other scenarios as well. This is because it explains, at an abstract level, how *collective* outcomes are shaped from *individual* choice. We note here that connecting individual actions to collective outcomes is a mainstream theme of research in collective action.

Schelling introduces two basic models to study the dynamics of segregation among individuals of two different types [77]. In the first model, the *spatial proximity model*, individuals are initially positioned in a spatial configuration (such as a line or a stylized two-dimensional area) and individuals of the same type share a common "level of tolerance." This level of tolerance quantifies the upper limit on the percentage of an individual's opposite type in his local neighborhood that he can put up with. Here, we define an individual's local neighborhood with respect to the individual's position in the specified spatial configuration. In this model, we use a rule of movement for the "unhappy" individuals to study the dynamics of segregation. For example, an individual whose level of tolerance has been exceeded, moves to the closest location where the tolerance constraint can be satisfied. Assuming an equal number of individuals of each type and a fixed local neighborhood size, Schelling first studies how clusters evolve from the initial configuration of a random placement of the individuals on a straight line. He then generalizes the experimentation by varying different model parameters such as neighborhood size, level of tolerance, and the ratio of individuals of the two types. Notable findings are that decreasing the local neighborhood size leads to a decrease in the average cluster size and that for an unequal number of individuals of the two types, decreasing the relative size of the minority leads to an increase in the average minority cluster size.

Schelling extends this experimentation to a different setting of a two-dimensional checkerboard. The individuals are randomly distributed on the squares of the checkerboard, leaving some of the squares unoccupied. An individual's local neighborhood is defined by the squares around it and an unhappy individual moves to the "closest" unoccupied square (leaving its original square unoccupied) that can satisfy its tolerance constraint. In addition to studying clustering properties by varying different model parameters, two new classes of individual preferences have been studied: congregationist and integrationist. In a congregationist preference, an individual only wants to have at least a certain percentage of neighbors of its own type and does not care about the presence of individuals of the opposite type in its neighborhood. Experiments show that even when each individual is happy being a minority in its neighborhood (e.g., having three neighbors of its own type out of eight), the dynamics of segregation leads to a configuration as if the individuals wished to be majority in their neighborhoods. In an integrationist preference, individuals have both an upper and a lower limit on the level of tolerance. The dynamics is much more complex in this case and leads to clusters of unoccupied squares.

Schelling's second model, the *bounded-neighborhood model*, concerns one global neighborhood. An individual enters it if it satisfies its level of tolerance constraint and leaves it otherwise. The level of tolerance is no longer fixed for each type and the distribution of tolerances among individuals of each type is given. The emphasis is on the stability of equilibria as the distribution of tolerances and the population ratio of the two types vary. For example, under a certain linear distribution of tolerances and a population ratio of $2:1$, there exist only two stable equilibria, each consisting of individuals of one type only; whereas a mixture of individuals of both types can arise as a stable equilibrium under a different setting. This model has been adapted to study *tipping* phenomenon, with one notable constraint; that is, the capacity of the neighborhood is fixed. An example of a tipping phenomenon is when a neighborhood consisting of only one type of individuals is later inhabited by some individuals of the opposite type and as a result, the entire population of the original type evacuates the neighborhood. An important finding is that in the cases studied, the modal level of tolerance does not correspond to a tipping point.

Appendix C. Some basic concepts in game theory

C.1. Strategic complementarity

A formal, general definition of strategic complementarity is beyond the scope of this paper. Instead, we present a definition in the context of this paper and refer the reader to standard references for a general definition. We say that player i in a general-influence game $\mathcal{G}$ exhibits *strategic complementarity* if for every pair of the joint-actions $\mathbf{x}_{-i}$ and $\mathbf{x}'_{-i}$, if $\mathbf{x}_{-i} \geq \mathbf{x}'_{-i}$, element-wise, implies $u_i(1, \mathbf{x}_{-i}) \geq u_i(-1, \mathbf{x}'_{-i})$. We then say the influence game $\mathcal{G}$, as a whole, exhibits *strategic complementarity* if every player in the game does. Intuitively, it says that the action/behavior in the best-response correspondence of any player cannot "decrease" (i.e., move "down" from $\{+1\}$ to $\{-1\}$, or to $\{-1, +1\}$ for that matter) if the actions/behavior of the other players "increase" (i.e., at least one other player moves "up" from -1 to $+1$); and *vice versa*. Thus, roughly speaking, we can say that the players' actions "complement" each other "strategically;" or said differently, in general, each player prefers to "play along" by choosing an action "consistent" with that chosen by the other players: if the other players "move up" or "move down" then the player would like to "follow along" with the other players by choosing the respective action, or "stay put."

C.2. Strategic substitutability

Strategic substitutability is essentially the opposite of strategic complementarity, as presented in the previous paragraph, *except* that one replaces the $\geq$ sign with a $\leq$ sign in the hypothesis condition in the definition (i.e., $\mathbf{x}_{-i} \leq \mathbf{x}'_{-i}$). Once again a formal, general definition is beyond the scope of this paper. Intuitively, it says that the action/behavior in the best-response correspondence of any player cannot "decrease" (i.e., move "down" from $\{+1\}$ to $\{-1\}$, or to $\{-1, +1\}$ for that matter) if the actions/behavior of the other players also "decrease" (i.e., at least one other player moves "down" from $+1$ to -1); and *vice versa*. Thus, roughly speaking, we can say that the players' actions are "substitutes" of each other "strategically;" or said differently, in general, each player prefers to play an action that is opposite to/different than that chosen by the other players: if the other players "move up" or "move down" then the player would like to "go against the crowd" by choosing an opposite action, or "stay put."

C.3. Potential games

A formal, general definition of potential games is beyond the scope of this paper. Instead, we present a definition in the context of this paper and refer the reader to the standard reference for a more general definition [63].

We say an influence game $\mathcal{G}$ is a *ordinal potential game* if there exists a function $\Phi : \{-1, +1\}^n \to \mathbb{R}$, called the *ordinal potential function*, independent of any specific player, such that for every joint-action $\mathbf{x}$, for every player i, and for every possible action/pure-strategy x'_i that player i can take, we have that $u_i(x'_i, \mathbf{x}_{-i}) - u_i(x_i, \mathbf{x}_{-i}) > 0$ if and only if $\Phi(x'_i, \mathbf{x}_{-i}) - \Phi(x_i, \mathbf{x}_{-i}) > 0$.[31]

When the overall condition above is the stricter condition $u_i(x'_i, \mathbf{x}_{-i}) - u_i(x_i, \mathbf{x}_{-i}) = \Phi(x'_i, \mathbf{x}_{-i}) - \Phi(x_i, \mathbf{x}_{-i})$, then we call $\mathcal{G}$ and Φ an *exact potential game and function*, respectively. For simplicity, we refer to such $\mathcal{G}$ and Φ simply as a *potential game and function*, when clear from context.

Intuitively, the potential function, a global quantity independent of any specific player, defines the best-response correspondence of each player in the potential game. Also, the PSNE of a potential game are essentially the local minima (or stationary points) of the potential function. Hence, one could view the players as trying to optimize the potential function as a "global group" but via "individual, local best-responses." By performing asynchronous/non-simultaneous best-response dynamics, the players are implicitly performing an axis-parallel optimization of the potential function. Because the domain of the potential function (i.e., the space of joint-actions) is finite, this process will always converge to a local maxima or stable point of the potential function, or equivalently, to a PSNE of the potential game. Hence, every potential game always has a PSNE.

C.4. Invariance of equilibrium concepts to affine/linear transformations

The reason the result in Proposition 3.8 generalizes beyond PSNE is that, as is well-known, two games with the same number of players and the same set of actions for each player have the same CE set if the set of payoff functions $\{u_i^1\}$ and $\{u_i^2\}$ of the games 1 and 2, respectively, satisfy the following condition: for all players i, there exist a positive constant $c_i > 0$ and an arbitrary real-valued function d_i of the actions of all the players except i, such that, $u_i^1(x_1, \mathbf{x}_{-i}) = c_i u_i^2(x_i, \mathbf{x}_{-i}) + d_i(\mathbf{x}_{-i})$.

It is known that the computation of MSNE of 2-action polymatrix games is PPAD-hard. (It follows from the result of Daskalakis et al. [19], although it is not explicitly mentioned there.) An implication of this is that the computation of MSNE of LIGs is also PPAD-hard.

Appendix D. On submodularity

As briefly discussed in Section 2.2.4, the computer-science community is enthusiastic about the submodularity of the so called "influence spread function." One reason for the intense interest is the important role submodularity plays in

[31] Note the abuse of notation by letting $\Phi(\mathbf{x}) \equiv \Phi(\mathbf{x}_i, \mathbf{x}_{-i})$ for each player i; this is consistent with the same abuse of notation we use for the payoff functions u_i throughout the paper, standard in the game-theory literature.

Table 2
PSNE computation on uniform random directed graphs. Offsets of 95% confidence intervals are shown in parenthesis.

p	# of equilibria Avg (95% CI)		# of node visits/equilibrium Avg (95% CI)		Avg CPU time (sec) for computing all equilibria
0.00	2.18	(0.16)	35379.73	(3349.77)	1.81
0.125	3.72	(0.50)	22756.15	(2673.56)	1.57
0.25	13.00	(1.92)	9796.30	(1748.76)	1.9
0.375	19.42	(2.88)	7380.97	(1870.11)	1.95
0.50	14.40	(2.17)	9826.61	(1696.52)	2.04
0.625	19.78	(3.40)	8167.60	(1450.48)	2.07
0.75	28.76	(4.34)	6335.18	(1963.11)	2.21
0.875	67.14	(9.52)	4064.06	(1539.33)	3.27
1.00	194.96	(28.47)	1879.45	(235.90)	5.23

Table 3
Computation of the most influential nodes: Comparison between the solution given by the **ApproximateUniqueHyperedge** algorithm and the optimal one in random directed graphs. Offsets of the 95% confidence intervals are shown in parenthesis.

p	Approx soln size Avg (95% CI)		Opt soln size Avg (95% CI)		Approx = Opt % of times	Approx $\leq$ Opt + 1 % of times	Approx $\leq$ Opt + 2 % of times
0.00	1.06	(0.05)	1.06	(0.05)	100	100	100
0.125	1.50	(0.11)	1.48	(0.10)	98	100	100
0.25	2.57	(0.18)	2.14	(0.11)	62	96	99
0.375	2.75	(0.18)	2.35	(0.13)	62	98	100
0.50	2.47	(0.16)	2.09	(0.10)	66	96	100
0.625	2.82	(0.18)	2.31	(0.13)	52	97	100
0.75	3.02	(0.19)	2.48	(0.14)	51	95	100
0.875	3.83	(0.20)	3.04	(0.15)	36	86	99
1.00	4.72	(0.23)	3.95	(0.18)	37	87	99

the algorithmic design and analysis of methods to solve the "most influential nodes" problem as formulated within the diffusion setting using the cascade model [49]. Because our approach abstracts away the dynamics, *it is not even clear what such "influence spread function" is in our context, or even whether it could be meaningfully defined.*

In this paragraph, we will consider a potential definition and discuss its meaning or implications, if any. We first introduce some additional notation to simplify the presentation. We denote the set involved in the uniqueness condition given in Definition 3.6 by $C_{g,\mathcal{G}}(S) \equiv \{\mathbf{x} \in \mathcal{NE}(\mathcal{G}) \mid \mathbf{x}_S = \mathbf{x}^*_S, \mathbf{x}^* \in \mathcal{X}^*_g(S)\}$. Perhaps the most direct attempt at defining a notion analogous to the "influence spread function" in the diffusion setting, which in our context we will denote by $f_{g,h,\mathcal{G}}$, may be to let $f_{g,h,\mathcal{G}}(S) \equiv h(S) - \lambda|C_{g,\mathcal{G}}(S)|$, for some constant $\lambda > 0$. We can interpret such an $f_{g,h,\mathcal{G}}$ as trying to minimize the PSNE consistent with assigning the nodes in S according to some goal PSNE, while maximizing a general preference over subsets as captured by h, modulo a "penalization constant" λ. Yet, we cannot say anything meaningful about such an "influence spread function" in general, even for the most common instantiations of g and h; except perhaps that it is at best unclear to us that the question of whether that function is submodular makes sense, or whether it could even have a reasonable answer. To start, one major reason for our inability to say anything meaningful at this moment is that we are unaware of any PSNE characterization result that would apply to our setting. It seems to us that such characterizations would be key in any study of the potential submodularity properties of that "influence spread function" $f_{g,h,\mathcal{G}}$ as we have just defined.

The study of other potential definitions of an "influence spread function" in our setting, as well as their properties, is beyond the scope of this paper. (In fact, we do not know of any other reasonable alternative, beyond simple variations of the function defined above.) More importantly, we leave open for future work the study of the potential *relevance* that submodularity may have within our approach, beyond the indirect connection through the characteristics of certain classes of network influence games discussed in Section 4 (i.e., strategic complementarity and substitutability).

Appendix E. Experimental results in tabular form

Table 2 shows experimental data of PSNE computation on uniform random directed graphs. Of particular interest is the result that the number of PSNE usually increases when the flip probability p is increased, i.e., when the number of arcs with negative influence factors is increased.

Table 3 illustrates the experimental result that the logarithmic-factor approximation algorithm for identifying the most influential individuals performs very well in practice.

Finally, Table 4 shows the influence factors and thresholds of the LIG among the U.S. Supreme Court justices, which are learned using the U.S. Supreme Court dataset.

Table 4
LIG learned from data. Each non-diagonal element in each row represents the influence factor the row player receives from the column players (for example, Justice Scalia's influence factor on Justice Thomas is 0.2930 and that in the opposite direction is 0.4282). The diagonal elements represent thresholds of the corresponding row players.

	Scalia	Thomas	Rehnquist	O'Connor	Kennedy	Breyer	Souter	Ginsburg	Stevens
Scalia	0.0120	0.4282	0.0317	0.0717	0.0721	0.0772	−0.0321	0.1362	−0.1388
Thomas	0.2930	−0.1020	0.1245	−0.0010	0.0183	−0.1497	0.0839	−0.1311	0.0965
Rehnquist	0.0671	0.1762	−0.0834	0.0973	0.1254	0.0921	−0.0861	0.1336	−0.1388
O'Connor	0.0580	0.1073	0.1045	−0.2522	−0.0537	0.2313	0.0325	−0.1245	−0.0359
Kennedy	0.0666	0.1236	0.1863	−0.0255	−0.2634	0.0548	−0.1115	−0.0149	0.1532
Breyer	0.1009	−0.2191	0.0570	0.2208	−0.0061	−0.0209	0.0627	0.1102	0.2023
Souter	0.0368	0.1192	−0.0476	0.0762	−0.0338	0.1429	−0.0619	0.2783	0.2034
Ginsburg	0.0779	−0.1000	0.1613	−0.0962	−0.0089	0.1199	0.1999	−0.0381	0.1978
Stevens	−0.1379	0.1088	−0.1721	−0.0568	0.1053	0.1374	0.0932	0.1274	−0.0611

References

[1] R. Albert, A.L. Barabási, Statistical mechanics of complex networks, Rev. Mod. Phys. 74 (2002) 47–97.

[2] R. Aumann, Subjectivity and correlation in randomized strategies, J. Math. Econ. 1 (1974).

[3] R. Aumann, Correlated equilibrium as an expression of Bayesian rationality, Econometrica 55 (1987).

[4] G. Ausiello, A. D'Atri, M. Protasi, On the structure of combinatorial problems and structure preserving reductions, J. Comput. Syst. Sci. 21 (1980) 136–153.

[5] C. Ballester, A. Calvó-Armengol, Y. Zenou, Who's who in crime network. Wanted the key player, Research Institute of Industrial Economics, 2004, Working Paper Series 617.

[6] C. Ballester, A. Calvó-Armengol, Y. Zenou, Who's who in networks. Wanted: the key player, Econometrica 74 (2006) 1403–1417.

[7] P. Bardsley, Q. Nguyen, Rent seeking and judicial bias in weak legal systems, Technical report 925, The University of Melbourne, 2005.

[8] G.S. Becker, A theory of competition among pressure groups for political influence, Q. J. Econ. 98 (1983) 371–400.

[9] R.A. Berk, A gaming approach to crowd behavior, Am. Sociol. Rev. 39 (1974) 355–373.

[10] H.G. Blumer, Collective behavior, in: R.E. Park (Ed.), Principles of Sociology, Barnes & Noble, New York, 1939, pp. 219–288.

[11] H.G. Blumer, Collective behavior, in: J.B. Gittler (Ed.), Review of Sociology: Analysis of a Decade, J. Wiley, New York, 1957, pp. 127–158, chapter 5.

[12] C. Budak, D. Agrawal, A. El Abbadi, Limiting the spread of misinformation in social networks, in: Proceedings of the 20th International Conference on World Wide Web, ACM, 2011, pp. 665–674.

[13] J.I. Bulow, J.D. Geanakoplos, P.D. Klemperer, Multimarket oligopoly: strategic substitutes and complements, J. Polit. Econ. 93 (1985) 488–511.

[14] W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincon, X. Sun, Y. Wang, W. Wei, Y. Yuan, Influence maximization in social networks when negative opinion may emerge and propagate, in: Proceedings of the Eleventh SIAM International Conference on Data Mining, 2011, pp. 379–390.

[15] N.A. Christakis, J.H. Fowler, The spread of obesity in a large social network over 32 Years, N. Engl. J. Med. 357 (2007) 370–379, http://content.nejm.org/cgi/reprint/357/4/370.pdf.

[16] N.A. Christakis, J.H. Fowler, The collective dynamics of smoking in a large social network, N. Engl. J. Med. 358 (2008) 2249–2258, http://content.nejm.org/cgi/reprint/358/21/2249.pdf.

[17] M.S.Y. Chwe, Structure and strategy in collective action, Am. J. Sociol. 105 (1999) 128–156.

[18] R.D. Congleton, A political efficiency case for federalism in multinational states: controlling ethnic rent-seeking, in: Competition and Structure: The Political Economy of Collective Decisions: Essays in Honor of Albert Breton, 2000, pp. 365–397.

[19] C. Daskalakis, P.W. Goldberg, C.H. Papadimitriou, The complexity of computing a Nash equilibrium, SIAM J. Comput. 39 (2009) 195–259.

[20] P. Domingos, M. Richardson, Mining the network value of customers, in: Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining, ACM Press, San Francisco, CA, 2001, pp. 57–66.

[21] P. Erdös, A. Rényi, On the evolution of random graphs, Bull. Inst. Int. Stat. 38 (1961) 343–347.

[22] E. Even-Dar, A. Shapira, A note on maximizing the spread of influence in social networks, in: X. Deng, F. Graham (Eds.), Internet and Network Economics, in: Lect. Notes Comput. Sci., vol. 4858, Springer, Berlin, Heidelberg, 2007, pp. 281–286.

[23] A. Fabrikant, C. Papadimitriou, K. Talwar, The complexity of pure Nash equilibria, in: STOC '04: Proceedings of the 36th Annual ACM Symposium on Theory of Computing, ACM, New York, NY, USA, 2004, pp. 604–612.

[24] J.H. Fowler, N.A. Christakis, Dynamic spread of happiness in a large social network: longitudinal analysis over 20 years in the Framingham Heart Study, Br. Med. J. 337 (2008) a2338.

[25] D. Fudenberg, D. Levine, The Theory of Learning in Games, Economics Learning and Social Evolution Series, MIT Press, 1998.

[26] D. Fudenberg, J. Tirole, Game Theory, The MIT Press, 1991.

[27] M. Garey, D. Johnson, Computers and Intractability: a Guide to the Theory of NP-Completeness, Series of Books in the Mathematical Sciences, W.H. Freeman, 1979.

[28] E. Goode, Collective Behavior, Saunders College Pub., 1992.

[29] G. Gottlob, G. Greco, F. Scarcello, Pure Nash equilibria: hard and easy games, J. Artif. Intell. Res. 24 (2005) 357–406.

[30] S. Goyal, M. Kearns, Competitive contagion in networks, in: Proceedings of the 44th Symposium on Theory of Computing, ACM, 2012, pp. 759–774.

[31] M. Granovetter, Threshold models of collective behavior, Am. J. Econ. Sociol. 83 (1978) 1420–1443.

[32] M. Granovetter, R. Soong, Threshold models of diffusion and collective behavior, J. Math. Sociol. 9 (1983) 165–179, http://www.tandfonline.com/doi/pdf/10.1080/0022250X.1983.9989941.

[33] A. Guille, H. Hacid, C. Favre, D.A. Zighed, Information diffusion in online social networks: a survey, SIGMOD Rec. 42 (2013) 17–28.

[34] X. He, G. Song, W. Chen, Q. Jiang, Influence blocking maximization in social networks under the competitive linear threshold model, in: Proceedings of the Twelfth SIAM International Conference on Data Mining, 2012, pp. 463–474.

[35] G. Heal, H. Kunreuther, You only die once: managing discrete interdependent risks, Working paper W9885. NBER. Available at SSRN, http://ssrn.com/abstract=430599, 2003.

[36] G. Heal, H. Kunreuther, You can only die once: interdependent security in an uncertain world, in: H.W. Richardson, P. Gordon, J.E. Moore II (Eds.), The Economic Impacts of Terrorism Attacks, Edward Elgar, 2005, pp. 35–56, Chapter 3.

[37] G. Heal, H. Kunreuther, Supermodularity and tipping, Working paper 12281, National Bureau of Economic Research, 2006.

[38] G. Heal, H. Kunreuther, Modeling interdependent risks, Risk Anal. 27 (2007) 621–634.

[39] J. Honorio, L.E. Ortiz, Learning the structure and parameters of large-population graphical games from behavioral data, CoRR abs/1206.3713, http://arxiv.org/abs/1206.3713, 2013.

[40] M.T. Irfan, Causal strategic inference in social and economic networks, Ph.D. thesis, Stony Brook University, 2013, http://www.bowdoin.edu/~mirfan/papers/Mohammad_Tanvir_Irfan_Dissertation.pdf.

[41] M.T. Irfan, L.E. Ortiz, A game-theoretic approach to influence in networks, in: Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, 2011, pp. 688–694.

[42] E.B. Janovskaja, Equilibrium situations in multi-matrix games, Liet. Mat. Rink. 8 (1968) 381–384.

[43] D.S. Johnson, Approximation algorithms for combinatorial problems, J. Comput. Syst. Sci. 9 (1974) 256–278.

[44] R. Karp, Reducibility among combinatorial problems, in: R. Miller, J. Thatcher (Eds.), Complexity of Computer Computations, Plenum Press, 1972, pp. 85–103.

[45] G. Karypis, V. Kumar, METIS – unstructured graph partitioning and sparse matrix ordering system, Version 2.0, Technical report, Department of Computer Science, University of Minnesota, 1995.

[46] M. Kearns, M. Littman, S. Singh, Graphical models for game theory, in: Proceedings of the Conference on Uncertainty in Artificial Intelligence, 2001, pp. 253–260.

[47] M. Kearns, L.E. Ortiz, Algorithms for interdependent security games, in: S. Thrun, L. Saul, B. Schölkopf (Eds.), Advances in Neural Information Processing Systems, vol. 16, MIT Press, Cambridge, 2004, pp. 561–568.

[48] D. Kempe, J. Kleinberg, E. Tardos, Maximizing the spread of influence through a social network, in: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 2003, pp. 137–146.

[49] J. Kleinberg, Cascading behavior in networks: algorithmic and economic issues, in: N. Nisan, T. Rougarden, Éva Tardos, V.V. Vazirani (Eds.), Algorithmic Game Theory, Cambridge University Press, 2007, pp. 613–632, chapter 24.

[50] H. Kunreuther, G. Heal, Interdependent security, J. Risk Uncertain. 26 (2003) 231–249.

[51] H.C. Kunreuther, E.O. Michel-Kerjan, Assessing, managing and benefiting from global interdependent risks: the case of terrorism and national disasters, in: H.W. Richardson, P. Gordon, J.E. Moore II (Eds.), Global Business and the Terrorist Threat, 2009, pp. 42–73, chapter 4.

[52] G. Le Bon, The Crowd: A Study of the Popular Mind, Macmillian, 1897.

[53] G. Levy, R. Razin, Gradualism in Dynamic Influence Games, London School of Economics and Political Science, 2011.

[54] D.A. Locher, Collective Behavior, Prentice Hall, 2001.

[55] R.D. Luce, H. Raiffa, Games and Decisions: Introduction and Critical Survey, Courier Dover Publications, 1957.

[56] G.T. Marx, D. McAdam, Collective Behavior and Social Movements: Process and Structure, Prentice Hall Foundations of Modern Sociology Series, Prentice Hall, 1994.

[57] C. McPhail, The Myth of the Madding Crowd, Social Institutions and Social Change, A. de Gruyter, 1991.

[58] C. McPhail, Crowd behavior, in: G. Ritzer (Ed.), Blackwell Encyclopedia of Sociology, Blackwell, Oxford, England, 2007, pp. 880–884.

[59] T. Michalak, K. Aaditha, P.L. Szczepanski, B. Ravindran, N.R. Jennings, Efficient computation of the Shapley value for game-theoretic network centrality, J. Artif. Intell. Res. 46 (2013) 607–650.

[60] P. Milgrom, J. Roberts, Rationalizability, learning, and equilibrium in games with strategic complementarities, Econometrica 58 (1990) 1255–1277.

[61] D.L. Miller, Introduction to Collective Behavior and Collective Action, Waveland Press, 2000.

[62] X. Molinero, F. Riquelme, M.J. Serna, Power indices of influence games and new centrality measures for social networks, 2013, CoRR abs/1306.6929.

[63] D. Monderer, L. Shapley, Potential games, Games Econ. Behav. 14 (1996) 124–143.

[64] R.L. Montgomery, Introduction to the Sociology of Missions, Praeger, 1999.

[65] J. Moody, P.J. Mucha, Portrait of political party polarization, Netw. Sci. 1 (2013) 119–121.

[66] S. Morris, Contagion, Rev. Econ. Stud. 67 (2000) 57–78.

[67] R. Narayanam, Y. Narahari, A Shapley value-based approach to discover influential nodes in social networks, IEEE Trans. Autom. Sci. Eng. (2010) 1–18.

[68] J. Nash, Non-cooperative games, Ann. Math. 54 (1951) 286–295.

[69] N. Nisan, Introduction to mechanism design (for computer scientists), in: N. Nisan, T. Rougarden, Éva Tardos, V.V. Vazirani (Eds.), Algorithmic Game Theory, Cambridge University Press, 2007, pp. 209–242, chapter 24.

[70] L.E. Ortiz, M. Kearns, Nash propagation for loopy graphical games, in: S.B. Becker, S.T. Thrun, K. Obermayer (Eds.), Advances in Neural Information Processing Systems, vol. 15, 2003, pp. 817–824.

[71] M.J. Osborne, An Introduction to Game Theory, Oxford University Press, USA, 2003.

[72] R.E. Park, E.W. Burgess, Introduction to the Science of Sociology, The University of Chicago Press, 1921.

[73] R. Raz, S. Safra, A sub-constant error-probability low-degree-test, and a sub-constant error-probability PCP characterization of NP, in: Proceedings of the 29th ACM Symposium on Theory of Computing, STOC, 1997, pp. 475–484.

[74] S.J. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, second edition, Pearson Education, 2003.

[75] S.J. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, third edition, Pearson Education, 2009.

[76] T. Schelling, Micromotives and Macrobehavior, Norton, 1978.

[77] T.C. Schelling, Dynamic models of segregation, J. Math. Sociol. 1 (1971) 143–186.

[78] A. Schrijver, Combinatorial Optimization: Polyhedra and Efficiency, Algorithms and Combinatorics: Study and Research Texts, Springer, 2003.

[79] L.S. Shapley, A value for n-person games, Technical report, 1952, DTIC Document.

[80] C. Tilly, From Mobilization to Revolution, Random House, 1978.

[81] D.M. Topkis, Minimizing a submodular function on a lattice, Oper. Res. 26 (1978) 305–321.

[82] D.M. Topkis, Equilibrium points in nonzero-sum n-person submodular games, SIAM J. Control Optim. 17 (1979) 773–787.

[83] R.H. Turner, L.M. Killian, Collective Behavior, Prentice-Hall Sociology Series, Prentice–Hall, 1957.

[84] J. Ugander, L. Backstrom, C. Marlow, J. Kleinberg, Structural diversity in social contagion, Proc. Natl. Acad. Sci. 109 (2012) 5962–5966.

[85] D. Vickrey, D. Koller, Multi-agent algorithms for solving graphical games, in: Proceedings of the Eighteenth National Conference on Artificial Intelligence, AAAI-02, 2002, pp. 345–351.

[86] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications, Cambridge University Press, 1994.