

# 社会网络中影响力传播的鲁棒抑制方法

李 劲<sup>1,2</sup> 岳 昆<sup>2,3</sup> 张德海<sup>1,2</sup> 刘惟一<sup>3</sup>

<sup>1</sup>(云南大学软件学院 昆明 650091)

<sup>2</sup>(云南省软件工程重点实验室 昆明 650091)

<sup>3</sup>(云南大学信息学院 昆明 650091)

(lijin@ynu.edu.cn)

## Robust Influence Blocking Maximization in Social Networks

Li Jin<sup>1,2</sup>, Yue Kun<sup>2,3</sup>, Zhang Dehai<sup>1,2</sup>, and Liu Weiyl<sup>3</sup>

<sup>1</sup>(School of Software, Yunnan University, Kunming 650091)

<sup>2</sup>(Key Laboratory of Software Engineering of Yunnan Province, Kunming 650091)

<sup>3</sup>(School of Information Science and Engineering, Yunnan University, Kunming 650091)

**Abstract** Influence blocking maximization is currently a problem of great interest in the research area of social networks. Existing influence blocking methods assume negative influence sources are definitely known or non-adversarial. However, in real applications, it is hard to obtain the accurate information of influence sources. In addition, adversarial spreader may strategically select the seeds to maximize negative spread. In this paper, we aim to tackle the problems of influence blocking maximization with uncertain and strategic negative influence sources respectively. First, in order to increase the efficiency of the algorithms for mining blocking seeds, we discuss approximate estimation of the influence spread of negative seeds under the competitive linear threshold model. Based on the estimation, we propose a blocking seeds mining algorithm under the situation of finite uncertain and negative influence sources to maximize the expected influence blocking utility. In adversarial influence spread situations, one entity strategically attempts to maximize negative influence spread while the other one tries to block the negative propagation of its competing entity as much as possible by selecting a number of seed nodes that could initiate its own influence propagation. We model the interaction as a game and propose a polynomial time approximate algorithm for solving random blocking strategy based on min-max principle to decrease the worst negative influence spread. Finally, we make experiments on real data sets of social networks to verify the feasibility and scalability of the proposed algorithms.

**Key words** social networks; influence blocking maximization; minmax principle; approximation algorithms; submodular function

**摘 要** 社会网络中影响力传播的有效抑制是当前社会网络影响力传播机制研究关注的问题之一. 针对不确定性、策略性负影响源的影响力传播抑制, 讨论社会网络中影响力传播的鲁棒抑制问题. 首先, 作为

收稿日期: 2014-12-09; 修回日期: 2015-06-23

基金项目: 国家自然科学基金项目(61562091, 61472345, 61263043); 云南省自然科学基金项目(2014FA023, 201501CF00022); 云南大学骨干教师培养计划基金项目(XT412003); 云南大学创新团队建设项目(XT412011)

This work was supported by the National Natural Science Foundation of China (61562091, 61472345, 61263043), the Natural Science Foundation of Yunnan Province of China (2014FA023, 201501CF00022), the Program for Backbone Teacher Development of Yunnan University (XT412003), and the Program for Innovative Research Team of Yunnan University (XT412011).

通信作者: 岳昆(kyue@ynu.edu.cn)

提高算法运行效率的有效途径,讨论在竞争性线性阈值传播模型下,负种子集传播能力的近似估计方法,以此为基础,提出不确定性负影响源情况下,期望抑制效果最大化的抑制种子集挖掘算法.然后,对于策略性传播源,以最小化最坏情况下的影响力传播范围为目标,基于极小极大优化作为抑制决策准则,提出了一个随机抑制策略的多项式时间近似求解算法.最后,在真实的社会网络数据集上,通过实验验证了所提出方法的有效性.

**关键词** 社会网络;影响力抑制最大化;极小极大原理;近似算法;次模函数

**中图法分类号** TP391

近年来,许多学者针对面向多信息源发布的信息全局传播机制及其预测模型,以影响力传播范围及传播速度最大化为目标的关键结点集选取等问题开展了积极探索,并取得了一系列成果,极大地促进了社会网络影响力传播机制问题的研究<sup>[1-7]</sup>.

然而,由于社会网络本身具有开放性和虚拟性,各种不良、虚假信息、反动言论可以跨地域、跨国界地散布和传播,严重危害社会稳定及国家安全.因此,为有效抑制万维网环境下社会网络负面影响传播,影响力传播抑制问题也引起了学术界的关注,取得了一些初步的研究成果,成为社会网络影响传播机制研究的一个重要方面<sup>[8-11]</sup>.

例如文献[8]扩展了经典的线性阈值模型,给出用于描述社会网络中影响力竞争性传播现象的竞争性线性阈值模型(competitive linear threshold model, CLT),定义影响力传播抑制最大化问题(influence blocking maximization, IBM),提出求解 IBM 问题的高效近似算法.文献[9]扩展了独立级联模型,定义多竞争独立级联模型(multi-campaign independent cascade model, MCICM),及影响力传播限制问题(eventual influence limitation problem, EIL),给出了 MCICM 模型下, EIL 问题目标函数满足次模性(submodular)的充分条件及 EIL 问题的近似求解算法.文献[8-9]的研究工作均关注影响力传播的抑制最大化问题,即最小化竞争对手的影响力传播范围.此外,文献[10]从另一角度研究了影响力传播的抑制,即限定影响力传播范围,求解最小抑制种子集.

不难看出,在解决影响力传播抑制问题时,文献[8-10]都假设在进行抑制决策时影响源已知,在此前提下,将影响力传播抑制问题建模为相应的优化问题,进而提出求解问题的近似优化算法.然而,在实际社会网络应用中,在做出抑制决策前,抑制方往往很难掌握影响源的准确信息,因此无法保证已有算法的抑制效果.其次,上述研究工作忽略了影响力传播方“策略性”传播行为,即传播方可能会根据抑

制方采取的抑制决策,有针对性地、策略性地选择影响力传播源,变更影响力传播的途径,改变影响力传播结果,最终导致抑制方无法取得好的抑制效果.

由于影响力传播与抑制行为的本质是传播方与抑制方之间的一种博弈关系,最近,文献[11]以传播方和抑制方可能选取的种子集作为策略空间,将影响力传播与抑制的对抗行为建模为一个零和博弈(zero sum game)<sup>[12]</sup>,最优抑制决策就是该博弈的 Nash 均衡解.然而,该博弈模型具有指数级的策略空间.为避免对博弈的策略空间进行枚举,文献[11]提出了基于策略生成技术的博弈 Nash 均衡的近似求解算法.虽然这一研究工作首次从博弈论的角度出发研究和分析传播和抑制对抗行为,充分考虑到了影响力传播行为的策略性,然而,基于策略生成技术的求解算法在最坏情况下需要穷举整个策略空间,同时算法不能保证最后收敛到博弈的 Nash 均衡,也无法得到保证近似下界的近似解,因此只适用于规模较小的社会网络.

综上,针对社会网络中影响力传播抑制问题,已有研究工作取得了积极的成果.但是面对不确定性和策略性影响源,如何有效、鲁棒地进行影响力传播抑制,仍是我们面临的研究挑战.对此,针对不确定性影响源、策略性影响源 2 种情况,本文提出社会网络中影响力传播的鲁棒抑制问题以及影响力传播鲁棒抑制的有效解决方法.

本文的研究内容主要包括:

1) 用影响源上的概率分布来描述影响传播源具有的不确定性.为有效提高算法的执行效率,采用基于局部有向无环图(local directed acyclic graph, LDAG)结构的估计方法对影响源传播范围进行高效地近似估计,基于此,以期望抑制效果最大化为目标,提出了针对不确定影响源的抑制种子集挖掘算法.

2) 针对策略性影响源的影响传播抑制问题,基于极小极大决策准则,以最小化最坏情况下的影响

力传播范围为目标, 提出用于影响力传播抑制的多项式时间近似算法.

3) 在真实的社会网络数据集上, 通过实验验证了本文提出方法的有效性. 对于不确定性影响源, 相比较已有方法, 本文的算法在执行效率和抑制效果之间取得了很好的折中; 对于策略性影响源, 本文的算法能够有效抑制最坏情况下的影响力传播范围, 达到了影响力传播的鲁棒抑制效果.

## 1 背景知识

线性阈值模型(linear threshold model, LT)<sup>[13]</sup>是社会网络影响力传播的经典模型. 为描述影响力的竞争性传播现象, He 等人<sup>[8]</sup>对 LT 进行扩展, 提出了竞争性的线性阈值模型. 令  $\theta^+ = (\theta_1^+, \theta_2^+, \dots, \theta_{|V|}^+)$  和  $\theta^- = (\theta_1^-, \theta_2^-, \dots, \theta_{|V|}^-)$  分别代表 CLT 模型中结点集  $V$  上的正和负影响阈值随机向量. 负、正影响分别从种子集  $D, C$  出发, 按照 CLT 模型开始传播, 最终, 被负激活的结点集记为  $V^-(D, C | \theta^-, \theta^+)$ . 由于  $\theta^+$  和  $\theta^-$  是随机向量, 因此, 负激活的结点数  $|V^-(D, C | \theta^-, \theta^+)|$  是一个随机变量, 用该随机变量的期望值作为传播方的负影响传播效用, 记为  $\inf(D, C)$ , 定义如下:

$$\inf(D, C) = E_{\theta^-, \theta^+}(|V^-(D, C | \theta^-, \theta^+)|). \quad (1)$$

在社会网络影响力传播抑制问题中, 存在决策目标对立的双方: 传播方(记为 MAX)旨在通过选择相应的负种子集合  $D$ (也称为负影响源), 使得从  $D$  出发的负影响力传播效用最大化; 抑制方(记为 MIN), 其通过选取相应的正种子集  $C$ , 正影响从  $C$  出发传播, 负、正影响力按照 CLT 模型在网络中传播, 最终, 最小化 MAX 的负影响力传播效用.

令  $\{D\}$  和  $\{C\}$  分别为传播方和抑制方所有可能的负、正种子集组成的集合, 也分别称作传播方、抑制方的策略集. 基于此, 社会网络中影响传播抑制问题定义如下<sup>[8]</sup>:

**定义 1.** 在社会网络  $G$  中, 给定 MAX 的负种子集  $D \in \{D\}$ , 求解 MIN 的最优抑制种子集  $C^* \in \{C\}$ , 使得 MAX 方的负影响力传播效用最小化:

$$C^* = \arg \min_{C \in \{C\}} \inf(D, C). \quad (2)$$

定义 1 是最小化 MAX 的影响传播效用. 此外, 也可用传播效用的抑制最大化来等价定义该问题. 给定  $D$ , 定义种子集  $C$  的抑制效用函数:

$$\sigma(C | D) = \inf(D, \emptyset) - \inf(D, C), \quad (3)$$

于是, 影响力传播抑制问题定义为求解 MIN 的最优

抑制种子集  $C^* \in \{C\}$ , 使得负影响力传播抑制效果最大化:

$$C^* = \arg \max_{C \in \{C\}} \sigma(C | D). \quad (4)$$

文献[8]证明, 给定  $D, \sigma(C | D)$  是结点集  $V$  上的次模函数.

## 2 问题定义

在影响力传播抑制问题中, 抑制方不一定能准确地掌握负影响源的信息, 其所获得的负源信息往往带有不确定性. 设  $\mathcal{P}$  是  $\{D\}$  上的概率分布空间, 用  $\{D\}$  上的概率分布  $P \in \mathcal{P}$  来描述抑制方对于负影响源信息的不确定性(在后文中, 在强调  $P$  是 MIN 所掌握的关于的先验概率信息时, 将  $P$  记为  $P_{\text{prior}}$ ). 于是, 面向不确定性影响源的影响力传播抑制问题, 记为  $RIB_{\text{prior}}$ .

**定义 2.** 在社会网络  $G$  中, 给定 MAX 的负种子集  $\{D\}$  上的概率分布  $P_{\text{prior}}, RIB_{\text{prior}}$  定义为求解 MIN 的最优抑制种子集  $C^* \in \{C\}$ , 使得 MAX 方的影响力传播效用最小化, 即:

$$C^* = \arg \min_{C \in \{C\}} \inf(P_{\text{prior}}, C) = \arg \min_{C \in \{C\}} \sum_D P_{\text{prior}}(D) \inf(D, C). \quad (5)$$

在影响力传播抑制问题中, 双方的决策效果是相互依存的, 最终影响力传播的结果不仅决定于 MAX 选择的影响源, 同时也决定于 MIN 采取的抑制策略. 那么, 在决策效用相互依存的情况下, MIN 应如何决策? 从而实现对 MAX 的影响传播进行有效抑制. 从博弈论的角度来说, 影响传播、抑制对抗行为构成 MAX 和 MIN 之间的一个博弈. 对于 MIN 来说, 如果不能预先获知 MAX 决策行为, 那么, 采取极小极大准则进行抑制决策是合理、有效的选择. 另外, MIN 采取随机抑制策略可以获得更好的期望抑制效用.

综上, 面向策略性影响源的影响力传播抑制问题, 记为  $RIB_{\text{minmax}}$ .

**定义 3.** 令  $\{C\}$  是 MIN 的抑制种子集集合,  $\mathcal{Q}$  是  $\{C\}$  上的概率分布空间. 任意概率分布  $Q \in \mathcal{Q}$  称作 MIN 的随机抑制策略, 即 MIN 按照  $Q$  从  $\{C\}$  中选择种子集  $C$  进行抑制. 面对策略性影响力传播源, MIN 按照极小极大决策准则, 采取随机策略  $Q^* \in \mathcal{Q}$  对 MAX 负影响力传播进行抑制, 使得:

$$Q^* = \arg \min_{Q \in \mathcal{Q}} \max_{P \in \mathcal{P}} \inf(P, Q). \quad (6)$$

需要说明的是: 式(6)假定 MAX 采取的是随机

策略  $P$ , 即 MAX 按照  $P$  选择相应的负种子集作为影响源. 其实, 最优负种子集  $D^* = \arg \max_D \inf(D, Q)$ ,  $Q$  是  $P$  的一个特例, 因此, 求解  $D^*$  的问题:

$$Q^* = \arg \min_{Q \in Q} \max_{D \in \{D\}} \inf(D, Q), \quad (7)$$

只是式(6)的一个特例.

### 3 影响力传播抑制优化算法

#### 3.1 影响力传播效用的近似估计

在  $RIB_{\text{prior}}$  和  $RIB_{\text{minmax}}$  问题中, 均涉及对影响力传播效用  $\inf(D, C)$  进行计算. 由式(1)可知,  $\inf(D, C)$  是关于阈值随机向量  $\theta^+$  和  $\theta^-$  的期望值, 准确估计  $\inf(D, C)$  需要进行大量的随机实验, 计算代价较高.

为有效提高算法的运行效率, 本文采用文献[8]提出的基于 LDAG 的方法对  $\inf(D, C)$  进行近似估计, 即: 基于网中除种子结点之外的所有结点的负激活概率之和作为负影响传播效用的度量标准:

$$\inf(D, C) = \sum_{v \in V \setminus (D \cup C)} p^-(v | D, C), \quad (8)$$

其中,  $p^-(v | D, C)$  是结点  $v$  在负种子集  $D$ , 抑制种子集  $C$  下的负激活概率. 关于  $p^-(v | D, C)$  的计算方法详见文献[8].

在此, 需要说明: 式(1)的值是负激活结点集大小的期望值, 而式(8)的值是负激活结点的负激活概率值之和. 当然, 由结点负激活概率可以获得负激活结点集大小的期望值. 另外, 在式(8)中,  $p^-(v | D, C)$  的计算无需随机仿真, 且计算只涉及结点的有向无环子图  $LDAG_{\varphi}^-(v)$  及  $LDAG_{\varphi}^+(v)$ , 估计的精度可由参数  $\varphi$  进行控制. 因此, 基于 LDAG 对  $\inf(D, C)$  进行估计可以实现算法效率和估计质量之间的有效折中.

#### 3.2 $RIB_{\text{prior}}$ 问题的抑制种子集挖掘算法

首先分析  $RIB_{\text{prior}}$  问题目标函数的性质. 由式(3)定义的抑制效用函数  $\sigma(C | D)$  可知, 给定  $P_{\text{prior}}$ , 有:

$$\begin{aligned} \sigma(C | P_{\text{prior}}) &= \sum_{D \in \{D\}} P_{\text{prior}}(D) \sigma(C | D) = \\ &= \sum_{D \in \{D\}} P_{\text{prior}}(D) (\inf(D, \emptyset) - \inf(D, C)) = \\ &= \inf(P_{\text{prior}}, \emptyset) - \inf(P_{\text{prior}}, C). \end{aligned}$$

由于  $\inf(P_{\text{prior}}, \emptyset)$  是常数,  $RIB_{\text{prior}}$  可等价地描述为

$$C^* = \arg \max_{C \in \{C\}} \sum_{D \in \{D\}} P_{\text{prior}}(D) \sigma(C | D).$$

由文献[8]可知, 函数  $\sigma$  具有次模性, 由于次模函数的非负线性组合仍具有次模性<sup>[14]</sup>, 因此,  $RIB_{\text{prior}}$  问题是次模函数的最大化问题. 于是, 由次

模函数优化方法<sup>[15]</sup>, 基于贪心法进行  $k$  次迭代可求得保证质量下界的近似最优抑制种子集  $\tilde{C}^*$ . 然而, 基于贪心算法求解需要评价结点的边际抑制效用, 这需要进行大量的随机仿真实验, 导致算法效率低.

基于 3.1 节介绍的影响力传播效用的近似估计方法, 当 MAX 的负种子集  $\{D\}$  是有限 (设包含  $m$  个负种子集) 的情况下, 讨论期望抑制效用最大化的抑制种子集挖掘算法. 设  $\{D\}_m = \{D_1, D_2, \dots, D_m\}$  ( $D_i \subseteq V$ ) 是包含  $m$  个负种子集的集合.  $P_m$  是  $\{D\}_m$  上的一个概率分布. 随机策略  $P_m$  的期望影响力传播效用可用负激活概率表示为

$$\begin{aligned} \inf(P_m, C) &= \sum_{D \in \{D\}_m} P_m(D) \inf(D, C) = \\ &= \sum_{D \in \{D\}_m} P_m(D) \sum_{v \in V \setminus (D \cup C)} p^-(v | D, C) = \\ &= \sum_{v \in V \setminus C} \sum_{D \in \{D\}_m} P_m(D) p^-(v | D, C). \end{aligned}$$

为算法描述方便, 引入记号  $\gamma^-(v | P_m, C)$ , 即在  $P_m$  和  $C$  下, 任意结点  $v$  的负激活概率的期望值:

$$\gamma^-(v | P_m, C) = \sum_{D \in \{D\}_m} P_m(D) (p^-(v | D, C)),$$

于是, 可重写:

$$\inf(P_m, C) = \sum_{v \in V \setminus C} \gamma^-(v | P_m, C).$$

也就是说,  $\inf(P_m, C)$  是结点负激活概率的期望值之和. 算法 1 给出了期望抑制效用最大化的抑制种子集挖掘算法, 其基本思想如下:

步骤 1. 初始化阶段. 对于网中每个结点  $v$ , 初始化与其负激活概率计算相关的子图结构. 对于网中的每个结点  $u$ , 初始化其单独作为抑制结点时的抑制效用.

步骤 2. 抑制种子挖掘阶段. 本阶段从网中挖掘  $k$  个抑制种子. 算法的执行过程与采用贪心法挖掘抑制种子集是一致的, 即每次迭代, 从  $V \setminus C$  中选取最大边际抑制效用值的结点作为抑制种子加入到  $C$  中. 每次迭代完成后, 更新各个结点的边际抑制效用值. 经过  $k$  次迭代完成抑制种子集的挖掘. 不过, 与贪心法不同的是: 算法 1 采用  $\gamma^-$  值来度量抑制效用值.

**算法 1.**  $RIB_{\text{prior}}$  问题的抑制种子集挖掘算法  $A_{\text{prior}}$ .

输入:  $G = [V, E]$ ,  $\{D\}_m$ ,  $\{D\}_m$  上的概率分布  $P_m$ , LDAG 结构控制参数  $\varphi$ , 抑制种子集大小  $k$ ;

输出: 期望抑制效用最大化的抑制种子集  $C_m^*$ .

①  $C \leftarrow \emptyset$ ;

② for each  $v \in V$  do

③ 基于  $\varphi$  分别构建  $LDAG_{\varphi}^{+}(v)$ ,  
 $LDAG_{\varphi}^{-}(v), OLS_{\varphi}^{+}(v)$ ;  
 ④ end for  
 ⑤ for each  $u \in V$  do  
 ⑥  $\Delta^{-}(u) \leftarrow 0$ ;  
 ⑦ end for  
 ⑧ for each  $v \in V$  do  
 ⑨ for each  $u \in LDAG_{\varphi}^{+}(v)$  do  
 ⑩  $before \leftarrow \gamma^{-}(v | P_m, \emptyset), after \leftarrow \gamma^{-}(v | P_m, \{u\}), \Delta^{-}(u) \leftarrow \Delta^{-}(u) + (before - after)$ ;  
 ⑪ end for  
 ⑫ end for  
 ⑬ for each  $i = 1, 2, \dots, k$  do  
 ⑭  $c \leftarrow \arg \max_{v \in V \setminus C} (\Delta^{-}(v))$ ;  
 ⑮ for each  $s \in OLS_{\varphi}^{+}(c)$  do /\* 更新  
 $LDAG_{\varphi}^{+}(s)$  中结点的边际抑制效用 \*/  
 ⑯ for each  $u \in LDAG_{\varphi}^{+}(s)$  do  
 ⑰  $\Delta^{-}(u) \leftarrow \Delta^{-}(u) - (\gamma^{-}(s | P_m, C \cup \{u\}) - \gamma^{-}(s | P_m, C))$ ,  
 $\Delta^{-}(u) \leftarrow \Delta^{-}(u) + (\gamma^{-}(s | P_m, C \cup \{c, u\}) - \gamma^{-}(s | P_m, C \cup \{c\}))$ ;  
 ⑱ end for  
 ⑲ end for  
 ⑳  $C \leftarrow c$ ; /\* 添加新的抑制正种子 \*/  
 ㉑ end for  
 ㉒ return  $C$ .

算法 1 中子图  $OLS_{\varphi}^{+}(v)$  是以结点  $v$  作为源点的有向无环子图. 任意结点  $s \in OLS_{\varphi}^{+}(v)$ ,  $v$  对  $s$  有正影响.  $\Delta^{-}(u)$  是任意结点  $u$  作为正种子时的边际抑制效用值. 行 ⑥~⑩ 初始化每一个结点  $u \in V$  单独作为正种子时的抑制效用.

行 ⑪~⑲ 是  $k$  个正种子的挖掘过程, 每次迭代, 先从  $V \setminus C$  中选取当前具有最大边际抑制效用值的结点  $c$  加入到正种子集  $C$ . 当  $c$  加入  $C$  后, 所有  $c$  对其有正影响的结点  $s \in OLS_{\varphi}^{+}(c)$  的所有正影响父亲结点  $u \in LDAG_{\varphi}^{+}(s)$  的边际期望抑制效用均可能发生变化, 于是, 通过行 ⑬~⑰ 更新  $u \in LDAG_{\varphi}^{+}(s)$  结点的  $\Delta^{-}(u)$  值. 图 1 给出了结点  $c$  的  $LDAG$  子图、 $OLS$  子图以及边际抑制效用更新示意图.

### 3.3 $RIB_{\minmax}$ 问题的随机抑制策略求解算法

由定义 3 可知,  $RIB_{\minmax}$  问题归结为一个极小极大(minmax)优化问题. 求解 MIN 的极小极大随机抑制策略  $Q^*$  等价于: MIN 和 MAX 分别采用随机策略  $P$  和  $Q$ , 并以  $\inf(P, Q)$  作为决策效用函数进行零和博弈, 而  $Q^*$  就是 MIN 的 Nash 均衡解.

零和博弈的 Nash 均衡解可以通过 2 种方法进行求解, 即基于线性规划求解<sup>[12]</sup> 和基于迭代求解<sup>[16]</sup>. 然而, 采用上述 2 种方法求解  $RIB_{\minmax}$  问题, 存在 2 个困难.

1) 基于线性规划的求解方法. 首先, 在  $RIB_{\minmax}$  问题中, 负种子集  $D \in \{D\}$  和抑制种子集  $C \in \{C\}$  的数目与网络中的结点呈指数关系增长. 于是, 基于线性规划求解  $Q^*$  需要枚举指数级的约束关系且线性方程中含有指数级个数的变量. 其次, 线性规划方法需要预先确定博弈效用矩阵中的效用值, 也就是说在求解  $RIB_{\minmax}$  问题时, 对于任意  $D \in \{D\}$  和  $C \in \{C\}$ , 需先确定  $\inf(D, C)$  值, 当  $\{D\}$  或  $\{C\}$  的数目大时, 显然是不现实的.

2) 基于迭代的求解方法. 迭代方法在算法迭代求解过程中逐步生成效用矩阵, 克服了规划求解方法面临的困难. 然而, 该方法的收敛性无法得到保证. 同时, 即便算法收敛, 也无法保证所得解的求解质量<sup>[16]</sup>.

#### 算法 2. 随机抑制策略的求解算法 $A_{\minmax}$ .

输入:  $G = \langle V, E \rangle, 0 < \mu \leq 1$ ,  $LDAG$  结构控制参数  $\varphi$ 、拟挖掘的正种子集大小  $k$ 、算法终止的阈值  $\epsilon > 0$ ;

输出: MIN 的近似随机抑制策略  $\hat{Q}$ .

① 初始化 MAX 的传播策略权重:

$$\forall D \in \{D\}_m, w_D^{(1)} = 1;$$

② for  $t = 1$  to  $T = \lceil 4 \ln(m) / \epsilon^2 \rceil$

③ 生成 MAX 随机策略  $P^{(t)}$ , 即:

$$\Theta^{(t)} = \sum_D w_D^{(t)}, \forall D \in \{D\}_m,$$

$$P^{(t)}(D) = w_D^{(t)} / \Theta^{(t)};$$

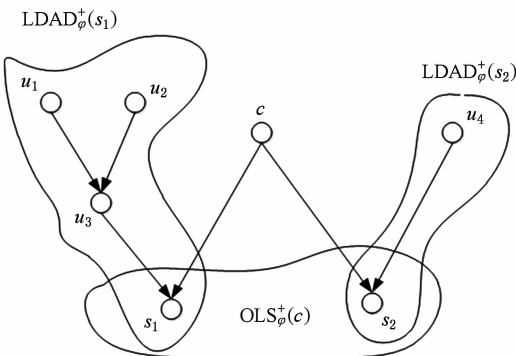


Fig. 1 Update of the nodes' marginal blocking utility.

图 1 边际抑制效用更新示意图

- ④ 针对随机策略  $P^{(t)}$ , 基于算法 1 求解 MIN 的近似抑制种子集  
$$\tilde{C}^{(t)} = \arg \min_C \inf(P^{(t)}, C);$$
- ⑤ 计算  $\forall D \in \{D\}_m$  在  $\tilde{C}^{(t)}$  下的传播效用  $\inf(D, \tilde{C}^{(t)})$ , 并进行归一化处理, 即:  
$$\inf(D, \tilde{C}^{(t)}) \leftarrow \inf(D, \tilde{C}^{(t)}) / |V|;$$
- ⑥ 更新 MAX 传播策略的权重:  
$$\forall D \in \{D\}_m, w_D^{(t+1)} = w_D^{(t)} (1 - \mu \times \inf(D, \tilde{C}^{(t)})), \Theta^{(t+1)} = \sum_D w_D^{(t+1)};$$
- ⑦  $t = t + 1$ ;
- ⑧ end for
- ⑨ 基于抑制种子集序列  $\tilde{C}^{(1)}, \tilde{C}^{(2)}, \dots, \tilde{C}^{(T)}$  生成 MIN 的近似抑制策略  $\hat{Q}$ , 其中,  $\hat{Q}(C) = |t: \tilde{C}^{(t)} = C| / T$ ; (式中分子是种子集  $C$  在序列中出现的次数)
- ⑩ return  $\hat{Q}$ .

在假定 MAX 的传播策略集  $\{D\}$  含有有限的  $m$  个负种子集的情况下 (记为  $\{D\}_m$ ), 提出一个求解  $Q^*$  的多项式时间的近似算法, 即算法 2. 其基本思想是: 时刻  $t$ , MAX 按照一定的随机策略选择负种子集, 针对该随机策略, MIN 选择最优抑制种子集  $C^{(t)} \in \{C\}$  进行抑制. 在时刻  $t+1$ , 根据  $D \in \{D\}_m$  在  $C^{(t)}$  的传播效用, MAX 更新  $D$  的权重, 即传播效用越大的  $D$ , 获得越大的权重, 并按照各策略的在前  $t$  个时刻的累积权重来确定时刻  $t+1$  的随机传播策略  $P^{(t)}$ . 完成  $T$  次博弈后, 由  $T$  个抑制种子集  $C^{(1)}, C^{(2)}, \dots, C^{(T)}$  确定  $\{C\}$  上的一个随机抑制策略, 该随机抑制策略就是最优随机抑制策略的近似解.

对算法 2 进行说明: 步骤④即 MIN 对  $P^{(t)}$  的“最优响应”. 在双方进行第  $t$  次博弈时, MAX 采取  $P^{(t)}$  作为随机策略 (初始时  $P^{(t)}$  是均匀分布, 即  $P^{(1)} = 1/m$ ). MIN 对  $P^{(t)}$  进行最优响应, 即 MIN 采取  $C^{(t)}$  对 MAX 进行抑制:

$$C^{(t)} \leftarrow \arg \min_C \inf(P^{(t)}, C).$$

(9)

由算法 1 可知, 求解  $C^{(t)}$  归结为一个次模函数的最大化问题, 因此, 存在保证近似下界的近似算法, 例如基于贪心法的次模函数的最大化求解方法. 于是, 由算法 1 可获得  $C^{(t)}$  的  $1/\beta$  (近似抑制策略  $\tilde{C}^{(t)}$  ( $\beta = 1 - 1/e$ ), 满足:

$$\inf(P^{(t)}, \tilde{C}^{(t)}) \leq (1/\beta) \inf(P^{(t)}, C^*),$$

(10)

其中,  $C^*$  是对  $P^{(t)}$  的最优响应.

算法 2 的优势在于: 1) 无需枚举 MIN 的策略集  $\{C\}$  (对于含  $n$  个结点的网络, 选取  $k$  个抑制结点作为抑制种子集, 那么  $\{C\}$  大小是从  $n$  中选  $k$  的组合数), 且近似解  $\hat{Q}$  中的非零元是有限的, 至多有  $T$  个, 相比较  $\{C\}$  的大小,  $\hat{Q}$  是  $\{C\}$  上的一个稀疏的概率分布; 2) 求解过程无需预先计算所有的  $\inf(D, C)$  值; 3) 在多项式时间内结束, 解的近似质量取决于算法的迭代次数.

综上, 算法 2 克服了已有的线性规划和迭代方法求解  $Q^*$  所面临的困难, 给出了求解最优随机抑制策略的有效近似方法.

定理 1 证明了算法 2 在多项时间内可求得  $RIB_{\min\max}$  问题的随机近似抑制策略.

**定理 1.** 给定近似因子  $0 < \epsilon < 1$ . 设  $m$  是  $\{D\}_m$  中负种子集的个数. 在算法 2 中, 时刻  $t$ , MAX 采取随机传播策略  $P^{(t)}$ , MIN 基于  $1/\beta$  (近似算法求解抑制种子集  $\tilde{C}^{(t)}$  对  $P^{(t)}$  响应. 那么, 经过  $T = \lceil 4 \ln m / \epsilon^2 \rceil$  次迭代后可获得求解  $RIB_{\min\max}$  问题的近似解  $\hat{Q}$ , 使得:

$$\max_P \inf(P, \hat{Q}) \leq (1/\beta) \min_{Q \in \mathcal{Q}} \max_{P \in \mathcal{P}} \inf(P, Q) + \epsilon,$$

其中,  $\beta = 1 - 1/e$ .

定理 1 的证明见附录 A.

## 4 实验结果

### 4.1 实验环境设置

所有实验在 NetHEPT 和 NetPHY 两个真实数据集上完成 (数据下载自文献[17]). 按照文献[8]描述的方式, 我们对数据集进行了预处理, 并采用广度优先搜索的方法, 从完成预处理后的 NetHEPT 和 NetPHY 中抽取了相应的连通子图. 表 1 给出了实验中所用到的子图数据的统计信息:

Table 1 Statistics Information of the Experimental Data

表 1 实验数据集的统计信息

| Parameter       | HEPT-600 | HEPT-6000 | PHY-600 | PHY-6000 |
|-----------------|----------|-----------|---------|----------|
| Number of Nodes | 600      | 6 000     | 600     | 6 000    |
| Number of Edges | 1102     | 16286     | 957     | 31 201   |
| Min in-Degree   | 0        | 0         | 0       | 0        |
| Max in-Degree   | 17       | 60        | 92      | 152      |
| Ave in-Degree   | 1.903    | 2.874     | 1.564   | 4.822    |
| Min out-Degree  | 0        | 0         | 0       | 0        |
| Max out-Degree  | 32       | 44        | 96      | 131      |
| Ave out-Degree  | 1.903    | 2.874     | 1.564   | 4.822    |

所有实验程序采用 C++ 语言编写, 编译器版本 gcc 4.8.2. 实验运行环境: Intel Xeon CPU E5620 2.4 GHz, 8GB 内存的服务器、操作系统 Linux Ubuntu 14.04 LTS.

4.2 算法的抑制效果

4.2.1 抑制效果的度量标准

在算法 1 和算法 2 中, 算法的目标函数定义为所有结点的负激活概率的期望值之和. 在求解抑制种子集(或随机抑制策略)后, 为将抑制效用与式(1)的定义保持一致, 求得每个结点在给定正、负种子情况下的负激活概率, 并以此为基础进行重复实验 1000 次, 以随机实验时负激活的结点集大小的平均值, 作为影响传播的抑制效果.

4.2.2  $A_{prior}$  算法的抑制效果

为验证算法 1 的有效性, 将其与其他抑制算法进行了对比实验. 1) Greedy 算法. 基于贪心法挖掘抑制种子集, 在每次挖掘抑制种子时, 采用 10 000 次蒙特卡罗仿真来估计结点的期望边际抑制效用. 2) Random 算法. 等概率从图中选取大小为  $k$  的抑制种子集. 3) Degree 算法以结点度为概率权重, 随机选取  $k$  个抑制种子构成抑制种子集.

考虑到 Greedy 算法的运行效率, 只在 2 个子图: NetHEPT-600 和 NetPHY-600 进行实验. 首先, 按照均匀分布, 随机地从子图结点集  $V$  中抽取 20 个结点生成负种子集  $D_i$ , 一共生成 100 个负种子集  $D_i$  构成  $\{D\}_{m=100}$ , 记  $P_{prior} = 1/m$  是  $\{D\}_{m=100}$  上的均匀分布. 其次, 对抑制种子集  $C = 10, 20, 30, 40, 50, 60$  执行 4 种抑制种子集挖掘算法, 并对比期望抑制效果. 其中,  $A_{prior}$  算法中参数  $\varphi = 0.01$ .

图 2 和图 3 分别给出了 4 种算法在 NetHEPT-600 和 NetPHY-600 数据集上的期望抑制效果对比

结果. 图 4 给出了 Greedy 算法和  $A_{prior}$  算法分别在 HEPT 和 PHY 数据集上的运行时间对比结果.

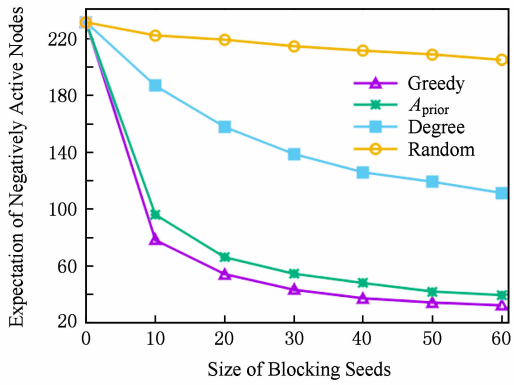


Fig. 3 Comparison of the blocking results on NetPHY-600.

图 3 NetPHY-600 上不同算法抑制效果比较

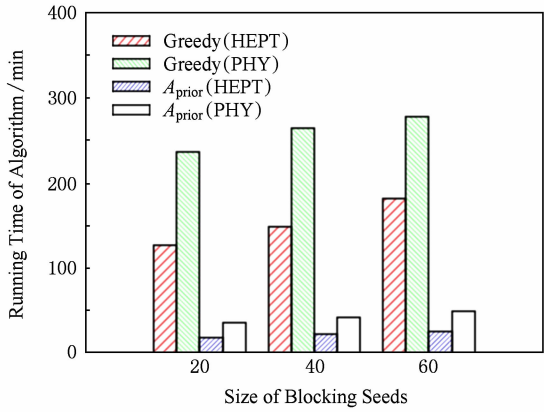


Fig. 4 Running time comparison of Greedy and  $A_{prior}$  algorithms.

图 4 Greedy 与  $A_{prior}$  算法运行时间比较

由实验结果可知, 相比较 Random, Degree 两种算法, Greedy,  $A_{prior}$  算法在抑制效果上均有显著的提升. 其中, Greedy 算法具有最好的抑制效果, 但该算法在执行过程中需要通过随机仿真来评价结点的边际抑制效用, 运行时间较长, 不适用于大尺寸数据集.  $A_{prior}$  算法在抑制效果上与 Greedy 算法差距不大, 且该算法采用基于 LDAG 结构的方法对结点的边际抑制效用进行近似评估, 算法运行时间显著减少. 因此,  $A_{prior}$  算法在抑制效果和算法运行时间 2 方面取得了很好的折中.

4.2.3  $A_{minmax}$  算法的抑制效果

首先, 为直观展示  $A_{minmax}$  算法的抑制效果, 将其与  $A_{prior}$  算法进行了对比. 其次, 为说明  $A_{minmax}$  算法在求解抑制策略上的有效性和优势, 将其与文献[11]提出的基于迭代的零和博弈求解算法(double oracle

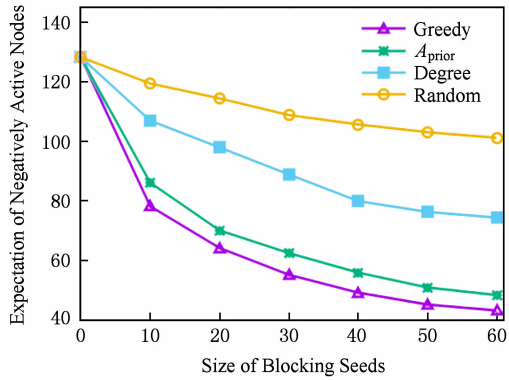


Fig. 2 Comparison of the blocking results on NetHEPT-600.

图 2 NetHEPT-600 上不同算法抑制效果比较



algorithm,  $A_{\text{oracle}}$ ) 进行对比. 实验采用 NetHEPT-6000 和 NetPHY-6000 数据集.

对比  $A_{\text{prior}}$  和  $A_{\text{minmax}}$  算法的抑制效果的实验过程如下: 首先, 按照均匀分布, 随机地从结点集  $V$  中抽取 100 个结点组成负种子集  $D_i$ , 共生成 100 个负种子集  $D_i$  构成  $\{D\}_{m=100}$ . 基于  $A_{\text{prior}}$ ,  $A_{\text{minmax}}$  算法分别挖掘  $C$  取 50, 100, 150, 200, 250 大小的抑制种子集或随机抑制策略. 其中, 在  $A_{\text{prior}}$  算法中, 参数  $\varphi = 0.01$ ; 在  $A_{\text{minmax}}$  算法中, 参数  $\varphi = 0.01, \epsilon = 0.1$ . 由于  $A_{\text{minmax}}$  算法得到的是随机抑制策略  $\hat{Q}$ , 在评价  $A_{\text{minmax}}$  算法的抑制效果时, 重复实验 1000 次, 每次实验中, 以  $\hat{Q}$  为概率分布选择相应的抑制种子集  $C$ . 计算每一个  $D_i \in \{D\}_{m=100}$  在  $C$  下的负激活的结点数目. 最后, 以 1000 次重复实验后每一个  $D_i \in \{D\}_{m=100}$  的负激活的结点数目的平均值来作为  $D_i$  的传播效用.

图 5 直观地展示  $A_{\text{minmax}}$  算法对最具传播能力的负种子集的传播抑制效果. 对于 NetPHY-6000 数据集, 当每个抑制种子集大小  $k = 50$  时,  $\{D\}_m$  中 100 个不同的负种子集分别在: 无抑制种子集、 $A_{\text{prior}}$  算法的抑制种子集以及  $A_{\text{minmax}}$  算法的随机抑制策略下的负激活结点(期望)数目分布的情况. 3 种情况下负激活结点数目的平均值分别为: 1003.15, 597.54, 628.21. 此外, 图 5 中用箭头标示了  $A_{\text{prior}}$ ,  $A_{\text{minmax}}$  算法进行抑制时, 最大的负激活结数目(其中,  $A_{\text{prior}}$  算法抑制时, 最大负激活结数目为 1121, 而  $A_{\text{minmax}}$  算法抑制数目为 896).

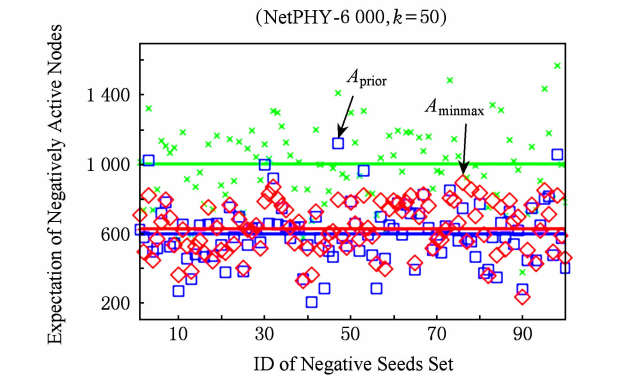


Fig. 5 Demonstration of blocking result of  $A_{\text{prior}}$  and  $A_{\text{minmax}}$  on NetPHY-6000.

图 5  $A_{\text{prior}}$  和  $A_{\text{minmax}}$  算法在 NetPHY-6000 的抑制结果示例

文献[11]采用线性规划和  $A_{\text{oracle}}$  算法来求解博弈的 Nash 均衡解. 由于数据集包含 6000 个结点, 因此, 采用函数 pagerank-oracle 作为  $A_{\text{oracle}}$  算法, 线性规划则采用软件包 CPLEX 12.6<sup>[18]</sup> 完成. 在 NetHEPT-6000 和 NetPHY-6000 数据集上, 对比实验的结果

相似, 因此, 只给出在 NetPHY-6000 的结果. 此外, 当抑制种子集大小超过 150 时,  $A_{\text{oracle}}$  算法无法在有限时间内输出结果, 于是只给出抑制种子集在 150 以内的对比结果.

图 6 给出了  $A_{\text{oracle}}$  和  $A_{\text{minmax}}$  算法的抑制效果对比结果. 当抑制结点数少于 70 时,  $A_{\text{oracle}}$  算法的平均抑制效果和最差情况的抑制效果要优于  $A_{\text{minmax}}$  算法. 随着抑制结点数增加 2 个算法的期望抑制效果相近, 而对于最差情况的抑制效果  $A_{\text{minmax}}$  算法明显比  $A_{\text{oracle}}$  算法稳定. 主要原因是随着抑制结点数的增加,  $A_{\text{oracle}}$  算法不能保证收敛, 这影响了算法的平均抑制效果. 图 7 给出了  $A_{\text{oracle}}$  和  $A_{\text{minmax}}$  算法的求解时间对比结果.  $A_{\text{minmax}}$  算法运行时间基本随抑制结点集的大小线性增长, 然而,  $A_{\text{oracle}}$  算法的运行时间随抑制结点集大小快速增长. 综上, 对于小数据集而言,  $A_{\text{oracle}}$  算法有着不错的抑制效果和合理的求解时间, 然而, 对于大数据集来说, 在抑制效果和求解时间上  $A_{\text{minmax}}$  算法更有优势.

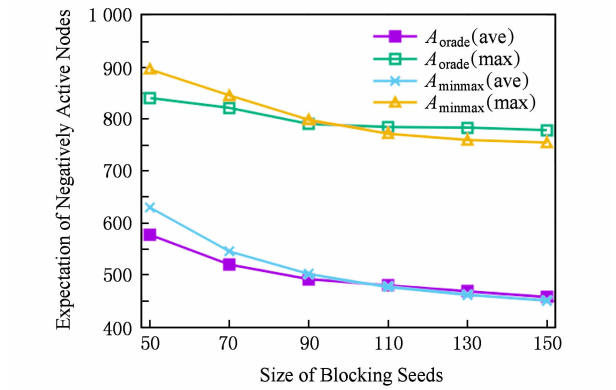


Fig. 6 Comparison of the worst-case and average-case blocking results of  $A_{\text{oracle}}$  and  $A_{\text{minmax}}$  on NetPHY-6000.

图 6  $A_{\text{oracle}}$  和  $A_{\text{minmax}}$  算法 NetPHY-6000 平均及最差抑制效果对比

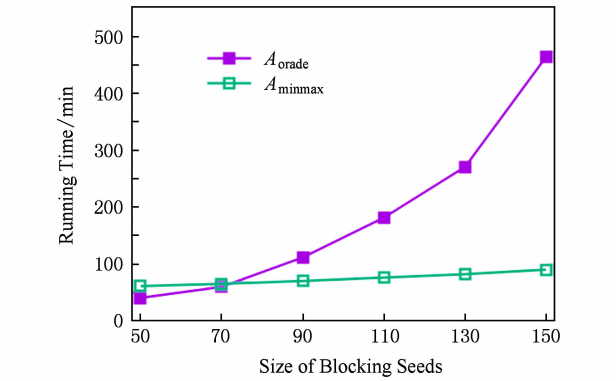


Fig. 7 Running time comparison of  $A_{\text{oracle}}$  and  $A_{\text{minmax}}$  on NetPHY-6000.

图 7  $A_{\text{oracle}}$  和  $A_{\text{minmax}}$  算法在 NetPHY-6000 运行时间对比



## 5 结束语

影响力传播的建模和分析及其控制是当前社会网络研究的重要内容. 其中, 影响传播的有效抑制是当前研究面临的一个新的挑战. 针对实际应用环境中, 不确定性负影响源、策略性负影响源 2 种情况, 本文提出了社会网络中负面影响传播的鲁棒抑制问题. 针对不确定性负影响源的情况, 以期望抑制效果最大化为目标, 提出了抑制种子集挖掘算法; 针对策略性负影响源的情况, 以极小极大优化作为决策准则, 提出了多项式时间的随机抑制策略的近似求解算法. 最后, 在实际社会网络数据集的实验结果验证了本文提出算法的有效性.

在本文研究的基础上, 未来可以在 3 个方面继续开展研究. 1) 本文中影响力传播模型是对线性阈值模型扩展而得, 而对于其他典型的传播模型, 例如独立级联模型进行扩展, 进而提出相应的影响力竞争性传播模型, 并提出该模型下的针对不确定性负影响源、策略性负影响源的影响传播抑制算法; 2) 社会网络的图结构对影响传播和抑制有很大的影响, 深入研究图结构与影响传播、抑制之间的关系, 并提出基于图结构信息的负面影响传播的有效抑制算法; 3) 针对大尺寸社会网络数据, 如何基于图抽样的方法<sup>[19]</sup>进一步在抑制策略的求解效率和求解质量之间进行有效折中, 这些都是将来值得研究的课题.

## 参 考 文 献

- [1] Wu Xindong, Li Yi, Li Lei. Influence analysis of online social networks [J]. Chinese Journal of Computers, 2014, 37(4): 735-752(in Chinese)  
(吴信东, 李毅, 李磊. 在线社交网络影响力分析[J]. 计算机学报, 2014, 37(4): 735-752)
- [2] Yang Shiqiang, Cui Peng. The Information propagation model of social media [J]. Communications of CCF, 2011, 7(12): 20-24 (in Chinese)  
(杨士强, 崔鹏. 社区媒体信息传播机制[J]. 中国计算机学会通讯, 2011, 7(12): 20-24)
- [3] Leskovec J, Krause A, Guestrin C, et al. Cost-effective outbreak detection in networks [C] //Proc of the 13th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2007: 420-429
- [4] Tian Jiatang, Wang Yitong, Feng Xiaojun. A new hybrid algorithm for influence maximization in social networks [J]. Chinese Journal of Computers, 2011, 34(10): 1956-1965 (in Chinese)  
(田家堂, 王铁彤, 冯小军. 一种新型的社会网络影响最大化算法 [J]. 计算机学报, 2011, 34(10): 1956-1965)
- [5] Chen Hao, Wang Yitong. Threshold-based heuristic algorithm for influence maximization [J]. Journal of Computer Research and Development, 2012, 49(10): 2181-2188 (in Chinese)  
(陈浩, 王铁彤. 基于阈值的社交网络影响力最大化算法 [J]. 计算机研究与发展, 2012, 49(10): 2181-2188)
- [6] Chen W, Yuan Y, Zhang L. Scalable influence maximization in social networks under the linear threshold model [C] //Proc of the 10th IEEE Int Conf on Data Mining. Piscataway, NJ: IEEE, 2010: 88-97
- [7] Goyal A, Lu W, Lakshmanan L V S. Simpath: An efficient algorithm for influence maximization under the linear threshold model [C] //Proc of the 11th IEEE Int Conf on Data Mining. Piscataway, NJ: IEEE, 2011: 211-220
- [8] He X, Song G, Chen W, et al. Influence blocking maximization in social networks under the competitive linear threshold model [C] //Proc of the SIAM Int Conf on Data Mining. Philadelphia, PA: SIAM, 2012: 463-474
- [9] Nguyen N P, Yan G, Thai M T, et al. Containment of misinformation spread in online social networks [C] //Proc of the 3rd Annual ACM Web Science Conf. New York: ACM, 2012: 213-222
- [10] Budak C, Agrawal D, El Abbadi A. Limiting the spread of misinformation in social networks [C] //Proc of the 20th Int Conf on World Wide Web. New York: ACM, 2011: 665-674
- [11] Tsai J, Nguyen T H, Weller N, et al. Game-theoretic target selection in contagion-based domains [J]. The Computer Journal, 2014, 57(6): 893-905
- [12] Myerson R B. Game Theory: Analysis of Conflict [M]. Cambridge, MA: Harvard University Press, 1997
- [13] Kempe D, Kleinberg J, Tardos É. Maximizing the spread of influence through a social network [C] //Proc of the 9th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2003: 137-146
- [14] Fujishige S. Submodular Functions and Optimization [M]. Amsterdam: Elsevier Science Press, 2005
- [15] Nemhauser G L, Wolsey L A, Fisher M L. An analysis of approximations for maximizing submodular set functions [J]. Mathematical Programming, 1978, 14(1): 265-294
- [16] Halvorson E, Conitzer V, Parr R. Multi-step multi-sensor hide-seeker games [C] //Proc of the 21st Int Joint Conf on Artificial Intelligence. San Francisco: Morgan Kaufmann, 2009: 159-166
- [17] Chen W. Social network dataset [EB/OL]. Beijing: Microsoft Research Laboratory, 2010 [2014-12-09]. <http://research.microsoft.com/en-us/people/weic/projects.aspx>
- [18] IBM Corporation. CPLEX Optimizer [CP]. New York: IBM Corporation, 2009 [2014-12-09]. <http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/index.html>
- [19] Ribeiro B, Towsley D. Estimating and sampling graphs with multidimensional random walks [C] //Proc of the 10th ACM SIGCOMM Conf on Internet Measurement. New York: ACM, 2010: 390-403
- [20] Freund Y, Schapire R E. Adaptive game playing using multiplicative weights [J]. Games and Economic Behavior, 1999, 29(1): 79-103



**Li Jin**, born in 1975. PhD and associate professor. Member of China Computer Federation. His current research interests include massive data analysis and machine learning.



**Yue Kun**, born in 1979. PhD and professor. Member of China Computer Federation. His current research interests include massive data analysis and uncertainty in artificial intelligence.



**Zhang Dehai**, born in 1977. PhD and associate professor. His current research interests include knowledge engineering, machine learning.



**Liu Weiyi**, born in 1950. Professor and PhD supervisor in the Yunnan University. His current research interests include intelligent data analysis, social computing, and knowledge engineering.

#### 附录 A. 正文中定理 1 的证明.

**定理 A1.** 即正文中的定理 1. 给定近似因子  $0 < \epsilon < 1$ . 设  $m$  是  $\{D\}_m$  中负种子集的个数. 在算法 2 中, 时刻  $t$ , MAX 采取随机传播策略  $P^{(t)}$ , MIN 基于  $1/\beta$  (近似算法求解抑制种子集  $\tilde{C}^{(t)}$  对  $P^{(t)}$  响应. 那么, 经过  $T = \lceil 4 \ln m / \epsilon^2 \rceil$  次迭代后可获得求解  $RIB_{\min\max}$  问题的近似解  $\hat{Q}$ , 使得:

$$\max_P \inf(P, \hat{Q}) \leq (1/\beta) \min_{Q \in \mathcal{Q}} \max_{P \in \mathcal{P}} \inf(P, Q) + \epsilon,$$

其中,  $\beta = 1 - 1/e$ .

证明. 算法 2 可以看作是 MAX, MIN 双方  $T$  轮重复博弈的过程. 因此, 由文献[20]结论可知, 对于 MAX, 如果其采取算法 2 与 MIN 进行重复博弈, 其  $T$  轮博弈的累积期望传播效用:

$$\sum_{t=1}^T \inf(P^{(t)}, \tilde{C}^{(t)}) \geq (1 - \mu) \sum_{t=1}^T \inf(P, \tilde{C}^{(t)}) - \ln m / \mu. \quad (A1)$$

其中,  $0 < \mu < 1$ ,  $P$  是  $\{D\}$  上的任意概率分布, 也就是 MAX 的任意随机策略.

由于算法 2 中传播效用是归一化的, 即  $\forall P, \tilde{C}^{(t)}, 0 \leq \inf(P, \tilde{C}^{(t)}) < 1$ . 将式(A1)两边同时除以  $T$ , 可得:

$$\begin{aligned} \frac{1}{T} \sum_{t=1}^T \inf(P^{(t)}, \tilde{C}^{(t)}) &\geq \frac{1}{T} \sum_{t=1}^T \inf(P, \tilde{C}^{(t)}) - \\ &\mu \frac{1}{T} \sum_{t=1}^T \inf(P, \tilde{C}^{(t)}) - \frac{\ln m}{\mu T} \geq \\ &\frac{1}{T} \sum_{t=1}^T \inf(P, \tilde{C}^{(t)}) - \mu - \frac{\ln m}{\mu T}. \end{aligned} \quad (A2)$$

由于式(A2)对于  $\{D\}$  上任意概率分布  $P$  均成立, 于是, 将式(A2)中随机策略  $P$  替换为 MAX 的

最优随机传播策略  $P^* = \arg \max_{P \in \mathcal{P}} \inf(P, \tilde{C}^{(t)})$ , 有:

$$\begin{aligned} \frac{1}{T} \sum_{t=1}^T \inf(P^{(t)}, \tilde{C}^{(t)}) &\geq \\ \frac{1}{T} \max_{P \in \mathcal{P}} \sum_{t=1}^T \inf(P, \tilde{C}^{(t)}) - \mu - \frac{\ln m}{\mu T}. \end{aligned} \quad (A3)$$

由于  $\tilde{C}^{(t)}$  是 MIN 对 MAX 的随机策略  $P^{(t)}$  作出的  $1/\beta$  近似响应, 因此:

$$\inf(P^{(t)}, \tilde{C}^{(t)}) \leq (1/\beta) \inf(P^{(t)}, C^{(t)}), \quad (A4)$$

其中,  $C^{(t)}$  是时刻  $t$ , MIN 对  $P^{(t)}$  的最优响应. 此外, 任意时刻  $t$ , 有:

$$\inf(P^{(t)}, C^{(t)}) \leq \min_Q \max_P \inf(P, Q) = v^*, \quad (A5)$$

因此, 将式(A5)两边同时乘以  $1/\beta$ , 并综合式(A4)的结果, 可得:

$$\inf(P^{(t)}, \tilde{C}^{(t)}) \leq (1/\beta) \inf(P^{(t)}, C^{(t)}) \leq (1/\beta) v^*. \quad (A6)$$

结合不等式(A3)的结果, 有:

$$\begin{aligned} (1/\beta) v^* &\geq \frac{1}{T} \sum_{t=1}^T \inf(P^{(t)}, \tilde{C}^{(t)}) \geq \\ \frac{1}{T} \max_{P \in \mathcal{P}} \sum_{t=1}^T \inf(P, \tilde{C}^{(t)}) - \mu - \frac{\ln m}{\mu T}. \end{aligned} \quad (A7)$$

令  $\mu = \epsilon/2$ ,  $T = 4 \ln m / \epsilon^2$ , 于是整理后, 有:

$$(1/\beta) v^* + \epsilon \geq \frac{1}{T} \max_{P \in \mathcal{P}} \sum_{t=1}^T \inf(P, \tilde{C}^{(t)}), \quad (A8)$$

记  $\hat{Q}(C) = |t: \tilde{C}^{(t)} = C| / T$ , 那么, 式(A8)重写为

$$\begin{aligned} \max_{P \in \mathcal{P}} \inf(P, \hat{Q}) &\leq (1/\beta) v^* + \epsilon \Leftrightarrow \\ \max_{P \in \mathcal{P}} \inf(P, \hat{Q}) &\leq (1/\beta) \min_{Q \in \mathcal{Q}} \max_{P \in \mathcal{P}} \inf(P, Q) + \epsilon. \end{aligned}$$

证毕.