Data Mining and Knowledge Discovery manuscript No.
(will be inserted by the editor)

Reducing polarization and increasing diverse navigability in graphs by inserting edges and swapping edge weights

Shahrzad Haddadan · Cristina Menghini · Matteo Riondato · Eli Upfal

Received: date / Accepted: date

Abstract The sets of hyperlinks in web pages, relationship ties in social networks, or sets of recommendations in recommender systems, have a major impact on the diversity of content accessed by the user in a browsing session. Bias induced by the graph structure may trap a reader in a polarized bubble with no access to other opinions. It is widely accepted that exposure to diverse opinions creates more informed citizens and consumers. We introduce the concept of the *polarized bubble radius* of a node, as the expected length of a random walk from it to a node of different opinion. Using the bubble radius, we define the measures of *structural bias* and *diverse navigability* to quantify the effect of links and recommendations on the diversity of content visited in a browsing session. We then propose algorithmic techniques to reduce the structural bias of the graph or improve the diverse navigability of the system through minimal modifications, such as edge insertions or flipping the order of existing links or recommendations, corresponding to switching the edge traversal probabilities. Under mild conditions, our techniques obtain a constant factor-approximation of their respective tasks. In our extensive ex-

A preliminary version of part of this work appeared in the proceedings of ACM WSDM'21 (Haddadan et al. 2021).

S. Haddadan
Dept. of Computer Science & Data Science Initiative, Brown University
E-mail: shahrzad_haddadan@brown.edu

C. Menghini
Dept. of Computer Science, Brown University
E-mail: cristina_menghini@brown.edu

M. Riondato
Dept. of Computer Science, Amherst College
E-mail: mriondato@amherst.edu

E. Upfal
Dept. of Computer Science, Brown University
E-mail: eli@cs.brown.edu

perimental evaluation, we show that our algorithms reduce the structural bias or improve the diverse navigability faster than appropriate baselines, including some designed with the goal of reducing the polarization of a graph.

Keywords Bias · Fairness · Polarization · Random Walks

1 Introduction

The World Wide Web contains thousands or even millions of pages on every topic, covering the whole spectrum of opinions. The fact that diverse information is easily *available* does not imply that *exploring* such diverse information is easy. Exposure to *diverse content* is necessary to obtain a complete picture about a topic. Such exposure depends on the hyperlinks connecting the pages to each other. It can be argued that enabling easier access to diverse content improves society as it creates a more informed and less polarized general public (Benhabib 1996). Indeed politicians have strongly promoted and even requested that audiences are exposed to varied content (LeFebvre 2017).

The progressive polarization of content presented to users of online platforms (social networks, microblogging websites, discussion boards) is a worrisome societal phenomenon with ample evidence (Ribeiro et al. 2019; O'Callaghan et al. 2015). In addition to polarizing content that influences online and offline discourse and "agitates the masses", the sequence of browsing options (e.g., recommendations) dictated by the *structure* of the graphs underlining these platforms, i.e., the hyperlink graph, is in part responsible for trapping the user in echo chambers and polarized bubbles (Pariser 2011; Adamic and Glance 2005; Conover et al. 2011; Flaxman et al. 2016), exposing them only to agreeable information (Bakshy et al. 2015) when not in a downward spiral of more and more extreme opinions, or leading to conflicts between users in different bubbles (Kumar et al. 2018; Cossard et al. 2020).. The gravity of this phenomenon is exacerbated by the fact that the user may not even realize that they entered such a bubble (Ribeiro et al. 2020; Menghini et al. 2020). There is also evidence that recommender systems may worsen these tendencies because they act on these graphs, by suggesting new edges for the social graph, or directly injecting into webpages hyperlinks to more and more extreme content (Baeza-Yates 2020; Ge et al. 2020; Aridor et al. 2020; Nguyen et al. 2014; Castells et al. 2015). Recommender systems may also reduce serendipity (Ge et al. 2010; Kotkov et al. 2016; Anagnostopoulos et al. 2020), i.e., the possibility of "stomping" on content/users expressing different opinions.

A web user can freely click on any hyperlink on the page they are currently visiting, but the choice of which hyperlinks to include in the page is with the website owner or editor, who, if not careful, may stop the user from being exposed to diverse opinions. Similar issues arise in the context of recommender systems: the user can only choose among the recommended items. In other words, the hyperlink topology of a website may suffer from *structural bias* that traps the user in a bubble of one-sided content without them knowing (Ribeiro et al. 2020; Menghini et al. 2020). For example, structural bias on topic-induced networks, such as Wikipedia topic-induced subgraphs, prevents users from building a well-rounded knowledge about the topic. On query-/user-induced recommendation networks such as those on Amazon and YouTube, structural bias hinders the discovery of diversified content, reducing serendipity (Ge et al. 2010; Kotkov et al. 2016; Anagnostopoulos et al. 2020). Structural bias thus limits the user's freedom while navigating the Web.

We model the network as a directed weighted graph, where each node represent a content item (webpage, news feed, blog entry, etc.), an edge represents a hyperlink or a recommendation for the next item to browse. The nodes of the graph are colored. Two nodes have the same color if they present the same opinion on a polarizing topic, the same product category, etc. The behavior of a web surfer is modeled as a random walk on the graph, with the weights on the directed edges representing their transition probabilities. We measure the bubble radius of a node in terms of the expected number of steps till a random walk reaches a node of a different color.

It has been suggested that, in order to increase the exposure of users to diverse content, every vertex (i.e., webpage, item) should link to a vertex of different color. For example, some methods for recommender systems add a constraint to ensure that a diverse set of recommendations is presented to the user, or at least that the probability of this event is maximized (Kunaver and Porl 2017). We argue that it is not sufficient to consider only the possibility that the user accesses diverse content "at the next step" of their surfing, and such "positive bias" may not even be sufficient (Blex and Yasseri 2022). Rather, we suggest that it is important to evaluate the content accessed during the *entire browsing session*, i.e., measure the expected number of steps needed for a user navigating this graph starting from a vertex of one class to visit a vertex of a different class.

We introduce novel measures of the polarization in a graph, namely the *structural bias and the diverse navigability*, that capture the above "long-horizon" point of view. The former is more tailored to the Web graphs, while the latter is designed for graphs arising from content recommendation systems. We then propose algorithmic techniques to improve these measures. Our techniques are based on either inserting new edges in the graph, which is relatively easy to do in web pages for the owner/administrator of the network (e.g., for Wikipedia editors, or for the company owning the content website and controlling the recommendation system), or swapping the transition probabilities of edges with the same source. The assignment of a weight to an edge depends on presentation and appearance factors (e.g., location, font size, order in a list) of the hyperlink corresponding to the edge in the page related to the item v, w.r.t. the other hyperlinks. For example, observational studies on hyperlink graphs (Lerman and Hogg 2014; Craswell et al. 2008; Lamprecht et al. 2016; Dimitrov et al. 2017; Richardson et al. 2007) have shown that a link closer to the top of the page has higher probability (i.e., weight) of being clicked than a link lower in the page, but the set of probabilities depends only on the set of hyperlinks (i.e., on the set of targets) and on the page itself (i.e., on v).

Contributions

- We consider directed graphs with vertices of two colors, representing a network of webpages on the same topic, with the two colors identifying the two opposite opinions on the topic, and edges representing links between pages (we also show how to generalize our approaches to more than two colors). We define the *(Polarized) Bubble Radius* (BR) of a vertex p as a novel measure to quantify the structural bias of p (see Def. 1), based on a task-specific variant of the hitting time for random walks, which models the navigation of a user on the web (Fagin et al. 2001; Dumitriu et al. 2003). The BR is the expected number of steps to go from p to a page of different opinion, and can be easily estimated with a sampling-based approach with probabilistic guarantees (Lemma 4).
- We define the *structural bias* of a graph G as the sum of the BRs of vertices with high BR (Eq. 1). Completely removing the bias is APX-hard by reduction from set cover (see Lemma 5). We therefore state the *k-edge structural bias decrease* problem as the task of finding the set of k pairs of vertices of different color such that adding the edge between the vertices in each pair would *maximally* decrease the structural bias, over all possible sets of k pairs (see Prob. 2 and Thm. 1). This problem connects two areas: link recommendation and polarization reduction.
- We also define the measure of *diverse navigability* (Def. 3), to quantify the diversity of bounded *navigation sessions*. Our measures capture the importance of evaluating diversity in an entire navigation session, in contrast with almost all prior work, which focused on diversity within one click. We formulate the problem of improving the diverse navigability of a graph by swapping the transition probabilities of some edges outgoing from the same source, which is a realistic inexpensive operation corresponding to, e.g., swapping the corresponding hyperlinks on a webpage or changing the ordering of a list of recommendations.
- We present RePBubLik, an efficient approximation algorithm for the k-edge structural bias decrease problem, that recommends the addition of k edges between vertices of different color. Under mild conditions, the resulting decrease of the structural bias is within a constant factor of the optimal (Thm. 2. Website editors have limited control on the probability that a newly added edge will be traversed by the users, so our algorithm makes no assumption or impose any restriction on it, as this probability is essentially external. At the core of RePBubLik is an analysis of the submodularity of the objective function (see Lemma 11), combined with the use of a task-specific variant of random-walk closeness (White and Smyth 2003), a well-established centrality measure. RePBubLik requires good estimations of the random walk closeness, so we also give an approximation algorithm for this quantity (see Lemma 1).
- We present ShuffLik, an approximation algorithm which aims at maximizing the diverse navigability using only k link swaps. We present a rigorous theoretical analysis of the performance of ShuffLik and show it

obtains a constant approximation under mild conditions (see Thm. 2). We then study the more general settings in which the cost of swapping two edge probabilities is not the same for all edge pairs, and there is a limit (budget) to the total costs of the swaps. We show a variant ShuffLik+ that works in this setting.
- We evaluate RePBubLik on eight real datasets. We compare it to baselines and existing methods for edge recommendation either designed with the goal of reducing the controversy of a graphs (Garimella et al. 2017a) or with the more general purpose of completing the network's link structure (Grover and Leskovec 2016). Our algorithm leads to a faster reduction of the average BR (i.e., requiring fewer edge insertions) than existing contributions.
- We evaluate ShuffLik on recommendation networks built using the *25M MovieLens dataset* (Harper and Konstan 2015). Results show that ShuffLik produces a faster increase of diverse navigability compared to a reasonable baseline. ShuffLik is particularly effective when combined with diversity constraints on standard recommendation systems.

2 Related work

Polarization in social networks and the Web Polarization has long been studied in political science (Sunstein 2002; Isenberg 1986), and the diffusion of (micro-) blogs and social media platforms brought the issue to the attention of the broad computer science community. Many works focused on showing the existence of polarization on these platforms (Morales et al. 2015; Adamic and Glance 2005; Cossard et al. 2020; Conover et al. 2011; Flaxman et al. 2016), and on modeling, quantifying, and reducing polarization (Garimella et al. 2018b, 2017a; Musco et al. 2018; Chitra and Musco 2020; Matakos et al. 2017; Becker et al. 2020; Garimella et al. 2017b; Matakos et al. 2020; Aslay et al. 2018; Akoglu 2014; Garimella et al. 2018a; Matakos et al. 2017; Nelimarkka et al. 2018; Liao and Fu 2014a,b; Munson et al. 2013), or the glass ceiling effect (Stoica and Chaintreau 2019; Stoica et al. 2018, 2020). The literature is rich, to the point that the times seem ripe for an in-depth survey on the topic. We discuss here the relationship between our work and the most relevant algorithmic contributions to polarization reduction (Garimella et al. 2017a; Chitra and Musco 2020; Musco et al. 2018; Aslay et al. 2018; Becker et al. 2020; Garimella et al. 2017b; Matakos et al. 2020; Menghini et al. 2019; Menghini et al. 2020; Stoica et al. 2018).

A first important difference of our work with respect to most previous contributions is that they consider a network of *users*, with edges representing notions such as friendship or endorsement (e.g., retweets) (Garimella et al. 2017a; Chitra and Musco 2020; Musco et al. 2018; Aslay et al. 2018; Becker et al. 2020; Garimella et al. 2017b; Matakos et al. 2020; Stoica et al. 2018). We focus instead on networks of *content*, such as web pages linked to each other, or products that are connected when similar. This deep difference makes our contribution quite orthogonal to previous ones: we focus on the polarization that is introduced by the topology of the network, rather than on the polarizing effect of content on users or on the effect of users on each other. Both aspects are important and they lead to different kinds of bubbles: the *filter* bubble, due to the network structure, and the *epistemic* bubble, due to the user behavior. The structural bias we focus on has only been the subject of only few studies (Menghini et al. 2019; Menghini et al. 2020). These works, relying on the notion of weighted reciprocity, propose a static and dynamic analysis of structural bias on Wikipedia. The measures of structural bias and diverse navigability that we introduce are not tailored to a specific website.

A second relevant difference from many previous works is that we consider the "opinion" of a page (i.e., a vertex) to be fixed, as it depends on its content, while past contributions consider different models of user opinion dynamics (Mossel and Tamuz 2017; Das et al. 2014) to study the evolution of such opinions as the users are exposed to different content or recommended different friendships. The problem of recommending changes to the content of a page to modify the opinion expressed in it is interesting but outside the scope of our work. Instead, we focus on recommending the addition of links between pages, to reduce the structural bias, or to swapping transition probabilities between edges with the same source, to increases the diverse navigability.

An interesting line of work studies how to reduce polarization in the content seen by the users, by adapting information diffusion approaches through better selection of the seed set for cascades (Aslay et al. 2018; Becker et al. 2020; Garimella et al. 2017b; Matakos et al. 2020; Stoica et al. 2020), or by directly acting on recommendation systems (Rastegarpanah et al. 2019). These methods can not be adapted to the problem we study, as they do not act on the graph of content, but on that of users.

The most similar methods to ours are those that act on the structure of the graph (Chitra and Musco 2020; Musco et al. 2018; Garimella et al. 2017a; Stoica et al. 2018), although as we mentioned, they consider a network of users, not of content. Musco et al. (2018) propose a network-design approach: they aim to find the best set of edges between vertices such that the resulting graph would minimize both disagreement and polarization. Rather than a "design-from-scratch" approach, which seems mostly of theoretical relevance, we consider instead a practical incremental approach that suggests modifications to an existing network. Like us, Garimella et al. (2017a) consider a graph polarization measure based on random walks (Garimella et al. 2018b). Their measure essentially quantifies the probability that a user of one opinion is exposed to content from a user of a different opinion, thanks to a chain of retweets (represented by the random walks). The measure is based on a variant of personalized PageRank for sets of users with different opinions. The task requires to recommend new edges, i.e., retweets, to increase this probability. Our measures are instead defined on the basis of the (Polarized) Bubble Radius (BR) (Def. 1), which is a vertex-dependent measure that represents the expected number of steps, for a user starting at the page represented by vertex v, to reach, with a random walk, a vertex with color different from v, representing a page expressing a different opinion. Our measures are appropriate for making it easier for users to reach pages of different opinions. In Sect. 6 we compare our approach to that of Garimella et al. (2017a).

An important line of work in graph analysis and mining looked at manipulating the topology to modify different interesting characteristic quantities of the graph, such as shortest paths and related measures (Parotsidis et al. 2015; Papagelis et al. 2011; Demaine and Zadimoghaddam 2010; Perumal et al. 2013), various forms of centrality (Parotsidis et al. 2016; Bergamini et al. 2018; D'Angelo et al. 2019; Wąs et al. 2020; Medya et al. 2018; Mahmoody et al. 2016; Angriman et al. 2020), and more (Arrigo and Benzi 2016a,b; Chan et al. 2014; Tong et al. 2012; Zeng et al. 2012). Despite the fact that we consider a specific centrality to choose the source of the added edges, these methods cannot be used to solve our task of interest.

Another body of work related to ours are those which estimate graph properties using random walks (Bera and Seshadhri 2020; Chierichetti and Haddadan 2018; Ben-Hamou et al. 2018; Dasgupta et al. 2014). The studied properties are not defined based on random walks, rather random walks are used as a tool to estimate them. Here based on random walks, we define a new property for networks: *the structural bias*, and we use random walks to estimate it.

Polarization and bias in recommender systems Many works have identified different types of algorithmic bias, lack of diversity, and lack of fairness in recommender systems (Baeza-Yates 2020; Fu et al. 2020; Resnick et al. 2013; Aridor et al. 2020; Ge et al. 2020; Geyik et al. 2019; Singh and Joachims 2018; Zhu et al. 2018; Yao and Huang 2017; Jiang et al. 2019; Sîrbu et al. 2019; Blex and Yasseri 2022). Their effects are usually framed in terms of having a negative impact on underrepresented/underserved groups, whose members progressively get less recommended over time, while "the popular ones get more popular". But these same effects can also be seen as preventing a web surfer, who is genuinely interested in learning in depth or widely exploring a topic (broadly defined to include both knowledge topics and music/movie genre), from ever reaching less popular points of view, or from being exposed to more niche details of the topic. Essentially, they can be seen as trapping the user in a "filter bubble" (Pariser 2011; Nguyen et al. 2014), i.e., in a limited set of items (pages/products/songs/movies/...) with little or no possibility to reach even slightly different items. It is necessary to enable the user to escape such a bubble, as the desire of the user to explore widely and serendipitously can be seen as empathy, as a desire to learn and experience content from the point of view of another individual (Taramigkou et al. 2013).

The existence of these issues suggested that exposing users to diverse content should be made "a design principle for recommender systems" (Helberger et al. 2018), possibly even a legally-required property. Many algorithmic solutions have been proposed to address these issues (Kunaver and Porl 2017). These solutions usually involve changing the optimization problem solved by the recommender systems, with the goal of taking into consideration fairness and diversity (both defined in a variety of ways) when producing the recommendations and their ranking. Among such changes, there is the possibility of introducing diversity/fairness constraints (Celis et al. 2017; Geyik et al. 2019; Singh and Joachims 2018; Zehlike et al. 2017; Tabibian et al. 2020; Biega et al. 2018; Singh and Joachims 2019; Celis et al. 2017), optimizing diversity/fairness with a lower bound on the relevance or other utility of the recommended items (Zhu et al. 2018), or jointly optimizing utility and diversity/fairness (Celis et al. 2019).

All these works only consider a "single step" in the web surfer exploration of the graph of content/recommendations: their aim is to ensure that, for every item, the items that can be reached from it, i.e., the items that would be recommended, have sufficient utility and are sufficiently diverse. But exploration is rarely, if ever, limited to a single step. Rather, users follow sequences of recommendations and/or web links, taking multiple "hops" on the graph.

Our measures take this aspect into account and are defined on the basis of how many steps a user walking on the graph following the recommendations would need, in expectation, to eventually reach an item that is different (i.e., has different color) than all the ones they visited so far. Our algorithm ShuffLik (see Sect. 5) is designed to optimize this measure by swapping the weights, i.e., the traverse probabilities, of edges outgoing from the same node. As discussed in Sect. 1 this action can be seen as a re-ranking of the items recommended from the visited one, without affecting the set of recommended items. Thus, we take a more "far-sighted" and "inclusive" approach that takes multiple steps into consideration. Additionally, our method does not change the topology of the graph, so it can also be applied as a post-processing steps after the above methods have been used to generate the graph itself.

This version of the work differs from the conference version (Haddadan et al. 2021) in multiple ways:

- We introduce a new approach that, instead of adding new edges to the graph, reduces the bias in the network by swapping the edge weights, corresponding to the transition probabilities of the random walk (see Sect. 5). This approach is less invasive and can be applied in different situation than the previous approach.
- We evaluated the above approach, showing how it performs much better than existing baselines.
- We strengthen and streamline our theoretical results, present all their proofs, and add examples and intuitions that allow the reader to better understand our approach.
- We present new experimental results on additional datasets.

3 Preliminaries

Let $G = (V, E)$ be a directed weighted graph with $|V| = n$ vertices, such that no vertex $v \in V$ has only incoming edges and no outgoing edges. V is partitioned in two disjoint sets R and B (i.e., $R \cap B = \emptyset$ and $R \cup B = V$), called "red" or "blue" vertices, respectively. These colors either correspond to opposing viewpoints of a polarizing topic (e.g., on YouTube) or, when considering several categories of items (e.g., movie genres), red corresponds to one category and blue to all other categories. At the end of Sect. 3.2 we discuss how to extend our definitions and approaches to more than two colors. For any $v \in V$, the set of all *other* vertices of the same color as v is denoted as C_v and the sets of all vertices of colors different than v is denoted as $\bar{C}_v$.

The edge weights M are *transition probabilities*: M is a $n \times n$ right-stochastic *transition matrix* such that each entry $M(i,j)$ is a probability, with $M(i,j) = 0$ if $(i,j) \notin E$, and such that $\sum_{j=1}^{n} M(i,j) = 1$. We sometimes abuse notation and use $M(e)$ to denote the transition probability of the edge e.

We are interested in random walks (r.w.'s) on the graph G using the transition matrix M. Intuitively, a random walk starting at a vertex v explores the graph by choosing at each step an outgoing edge from the current vertex, with probability equal to the weight of such edge, independently from previous choices. Let $S \subseteq V$ and $v \in V$. Let $T_v(S)$ be the random variable indicating the first instant when a random walk from v hits (i.e., reaches) any vertex in S. The quantity $\mathbb{E}_G[T_v(S)]$ is known as the *hitting time of S from v*, where the expectation is over the space of all random walks on G starting from v, with transition probabilities given by M.

Random walks and their variants have been used widely for network analysis (Fouss et al. 2007; Hua et al. 2020; Jung et al. 2019), and in particular to model network exploration behavior (Fagin et al. 2001; Dumitriu et al. 2003). It is realistic to assume that there is an upper bound t, which we call the *exploration factor*, on the length of a walk performed by the users. For example, we can assume that there is an upper limit on the number of pages that a user will visit one after the other in a browsing session. The value of the parameter t can be derived, for example, from traces of visits. In most practical cases, t is likely to be bounded by a polylogarithmic quantity in the number of nodes, if not a constant. For a random walk starting from $v \in V$, given a set $S \subseteq V$, we define the random variable $T_v^t(S)$ as $\min\{t, T_v(S)\}$. This variable is more appropriate for measuring the length of browsing sessions, which have bounded length, than the unbounded length classically used when discussing random walks.

For a graph Z, any vertex u, and any set S of vertices of Z, let $u \overset{\mathsf{cond}}{\underset{Z}{\rightsquigarrow}} S$, denote the event that a random walk in Z from u hits a vertex in S without first visiting any vertex in $\bar{C}_u$ and while satisfying the condition cond on the number of steps needed to hit S. For example, $u \overset{<t}{\underset{Z}{\rightsquigarrow}} S$ is the event that a random walk in Z from u hits a vertex in S in *less than t steps*, without first visiting any vertex in $\bar{C}_u$. We denote the complementary event as $u \overset{\mathsf{cond}}{\underset{Z}{\not\rightsquigarrow}} S$.

3.1 Random-Walk Closeness Centrality

Our algorithms choose the sources of newly added edges and the sources of the edges whose weights to swap on the basis of a task-tailored measure of centrality based on random walks. Specifically, we adapt the definition of *random-walk closeness centrality* (White and Smyth 2003) to bounded random walks so that the contribution to the centrality of v by vertices that do not reach v in less than t' steps (in expectation) is zero, for any t'.

Random-walk closeness centrality (bounded form). For a vertex $v \in V$, a subset $S \subseteq V$, and any t', the t'-*bounded* Random Walk Closeness Centrality (RWCC) of v with respect to S is

$$\mathsf{r}^{t'}(v;S) \doteq \frac{1}{|S|}\sum_{w\in S}\left(t' - \mathbb{E}_G\left[T_w^{t'}(v)\right]\right) = \frac{1}{|S|}\sum_{w\in S}\sum_{i=1}^{t'-1}(t'-i)\mathbb{P}\left(w \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \enspace .$$

Computing the exact RWCC is expensive. To estimate $\mathsf{r}^{t'}(v;S)$, we pick z vertices $\{w_i\}_{i=1}^{z}$ u.a.r. from S, and run some κ random walks to obtain an estimate $\bar{h}_{w_i}$ of $\mathbb{E}_G\left[T_{w_i}^{t'}(v)\right]$ for each w_i. The quantity

$$\bar{r}(v) \doteq t' - \frac{1}{z}\sum_{i=1}^{z}\bar{h}_{w_i}$$

is a good approximation of $\mathsf{r}^{t'}(v;S)$.

Lemma 1 *Let* $z \geq ({t'}/{2\varepsilon})^2\delta^{-1}$. *Then*

$$\mathbb{P}\left(|\bar{r}(v) - \mathsf{r}^{t'}(v;S)| \geq \varepsilon\right) \leq \delta \enspace .$$

Proof We can write

$$\mathsf{r}^{t'}(v;S) = t' - \frac{1}{|S|}\sum_{w\in S}\mathbb{E}_G\left[T_w^{t'}(v)\right] \enspace .$$

We apply Chebyshev's inequality to the r.v. ${1}/{z}\sum_{i=1}^{z}\bar{h}_{w_i}$, to bound the deviation from its expectation

$$\frac{1}{|S|}\sum_{w\in S}\mathbb{E}_G\left[T_w^{t'}(v)\right] \enspace .$$

To get an upper bound to the variance of this r.v., we use the fact that the r.v.'s $\bar{h}_{w_i}$, $i = 1, \ldots, z$, are independent, and, from Popoviciu's inequality, the fact that each has a variance at most ${t'^2}/{4}$, as $\bar{h}_{w_i} \in [0, t']$.

3.2 The Bubble Radius

We introduce the *(Polarized) Bubble Radius* to quantify how many steps it would take, in expectation, for users starting their random walk on a vertex $v \in V$ of one color, to hit a vertex of the other color.

Definition 1 The *(Polarized) Bubble Radius (BR)* $\mathsf{B}_G^t(v)$ *of* v *with exploration parameter* t is

$$\mathsf{B}_G^t(v) \doteq \mathbb{E}_G\left[T_v^t\left(\bar{C}_v\right)\right] \quad .$$

A random walk starting at a vertex v with high BR is unlikely to hit a vertex in $\bar{C}_v$ in fewer-than-or-exactly t steps. The following lemma formalizes this idea on common models for web browsing (random walks with restarts or with back button (Fagin et al. 2001; Dumitriu et al. 2003)).

Lemma 2 *Let* $r \in \mathbb{N}$*, and consider a user who starts their random walk at* $v \in V$ *and may either restart their walk from* v *or hit the back button up to* z *times. Let* $\mathcal{T}_v$ *be the random variable denoting the number of steps such user takes to hit a vertex in* $\bar{C}_v$*. If* $\mathsf{B}_G^t(v) \geq t(1 - {}^1\!/_{8z})$*, then* $\mathbb{P}(\mathcal{T}_v \leq t/2) \leq {}^1\!/_4$*. If instead* $\mathsf{B}_G^t(v) \leq b$ *for some* $b > 0$*, then* $\mathbb{P}(\mathcal{T}_v > 4bz) \leq {}^1\!/_4$*.*

In the proof, we use the following technical result (proof in App. A).

Lemma 3 *Let* X *be a random variable satisfying* $0 \leq X \leq t$*. We have:*

$$\mathbb{P}(X \leq k) \leq \frac{t - \mathbb{E}[X]}{t-k} \quad .$$

Proof (Lemma 2) Assume first that $\mathsf{B}_G^t(v) \geq t(1 - 1/(8z))$. Consider a set of z independent random walkers, $w_1, \ldots, w_r$, each starting from v. We can see the trace of the partial walks taken by our random walker with restarts as the union of the traces of these walkers. The event $\mathcal{E}' \doteq$ "$\mathcal{T}_v \leq {}^t\!/_2$" is a strict subset of the event $\mathcal{E}'' \doteq$ "there is (at least) a walker w_i for which $T_v^t \leq {}^t\!/_2$", as the condition in $\mathcal{E}'$ implies the condition in $\mathcal{E}''$, but not vice versa. Thus, $\mathbb{P}(\mathcal{E}') < \mathbb{P}(\mathcal{E}'')$. By Lemma 3 we have, for each walker, that

$$\mathbb{P}\left(T_v^t \leq \frac{t}{2}\right) \leq \frac{t - \mathbb{E}\left[T_v^t(S)\right]}{t - \frac{t}{2}} \leq \frac{\frac{t}{8z}}{\frac{t}{2}} \leq \frac{1}{4z}.$$

Thus, using the union bound over the z walkers, we get $\mathbb{P}(\mathcal{E}'') \leq {}^1\!/_4$. Equivalently $\mathbb{P}(\mathcal{T}_v \leq {}^t\!/_2) \leq {}^1\!/_4$.

When $\mathsf{B}_G^t(v) \leq b$, it holds, from Markov's inequality, that $\mathbb{P}(\mathcal{T}_v > 4bz) \leq {}^1\!/_4$. □

Given t, it is easy to estimate $\mathsf{B}_G^t(v)$ for each vertex $v \in V$ by sampling random walks from v. The following result, whose proof uses the Hoeffding's bound and the union bound, shows the trade-off between the number of sampled random walks and the accuracy in estimating the BR of v.

Lemma 4 *For each $v \in V$, let $w_1^{(v)}, w_2^{(v)}, \ldots, w_r^{(v)}$ be r random walks from v and stopped either when they hit a vertex of color $\bar{C}_v$ or when they run for t steps, whichever happens first. For $i = 1, \ldots, r$, let $b_i^{(v)}$ be the length of random walk $w_i^{(v)}$. Let*

$$\bar{B}(v) \doteq \frac{1}{r} \sum_{i=1}^{r} b_i^{(v)} \quad .$$

Let $\varepsilon, \delta \in (0,1)$. If $r \geq \frac{t^2}{\varepsilon^2} \ln \frac{2n}{\delta}$, then

$$\mathbb{P}\left(\exists v \in V \ \ s.t. \ |\mathsf{B}_G^t(v) - \bar{B}(v)| > \varepsilon\right) < \delta,$$

where the probability is over the choice of the random walks.

In the rest of the work, we assume for simplicity to have access to the *exact* BR of every vertex. Lemma 4 makes this assumption reasonable because computing approximations of extremely high quality is relatively inexpensive.

Extension to more than two colors We assume that there are only two colors, R and B. Our approach can be extended to the case of $k > 2$ groups representing different opinions. One possible extension involves considering, in turn, the vertices in one group to have color R and all other vertices to have color B, independently from their group. Another possibility is to assign a different color C_i, $1 \leq i \leq k$, to each group, and then, for each $1 \leq i \leq k$, define, for each $v \in C_i$, the *BR of v w.r.t. color C_j*, $j \neq i$, using the hitting time from v to vertices of color C_j. We can then redefine the BR of v as the minimum, over all C_j, $j \neq i$, of the BR of v w.r.t. color j.

When one of the groups is supposed to represent a neutral opinion, one approach could be to assign weights to the vertices, representing the extremeness of their position, and then define a weighted version of the bubble radius. Such an extension goes beyond the scope of our work, but it is an interesting direction for future work.

Another interesting direction, in the presence of multiple groups, would be to study "multi-chromatic exploration", by considering hitting times defined as the number of steps needed for a r.w. to traverse vertices of $h > 2$ colors.

4 Reducing the structural bias with edge insertions

On the basis of the BR, we define two sets of vertices: *cosmopolitan* and *parochial*. Given two reals b and r with $1 \leq b < r \leq t$, the set $\mathcal{Z}(G)$ of *cosmopolitan* vertices contains all and only the vertices in G with BR *at most* b, and the set $\mathcal{P}(G)$ of *parochial* vertices contains all and only the vertices in G with BR at least r. The intuition is that it is easy to reach the other color from cosmopolitan nodes, while it is hard from parochial nodes. For ease of notation, we do not include b and r in the notation for $\mathcal{Z}(G)$ and $\mathcal{P}(G)$. For technical reasons (see the proof of Lemma 8), we require $r \geq t/2$. $\mathcal{Z}(G)$ and

$\mathcal{P}(G)$ are *disjoint*, but they do not necessarily form a partitioning of V. We will often consider the partitioning of $\mathcal{P}(G)$ by color, i.e., the two sets $\mathcal{P}_R(G)$ and $\mathcal{P}_B(G)$, containing the parochial vertices of color R or B respectively.

Definition 2 The *structural bias* $\varrho(G)$ of G is the sum of the BRs of the parochial nodes of G, i.e.,

$$\varrho(G) \doteq \sum_{v \in \mathcal{P}(G)} \mathsf{B}^t_G(v) \quad . \tag{1}$$

It is reasonable to consider only the parochial nodes in the definition of structural bias because they are the ones such that a random walk from them is very unlikely to hit any vertex of color different than the starting vertex (see also Lemma 2). Additionally, the choice of considering the sum, and not, for example, the average of the BRs is due to the fact that it makes it easy to compare the structural bias of a graph vs. that of the same network with some added edges: such a network may have *higher* average BR of the parochial nodes (because, e.g., there are fewer parochial nodes with higher BR), but the "total" bias is lower (because there are fewer parochial nodes and/or the BR of some parochial node decreased as a consequence of the edge insertions), and that is the quantity we want to measure.

Our goal in this section is to find a set of edges with extrema of different color whose addition to G would decrease the structural bias of the network. It is reasonable to only consider edge with end points of different color, as they are always preferable (i.e., their insertion will result in a larger decrease of the structural bias) than edges with monochromatic extrema: the addition of the new edge can only have positive impact on the parochial vertices of the same color as the edge source, and has no impact on the parochial vertices of the other color. If we could add *any number* of such edges to G, it would be easy to bring the structural bias of G to zero, as there would be no parochial nodes left. This assumption is not realistic: the number of links that a website editor can add to a single page and to the whole graph is limited by many factors, such as the fact that a human-readable page cannot have too many links, and the fact that the editor can only spend a limited time on this activity. Nevertheless, ideally one would want to solve Prob. 1, defined as follows. Given a set Σ of edges not currently in G, we denote with G_Σ the graph $G_\Sigma \doteq (V, E \cup \Sigma)$.

Problem 1 Given $C \in \{R, B\}$, find the smallest set Σ of pairs of distinct vertices $(v, w) \notin E$ with $C_v = C$ and $C_w \neq C$ such that $\mathcal{P}_C(G_\Sigma) = \emptyset$.

Lemma 5 *Problem 1 is NP-hard and APX-hard.*

Proof We show an approximation-preserving polynomial time reduction from the minimum set cover problem to Prob. 1.

Let $U = \{u_1, u_2, \ldots, u_n\}$ be a domain and let $S_1, S_2, \ldots, S_m \subseteq U$ be an instance of the set cover problem. We construct an instance of Prob. 1 as follows. Fix $t \geq 3$. Let V be union of the following sets: U, $S = \{s_i\}_{i=1}^m$

representing the sets, $T = \bigcup_{j=1}^{m} T_j$ where each T_j is a set of $r-1$ distinct vertices, and $\{g\}$. Assume all vertices except g have color red and g is blue, i.e., $R = V \setminus \{g\}$ and $B = \{g\}$. For each $i \in [n]$ and $j \in [m]$, place an edge from u_i to s_j if and only if $u_i \in S_j$. For each $j \in [m]$, using the vertices in T_j, place a path of length $r-1$ going from s_j to g. For each $1 \le j \le m$, it holds $\mathsf{B}_G^t(s_j) = r-1$, and for each $1 \le i \le n$,

$$\mathsf{B}_G^t(u_i) = \frac{1}{|\{j \ : \ u_i \in S_j\}|} \sum_{j \text{ s.t. } u_i \in S_j} \mathsf{B}_G^t(s_j) + 1 = r \ .$$

Clearly the bubble radius of vertices in T is strictly less than r. Thus the parochial vertices are all and only those in U. Assume there is a polynomial-time algorithm for Prob. 1. For any (optimal) solution $\Sigma \subseteq V \times V$, it holds $\mathsf{B}_{G_{\text{new}}}^t(u_i) < r$ if and only if Σ contains an edge whose source is in $\{u_i\} \cup \bigcup_{j \text{ s.t. } u_i \in S_j}(\{s_j\} \cup T_j)$, for each $i \in [n]$. The source vertices of the edges in Σ must be distinct, as any solution containing two edges originating from the same vertex cannot be optimal. Denote with Z the set of the source vertices of the edges in Σ. Consider now the solution Σ' obtained by changing (in polynomial time) Σ as follows: *1.* each edge in Σ whose source is in T_i is modified to have source s_i, for each $i \in [n]$; and *2.* each edge in Σ whose source is $u \in U$ is changed to have source s_j where j is such that $u \in S_j$. Clearly Σ' is still an (optimal) solution to Prob. 1. Let OPT be the set of source vertices of the edges in Σ'. Clearly it must be $\mathsf{OPT} \subseteq S$. We now show that Σ' is an (optimal) solution to Prob. 1 if and only if OPT is such that $\{S_j \ : \ s_j \in \mathsf{OPT}\}$ is a minimum set cover for the considered instance. It is evident that $\{S_j \ : \ s_j \in \mathsf{OPT}\}$ is a set cover, which can be obtained in polynomial time from Σ'. We now show that this set cover is minimal. Consider now any set cover $Y \subseteq \{S_1, \ldots, S_m\}$, and consider the set of edges $\{(s_i, g) \ : \ S_i \in Y\}$. Adding these edges to G would result in all the vertices in U to no longer be parochial. This holds in particular for any *minimal* set cover Y, from which we can create an (optimal) solution Σ_Y to Prob. 1. Thus we found a bijection between (optimal) solutions to Prob. 1 and minimal set covers for the considered instance, and computing one from the other can be done in polynomial time, showing the NP-hardness of Prob. 1. The APX-hardness follows because, for any minimum set cover Y, the corresponding optimal solution Σ_Y to Prob. 1, built as above, is such that $|\Sigma_Y| = |Y|$, thus if we had a constant-factor polynomial-time approximation algorithm for Prob. 1 we would have an algorithm with the same properties for the minimum set cover problem. □

Since Prob. 1 is hard to even approximate, we seek to answer a close relative (see Prob. 2 below). We first introduce a set of measures to capture the change in the BRs of the (original) parochial nodes of G after edge insertions. Assume to add to G all edges in a set Σ of non-existing directed edges between nodes of different colors, with each inserted edge $e = (v, w)$ having weight $M(e)$. For a set U of vertices, we define the *gain* of U due to Σ as

$$\Delta(G, U, \Sigma, \{M(e)\}_{e\in\Sigma}, t') \doteq \frac{1}{|U|}\sum_{u\in U}\left(\mathsf{B}^{t'}_{G}(u) - \mathsf{B}^{t'}_{G_\Sigma}(u)\right), \quad (2)$$

i.e., as the average change of the BR of vertices in U. When $U = \{v\}$, clearly that is just the change in the BR of v.

When adding an edge to the graph, we also have to decide its weight. It seems excessive to assume complete freedom in choosing the weight. We make the assumption that the weight $M(v, w)$ of an edge (v, w) that we would like to add is given to us by an oracle which computes $M(v, w)$ only as a function of v and of information *local* to v (e.g., its out-degree) obtained from G and potentially a set of other edges (and their weights) that we want to add from v. In other words, the oracle returns the same quantity $q_v = M(v, w)$ no matter what w is. When adding (v, w) with weight $M(v, w)$, the other edges outgoing from v have their weights multiplied by $1 - M(v, w)$ to ensure that the sum of the weights of the edges leaving v is 1.

The problem we want to solve then is the following.

Problem 2 Let $C \in \{R, B\}$. Find a set $\Sigma = \{(v_i, w_i)\}_{i=1}^{k}$ of k edges with $v_i \in C$ and $w_i \notin C$, for $1 \le i \le k$, that maximizes $\Delta(G, \mathcal{P}_C(G), \Sigma, \{M(e)\}_{e\in\Sigma}, t)$.

RePBubLik (Alg. 1) is our algorithm to approximate Prob. 2. Before describing it in detail, we give an intuition of its workings, and present the theoretical results that guides its design. Specifically, since the objective function from Prob. 2, i.e., the gain, is *monotonic and submodular* (Lemma 11), we can greedily choose, one by one, the edges to be added. Our oracle assumption on the weights ensures that *any* vertex of color different than the source can be picked as the target of the added edge, so the task essentially *reduces to finding the sources for the edges to be added.* Lemma 8 quantifies the gain when picking each source according to a specific measure depending on the bounded RWC and on the oracle-given weight that only depends on the source. In Lemma 10 we show that under mild conditions this choice is constantly close to an optimal choice. Theorem 1 states the approximation qualities of RePBubLik. Let us first introduce some needed quantities.

For any vertex v, and $0 \le t' \le t$, let

$$\mathcal{F}_{t'}(v) \doteq \sum_{i=0}^{t'-1}\mathbb{P}\left(v \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) = 1 + \mathbb{P}\left(v \overset{<t'}{\underset{G}{\rightsquigarrow}} v\right), \quad (3)$$

where we assumed $\mathbb{P}\left(v \overset{=0}{\underset{G}{\rightsquigarrow}} v\right) = 1$, and define $\gamma_t \doteq \max_{v\in V}\mathcal{F}_{t'}(v)$. This quantity is one plus the maximum probability that a random walk from a vertex return to that vertex in at most t' steps, which is a constant for many graphs

Theorem 1 *Let Σ be the output of* RePBubLik *and* OPT *be the optimal solution to Prob. 2. Let $\Delta_\Sigma = \Delta(G, \mathcal{P}_C(G), \Sigma, \{M(e)\}_{e\in\Sigma}, t)$. Then*

$$\Delta(G, \mathcal{P}_C(G), \mathsf{OPT}, \{M(e)\}_{e\in\mathsf{OPT}}, t) \le \left(2\frac{t}{r}\gamma_{t-2} + 1\right)\left(1 + \frac{1}{e}\right)\Delta_\Sigma \ .$$

When γ_t is bounded by a constant, so is γ_{t-2}, and since we assumed $r \geq t/2$, we get that RePBubLik gives a *constant factor approximation*, under this mild condition.

We now proceed towards presenting lemmas which together provide a proof for Thm. 1. The following lemma shows upper and lower bounds to the change in the BR of a vertex v (i.e., to the gain for v) when a new edge from v is added to the graph.

Lemma 6 *Let $v \in \mathcal{P}(G)$, $w \in \bar{C}_v$ and $t' \leq t$. Assume to add $e = (v, w)$ to G with weight $M(e)$. Then,*

$$\left(\mathsf{B}_G^{t'}(v) - 1\right) M(e) \leq \Delta(G, v, e, M(e), t') \leq \mathcal{F}_{t'-1}(v) \left(\mathsf{B}_G^{t'}(v) - 1\right) M(e) \ .$$

Proof Let G_e be the graph obtained after adding e to G with weight $M(e)$. Consider the probability space of all random walks starting from v in G_e and G. We introduce a coupling between these two probability spaces as follows: consider a walk in G_e and couple every step of it to an identical step in G. If a walk in G_e never traverses (v, w) then the gain function is zero as it gets coupled to the identical walk in G. Assume that the walk in G_e traverses (v, w) at the i^{th} step without first visiting a vertex in $\bar{C}_v$. Before traversing (v, w), the two identical walks in G_e and G have the same probability, and the coupling works. We partition the state space by conditioning on the step i as follows.

Let $\mathcal{E}_i$, $1 \leq i \leq t'$, be the event that the walk in G_e traverses (v, w) at step i. Consider all such walks. These walks need, at step $i - 1$, one more step to reach the other color, and they are coupled to walks in G which in expectation need $\mathsf{B}_G^{t'-i+1}(v)$ steps to reach $\bar{C}_v$ (or terminate). Thus, assuming $\mathcal{E}_i$, the gain in bubble radius is equal to $\mathsf{B}_G^{t'-i+1}(v) - 1$. Using the law of total expectation and summing over all $1 \leq i \leq t'$, we can write

$$\begin{aligned}
\Delta(G, v, (v, w), M(v, w), t') &= \sum_{i=1}^{t'} \left(\mathsf{B}_G^{t'-i+1}(v) - 1\right) \mathbb{P}(\mathcal{E}_i) \\
&= \sum_{i=1}^{t'-1} \left(\mathsf{B}_G^{t'-i+1}(v) - 1\right) \mathbb{P}(\mathcal{E}_i),
\end{aligned}$$

where the second equality comes from the fact that $\mathsf{B}_G^1(v) = 1$ always. The left hand side of the thesis then follows from the fact that $\mathbb{P}(\mathcal{E}_1) = M(v, w)$ and that $\mathsf{B}_G^j(v) \geq 1$ for any $1 \leq j \leq t'$. The right-hand side is concluded from the fact that $\mathsf{B}_G^{t'-i+1}(v) \leq \mathsf{B}_G^{t'-1}(v)$ and that

$$\sum_{i=1}^{t'-1} \mathbb{P}(\mathcal{E}_i) = \sum_{i=0}^{t'-2} \mathbb{P}\left(v \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) M(v, w) = \mathcal{F}_{t'-1}(v) \, M(v, w) \ .$$

□

Adding an edge from v does not just decreases the BR of v, but it also decreases the BRs of vertices in C_v, and thus the structural bias of the whole network. Lemma 7 quantifies this change.

Lemma 7 *Let $e = (v, w)$ be an edge with weight $M(e)$ added to G. It holds*

$$\Delta(G, \mathcal{P}_{C_v}(G), e, M(e), t) = \frac{1}{|\mathcal{P}_{C_v}(G)|} \sum_{u \in \mathcal{P}_{C_v}(G)} \sum_{i=1}^{t-2} \Delta(G, v, e, M(e), t-i) \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \ . \tag{4}$$

Proof Using the law of total expectation, for any graph Z, it holds

$$\mathsf{B}_Z^t(u) = \left(\sum_{i=1}^{t-1} \left(i + \mathsf{B}_Z^{t-i}(v)\right) \mathbb{P}\left(u \overset{=i}{\underset{Z}{\rightsquigarrow}} v\right)\right) + \mathbb{E}_Z\left[T_u^t\left(\bar{C}_v\right) \mid u \overset{<t}{\underset{Z}{\not\rightsquigarrow}} v\right] \mathbb{P}\left(u \overset{<t}{\underset{Z}{\not\rightsquigarrow}} v\right). \tag{5}$$

Let G_e be the graph obtained after adding e. Between G and G_e, we are only adding an outgoing edge from v and modifying the weights of the edges outgoing from v, so

$$\mathbb{E}_G\left[T_u^t\left(\bar{C}_v\right) \mid u \overset{<t}{\underset{G}{\not\rightsquigarrow}} v\right] = \mathbb{E}_{G_e}\left[T_u^t\left(\bar{C}_v\right) \mid u \overset{<t}{\underset{G_e}{\not\rightsquigarrow}} v\right],$$

$$\mathbb{P}\left(u \overset{<t}{\underset{G}{\not\rightsquigarrow}} v\right) = \mathbb{P}\left(u \overset{<t}{\underset{G_e}{\not\rightsquigarrow}} v\right), \text{ and } \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) = \mathbb{P}\left(u \overset{=i}{\underset{G_e}{\rightsquigarrow}} v\right) \ .$$

Therefore, using (5),

$$\begin{aligned}\Delta(G, u, (v, w), M(v), t) &\doteq \mathsf{B}_G^t(u) - \mathsf{B}_{G_{\text{new}}}^t(u) \\ &= \sum_{i=1}^{t-1} \left(\Delta(G, v, (v, w), M(v, w), t-i)\right) \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \\ &= \sum_{i=1}^{t-2} \left(\Delta(G, v, (v, w), M(v, w), t-i)\right) \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \ .\end{aligned}$$

The last step follows from the fact that $\Delta(G, v, (v, w), M(v, w), 1) = 0$ because $\mathsf{B}_Z^1(u) = 1$ for every vertex u of any graph Z. The thesis then follows from the definition of gain for a set of vertices, rather than for a single node (see (2)). □

Recall that our greedy choice is to identify a node v that maximizes the gain $\Delta(G, \mathcal{P}(G), (v, w), M(v, w), t)$ where w is any vertex in $\bar{C}_v$, and $M(v, w)$ is give to us by an oracle, only on the basis of information "locally available" from v. Lemma 7 suggests that a good candidate v is a vertex that is likely to be reached by short random walks from many other vertices in $\mathcal{P}_{C_v}(G)$, a property that is captured by the bounded RWCC $\mathsf{r}^{t-2}(v; \mathcal{P}_{C_v}(G))$ (Sect. 3.1).

Now, we first quantify the gain for adding an edge from any vertex. Then we show that under mild conditions on the return time of vertices we get a constant approximation by greedily choosing a vertex v that maximizes $\mathsf{r}^{t-2}(v; \mathcal{P}_{C_v}(G))\, M(v, w)$ (Lemma 10).

Lemma 8 *Let $v \in \mathcal{P}(G)$. Let $w \in \bar{C}_v$, and assume to add the edge $e = (v, w)$ with weight $M(e)$. It holds*

$$\Delta(G, \mathcal{P}_{C_v}(G), e, M(e), t) \geq \frac{r}{t} M(e) \mathsf{r}^{t-2}(v; \mathcal{P}_{C_v}(G)) \quad .$$

We need the following technical result before proving Lemma 8 (proof in App. A).

Lemma 9 *Let $v \in \mathcal{P}_{C_v}(G)$, then, for any $t' \leq t$, it holds $\mathsf{B}_G^{t'}(v) \geq r\frac{t'}{t}$.*

Proof (Lemma 8) It holds from Lemmas 6 and 9 that

$$\Delta(G, v, e, M(e), t') \geq \left(r\frac{t'}{t} - 1\right) M(e) \text{ for every } 1 \leq t' \leq t \quad .$$

Using this fact, and the requirement that $r \geq t/2$, we can take (4), and conclude as follows

$$\begin{aligned}
&\Delta(G, \mathcal{P}_{C_v}(G), e, M(e), t) \\
&\geq \frac{1}{|\mathcal{P}_{C_v}(G)|} \sum_{u \in \mathcal{P}_{C_v}(G)} \sum_{i=1}^{t-2} \left(r\frac{t-i}{t} - 1\right) M(e) \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \\
&\geq \frac{r}{t} M(e) \underbrace{\frac{1}{|\mathcal{P}_{C_v}(G)|} \sum_{u \in \mathcal{P}_{C_v}(G)} \sum_{i=1}^{t-3} (t-i-2) \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right)}_{\mathsf{r}^{t-2}(v; \mathcal{P}_{C_v}(G))} \quad .
\end{aligned}$$

□

Lemma 8 suggests that inserting edges from a vertex v with the highest value of $M(v, w)\mathsf{r}^{t-2}(v; \mathcal{P}_C(G))$ *may* result in a larger improvement in the objective function than if we chose a different source. In the next lemma we compare the effect of choosing such a source to the effect of an optimal choice.

Lemma 10 *Consider the set $\mathcal{P}_C(G)$ where $C \in \{R, B\}$. Among all vertices in $\mathcal{P}_C(G)$ let*

$$\begin{aligned}
\mathrm{opt} &\doteq \arg\max\nolimits_{u \in \mathcal{P}_C(G)} \Delta(G, \mathcal{P}_C(G), e_u, M(e_u), t), \text{ and} \\
v &\doteq \arg\max\nolimits_{u \in \mathcal{P}_C(G)} M(e_u)\mathsf{r}^{t-2}(u; \mathcal{P}_C(G)),
\end{aligned}$$

where e_u is any non-existing edge connecting u to $\bar{C}_u$. It holds

$$\Delta(G, \mathcal{P}_C(G), e_{\mathrm{opt}}, M(e_{\mathrm{opt}}), t) \leq \left(2\frac{t}{r}\gamma_{t-2} + 1\right) \Delta(G, \mathcal{P}_C(G), e_v, M(e_v), t) \quad . \tag{6}$$

Proof It follows from Lemma 6 that, for any t',

$$\begin{aligned}\Delta(G, \mathrm{opt}, e_{\mathrm{opt}}, M(e_{\mathrm{opt}}), t') &\leq \mathcal{F}_{t'-1}(\mathrm{opt}) \left(\mathsf{B}_G^{t'}(\mathrm{opt}) - 1\right) M(e_{\mathrm{opt}}) \\ &\leq \mathcal{F}_{t'-1}(\mathrm{opt})\,(t'-1) M(e_{\mathrm{opt}}) \ .\end{aligned}$$

From Lemma 7, applying the above inequality, we get

$$\begin{aligned}
&\Delta(G, \mathcal{P}_C(G), e_{\mathrm{opt}}, M(e_{\mathrm{opt}}), t) \\
&= \frac{1}{|\mathcal{P}_C(G)|} \sum_{u \in \mathcal{P}_C(G)} \sum_{i=1}^{t-2} \left(\Delta(G, \mathrm{opt}, e_{\mathrm{opt}}, M(e_{\mathrm{opt}}), t-i)\right) \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \\
&\leq \frac{1}{|\mathcal{P}_C(G)|} \sum_{u \in \mathcal{P}_C(G)} \sum_{i=1}^{t-3} \left(\Delta(G, \mathrm{opt}, e_{\mathrm{opt}}, M(e_{\mathrm{opt}}), t-i)\right) \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \\
&\qquad + \Delta(G, \mathrm{opt}, e_{\mathrm{opt}}, M(e_{\mathrm{opt}}), 2) \mathbb{P}\left(u \overset{=t-2}{\underset{G}{\rightsquigarrow}} v\right) \\
&\leq \frac{1}{|\mathcal{P}_C(G)|} \sum_{u \in \mathcal{P}_C(G)} \sum_{i=1}^{t-3} (t-1-i) M(e_{\mathrm{opt}}) \mathcal{F}_{t-2}(\mathrm{opt})\, \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) + 1 \\
&\leq \frac{1}{|\mathcal{P}_C(G)|} \sum_{u \in \mathcal{P}_C(G)} \sum_{i=1}^{t-2} 2(t-2-i) M(e_{\mathrm{opt}}) \mathcal{F}_{t-2}(\mathrm{opt})\, \mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) + 1 \\
&\leq 2M(e_{\mathrm{opt}}) \mathsf{r}^{t-2}(\mathrm{opt}; \mathcal{P}_C(G))\, \mathcal{F}_{t-2}(\mathrm{opt}) + 1 \\
&\leq 2M(e_v) \mathsf{r}^{t-2}(v; \mathcal{P}_C(G))\, \gamma_{t-2} + 1 \\
&\leq \left(2\frac{t}{r}\gamma_{t-2} + 1\right) \Delta(G, \mathcal{P}_C(G), e_v, M(e_v), t),
\end{aligned}$$

where the last step follows from Lemma 8. □

When the probability $\mathbb{P}\left(u \overset{<t}{\underset{G}{\rightsquigarrow}} u\right)$ that a random walk starting at any vertex u gets back to u in less than t steps is less than α, for some constant α, then $\gamma_{t-2} \leq 1 + \alpha$. This assumption is realistic since t is usually small and the return time to u is often much larger than t. The multiplicative factor on the r.h.s. of (6) is then, in such cases, a constant, because $t/r \leq 2$, since we assumed $r \geq t/2$. Thus, the gain obtained by choosing v as in Lemma 10 is a *constant-factor* approximation to the optimal choice.

Finally, we show that the gain function is monotonic and sub-modular.

Lemma 11 *Let $C \in \{R, B\}$, $v, u \in \mathcal{P}_C(G)$, and $w_v, w_u \in \bar{C}_v$, such that $e_v = (v, w_v)$ and $e_u = (u, w_u)$ are not edges in G. Let $\Sigma = \{e_v, e_u\}$. It holds*

$$\Delta(G, \mathcal{P}_C(G), e_v, M(e_v), t) \leq \Delta(G, \mathcal{P}_C(G), \Sigma, \{M(e)\}_{e \in \Sigma}, t), \tag{7}$$

and

$$\begin{aligned}\Delta(G, \mathcal{P}_C(G), \Sigma, \{M(e)\}_{e \in \Sigma}, t) \leq\, &\Delta(G, \mathcal{P}_C(G), e_v, M(e_v), t) \\ &+ \Delta(G, \mathcal{P}_C(G), e_u, M(e_u), t) \ .\end{aligned} \tag{8}$$

Proof Let G_v be the graph after adding only the edge e_v, G_u be the graph after only adding the edge e_u, and G_{vu} be the graph after adding both edges.

We first show the monotonicity of the objective function, i.e., that (7) holds. For any $w \in \mathcal{P}_C(G)$, it holds

$$\begin{aligned}\Delta(G, w, e_v, M(e_v), t) &\doteq \mathsf{B}^t_G(w) - \mathsf{B}^t_{G_v}(w) \le \mathsf{B}^t_G(w) - \mathsf{B}^t_{G_{vu}}(w) \\ &\doteq \Delta(G, w, \{e_v, e_u\}, \{M(e_v), M(e_u)\}, t)\end{aligned}$$

because $\mathsf{B}^t_{G_v}(w) \ge \mathsf{B}^t_{G_{vu}}(w)$, as adding an edge from u, which is in C_w, to a vertex in $\bar{C}_w$ cannot increase the bubble radius of w. The result generalizes to (7) in a straightforward way.

We now show the sub-modularity of the objective function, i.e., that (8) holds. We start by showing that, for $w \in \mathcal{P}_C(G)$, it holds

$$\begin{aligned}\Delta(G, w, \{e_v, e_u\}, \{M(e_v), M(e_u)\}, t) \le &\Delta(G, w, e_v, M(e_v), t) \\ &+ \Delta(G, w, e_u, M(e_u), t) \ .\end{aligned}$$

With an expansion of the definition and a slight rearrangement of the terms, the above inequality is equivalent to

$$\underbrace{\mathsf{B}^t_{G_v}(w) - \mathsf{B}^t_{G_{vu}}(w)}_{\Delta(G_v, w, e_u, M(e_u), t)} \le \underbrace{\mathsf{B}^t_G(w) - \mathsf{B}^t_{G_u}(w)}_{\Delta(G, w, e_u, M(e_u), t)},$$

i.e., the gain of adding the same edge (in this case e_u) is smaller when the edge is added to a graph (in this case G_v) that has a superset of the edges of another graph (in this case G).

Consider all the walks from w that pass through v or through u or both. Among such walks, let $\mathcal{E}_v$ be the event of seeing v first and $\mathcal{E}_u$ be the event of seeing u first. If a walk does not pass through either v or u, its probability of hitting the other color is the same in all three graphs we are considering, as the graphs differ only in the outgoing edges from these two nodes and their weights. For the same reason, $\mathbb{P}(\mathcal{E}_v)$ and $\mathbb{P}(\mathcal{E}_u)$ do not change across the graphs. Thus,

$$\begin{aligned}\mathsf{B}^t_{G_v}(w) - \mathsf{B}^t_{G_{vu}}(w) = &\left(\mathbb{E}_{G_v}\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_v\right] - \mathbb{E}_{G_{vu}}\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_v\right]\right)\mathbb{P}(\mathcal{E}_v) \\ &+ \left(\mathbb{E}_{G_v}\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_u\right] - \mathbb{E}_{G_{vu}}\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_u\right]\right)\mathbb{P}(\mathcal{E}_u) \ .\end{aligned}$$

Similarly,

$$\begin{aligned}\mathsf{B}^t_G(w) - \mathsf{B}^t_{G_u}(w) = &\left(\mathbb{E}_G\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_v\right] - \mathbb{E}_{G_u}\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_v\right]\right)\mathbb{P}(\mathcal{E}_v) \\ &+ \left(\mathbb{E}_G\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_u\right] - \mathbb{E}_{G_u}\left[T^t_w\left(\bar{C}_w\right) \mid \mathcal{E}_u\right]\right)\mathbb{P}(\mathcal{E}_u) \ .\end{aligned}$$

We want to show that it holds

$$\begin{aligned}&\mathbb{E}_{G_v}\left[T^t_w\left(\bar{C}_w \mid \mathcal{E}_v\right)\right] - \mathbb{E}_{G_{vu}}\left[T^t_w\left(\bar{C}_w \mid \mathcal{E}_v\right)\right] \\ &\qquad \le \mathbb{E}_G\left[T^t_w\left(\bar{C}_w \mid \mathcal{E}_v\right)\right] - \mathbb{E}_{G_u}\left[T^t_w\left(\bar{C}_w \mid \mathcal{E}_v\right)\right] \ . \qquad (9)\end{aligned}$$

and

$$\mathbb{E}_{G_v}\left[T_w^t\left(\bar{C}_w \mid \mathcal{E}_u\right)\right] - \mathbb{E}_{G_{vu}}\left[T_w^t\left(\bar{C}_w \mid \mathcal{E}_u\right)\right] \\ \leq \mathbb{E}_{G}\left[T_w^t\left(\bar{C}_w \mid \mathcal{E}_u\right)\right] - \mathbb{E}_{G_u}\left[T_w^t\left(\bar{C}_w \mid \mathcal{E}_u\right)\right] \ . \quad (10)$$

We can write

$$\mathbb{E}_G\left[T_w^t\left(\bar{C}_w\right) \mid \mathcal{E}_v\right] = \sum_{i=1}^{t}\left(i + \mathbb{E}_G\left[T_v^{t-i}\left(\bar{C}_v\right)\right]\right)\mathbb{P}\left(w \overset{=i}{\underset{G}{\rightsquigarrow}} v \mid \mathcal{E}_v\right) \ .$$

The probabilities on the r.h.s. are the same on all graphs. Similar expressions hold for

$$\mathbb{E}_{G_v}\left[T_w^t\left(\bar{C}_w\right) \mid \mathcal{E}_v\right], \quad \mathbb{E}_{G_u}\left[T_w^t\left(\bar{C}_w\right) \mid \mathcal{E}_v\right], \quad \mathbb{E}_{G_{vu}}\left[T_w^t\left(\bar{C}_w\right) \mid \mathcal{E}_v\right],$$

and when conditioning on $\mathcal{E}_u$. To prove (9) and (10), we now show that, for every $t' \leq t$, it holds

$$\mathbb{E}_{G_v}\left[T_v^{t'}\left(\bar{C}_v\right)\right] - \mathbb{E}_{G_{vu}}\left[T_v^{t'}\left(\bar{C}_v\right)\right] \leq \mathbb{E}_{G}\left[T_v^{t'}\left(\bar{C}_v\right)\right] - \mathbb{E}_{G_u}\left[T_v^{t'}\left(\bar{C}_v\right)\right], \quad (11)$$

and

$$\mathbb{E}_{G_v}\left[T_u^{t'}\left(\bar{C}_u\right)\right] - \mathbb{E}_{G_{vu}}\left[T_u^{t'}\left(\bar{C}_u\right)\right] \leq \mathbb{E}_{G}\left[T_u^{t'}\left(\bar{C}_u\right)\right] - \mathbb{E}_{G_u}\left[T_u^{t'}\left(\bar{C}_u\right)\right].$$

We focus on showing (11), as the same steps, with simple modifications, can be followed to show the other inequality. For $Z \in \{G, G_u, G_v, G_{vu}\}$, let $\mathcal{A}_Z \doteq v \overset{\leq t}{\underset{Z}{\rightsquigarrow}} u$, i.e., the event that a random walk in Z starting at v reaches u in at most t steps before visiting any vertex in $\bar{C}_v$, and let $\bar{\mathcal{A}}_Z$ be the complementary event. It holds

$$\mathbb{P}(\mathcal{A}_{G_v}) = \mathbb{P}(\mathcal{A}_{G_{vu}}) \leq \mathbb{P}(\mathcal{A}_G) = \mathbb{P}(\mathcal{A}_{G_u}),$$

due to the insertion of e_v. It also holds

$$\mathbb{E}_{G_v}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \bar{\mathcal{A}}_{G_v}\right] = \mathbb{E}_{G_{vu}}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \bar{\mathcal{A}}_{G_{vu}}\right],$$

and

$$\mathbb{E}_{G}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \bar{\mathcal{A}}_{G}\right] = \mathbb{E}_{G_u}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \bar{\mathcal{A}}_{G_u}\right],$$

Using the law of total expectation (across $\mathcal{A}_Z$ and $\bar{\mathcal{A}}_Z$) and applying these facts, we can rewrite (11) as

$$\left(\mathbb{E}_{G_v}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \mathcal{A}_{G_v}\right] - \mathbb{E}_{G_{vu}}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \mathcal{A}_{G_{vu}}\right]\right)\mathbb{P}(\mathcal{A}_{G_v}) \\ \leq \left(\mathbb{E}_{G}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \mathcal{A}_{G}\right] - \mathbb{E}_{G_u}\left[T_v^{t'}\left(\bar{C}_v\right) \mid \mathcal{A}_{G_u}\right]\right)\mathbb{P}(\mathcal{A}_{G}) \ .$$

The differences between parentheses have the same value, as their corresponding terms have the same values. The inequality holds because $\mathbb{P}(\mathcal{A}_{G_v}) \leq \mathbb{P}(\mathcal{A}_G)$ due to the insertion of e_v in G to obtain G_v. □

Algorithm 1 RePBubLik

1: **Input**: Graph $G = (V, E)$, number of desired insertions k, oracle $\mathcal{W}_G : V \times 2^{V \times V} \rightarrow [0, 1]$, $C \in \{R, B\}$.
2: **Output**: Set Σ of k edges to be inserted, with their weights.
3: $\Sigma \leftarrow \emptyset$
4: **for** $i = 1$ **to** k **do**
5: $P \leftarrow$ `computeParochials`$(G_\Sigma,\ C)$
6: $\mathcal{R} \leftarrow$ `computeRWCentrality`$(P,\ G_\Sigma)$
7: $v_i \leftarrow \mathrm{argmax}_{v \in P} \mathcal{R}(v) \times \mathcal{W}_G(v, \Sigma)$
8: $u_i \leftarrow$ arbitrary in $\bar{C}_{v_i}$
9: $\Sigma \leftarrow \Sigma \cup \{(v_i, u_i)\}$
10: **end for**
11: **return** Σ

We are now ready to prove Thm. 1.

Proof (Thm. 1) Lemma 11 shows the monotonicity and submodularity of the objective function, i.e., of the gain. Thus, a greedy algorithm that picks, iteratively, the k best choices over all parochial vertices of color C as the sources of the added edges, would result in a $(1 + {}^{1}\!/_{e})$-approximation. Lemma 10 shows that by choosing a vertex v maximizing $M(e_v)\mathsf{r}^{t-2}(v; \mathcal{P}_C(G))$ among all parochial vertices of color C, we obtain a vertex such that the gain when adding an edge from this source is a $(2({}^{t}\!/_{r})\gamma_{t-2} + 1)$-approximation to the greedy choice. The correctness of our algorithm is concluded by putting these lemmas together. □

We can now describe RePBubLik in detail (pseudocode in Alg. 1). The algorithm takes as input the graph G, the number k of desired edge insertions, the oracle $\mathcal{W}$ that determines the weights of the new edges, and the set $C \in \{R, B\}$ of nodes of a color. It first creates the empty set Σ that will store the edges to be added and then enters a for-loop to be repeated for k times. At every iteration of the loop, it first computes the BR of every node in C in the graph G_Σ obtained by adding to G the edges in Σ (with their weights obtained from the oracle $\mathcal{W}_G$) (in practice, the BR is computed using the approximation algorithm outlined in Lemma 4). Thanks to this computation, the algorithm obtains (line 5) the set P of parochial nodes in this graph (at the first iteration of the loop $P = \mathcal{P}_C(G)$). It then obtains the centralities values $\mathsf{r}^{t-2}(v; P)$ of every node $v \in P$ (in practice, using the approximation algorithm outlined in Lemma 1), storing them in a dictionary $\mathcal{R}$ (line 6). The algorithm then selects the node $v_i \in P$ associated to the maximum quantity $\mathcal{R}(v_i) \times \mathcal{W}_G(v_i, \Sigma)$, and arbitrarily picks a node u_i of color $\bar{C}_{v_i}$ (i.e., of the color other than C). The directed edge (v_i, u_i) is added to the set Σ (lines 7–9). After k iterations of the loop, the algorithm returns Σ, together with the weights obtained from the oracle.

A practical algorithm RePBubLik would require a re-computation of the BRs and of the centralities of all vertices, at every iteration of the loop, which would require to run a very large number of random walks, making it computationally very expensive. We now propose a more practical alternative RePBubLik+, at the price of losing the approximation guarantees. RePBubLik+ only computes $\mathcal{P}_C(G)$ and $\mathcal{R}$ before entering the for-loop[1] and uses the same values throughout its execution, but trades off the consequences of this choice by adding a penalty factor to the objective function involved in the selection of the source vertices for the edges to be added. Specifically, RePBubLik+ chooses v_i (line 7) by maximizing the quantity ${\mathcal{R}(v)\times\mathcal{W}_G(v,\Sigma)}/{\eta_v}$, where η_v is a penalty factor equals to one plus the number of edges with source v in Σ (thus at iteration 1, $\eta_v = 1$ for every node). This penalty factor favours the insertion of edges from nodes that have not yet been altered. Consequently, it indirectly *(1)* handles the possibility that nodes with new edges are no longer parochial, thus we want to avoid to keep adding edges to them; and *(2)* avoids that the new edges are added from a restricted set of nodes, limiting the positive effect of the insertions on $\Delta(G, \mathcal{P}_C(G), \Sigma, \{M(e)\}_{e\in\Sigma}, t')$.

5 Enhancing the diverse navigability by swapping transition probabilities of links

Adding new edges to a graph is a very invasive operation, and not always possible, for example in recommender systems, where one can assume that the set of edges outgoing from a vertex v is the complete list of items to be recommended when visiting the item corresponding to v, and no item should reasonably be added to this list. Additionally we observe that, in recommender systems, parochial nodes are few, and mitigating their bubble radius has little effect on the entire graph. For these reasons, we now introduce a new measure, called *diverse navigability*, which takes into consideration the bubble radii of *all* nodes.

Definition 3 (Diverse navigability) Given $S \subseteq V$, the *diverse navigability* $\xi(S)$ of S is the opposite of the average BR of the vertices in S, i.e.,

$$\xi(S) \doteq -\frac{1}{|S|}\sum_{v\in S} \mathsf{B}_G^t(v) \quad .$$

When $S = V$, we talk about the diverse navigability of G, and we denote it with $\xi(G)$.

Our strategy to improve the diverse navigability is to *swap the transition probabilities* of edges with the same sources. This strategy is realistic: as demonstrated in observational studies, the transition probabilities of links (outgoing edges) in a given page (vertex) are often tightly correlated with their display features in the page, (e.g., location (Hofmann et al. 2014; Richardson et al. 2007; Collins et al. 2018; Lerman and Hogg 2014; Craswell et al. 2008),

[1] Lemmas 1 and 4 provide bounds of order $\Theta(nt^2)$ on the runtime of this computation. Therefore for small values of t, this approach is more efficient than algorithms that compute hitting times using the Laplacian, which need $\Omega(n^3)$ steps.

background color, font-size, and others (Lamprecht et al. 2016; Dimitrov et al. 2017)). Thus, to achieve the swap of the transition probabilities, one would swap the position (or other presentation factors, e.g., font color or size), on the page corresponding to v, of the two hyperlinks.

We now define the set of *diversifying swaps*, i.e., the set of pairs of nodes that it is reasonable to swap in order to obtain an improvement in the diverse navigability.

Definition 4 (diversifying swaps) For $C \in \{R, B\}$, the set $\Re_C$ of *diversifying swaps* is the set of ordered pairs of edges s.t., for every $(e, e') \in \Re_C$, it holds:

1. $(e, e') = ((v, w), (v, u))$, with $v, w \in C$ and $u \in \bar{C}_v$; and
2. $M(e) > M(e')$; and
3. $w \in \mathcal{P}_C(G)$ (we call w the *radicalizing end point*).[2]

Swapping the transition probabilities of $(e, e') \in \Re_C$ decreases the BR of the source vertex v, and possibly of other vertices in C_v as well, thus improving the diverse navigability $\xi(C_v)$ (and $\xi(G)$). We now introduce a function to quantify this improvement.

Definition 5 For $C \in \{R, B\}$, $u \in C$, and $t' \leq t$, the *gain* $\Gamma(G, u, (e, e'), t')$ on u's BR in G obtained by swapping the transition probabilities of $(e, e') \in \Re_C$, is defined as

$$\Gamma(G, u, (e, e'), t') \doteq \mathsf{B}^{t'}_G(u) - \mathsf{B}^{t'}_{G_{e,e'}}(u),$$

where $G_{e,e'}$ is the same graph as G with only the transition probabilities of e and e' swapped.

The definition extends immediately for a set of edge pairs $\Sigma \subseteq \Re_C$ to $\Gamma(G, u, \Sigma, t')$, and the whole network as follows.

Definition 6 (total gain) Let $\Sigma \subseteq \Re_C$ and G_Σ be the graph obtained by swapping the transition probabilities of every pair of edges in Σ. The *total gain* of Σ is defined as

$$\mathcal{G}(G, \Sigma) \doteq \frac{1}{|C|} \sum_{u \in C} \left(\mathsf{B}^t_G(u) - \mathsf{B}^t_{G_\Sigma}(u) \right) . \quad (12)$$

Although the practical cost of swapping two edge probabilities is small, certainly smaller than modifying the list of edge outgoing from v, it is clearly prohibitively expensive, and unrealistic, to swap the transition probabilities of every pair of edges in $\Re_C$. It is more reasonable to assume that the number of possible swaps is bounded by a parameter k.

Problem 3 Given G, a color C, and a parameter k, find a set $\Sigma \subseteq \Re_C$ of size k such that $\mathcal{G}(G, \Sigma)$ is maximized.

[2] This last requirement is not needed, but restricting to parochial nodes as the radicalizing end points is reasonable in practice, although not necessarily resulting in the best improvement in the diverse navigability.

We design an approximation algorithm SHUFFLIK for Prob. 3, with the following properties (proof in Sect. 5.1).

Theorem 2 *Let $\Sigma \subseteq \Re_C$ be the output of* SHUFFLIK *and* OPT *be the optimal solution to Prob. 3. Then*

$$\mathcal{G}\left(G, \mathsf{OPT}\right) \leq \left(2\frac{t}{r}\ \gamma_{t-2} + 1\right)\left(1 + \frac{1}{\mathrm{e}}\right)\mathcal{G}\left(G, \Sigma\right),$$

We recall that $\gamma_{t'} \doteq \max_{v\in V} \mathcal{F}_{t'}\left(v\right)$ is the maximum, over all $v \in V$, of one plus the probability that a random walk starting from v returns to v before step t', without having visited a vertex in $\bar{C}_v$, and this quantity is a constant for many graphs, and e is the Euler number, also a constant. Since we assume $r \geq t/2$, we obtain that SHUFFLIK gives a *constant-factor approximation.*

Problem 3 is based on the assumption that the *cost* of swapping the transition probabilities of any two edges with the same source is fixed. In practice, these costs may be dependent on arbitrary features, and each pair (e, e') has a *swap cost* $\mathcal{D}(e, e')$, for some function $\mathcal{D} : E \to \mathbb{R}^+$ (assumed known), which can be fixed, e.g., by a website administrator. For example, $\mathcal{D}$ can be the Kendall-tau or Spearman's distance of a swap corresponding to the order they appear in the source (Monjardet 1998; Ceberio et al. 2014; Fligner and Verducci 1986; Fagin et al. 2002; Chierichetti et al. 2018). Alternatively, $\mathcal{D}$ can be the semantic distance of pair of links (Gabrilovich and Markovitch 2007; Zhang et al. 2013; Camacho-Collados and Pilehvar 2018), the financial loss that swapping links would cause, or a combination of all of these measures. The problem then becomes to find a set of swaps that maximizes the diverse navigability within a given *budget* $\mathcal{B}$.

Problem 4 Given $C \in \{R, B\}$, the function $\mathcal{D} : E \times E \to \mathbb{R}^+$, and a budget $\mathcal{B} \in \mathbb{R}^+$, find a set $\Sigma \subseteq \Re_C$ satisfying $\sum_{(e,e')\in\Sigma} \mathcal{D}(e, e') \leq \mathcal{B}$ such that $\mathcal{G}\left(G, \Sigma\right)$ is maximized.

In the next sections, we describe our algorithms and present their theoretical analysis. We start with SHUFFLIK which provides a solution to Prob. 3 in Sect. 5.1. Then in Sect. 5.2, we propose SHUFFLIK+ for Prob. 4.

5.1 SHUFFLIK

Before describing SHUFFLIK, we show that the gain function from (12) is monotonic and sub-modular. This property guides the design of SHUFFLIK, and it is at the basis of its correctness.

Lemma 12 *Let C be a color, and $(e_1, e'_1), (e_2, e'_2) \in \Re_C$. Let $\Sigma = \{(e_1, e'_1), (e_2, e'_2)\}$, and $\delta_1 = |M(e_1) - M(e'_1)|$, $\delta_2 = |M(e_2) - M(e'_2)|$. It holds*

$$\Gamma(G, C, (e_1, e'_1), t) \leq \Gamma(G, C, \Sigma, t), \tag{13}$$

and

$$\Gamma(G, C, \Sigma, t) \leq \Gamma(G, C, (e_1, e'_1), t) + \Gamma(G, C, (e_2, e'_2), t)\ \ .$$

Proof Assume $u, v \in C$ (they can be the same vertex). Let G_v be the graph after swapping (e_v, e'_v), with source vertex v and G_u be the graph after swapping (e_u, e'_u), and G_{vu} be the graph after swapping both pairs.

We first show the monotonicity of the objective function, i.e., that (13) holds. For any $w \in C$, it holds

$$\begin{aligned}\Gamma(G, w, (e_v, e'_v), t) &\doteq \mathsf{B}^t_G(w) - \mathsf{B}^t_{G_v}(w) \le \mathsf{B}^t_G(w) - \mathsf{B}^t_{G_{vu}}(w) \\ &\doteq \Gamma(G, w, \{(e_v, e'_v), (e_u, e'_u)\}, t) \ .\end{aligned}$$

We now show the sub-modularity of the objective function. We start by showing that, for $w \in C$, it holds

$$\Gamma(G, w, \{(e_v, e'_v)(e_u, e'_u)\}, t) \le \Gamma(G, w, (e_v, e'_v), t) + \Gamma(G, w, (e_u, e'_u), t) \ .$$

With an expansion of the definition and a slight rearrangement of the terms, the above inequality is equivalent to

$$\underbrace{\mathsf{B}^t_{G_v}(w) - \mathsf{B}^t_{G_{vu}}(w)}_{\Gamma(G_v, w, (e_u, e'_u), t)} \le \underbrace{\mathsf{B}^t_G(w) - \mathsf{B}^t_{G_u}(w)}_{\Gamma(G, w, (e_u, e'_u), t)} \ .$$

The last inequality can be proved following the same steps as in the proof for Lemma 11. □

Lemma 12 would suggest to greedily choose the pair $(e, e') \in \Re_C$ that maximizes the gain, in order to obtain a constant factor approximation to Prob. 3, but finding such pair is not straightforward. In fact, we only manage to find a pair (e, e') which *approximates* the greedy choice (see Lemma 15).

The idea behind ShuffLik is to greedily choose the pair $(e, e') \in \Re_C$ with source vertex v, that maximizes $\mathsf{r}^{t-2}(v; C) \times |M(e) - M(e')|$.

The following lemma shows that the gain, on the source vertex v, of swapping the edge probabilities of a diversifying pair of edges, is proportional to the RWCC of v.

Lemma 13 *Let $C \in \{R, B\}$ and $(e, e') \in \Re_C$ with source vertex v. It holds*

$$\mathcal{G}(G, (e, e')) \ge \frac{r}{t} \delta \mathsf{r}^{t-2}(v; C) \ .$$

To prove it, we need the following result (proof in App. A).

Lemma 14 *Let $C \in \{R, B\}$, $v \in C$, $t' \le t$, and $(e, e') \in \Re_C$ with v the source vertex and w the radicalizing end point. Let $\delta = M(e) - M(e')$. It holds*

$$\delta \mathsf{B}^{t'-1}_G(w) \le \Gamma(G, v, (e, e'), t') \le \mathcal{F}_{t'-1}(v)\, \delta \mathsf{B}^{t'}_G(w) \ .$$

Proof (Lemma 13) Let $G_{e,e'}$ be the graph we obtain by swapping transition probabilities of e and e'. Using the definition of the gain function we get

$$\begin{aligned}\mathcal{G}\left(G,(e,e')\right) &= \Gamma(G,C,(e,e'),t) \\ &= \frac{1}{|C|}\sum_{u\in C}\sum_{i=1}^{t-1}\Gamma(G,v,(e,e'),t-i)\mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \\ &= \frac{1}{|C|}\sum_{u\in C}\sum_{i=1}^{t-2}\Gamma(G,v,(e,e'),t-i)\mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \quad . \end{aligned} \tag{14}$$

The last line comes from the fact that $\Gamma(G,v,(e,e'),1)=0$, as any walk needs at least one step to reach the other color.

From Lemma 9, we get that $\mathsf{B}_v^t(G)\geq r$ implies $\mathsf{B}_v^{t'}(G)\geq rt'/t$. We now plug-in this lower bound on BR of the source vertex v, and use the lower bound obtained in Lemma 14:

$$\begin{aligned}\mathcal{G}\left(G,(e,e')\right) &\geq \frac{1}{|C|}\sum_{u\in C}\sum_{i=1}^{t-2}\left((t-i)\frac{r}{t}\delta\right)\mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \\ &\geq \frac{r}{t}\delta\frac{1}{|C|}\sum_{u\in C}\sum_{i=1}^{t-2}(t-2-i)\mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v\right) \geq \frac{r}{t}\delta\mathsf{r}^{t-2}(v;C) \quad .\end{aligned}$$

□

We now use Lemma 13 in Lemma 15 to show that a pair of edges (e,e') with source vertex v which maximizes $\mathsf{r}^t(v;C)\times|M(e)-M(e')|$ has a gain that approximates the optimal gain of the greedy choice that maximizes the gain.

Lemma 15 *Let $C\in\{R,B\}$. Among all pairs of edges in $\Re_C$, let $(e_{\mathrm{opt}},e'_{\mathrm{opt}})$ be*

$$(e_{\mathrm{opt}},e'_{\mathrm{opt}}) \doteq \arg\max_{(e,e')\in\Re_C}\Gamma(G,C,(e,e'),t) \quad .$$

Let $(e_{\mathrm{RC}},e'_{\mathrm{RC}})$ be a pair of edges in $\Re_C$ with source vertex v_{RC} such that

$$(e_{\mathrm{RC}},e'_{\mathrm{RC}}) \doteq \arg\max_{(e,e')\in\Re_C}\mathsf{r}^{t-2}(v;C)\left(M(e)-M(e')\right) \quad .$$

It holds

$$\Gamma(G,V,(e_{opt},e'_{opt}),t) \leq \left(2\frac{t}{r}\gamma_{t-2}+1\right)\Gamma(G,V,(e_{RC},e'_{RC}),t) \quad .$$

Proof Let $\delta_{\mathrm{opt}}\doteq M(e_{\mathrm{opt}})-M(e'_{\mathrm{opt}})$, and $\delta_{\mathrm{RC}}\doteq M(e_{\mathrm{RC}})-M(e'_{\mathrm{RC}})$. Employing Lemma 14, for any t', it holds

$$\Gamma(G,v_{\mathrm{opt}},(e_{\mathrm{opt}},e'_{\mathrm{opt}}),t') \leq \mathcal{F}_{t'-1}\left(v_{\mathrm{opt}}\right)\left(\mathsf{B}_G^{t'}\left(w_{\mathrm{opt}}\right)\right)\delta_{\mathrm{opt}} \leq t'\delta_{\mathrm{opt}}\mathcal{F}_{t'-1}\left(v_{\mathrm{opt}}\right), \tag{15}$$

where v_{opt} and w_{opt} are respectively the source vertex and the radicalizing end point of $(e_{\text{opt}}, e'_{\text{opt}})$. From (14), we get

$$\begin{aligned}
&\Gamma(G, C, (e_{\text{opt}}, e'_{\text{opt}}), t) \\
=&\frac{1}{|C|}\sum_{u\in C}\sum_{i=1}^{t-3}\left(\Gamma(G, v_{\text{opt}}, (e_{\text{opt}}, e'_{\text{opt}}), t-i)\right)\mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v_{\text{opt}}\right) \\
&\qquad + \Gamma(G, v_{\text{opt}}, (e_{\text{opt}}, e'_{\text{opt}}), 2)\mathbb{P}\left(u \overset{=t-2}{\underset{G}{\rightsquigarrow}} v_{\text{opt}}\right) \ .
\end{aligned}$$

Continuing by using (15) and Lemma 13 we get

$$\begin{aligned}
&\Gamma(G, C, (e_{\text{opt}}, e'_{\text{opt}}), t) \\
\leq&\frac{1}{|C|}\sum_{u\in C}\sum_{i=1}^{t-3}(t-i)\delta_{\text{opt}}\mathcal{F}_{t-i-1}\left(v_{\text{opt}}\right)\mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v_{\text{opt}}\right) + 1 \\
\leq&\frac{1}{|C|}\sum_{u\in C}\sum_{i=1}^{t-2}2(t-2-i)\delta_{\text{opt}}\mathcal{F}_{t-2}\left(v_{\text{opt}}\right)\mathbb{P}\left(u \overset{=i}{\underset{G}{\rightsquigarrow}} v_{\text{opt}}\right) + 1 \\
\leq&2\frac{t}{r}\delta_{RC}\mathsf{r}^{t-2}(v_{RC}; C_{\text{opt}})\,\gamma_{t-2} + 1 \\
\leq&\left(2\frac{t}{r}\gamma_{t-2} + 1\right)\Gamma(G, C, (e_{\text{RC}}, e'_{\text{RC}}), t) \ .
\end{aligned}$$

□

We can now combine all the above results to prove the correctness of the algorithm.

Proof (Thm. 2) Lemma 12 shows the monotonicity and submodularity of the gain function. Thus, if k swaps are picked greedily over all the pairs in $\Re_C$ we will have a $(1 + {}^1/\mathrm{e})$-approximation. Lemma 15 shows that by choosing a pair $(e, e') \in \Re_C$ with source vertex v that maximizes $(M(e) - M(e')) \times \mathsf{r}^{t-2}(v; C)$, is a $(2({}^t/r)\gamma_{t-2} + 1)$-approximation to the greedy choice. Thus, the correctness of ShuffLik is concluded by putting together these results. □

We can now give the details to ShuffLik (pseudocode in Alg. 2). The algorithm takes as input the graph G, the edges transition probabilities M_G, the number k of desired swaps, and the set of nodes C. It first creates the empty set Σ to store the pairs of edges to be swapped, and then enters a for-loop to be repeated for k times. At every iteration of the loop, it first determines the set of diversifying swaps $\Re_C$ (Def. 4) applying the function `getDiversifyingSwaps` (line 5). The function takes in input G, the color C, and Σ, and returns the set $\Re_C$ of possible swaps in the graph G_Σ, obtained by performing the swaps in Σ. The computation of the radicalizing edges involves the computation of the BR of all vertices in C, using the approximation algorithm described in Lemma 4. Then, the algorithm computes the RWCC of each node in C, using the approximation algorithm from Sect. 3.1, and stores these values in a

Algorithm 2 SHUFFLIK

1: **Input**: Graph $G = (V, E)$, transition matrix M_G, color $C \in \{R, B\}$, desired number of swaps k.
2: **Output**: Set Σ of k pairs of edges to be swapped.
3: $\Sigma \leftarrow \emptyset$
4: **for** $i = 1 : k$ **do**
5: $\quad \Re_C \leftarrow$ `getDiversifyingSwaps`(G, C, Σ)
6: $\quad \mathcal{R} \leftarrow$ `computeRWCentrality`(C)
7: $\quad (e_i, e'_i) \leftarrow \text{argmax}_{(e,e')\in\Re_C} \mathcal{R}(v) \times (M_G(e) - M_G(e'))$
8: $\quad \Sigma \leftarrow \Sigma \cup \{(e_i, e'_i)\}$
9: **end for**
10: **return** Σ

dictionary $\mathcal{R}$ (line 7). At this point, SHUFFLIK selects the pair of edges (e_i, e'_i) associated to the maximum quantity $\mathcal{R}(v) \times |M_G(e) - M_G(e')|$, where v is the source vertex of (e, e'), and adds it to the solution set Σ (lines 7–8). After k iterations of the loop, the algorithm returns Σ.

5.2 SHUFFLIK+

We now present SHUFFLIK+ (Alg. 3), which solves Prob. 4, i.e., it covers the case when the cost of a swap is not a fixed constant and the total cost of the swaps can not exceed a given budget. In addition to the parameters taken by SHUFFLIK, it takes as input a cost function $\mathcal{D} : E \times E \to \mathbb{R}$, and a maximum budget $\mathcal{B} \in \mathbb{R}^+$ (in place of the number of swaps k taken by SHUFFLIK). SHUFFLIK+ builds its solution by greedily choosing the pair of edges to swap. Differently from SHUFFLIK, at each step, before swapping the pair of edges maximizing $\frac{\mathsf{r}^{t-2}(v;C)\times(M(e)-M(e'))}{\mathcal{D}(e,e')}$, it controls whether the cost of the swaps performed so far has exceeded the budget (lines 6–12). If it is the case, the algorithm stops, otherwise it iterates until the budget is exhausted.

We show the correctness of SHUFFLIK+ in the following corollary.

Corollary 1 *Let $\Sigma \subseteq \Re_C$ be the output of* SHUFFLIK+*, and assume it has a cost $\mathcal{B}' \leq \mathcal{B}$. Let* OPT *be the optimal solution to Prob. 4 with budget $\mathcal{B}'$. Then the gains of* OPT *and Σ satisfy Thm. 2.*

Proof Assume we could swap a fraction of two diversifying edges. Thus, for spending each dollar we can take a greedy choice of picking the pair of edges maximizing $\frac{\mathsf{r}^{t-2}(v;C)\times(M(e)-M(e'))}{\mathcal{D}(e,e')}$, and swap a fraction of $\frac{1}{\mathcal{D}(e,e')}$ of (e, e'). Using Thm. 2 this will be a $(1 + 1/e)(2(t/r)\gamma_{t-2} + 1)$-approximation. Note that the initial assumption is not prohibitive if we spend $\mathcal{B}'$ dollars. We thus conclude the result. $\square$

Algorithm 3 ShuffLik+

Input: Graph $G = (V, E)$, transition matrix M_G, color $C \in \{R, B\}$, cost function $\mathcal{D} : E \times E \to \mathbb{R}^+$, budget $\mathcal{B}$.
Output: Set Σ of pairs of edges to be swapped, whose total cost is at most $\mathcal{B}$.
$\Sigma \leftarrow \emptyset$
$\mathcal{B}_\Sigma \leftarrow 0$
while $\mathcal{B}_\Sigma < \mathcal{B}$ **do**
  $\Re_C \leftarrow$ `getDiversifyingSwaps`(G, C, Σ)
  $\mathcal{R} \leftarrow$ `computeRWCentrality`(C)
  $(e_i, e'_i) \leftarrow \arg\max_{(e,e') \in \Re_C} \frac{\mathcal{R}(v) \times |M_G(e) - M_G(e')|}{\mathcal{D}(e,e')}$
  $\mathcal{B}_\Sigma \leftarrow \mathcal{B}_\Sigma + \mathcal{D}(e_i, e'_i)$
  if $\mathcal{B}_\Sigma \leq \mathcal{B}$ **then** $\Sigma \leftarrow \Sigma \cup \{(e_i, e'_i)\}$
  end if
end while
return Σ

6 Experiments

To evaluate the proposed algorithms, namely RePBubLik and ShuffLik, on several graphs, we measure the reduction in structural bias and the improvement in the diverse navigability, respectively.

We conduct separate experiments for the two methods and present them in the following sections. Although the two approaches build upon the same set of concepts and definitions (e.g., the bubble radius and the random walk closeness centrality), we do not perform a comparison between the two because the two algorithms optimize different objective functions, i.e., the structural bias and the diverse navigability through different approaches, namely, edge insertions and traversal probabilities swaps. Because of these distinctions, we also pick distinct baselines and datasets. For reproducibility purposes, we make the code of the experiments available from `https://github.com/CriMenghini/RePBubLik`.

6.1 RePBubLik Evaluation

The goal of our experimental evaluation for RePBubLik is to understand how the addition of the set $\Sigma = \Sigma_R \cup \Sigma_B$ of $K = k_R + k_B$ edges output by the algorithm, run separately with $C = R$ and B, affects the structural bias of the network, by computing the gain in the structural bias reduction. In particular, we measure the gain $\Delta(G, \Sigma)$, introduced in Sect. 4, used here with a simpler notation. We also measure the change $|\mathcal{P}(G)| - |\mathcal{P}(G_\Sigma)|$ after adding Σ.

Datasets We create graphs obtained from *Wikipedia*, *Amazon*[3] and *PolBlogs*[4]. Table 1 shows the relevant statistics.

[3] `https://snap.stanford.edu/data/amazon-meta.html`
[4] `http://www-personal.umich.edu/~mejn/netdata/`

Table 1: Networks' statistics. The notation is consistent with the rest of the paper.

Wikipedia							
Topic	$\|R\|$	$\|B\|$	$\|E\|_{R\to B}$	$\|E\|_{B\to R}$	$\|E\|$	$\%\mathcal{P}_R(G)$	$\%\mathcal{P}_B(G)$
Abort.	208	413	80	170	1911	85.56	89.20
Guns	142	118	72	79	723	82.95	71.69
Pol.	10347	10129	17452	16484	141486	25.97	42.36
Sociol.	602	2283	284	192	10514	91.32	96.36
Amazon							
Topic	$\|R\|$	$\|B\|$	$\|E\|_{R\to B}$	$\|E\|_{B\to R}$	$\|E\|$	$\%\mathcal{P}_R(G)$	$\%\mathcal{P}_B(G)$
MaTe	302	132	25	42	675	90.91	79.63
MiHi	124	169	66	63	482	58.33	63.46
MaAs	293	117	11	6	680	97.31	95.15
PolBlogs							
Topic	$\|R\|$	$\|B\|$	$\|E\|_{R\to B}$	$\|E\|_{B\to R}$	$\|E\|$	$\%\mathcal{P}_R(G)$	$\%\mathcal{P}_B(G)$
Politics	545	488	902	781	17348	87.71	90.37

From *Wikipedia* we consider four bi-partitioned subgraphs related to controversial topics: *politics, abortion, guns* and *sociology* (Menghini et al. 2020). partitions, corresponding respectively to *democrats vs. republican, pro-life vs. pro-choice, control vs. right*, and *individualism vs. collectivism.* Each node in the graph is a page, and is assigned to one color according to Wikipedia's categorization. Directed edges denote links, and are weighted using Wikipedia's clickstream data.[5]

The *Amazon* dataset contains metadata about *books* (Leskovec et al. 2007). Given two book categories, the vertices are all the items in those categories, colored accordingly. There is a directed edge (u, v) if v appears in the list of items similar to u. The edge is weighted by v's sales rank.[6] We built three graphs by considering pairs of the following categories: *Mathematics & Technology (MaTe)*, *History of Technology & Military Science (MiHi)*, and *Mathematics & Astronomy (MaAs).*

The *Political Blogs* dataset is a directed network of hyperlinks between weblogs on US politics (Adamic and Glance 2005). Each node represents a blog and is colored according to its political leaning. Links between blogs were automatically extracted from a crawl of the front page of the blog and represent the edges of the graph. Each edge (v, u) has weight proportional to the out-degree of v.

[5] `https://dumps.wikimedia.org/other/clickstream/`

[6] Amazon sales rank is a metric of the relationship among products within one category based on their sales performance. It expresses how well a product is selling relative to other products in the same category.

Baselines We compare RePBubLik+ to three different baselines (i.e., simplified variants of RePBubLik+) and to two existing algorithms, described in the following. The first baseline, *PureRandom* (PR) selects the source, and the target, nodes of the new edges uniformly at random from the set $\mathcal{P}_C(G)$ and $\bar{C}$, respectively. The second baseline *Random Top-N Central Nodes (N-RCN)*, given a parameter $N \in (0, 100)$, sorts the nodes in $\mathcal{P}_C(G)$ by descending centrality, and picks, uniformly at random, k_C edges with source in the top-N percent of nodes in $\mathcal{P}_C(G)$. The last baseline, *Random Top-N Weighted Central Nodes (N-RWCN)*, differs from N-RCN as the nodes in $\mathcal{P}_C(G)$ are sorted in descending order by $\mathcal{R}(v) \times M(v, u)$.

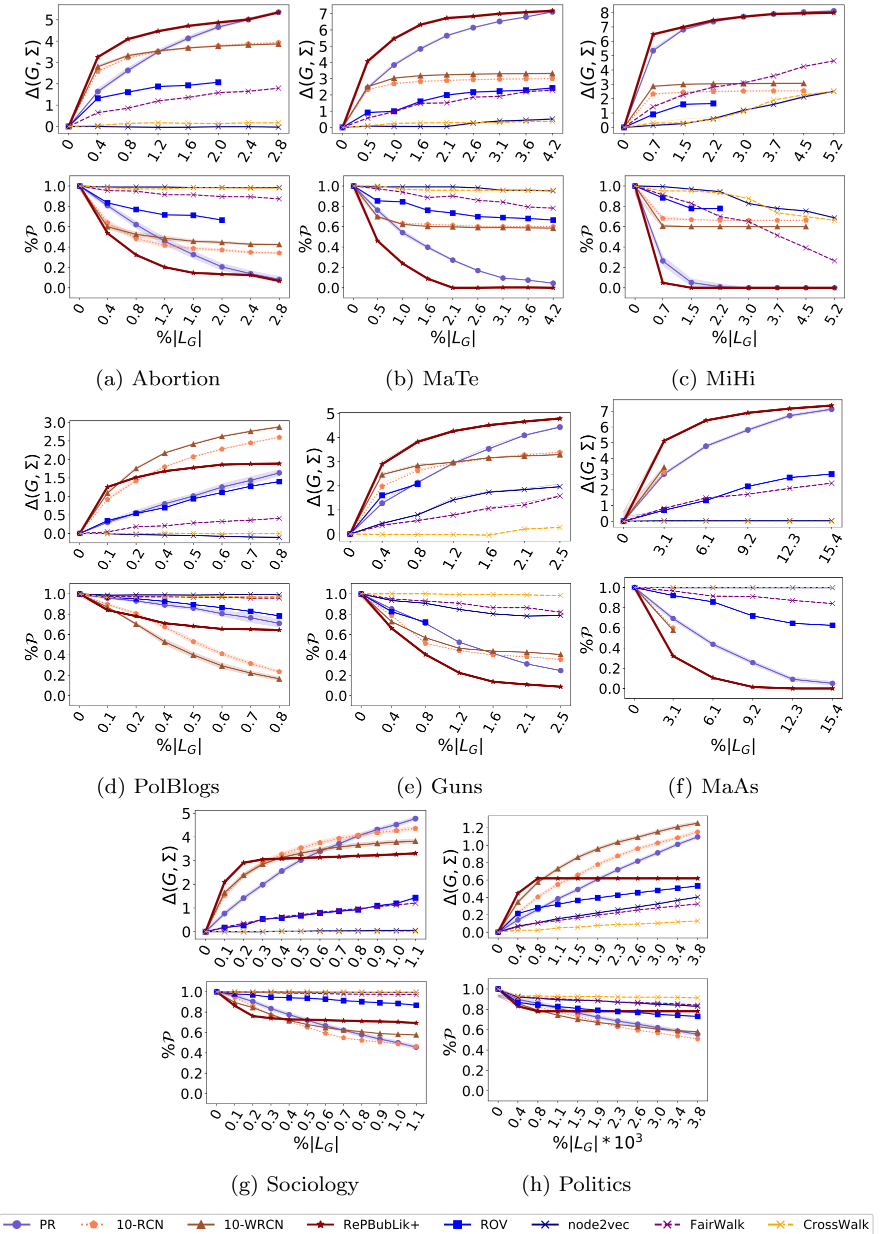
We compare RePBubLik+ also to three existing methods, ROV (Garimella et al. 2017a), node2vec (Grover and Leskovec 2016), and FairWalk (Rahman et al. 2019). The *ROV* algorithm outputs a set of k edges to be added to G to minimize the controversy score (RWC) (Garimella et al. 2018b). The RWC is a metric that characterizes how controversial a topic is by capturing how well separated the two colors are. ROV considers as candidates the edges between the high-degree vertices of each color (Garimella et al. 2017a, Algorithm 1). These edges are sorted by descending impact on the graph controversy score, and the top-k edges are added to the graph. The objective of the comparison between ROV and RePBubLik+ is to verify whether an algorithm developed to minimize the RWC can be used to minimize the structural bias. *Node2vec* is a graph embedding technique that encodes a network in a low-dimensional space retaining characteristics like the nodes' similarity (Grover and Leskovec 2016). The generation of the embedding is based on random walks. One of the main applications of node2vec is to employ the embedding as the feature space to train link recommendation algorithms. The goal of comparing node2vec to RePBubLik+ is to understand how the predictions of widely-used link recommendation algorithms affect the network's structural bias. In the experiments, we create for each network a 128-dimensional space, then we train a logistic regression (avg. AUC 85%) over these features, and we predict the existence probabilities of edges from $\mathcal{P}(G)$. We add to the graph the top k edges according to these probabilities. FairWalk (Rahman et al. 2019) is a variation of node2vec, that imposes the same probability of sampling nodes from the two partitions throughout the walks. We use it to obtain a fair graph embedding space, where the nodes' neighborhoods are expected to be composed of nodes belonging to different partitions of the graph. The link recommendation strategy (avg. AUC 73%) follows the one utilized using node2vec as embedding generator. We compare RePBubLik+ to link recommendation based on FairWalk to understand whether fairified embedding space can help in recommending connections that reduce graphs' polarization. Finally, we also compare RePBubLik+ to CrossWalk (Khajehnejad et al. 2022). CrossWalk creates an embedding to *(1)* increase weights in the peripheries of groups; and *(2)* to generally connect different groups. We want to understand if creating nodes embeddings with this technique favours connections among groups that also help reducing the network's structural bias.

Setup Given a network, we run RePBubLik and the other algorithms on it for increasing values of K, with $K = 1, 2, 4, 6, \ldots, 400$ or 2000 for larger graphs (*Sociology* and *Politics*). These values of K represent only a small percentage of the set of possible edges to insert and correspond to the total number of edges to add to the graph. Once we set the value of K, accordingly, we allocate k_B and k_R of the K edge insertions to each color proportionally to the sum of the BRs of the parochial vertices in each color. That is, we define $Y_C = \sum_{v \in \mathcal{P}_C(G)} \mathsf{B}^t_G(v)$, for $C \in R, B$, then $k_B = \left\lceil k \frac{Y_B}{Y_B + Y_R} \right\rceil$ and $k_R = K - k_B$. This allocation strategy is a simple but reasonable heuristic that ensures that more edges are added from nodes with the color whose parochial nodes contribute more to the structural bias of the network.

We assign the weight $M(v, u) = 1/(d(v) + 1)$ to the added edge (v, u), where $d(v)$ is the out-degree of v before the insertion, and then we re-normalize the weights of the other edges by multiplying each of them by $1 - M(v, u)$. Furthermore, we set $r = 5$ and $b = 2$. Moreover, for the algorithms picking the top-N central nodes, we set $N = 10$. To account for variability of the algorithms, we run them 10 times. The variance of the results is low, overall.

Experiment results In Fig. 1, the plots in the first row show how the structural bias is affected by the insertion of an incrementally larger set of edges, while the ones on the second row show the reduction in the number of parochial nodes. There is a curve for the gain of each algorithm. We draw the following observations, which we comment on in detail below: *(1)* RePBubLik+ performs better than the baselines and the competitors: the gain increases faster after the insertion of just a few edges. *(2)* N-RCN, N-WRC, and ROV after a certain point become flat. *(3)* Overall, RePBubLik+ is the best algorithm. *(4)* The values of RePBubLik+ and PR converge, at different speed, to the same value when we add more edges. *(5)* node2vec, in the best cases, shows little improvement of the structural bias that, in the remaining cases, stays flat or even increases. (6) FairWalk improves over node2vec, but still does not guarantee a significant reduction of the structural bias. (7) CrossWalk behaves similarly to node2vec. We now explain these behaviours using the plots on the second row of Fig. 1.

1. RePBubLik+ chooses edges that directly affect the BR of central nodes and, with a chain effect, the BR of nodes connected to them. More central are the nodes we attach the edges to, higher the structural bias drop is. RePBubLik+achieves a high gain even just after inserting a small set of edges. There is also a significant drop in the number of parochial nodes.
2. N-RCN, N-WRC, and ROV attach edges only to a subset of $\mathcal{P}(G)$ and as k increases, so does the probability of adding multiple edges to the same nodes. These facts imply respectively that, especially on disconnected graphs (see MiHi in Fig. 1c), the addition of edges may affect few nodes, and that even the insertion of more edges does not modify the set of nodes on which the new edges have effect. Thus, the curves of N-RCN, N-WRCN and ROV reach an early saturation that expresses the scarce impact of



(a) Abortion (b) MaTe (c) MiHi

(d) PolBlogs (e) Guns (f) MaAs

(g) Sociology (h) Politics

Fig. 1: The first row shows the $\Delta(G, \Sigma)$ (y-axis) for increasing value of k, reported in terms of $\%\mathcal{L}_G$, the union of possible edges across $\mathcal{P}_C(G)$ and $\bar{C}$ for $C \in R, B$, (x-axis) for each algorithm. Higher values of Δ show more significant reduction of the structural bias. In the second row, we show the percentage of nodes that are still parochial, $\%\mathcal{P} = \frac{|\mathcal{P}(G)|-|\mathcal{P}(G_\Sigma)|}{|\mathcal{P}(G)|}$ after k additions.

subsequent edge additions. This explanation is confirmed by the percentage of parochial nodes, which does not decrease after the saturation point. Furthermore, the ROV shows a stepping behaviour due to it selecting edges between high-degree central nodes that minimize the RWC without imposing diversity constraints on nodes. And resulting in many selected edges being attached to the same node. Last, we see that on *Polblogs* the best algorithms are N-RCN, N-WRCN. This surprising superiority of the random approaches can be explained by the fact that *Polblogs* is a connected graph, thus edges added to the top-central nodes potentially affect all the nodes in $\mathcal{P}(G)$. Thus, even when N-RCN and N-WRCN add multiple new edges to the same set of nodes, Δ continues to increase.

3. RePBubLik+ shows a consistent behaviour, indeed it increases the gain faster than other methods, requiring fewer insertions. The penalty factor η allows the algorithm to diversify the set of nodes to which the new edges attach, raising the chances of lowering the BR of a larger number of parochial nodes, thus increasing the gain. This feature is important especially on disconnected graphs, where the vertices in tiny connected components always have lower centrality compared to those in huge ones. More importantly, we observe that the size of $\mathcal{P}(G)$ is often reduced to 0: RePBubLik+ is able to "heal" all the bad vertices, and if we measured the structural bias on the obtained graph it would be zero. When adding a very large number of edges, other methods do better than RePBubLik+ on some graphs, but this situation is not very realistic: usually one can really only add a small fraction of edges.
4. The variants of RePBubLik: RePBubLik+ and PR, pick edges from the same candidate set, thus the more edges they can pick, the more likely they choose edges with similar effect, thus the average parochial nodes' BR converges. This is the main explanation why the random algorithm performs so well.
5. Generally, link recommendation algorithms tend to suggest edges between similar nodes. Node2vec captures this similarity through the nodes' neighborhood. In this context, graphs partitions have high within- and low between-density. Nodes in the same partition then lie close in the embedding space. Edges suggested by node2vec with high probability connect nodes close to each other in the embedding, which often are in the same partition. Thus, node2vec has a hard time reducing the structural bias, and in some cases increases it.
6. Aligned with our expectations, link recommendation algorithms based on FairWalk reduce the structural bias more than node2vec. This suggests that a fair embedding space improve the connectivity between the graph's partitions. However, this is insufficient to significantly reduce the structural bias.
7. CrossWalk does not reduce the structural bias as much as FairWalk, indeed it behaves similarly to node2vec. We believe that this is a consequence of the algorithm design to create the embeddings. Specifically, the strategy of increasing the weights in the peripheries of groups is quite the opposites of

the goal of RepBubLiK, which adds connections from nodes at the core of a group.

6.2 ShuffLik Evaluation

We report here the results of the evaluation of ShuffLik, to assess its effect on the diverse navigability on a set of graphs. We measure $\xi(C)$ for an increasing number k of swaps. Another goal of our experiments is to compare the diverse navigability of recommendation networks built imposing a diversity constraint to diverse navigability obtained by applying ShuffLik to an unconstrained graph of recommendations.

Data The *MovieLens* dataset gathers 25M ratings provided by $u \sim 162,000$ users about $m \sim 60,000$ movies. We consider only the movies whose genre is known. Starting from the *25M MovieLens dataset*[7] (Harper and Konstan 2015), we generate multiple recommendation networks. Given the set of all movies, we fix a genre g and assign the same color to all movies of genre g, obtaining the set C. The remaining items belong to $\bar{C}$. For each genre g, we build two graphs of recommendations whose nodes are movies and edges indicate a directed recommendation relationship. For each movie i, let $\mathcal{L}_i$ be the set of its neighbors. In the first kind of graph, named *vanilla-RecNet*, $\mathcal{L}_i$ contains the ℓ most similar movies to i, obtained with an Item Collaborative Filtering (CF) based on the Alternative Least Square (ALS) (Hu et al. 2008). In the second kind of graph, *div-RecNet*, a constraint on the diversity of the neighbors is imposed, so that at least the 50% of the neighborhood of each $i \in C$ belongs to $\bar{C}$. This notion of diversity is a variant, on our data, of the metrics defined by Vargas et al. (2014); Anagnostopoulos et al. (2020). Both networks have weighted edges as follows. Given an edge (i, j) its weight is directly proportional to *(1)* its position within the list of suggested items, *(2)* the portion of users rating that item over the number of total users, *(3)* the number of reviews it received. Items *(2)* and *(3)* are quantities we infer from the original dataset, while, to obtain *(1)*, we evaluate a power-law density on ℓ equidistant points. By using the power-law distribution, we model the probability of clicking links differently positioned within the page. In particular, we assign higher likelihood to items at the top of the list (Hofmann et al. 2014; Richardson et al. 2007; Collins et al. 2018). We pick as parameter of the power-law distribution $\alpha = 0.3$. This way, suggestions at the top of the list are more likely to be clicked. In Table 2, we show the size of the genres and the number of possible *diversifying swaps*, $\Re_C$ (Def. 4). We indicate the number of swaps for both G and *div-RecNet*. We remark that the size of the graph is invariant for all topics and all kinds of recommendation graphs. In particular, the vertices are 54479 and the edges $\ell \times 54479$.

[7] `https://grouplens.org/datasets/movielens/25m/`

Table 2: Recommendation Networks' statistics.

	Action	*Adventure*	*Animation*	*Children*	*Crime*	*Documentary*
$\|C\|$	16051	349	3502	7305	195	5024
$\Re_C^G$	91689	57735	45948	34710	73064	46299
$\Re_C^{div-RecNet}$	156846	65043	80095	57995	77442	84309
	Drama	*Film-Noir*	*Horror*	*Romance*	*Sci-Fi*	*Thriller*
$\|C\|$	16784	5453	5746	1770	8330	3868
$\Re_C^G$	377669	8190	59857	100628	58720	137137
$\Re_C^{div-RecNet}$	724682	9068	155949	129147	73360	180766

Baseline We compare ShuffLik to a baseline *WeightDifference* (WD) which chooses the pair of edges whose weight to swap based only on absolute difference between their edge weights. It differs from ShuffLik because it ignores the centrality of the edges' source.

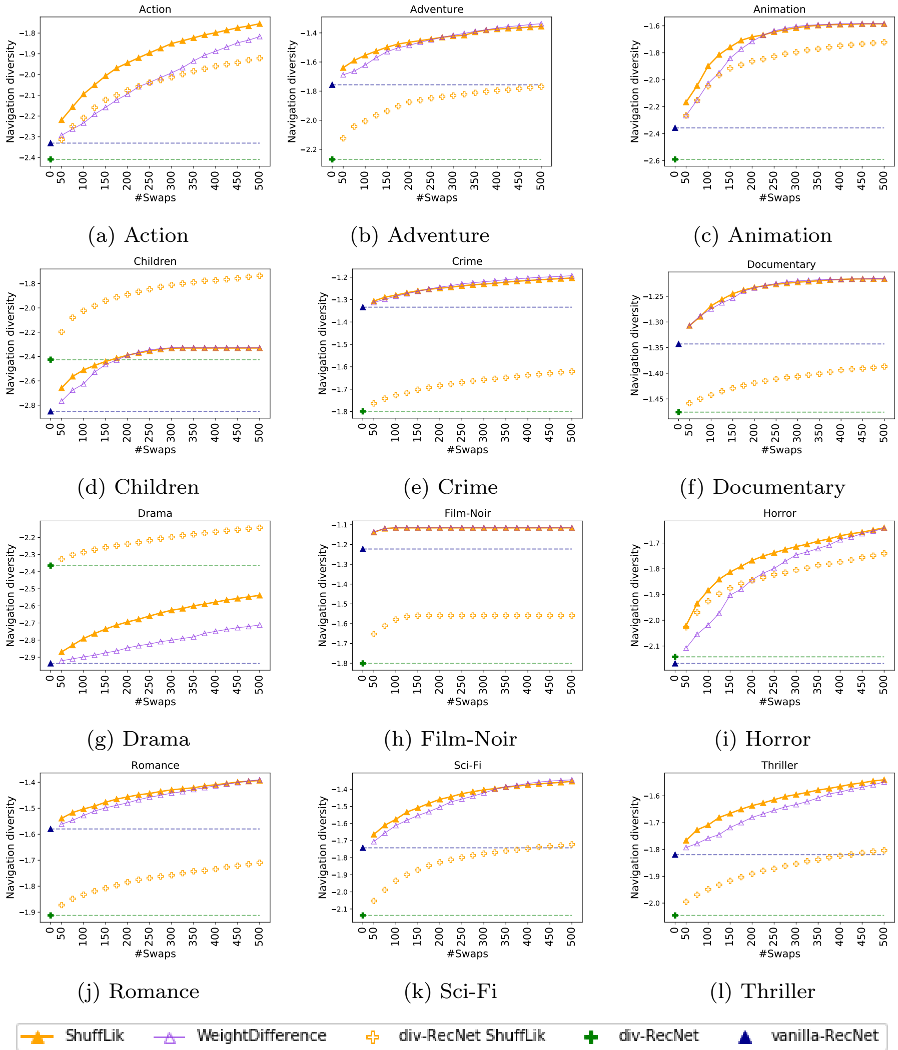
Setup We set the number of recommendation ℓ to 20. The parameter t used to bound the random walks is 10. The number k of weight swaps varies between 1 and 500. Even 500 is a very small fraction of the possible swaps for all graphs.

Results Figure 2 shows the effect of weight swapping on the diverse navigability (additional results for remaining genres are available inside the code folder).

We draw the following observations from Fig. 2 (detailed below): *(1)* in a few cases, initially $\xi(C)$ on *div-RecNet* is higher (i.e., better) than the one measured on *vanilla-RecNet*; *(2)* ShuffLik, even if applied to *vanilla-RecNet* achieves, a better diverse navigability for C than the one obtained by the diversity constrained construction of *div-RecNet*; *(3)* ShuffLik works as well or better than the baseline *WD*, especially for a lower number of swaps, but the values of $\xi(C)$ achieved by ShuffLik and WD sometimes are similar or converge as the number of allowed swaps increases; *(4)* when $\xi(C)$ is initially higher on *div-RecNet*, the overall best diverse navigability $\xi(C)$ is obtained by applying ShuffLik on the network *div-RecNet* obtained by imposing the diversity constraint (obviously, this solution is also the most expensive).

We now give the details about each observation.

1. Generally, imposing a diversity constraint on the set of recommended items can have a beneficial impact on diverse navigability, as at every step of the random walk there is a non-zero probability of traversing an edge to a node of different colors. We stress that imposing the diversity constraint to build *div-RecNet* has a significant cost in terms of the quality of the recommendations, i.e., the outgoing edges from each vertex i, because an item that is among the ℓ-most similar to i is replaced with one of smaller similarity in order to increase the diversity of the whole recommendation list. On the contrary, the cost of ShuffLik is quite limited: the suggested items loose their initial order but the goodness of set of items in $\mathcal{L}_i$ stays the



(a) Action (b) Adventure (c) Animation

(d) Children (e) Crime (f) Documentary

(g) Drama (h) Film-Noir (i) Horror

(j) Romance (k) Sci-Fi (l) Thriller

Fig. 2: Diverse navigability. On the x-axis is the number of edge swaps. We report $\xi(C)$ as it changes after applying ShuffLik or the baselines, the higher the better. Marker shapes identify the kind of graphs, while curve colors identify algorithms. The blue triangle and its corresponding dashed line indicates the diverse navigability on the initial *vanilla-RecNet* network. The green thick cross and its corresponding dashed line $\xi(C)$ are for *div-RecNet*.

same. Additionally, we note that in most of the cases, imposing the diversity constraint does not imply better graph navigability and a sufficient increase in the diverse navigability.

2. Swapping edge weights increases the probability of reaching the other color. With a relative small number of swaps, ShuffLik achieves, in most cases, the same or higher diverse navigability obtained by imposing a diversity constraint, and does so without altering the set of items recommended by an unconstrained system. This results shows the major effectiveness of ShuffLik over imposing a diversity constraint, since the weight swaps are much cheaper than the changes in the recommendation list required by imposing a diversity constraint. In one case (e.g., Fig. 2g) ShuffLik does not achieve the initial value of $\xi(C)$ in *div-RecNet*, but this fact is more a witness of the fact that the initial diverse navigability of *vanilla-RecNet* was particularly bad, than of the performance of ShuffLik.
3. Often, ShuffLik reaches a higher diverse navigability within fewer swaps than WD. This result essentially demonstrates that switching edge weights within pages that are central has a direct and significant impact on their BR and, with a domino effect, on all the nodes reaching them. On the contrary, WD can easily swap edges that do not have impact on many nodes. The two values start to converge when the effect of centrality dissipates. The latter happen because we do not update the nodes centralities for computational reasons.
4. It is pretty obvious that applying ShuffLik on *div-RecNet* when the initial navigability is higher results in the best diverse navigability, but it is important to remark that the cost of this operation would include both the cost of the changes in the recommendations due to the diversity constraint imposed in the construction of *div-RecNet*, and the cost in the change of ordering due to ShuffLik. In some cases, the performance of ShuffLik on *vanilla-DivRec* and ShuffLik on *div-RecNet* are similar (see, e.g., Fig. 2a). Still, the first solution has a much lower cost, only determined by swapping a few edges. We can generalize this observation by saying that, using ShuffLik, we can aim to the same level of diverse navigability without the need of imposing diversity constraints on the set of recommended items.

7 Conclusion

We define and study the problems of reducing the bubble diameter and increasing the diverse navigability of graphs by either inserting new edges or swapping edge traversal probabilities.

Our algorithms perform at most k iterations (either insertions or swaps). They exploit the monotonicity and submodularity of the objective functions, to take a greedy approach that is based on a task-specific variant of the random walk closeness centrality. Under mild conditions, they offer a constant factor approximation guarantee.

The results of our experimental evaluation show that the edge insertions suggested by our algorithm for this task result in a much quicker decrease of the structural bias than existing methods and reasonable baselines. Our swap-based algorithm is able to increase the diverse navigability more than an appropriate baseline and it performs particularly well when combined with diversity constraints on top of existing standard recommendation algorithms.

Acknowledgements Shahrzad Haddadan was supported by NSF Award CCF-1740741. Part of Cristina Menghini's work was done while visiting Brown University and is supported by the ERC Advanced Grant 788893 AMDROMA@. Matteo Riondato is supported in part by NSF award IIS-2006765. Eli Upfal was supported in part by NSF awards RI-1813444, and CCF-1740741. We thank an anonymous WSDM'21 reviewer for correcting one of our lemmas.

Conflict of interest

The authors declare that they have no conflict of interest.

References

Adamic LA, Glance N (2005) The political blogosphere and the 2004 U.S. election: Divided they blog. In: Proceedings of the 3rd International Workshop on Link Discovery, Association for Computing Machinery, New York, NY, USA, LinkKDD '05, pp 36–43, DOI 10.1145/1134271.1134277, URL https://doi.org/10.1145/1134271.1134277

Akoglu L (2014) Quantifying political polarity based on bipartite opinion networks. In: Eighth International AAAI Conference on Weblogs and Social Media

Anagnostopoulos A, Becchetti L, Fazzone A, Menghini C, Schwiegelshohn C (2020) Spectral Relaxations and Fair Densest Subgraphs, Association for Computing Machinery, New York, NY, USA, pp 35–44. URL https://doi.org/10.1145/3340531.3412036

Angriman E, van der Grinten A, Bojchevski A, Zügner D, Günnemann S, Meyerhenke H (2020) Group centrality maximization for large-scale graphs. In: 2020 Proceedings of the Twenty-Second Workshop on Algorithm Engineering and Experiments (ALENEX)

Aridor G, Goncalves D, Sikdar S (2020) Deconstructing the filter bubble: User decision-making and recommender systems. In: Fourteenth ACM Conference on Recommender Systems, Association for Computing Machinery, New York, NY, USA, RecSys '20, pp 82–91, DOI 10.1145/3383313.3412246, URL https://doi.org/10.1145/3383313.3412246

Arrigo F, Benzi M (2016a) Edge modification criteria for enhancing the communicability of digraphs. SIAM Journal on Matrix Analysis and Applications 37(1):443–468

Arrigo F, Benzi M (2016b) Updating and downdating techniques for optimizing network communicability. SIAM Journal on Scientific Computing 38(1):B25–B49

Aslay C, Matakos A, Galbrun E, Gionis A (2018) Maximizing the diversity of exposure in a social network. In: 2018 IEEE International Conference on Data Mining (ICDM), pp 863–868

Baeza-Yates R (2020) Bias in search and recommender systems. In: Fourteenth ACM Conference on Recommender Systems, Association for Computing Machinery, New York, NY, USA, RecSys '20, p 2, DOI 10.1145/3383313.3418435, URL https://doi.org/10.1145/3383313.3418435

Bakshy E, Messing S, Adamic LA (2015) Exposure to ideologically diverse news and opinion on facebook. Science 348(6239):1130–1132

Becker R, Corò F, D'Angelo G, Gilbert H (2020) Balancing spreads of influence in a social network. Proceedings of the AAAI Conference on Artificial Intelligence 34(01):3–10

Ben-Hamou A, Oliveira RI, Peres Y (2018) Estimating graph parameters via random walks with restarts. In: Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, USA, SODA '18, pp 1702–1714

Benhabib S (1996) Toward a deliberative model of democratic legitimacy. In: Democracy and difference: Contesting the boundaries of the political, Princeton University Press, Princeton, N.J., pp 67–94

Bera SK, Seshadhri C (2020) How to count triangles, without seeing the whole graph. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, New York, NY, USA, KDD '20, pp 306–316, DOI 10.1145/3394486.3403073, URL https://doi.org/10.1145/3394486.3403073

Bergamini E, Crescenzi P, D'Angelo G, Meyerhenke H, Severini L, Velaj Y (2018) Improving the betweenness centrality of a node by adding links. Journal of Experimental Algorithmics (JEA) 23:1–32

Biega AJ, Gummadi KP, Weikum G (2018) Equity of attention: Amortizing individual fairness in rankings. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA, SIGIR '18, pp 405–414, DOI 10.1145/3209978.3210063, URL https://doi.org/10.1145/3209978.3210063

Blex C, Yasseri T (2022) Positive algorithmic bias cannot stop fragmentation in homophilic networks. The Journal of Mathematical Sociology 46(1):80–97

Camacho-Collados J, Pilehvar MT (2018) From word to sense embeddings: A survey on vector representations of meaning. Journal of Artificial Intelligence Research 63(1):743–788, DOI 10.1613/jair.1.11259, URL https://doi.org/10.1613/jair.1.11259

Castells P, Hurley NJ, Vargas S (2015) Novelty and Diversity in Recommender Systems, Springer, Boston, MA, pp 881–918. DOI 10.1007/978-1-4899-7637-6_26, URL https://doi.org/10.1007/978-1-4899-7637-6_26

Ceberio J, Irurozki E, Mendiburu A, Lozano JA (2014) A distance-based ranking model estimation of distribution algorithm for the flowshop scheduling problem. IEEE Transactions on Evolutionary Computation 18(2):286–300, DOI 10.1109/TEVC.2013.2260548

Celis LE, Straszak D, Vishnoi NK (2017) Ranking with fairness constraints. arXiv preprint arXiv:170406840

Celis LE, Kapoor S, Salehi F, Vishnoi N (2019) Controlling polarization in personalization: An algorithmic framework. In: Proceedings of the conference on fairness, accountability, and transparency, pp 160–169

Chan H, Akoglu L, Tong H (2014) Make it or break it: Manipulating robustness in large networks. In: Proceedings of the 2014 SIAM International Conference on Data Mining, SIAM, pp 325–333

Chierichetti F, Haddadan S (2018) On the complexity of sampling vertices uniformly from a graph. In: 45th International Colloquium on Automata, Languages, and Programming, ICALP 2018

Chierichetti F, Dasgupta A, Haddadan S, Kumar R, Lattanzi S (2018) Mallows models for top-k lists. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS'18, pp 4387–4397

Chitra U, Musco C (2020) Analyzing the impact of filter bubbles on social network polarization. In: Proceedings of the 13th International Conference on Web Search and Data Mining, ACM

Collins A, Tkaczyk D, Aizawa A, Beel J (2018) A study of position bias in digital library recommender systems. arXiv preprint arXiv:180206565

Conover MD, Ratkiewicz J, Francisco M, Gonçalves B, Menczer F, Flammini A (2011) Political polarization on Twitter. In: Fifth international AAAI conference on weblogs and social media

Cossard A, De Francisci Morales G, Kalimeri K, Mejova Y, Paolotti D, Starnini M (2020) Falling into the echo chamber: The Italian vaccination debate on Twitter. In: Proceedings of the International AAAI Conference on Web and Social Media

Craswell N, Zoeter O, Taylor M, Ramsey B (2008) An experimental comparison of click position-bias models. In: Proceedings of the 2008 International Conference on Web Search and Data Mining, Association for Computing Machinery, New York, NY, USA, WSDM '08, pp 87–94, DOI 10.1145/1341531.1341545, URL https://doi.org/10.1145/1341531.1341545

D'Angelo G, Olsen M, Severini L (2019) Coverage centrality maximization in undirected networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 33, pp 501–508

Das A, Gollapudi S, Munagala K (2014) Modeling opinion dynamics in social networks. In: Proceedings of the 7th ACM international conference on Web search and data mining, pp 403–412

Dasgupta A, Kumar R, Sarlos T (2014) On estimating the average degree. In: Proceedings of the 23rd International Conference on World Wide Web, Association for Computing Machinery, New York, NY, USA, WWW '14, pp 795–806, DOI 10.1145/2566486.2568019, URL https://doi.org/10.1145/2566486.2568019

Demaine ED, Zadimoghaddam M (2010) Minimizing the diameter of a network using shortcut edges. In: Scandinavian Workshop on Algorithm Theory, Springer, pp 420–431

Dimitrov D, Singer P, Lemmerich F, Strohmaier M (2017) What makes a link successful on wikipedia? In: Proceedings of the 26th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, WWW '17, pp 917–926, DOI 10.1145/3038912.3052613, URL https://doi.org/10.1145/3038912.3052613

Dumitriu I, Tetali P, Winkler P (2003) On playing golf with two balls. SIAM J Discrete Math 16:604–615

Fagin R, Karlin A, Kleinberg J, Raghavan P, Rajagopalan S, Rubinfeld R, Tomkins A (2001) Random walks with "back buttons". The Annals of Applied Probability 11

Fagin R, Kumar R, Sivakumar D (2002) Comparing top k lists. SIAM Journal on Discrete Mathematics 17, DOI 10.1137/S0895480102412856

Flaxman S, Goel S, Rao JM (2016) Filter bubbles, echo chambers, and online news consumption. Public opinion quarterly 80(S1):298–320

Fligner MA, Verducci JS (1986) Distance based ranking models. Journal of the Royal Statistical Society Series B (Methodological) 48(3):359–369

Fouss F, Pirotte A, Renders Jm, Saerens M (2007) Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. IEEE Transactions on Knowledge and Data Engineering 19(3):355–369, DOI 10.1109/TKDE.2007.46

Fu Z, Xian Y, Gao R, Zhao J, Huang Q, Ge Y, Xu S, Geng S, Shah C, Zhang Y, de Melo G (2020) Fairness-aware explainable recommendation over knowledge graphs. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, DOI 10.1145/3397271.3401051, URL https://doi.org/10.1145%2F3397271.3401051

Gabrilovich E, Markovitch S (2007) Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, IJCAI'07, pp 1606–1611

Garimella K, De Francisci Morales G, Gionis A, Mathioudakis M (2017a) Reducing controversy by connecting opposing views. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM '17

Garimella K, Gionis A, Parotsidis N, Tatti N (2017b) Balancing information exposure in social networks. In: Advances in Neural Information Processing Systems, pp 4663–4671

Garimella K, De Francisci Morales G, Gionis A, Mathioudakis M (2018a) Political discourse on social media: Echo chambers, gatekeepers, and the price of bipartisanship. In: Proceedings of the 2018 World Wide Web Conference, pp 913–922

Garimella K, Morales GDF, Gionis A, Mathioudakis M (2018b) Quantifying controversy on social media. ACM Transactions on Social Computing

Ge M, Delgado-Battenfeld C, Jannach D (2010) Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In: Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10, pp 257–260

Ge Y, Zhao S, Zhou H, Pei C, Sun F, Ou W, Zhang Y (2020) Understanding echo chambers in e-commerce recommender systems. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA, pp 2261–2270, URL https://doi.org/10.1145/3397271.3401431

Geyik SC, Ambler S, Kenthapadi K (2019) Fairness-aware ranking in search & recommendation systems with application to LinkedIn talent search. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, DOI 10.1145/3292500.3330691, URL https://doi.org/10.1145%2F3292500.3330691

Grover A, Leskovec J (2016) node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, pp 855–864

Haddadan S, Menghini C, Riondato M, Upfal E (2021) RePBubLik: Reducing polarized bubble radius with link insertions. In: Lewin-Eytan L, Carmel D, Yom-Tov E, Agichtein E, Gabrilovich E (eds) WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021, ACM, pp 139–147, DOI 10.1145/3437963.3441825

Harper FM, Konstan JA (2015) The movielens datasets: History and context. ACM Trans Interact Intell Syst 5(4), DOI 10.1145/2827872, URL https://doi.org/10.1145/2827872

Helberger N, Karppinen K, D'Acunto L (2018) Exposure diversity as a design principle for recommender systems. Information, Communication & Society 21(2):191–207, DOI 10.1080/1369118X.2016.1271900, URL https://doi.org/10.1080/1369118X.2016.1271900, https://doi.org/10.1080/1369118X.2016.1271900

Hofmann K, Schuth A, Bellogín A, de Rijke M (2014) Effects of position bias on click-based recommender evaluation. In: de Rijke M, Kenter T, de Vries AP, Zhai C, de Jong F, Radinsky K, Hofmann K (eds) Advances in Information Retrieval, Springer International Publishing, Cham, pp 624–630

Hu Y, Koren Y, Volinsky C (2008) Collaborative filtering for implicit feedback datasets. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, IEEE Computer Society, USA, ICDM '08, pp 263–272, DOI 10.1109/ICDM.2008.22, URL https://doi.org/10.1109/ICDM.2008.22

Hua JL, Yu J, Yang M (2020) Fast clustering for signed graphs based on random walk gap. Soc Networks 60:113–128

Isenberg DJ (1986) Group polarization: A critical review and meta-analysis. Journal of personality and social psychology 50(6):1141

Jiang R, Chiappa S, Lattimore T, György A, Kohli P (2019) Degenerate feedback loops in recommender systems. In: Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, pp 383–390

Jung J, Jin W, Kang U (2019) Random walk-based ranking in signed social networks: model and algorithms. Knowledge and Information Systems 62:571–610

Khajehnejad A, Khajehnejad M, Babaei M, Gummadi KP, Weller A, Mirzasoleiman B (2022) Crosswalk: Fairness-enhanced node representation learning. In: AAAI

Kotkov D, Veijalainen J, Wang S (2016) Challenges of serendipity in recommender systems. In: WEBIST 2016: Proceedings of the 12th International conference on web information systems and technologies.

Kumar S, Hamilton WL, Leskovec J, Jurafsky D (2018) Community interaction and conflict on the web. In: Proceedings of the 2018 World Wide Web Conference, pp 933–943

Kunaver M, Porl T (2017) Diversity in recommender systems a survey. Know-Based Syst 123(C):154–162, DOI 10.1016/j.knosys.2017.02.009, URL https://doi.org/10.1016/j.knosys.2017.02.009

Lamprecht D, Dimitrov D, Helic D, Strohmaier M (2016) Evaluating and improving navigability of wikipedia: A comparative study of eight language editions. In: Proceedings of the 12th International Symposium on Open Collaboration, ACM, New York, NY, USA, OpenSym '16, pp 17:1–17:10, DOI 10.1145/2957792.2957813, URL http://doi.acm.org/10.1145/2957792.2957813

LeFebvre R (2017) Obama Foundation taps social media to fight online echo chambers, https://www.engadget.com/2017-07-05-obama-foundation-social-media-echo-chambers.html

Lerman K, Hogg T (2014) Leveraging position bias to improve peer recommendation. In: PLoS One, URL doi:10.1371/journal.pone.0098914

Leskovec J, Adamic LA, Huberman BA (2007) The dynamics of viral marketing. ACM Transactions on the Web (TWEB) 1(1):5–es

Liao QV, Fu WT (2014a) Can you hear me now? mitigating the echo chamber effect by source position indicators. In: Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing, pp 184–196

Liao QV, Fu WT (2014b) Expert voices in echo chambers: effects of source expertise indicators on exposure to diverse opinions. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp 2745–2754

Mahmoody A, Tsourakakis CE, Upfal E (2016) Scalable betweenness centrality maximization via sampling. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining

Matakos A, Terzi E, Tsaparas P (2017) Measuring and moderating opinion polarization in social networks. Data Mining and Knowledge Discovery 31:1480–1505

Matakos A, Tu S, Gionis A (2020) Tell me something my friends do not know: diversity maximization in social networks. Knowledge and Information Systems (9):3697–3726

Medya S, Silva A, Singh A, Basu P, Swami A (2018) Group centrality maximization via network design. In: Proceedings of the 2018 SIAM International Conference on Data Mining, SIAM, pp 126–134

Menghini C, Anagnostopoulos A, Upfal E (2019) Wikipedia polarization and its effects on navigation paths. In: 2019 IEEE International Conference on Big Data (Big Data), pp 6154–6156

Menghini C, Anagnostopoulos A, Upfal E (2020) Wikipedia's network bias on controversial topics. URL https://arxiv.org/abs/2007.08197

Monjardet B (1998) On the comparison of the spearman and kendall metrics between linear orders. Discrete Mathematics 192(1):281 – 292, DOI https://doi.org/10.1016/S0012-365X(98)00076-4, URL http://www.sciencedirect.com/science/article/pii/S0012365X98000764

Morales AJ, Borondo J, Losada JC, Benito RM (2015) Measuring political polarization: Twitter shows the two sides of Venezuela. Chaos: An Interdisciplinary Journal of Nonlinear Science 25(3):033114

Mossel E, Tamuz O (2017) Opinion exchange dynamics. Probability Surveys 14:155–204

Munson SA, Lee SY, Resnick P (2013) Encouraging reading of diverse political viewpoints with a browser widget. In: Seventh International AAAI Conference on Weblogs and Social Media

Musco C, Musco C, Tsourakakis CE (2018) Minimizing polarization and disagreement in social networks. In: Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18

Nelimarkka M, Laaksonen SM, Semaan B (2018) Social media is polarized, social media is polarized: towards a new design agenda for mitigating polarization. In: Proceedings of the 2018 Designing Interactive Systems Conference, pp 957–970

Nguyen TT, Hui PM, Harper FM, Terveen L, Konstan JA (2014) Exploring the filter bubble: The effect of using recommender systems on content diversity. In: Proceedings of the 23rd International Conference on World Wide Web, Association for Computing Machinery, New York, NY, USA, WWW '14, pp 677–686, DOI 10.1145/2566486.2568012, URL https://doi.org/10.1145/2566486.2568012

O'Callaghan D, Greene D, Conway M, Carthy J, Cunningham P (2015) Down the (white) rabbit hole: The extreme right and online recommender systems. Social Science Computer Review 33(4):459–478, DOI 10.1177/0894439314555329, URL https://doi.org/10.1177/0894439314555329, https://doi.org/10.1177/0894439314555329

Papagelis M, Bonchi F, Gionis A (2011) Suggesting ghost edges for a smaller world. In: Proceedings of the 20th ACM international conference on Information and knowledge management, pp 2305–2308

Pariser E (2011) The filter bubble: What the Internet is hiding from you. Penguin

Parotsidis N, Pitoura E, Tsaparas P (2015) Selecting shortcuts for a smaller world. In: Proceedings of the 2015 SIAM International Conference on Data Mining, SIAM, pp 28–36

Parotsidis N, Pitoura E, Tsaparas P (2016) Centrality-aware link recommendations. In: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, pp 503–512

Perumal S, Basu P, Guan Z (2013) Minimizing eccentricity in composite networks via constrained edge additions. In: MILCOM 2013-2013 IEEE Military Communications Conference, pp 1894–1899

Rahman TA, Surma B, Backes M, Zhang Y (2019) Fairwalk: Towards fair graph embedding. In: IJCAI

Rastegarpanah B, Gummadi KP, Crovella M (2019) Fighting fire with fire: Using antidote data to improve polarization and fairness of recommender systems. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM '19

Resnick P, Garrett RK, Kriplean T, Munson SA, Stroud NJ (2013) Bursting your (filter) bubble: Strategies for promoting diverse exposure. In: Proceedings of the 2013 Conference on Computer Supported Cooperative Work Companion, Association for Computing Machinery, New York, NY, USA, CSCW '13, pp 95–100, DOI 10.1145/2441955.2441981, URL https://doi.org/10.1145/2441955.2441981

Ribeiro MH, Ottoni R, West R, Almeida VAF, Meira W (2019) Auditing radicalization pathways on YouTube. 1908.08313

Ribeiro MH, Ottoni R, West R, Almeida VAF, Meira W (2020) Auditing radicalization pathways on youtube. In: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, FAT* '20, pp 131–141

Richardson M, Dominowska E, Ragno R (2007) Predicting clicks: estimating the click-through rate for new ads. In: Proceedings of the 16th international conference on World Wide Web, pp 521–530

Singh A, Joachims T (2018) Fairness of exposure in rankings. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, New York, NY, USA, pp 2219–2228, URL https://doi.org/10.1145/3219819.3220088

Singh A, Joachims T (2019) Policy learning for fairness in ranking. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 32, pp 5426–5436, URL https://proceedings.neurips.cc/paper/2019/file/9e82757e9a1c12cb710ad680db11f6f1-Paper.pdf

Sîrbu A, Pedreschi D, Giannotti F, Kertész J (2019) Algorithmic bias amplifies opinion fragmentation and polarization: A bounded confidence model. PloS one 14(3)

Stoica AA, Chaintreau A (2019) Hegemony in social media and the effect of recommendations. In: Companion Proceedings of The 2019 World Wide Web Conference

Stoica AA, Riederer C, Chaintreau A (2018) Algorithmic glass ceiling in social networks. In: Proceedings of the 2018 World Wide Web Conference, ACM Press

Stoica AA, Han JX, Chaintreau A (2020) Seeding network influence in biased networks and the benefits of diversity. In: Proceedings of The Web Conference 2020, ACM

Sunstein CR (2002) The law of group polarization. Journal of Political Philosophy 10(2):175–195

Tabibian B, Gómez V, De A, Schölkopf B, Gomez Rodriguez M (2020) On the design of consequential ranking algorithms. In: Peters J, Sontag D (eds) Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI), PMLR, Proceedings of Machine Learning Research, vol 124, pp 171–180, URL http://proceedings.mlr.press/v124/tabibian20a.html

Taramigkou M, Bothos E, Christidis K, Apostolou D, Mentzas G (2013) Escape the bubble: Guided exploration of music preferences for serendipity and novelty. In: Proceedings of the 7th ACM Conference on Recommender Systems, Association for Computing Machinery, New York, NY, USA, RecSys '13, pp 335–338, DOI 10.1145/2507157.2507223, URL https://doi.org/10.1145/2507157.2507223

Tong H, Prakash BA, Eliassi-Rad T, Faloutsos M, Faloutsos C (2012) Gelling, and melting, large graphs by edge manipulation. In: Proceedings of the 21st ACM international conference on Information and knowledge management, pp 245–254

Vargas S, Baltrunas L, Karatzoglou A, Castells P (2014) Coverage, redundancy and size-awareness in genre diversity for recommender systems. In: Proceedings of the 8th ACM Conference on Recommender systems, pp 209–216

Wąs T, Waniek M, Rahwan T, Michalak T (2020) The manipulability of centrality measures-an axiomatic approach. In: Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems, pp 1467–1475

White S, Smyth P (2003) Algorithms for estimating relative importance in networks. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '03, pp 266–275

Yao S, Huang B (2017) Beyond parity: Fairness objectives for collaborative filtering. arXiv preprint arXiv:170508804

Zehlike M, Bonchi F, Castillo C, Hajian S, Megahed M, Baeza-Yates R (2017) Fa* ir: A fair top-k ranking algorithm. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp 1569–1578

Zeng A, Lü L, Zhou T (2012) Manipulating directed networks for better synchronization. New Journal of Physics 14(8):083006

Zhang z, Gentile AL, Ciravegna F (2013) Recent advances in methods of lexical semantic relatedness – a survey. Natural Language Engineering 19(4):411–479, DOI 10.1017/S1351324912000125

Zhu Z, Hu X, Caverlee J (2018) Fairness-aware tensor-based recommendation. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, ACM, DOI 10.1145/3269206.3271795, URL `https://doi.org/10.1145%2F3269206.3271795`

A Postponed Proofs

Lemma 3 *Let X be a random variable satisfying $0 \leq X \leq t$. We have:*

$$\mathbb{P}(X \leq k) \leq \frac{t - \mathbb{E}[X]}{t-k} \quad .$$

Proof It holds

$$\mathbb{E}[X] = \int_0^k xp(x)dx + \int_k^t xp(x)dx \leq k\left(1 - \mathbb{P}(X \geq k)\right) + t\mathbb{P}(X \geq k) \quad .$$

Thus,

$$\mathbb{P}(X \geq k) \geq \frac{\mathbb{E}[X] - k}{t-k},$$

and

$$\mathbb{P}(X \leq k) = 1 - \mathbb{P}(X \geq k) \leq 1 - \frac{\mathbb{E}[X] - k}{t-k} = \frac{t - \mathbb{E}[X]}{t-k} \quad .$$

□

Lemma 9 *Let $v \in \mathcal{P}_{C_v}(G)$, then, for any $t' \leq t$, it holds $\mathsf{B}^{t'}_G(v) \geq r\frac{t'}{t}$.*

Proof From the hypothesis, using the definition of BR, it holds

$$\begin{aligned} r \leq \mathsf{B}^t_G(v) &= t\mathbb{P}\left(v \overset{\geq t}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right) + \sum_{i=1}^{t-1} i\mathbb{P}\left(v \overset{=i}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right) \\ &\leq t\mathbb{P}\left(v \overset{\geq t'}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right) + \sum_{i=1}^{t'-1} i\mathbb{P}\left(v \overset{=i}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right) \quad . \end{aligned}$$

By rearranging the terms, we obtain

$$\left(r - \sum_{i=1}^{t'-1} i\mathbb{P}\left(v \overset{=i}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right)\right)\frac{1}{t} \leq \mathbb{P}\left(v \overset{\geq t'}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right) \ .$$

Thus, for $t' \leq t$, it holds

$$\begin{aligned}\mathsf{B}_G^{t'}(v) &= t'\mathbb{P}\left(v \overset{\geq t'}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right) + \sum_{i=1}^{t'-1} i\mathbb{P}\left(v \overset{=i}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right) \\ &\geq rj\frac{t'}{t} - \sum_{i=1}^{t'-1} i\left(1 - \frac{t'}{t}\mathbb{P}\left(v \overset{\geq t'}{\underset{G}{\rightsquigarrow}} \bar{C}_v\right)\right) \geq \frac{t'}{t}r \ .\end{aligned}$$

□

Lemma 14 *Let $C \in \{R, B\}$, $v \in C$, $t' \leq t$, and $(e, e') \in \Re_C$ with v the source vertex and w the radicalizing end point. Let $\delta = M(e) - M(e')$. It holds*

$$\delta\mathsf{B}_G^{t'-1}(w) \leq \Gamma(G, v, (e, e'), t') \leq \mathcal{F}_{t'-1}(v)\,\delta\mathsf{B}_G^{t'}(w) \ .$$

Proof Let $G_{e,e'}$ be the graph obtained after swapping the probabilities of e and e'. Consider the probability space of all random walks starting from v in $G_{e,e'}$ and G. We introduce a coupling between these two probability spaces as follows: consider a walk in G, for any step that is not traversing e couple it to an identical step in G. If a step traverses e, then, with probability $1 - \delta$, couple it to the same step in $G_{e,e'}$, else couple it to e' in $G_{e,e'}$.

Let $\mathcal{E}_i$, $1 \leq i \leq t'$ be the event that the coupling diverges at the i-th step, which is equivalent to being at v at step $i - 1$ and the first r.w. taking e, the second taking e'.

When $\mathcal{E}_i$ happens, then the walk in $G_{e,e'}$ has reached the other color by taking e' while the walk in G still needs, in expectation, $\mathsf{B}_G^{t'-i}(w)$ steps to reach the other color. Using the law of total expectation and summing over all $1 \leq i \leq t'$, we can write

$$\Gamma(G, v, (e, e'), t') = \sum_{i=1}^{t'} \mathsf{B}_G^{t'-i}(w)\,\mathbb{P}(\mathcal{E}_i) = \sum_{i=1}^{t'-1} \mathsf{B}_G^{t'-i}(w)\,\mathbb{P}(\mathcal{E}_i) \ .$$

It holds

$$\mathbb{P}(\mathcal{E}_i) = \mathbb{P}\left(v \overset{i-1}{\underset{G}{\rightsquigarrow}} v\right)\delta \ .$$

Since $\mathsf{B}_G^{t'-i}(w) \geq 1$ for every $1 \leq i < t'$, and clearly $\mathbb{P}(\mathcal{E}_1) = \delta$, we obtain the left hand side of the thesis.

The right hand side is concluded as follows. It holds $\mathsf{B}_G^{t'-i}(w) \leq \mathsf{B}_G^{t'-1}(w)$, for every $1 \leq i < t'$, and we can write, using (3),

$$\sum_{i=1}^{t'-1} \mathbb{P}(\mathcal{E}_i) = \sum_{i=1}^{t'-1} \mathbb{P}\left(v \overset{i-1}{\underset{G}{\rightsquigarrow}} v\right)\delta = \sum_{i=0}^{t'-2} \mathbb{P}\left(v \overset{i}{\underset{G}{\rightsquigarrow}} v\right)\delta = \mathcal{F}_{t'-1}(v)\,\delta \ .$$

□