

Random Surfing on Multipartite Graphs

Athanasios N. Nikolakopoulos, Antonia Korba and John D. Garofalakis
Department of Computer Engineering and Informatics, University of Patras
{nikolako, korba, garofala}@ceid.upatras.gr

Abstract—Equipping an imaginary *Random Surfer* of the Web with the ability to teleport, was Page *et al.*’s creative way to justify a mathematical necessity; the *Teleportation Matrix*. Despite being essential for ensuring the ergodicity of the underlying Markov chain, the standard definition of this matrix treats the nodes of the graph in a simplistic and “leveling” way that can prove counterintuitive – especially for applications of the method on graphs of heterogeneous data. In this work, we focus on such graphs and we propose a novel alternative teleportation model that yields a well-defined ranking vector, while being as easy to handle as the traditional teleportation. We explore the theoretical implications of our model, and we reveal a wealth of nice properties that result to direct computational advantages over PageRank. We conduct a set of experiments using real-world datasets and we verify both the useful computational characteristics of our model and its favorable qualitative performance. Our promising findings suggest there remain more to be explored, and maybe much to be gained, by revisiting the teleportation model; a neglected part of PageRank that is typically taken for granted by the majority of applications in the literature.

Keywords—Random Surfer Model; PageRank; k-partite Graphs; Markov Chains; Near Decomposability

I. INTRODUCTION & MOTIVATION

PageRank’s approach to ranking is based on a very intuitive metaphor. In their seminal paper, Page *et al.* [26] imagined of a *random surfer* who, with probability α follows the links of a Web page, and with probability $1 - \alpha$ jumps to a different page uniformly at random. The final ranking score of a page is then defined to be equal to the fraction of time this random surfer spends on it, in the long run. Mathematically, this coincides with the limiting distribution of a homogeneous Markov chain governed by an elegant transition probability matrix, \mathbf{G} , which is known as the Google matrix and is defined by:

$$\mathbf{G} \triangleq \alpha \mathbf{H} + (1 - \alpha) \mathbf{E} \quad (1)$$

Matrix \mathbf{H} is the, usually very sparse, *row-normalized adjacency matrix* of the graph, and the *teleportation matrix* \mathbf{E} – typically defined by $\mathbf{E} \triangleq \frac{1}{n} \mathbf{1} \mathbf{1}^\top$ – is the price PageRank pays for being able to produce a well-defined ranking.

Parameter α – commonly referred to as the *damping factor* – has received much attention since it controls the participation of the actual graph in the final Google matrix. Choosing a small damping factor ignores the link structure of the graph in favor of the artificial teleportation matrix, and inevitably leads to uninformative ranking vectors. On

the other hand, setting the damping factor very close to one, implies a prohibitively large number of iterations till convergence to the PageRank vector, and also makes the computation of the rankings numerically ill-conditioned [19]. Furthermore, from a qualitative point of view, several studies indicate that damping factors close to one result into counterintuitive PageRank vectors where all the ranking gets concentrated into mostly irrelevant nodes [5].

While there have been serious attempts to understand and generalize the mechanism of damping (see [17] for a discussion of the family of *functional rankings* [4] and their connection to the traditional random surfing view), little have been done towards a generalization of the teleportation matrix itself. Mathematically, the introduction of some sort of teleportation is necessary to ensure that the corresponding Markov chain becomes irreducible and aperiodic. The very existence of the standard teleportation matrix, however, gives incentive for direct manipulation of the ranking score through link spamming [9], [24] and is also known to impose limitations to the quality of the ranking vectors (sensitivity to sparsity, biased ranking of newly added nodes etc. [24], [30]). Moreover, from a modeling point of view, the egalitarian treatment of the nodes by the standard teleportation matrix can be restrictive – and sometimes even counterintuitive – especially when PageRank is applied to graphs of special structure, or graphs arising from heterogeneous data. Furthermore, from a purely computational perspective, choosing a teleportation model that is completely “blind” to the spectral characteristics of the underlying graph, could result in unnecessary burden for the extraction of the ranking vector that could otherwise be alleviated through smarter teleportation selection.

In this work, we focus on *multipartite* graphs, which are known to provide good models for heterogeneous data [11], [14], [15], [20] and we modify the traditional teleportation model trying to fulfill three goals: (a) achieve well-defined ranking using an intuitive teleportation scheme in terms of random surfing behavior, (b) ensure that the model can be handled efficiently in order to be applicable in very large datasets, and (c) provide a model which is more “in-tune” with known eigencharacteristics of the graph. To this end, we propose a simple model that provides different teleportation behavior of the random surfer depending on the node he is currently in. In particular, the teleportation model we propose assumes that at each step the random

surfer: with probability η follows the edges of the graph; and with probability $1 - \eta$ *teleports to a node belonging to the same partite set with the node he currently occupies*. The interesting fact is that this simple modification, implies a wealth of new theoretical properties, many of which result in significant advantages over the traditional PageRank.

Summary of Contributions. After we define formally the main components of our model, we prove that it results in an ergodic Markov chain and therefore yields a well-defined ranking vector. We prove that for reasonable values of the teleportation parameter, the corresponding Markov chain is *nearly completely decomposable* into aggregates that can be made stochastic solely through the teleportation model. This observation enables a nice time-scale analysis of the stochastic dynamics of the random surfing process that gives us useful information about the sought-after ranking vector as well as insights on how to compute it. The alternative teleportation matrix, can be handled very efficiently since it can be written in product-form in terms of two extremely sparse and low-rank components. To exploit this fact we propose computing the final vector by an iterative procedure which needs to access our matrix only through sparse matrix-vector (SpMV) products – and thus, allows the final stochastic matrix not to be explicitly computed and stored. We proceed to the rigorous analysis of our algorithm focusing on the most commonly occurring 2-chromatic graphs, for which we can predict analytically the existence of a special eigenvalue controlled by the parameters of our model, as well as some useful eigenvector properties arising from the theory of *lumpable* Markov chains. Our computational experiments indicate that the subtle computational overhead per iteration with respect to PageRank, is more than compensated by the drop of the number of iterations till convergence to the final ranking vector; our method was found to consistently outperform PageRank, in every dataset we experimented on, with the computational time difference being particularly emphatic, when we exploit lumpability. Finally, we test our method’s ranking quality in the Top-N recommendation task and we find that its performance is competitive with state-of-the-art ranking-based collaborative filtering methods.

II. BLOCK TELEPORTATION MODEL

A. Definitions

Let $\mathcal{G} \triangleq \{\mathcal{V}_1, \mathcal{V}_2, \dots, \mathcal{V}_K, \mathcal{E}\}$ be an undirected K -partite graph, i.e. a graph whose vertices can be partitioned into K disjoint independent sets, $\mathcal{V}_1, \mathcal{V}_2, \dots, \mathcal{V}_K$. In our model, each partite set \mathcal{V}_I will be also referred to as a *teleportation block*; the nodes belonging to each block, \mathcal{V}_I , are considered sharing a certain characteristic, chosen for the particular problem at hand (e.g. in a 3-partite graph model of a recommender database, the nodes of \mathcal{V}_1 may represent users, the nodes of \mathcal{V}_2 movies, the nodes of \mathcal{V}_3 genres etc.). Clearly, every edge in \mathcal{E} connects nodes that belong to different teleportation blocks; furthermore, we assume that the set

of blocks cannot be split into two non-empty sets, such that there exist no edge between them.

Normalized Adjacency Matrix \mathbf{H} . As in the traditional PageRank model, this matrix depicts the relations between the nodes as they arise directly from the dataset. In particular, matrix \mathbf{H} is defined to be the row-normalized version of the adjacency matrix of the graph, which we denote \mathbf{A}_G ¹. Concretely, matrix \mathbf{H} is defined as follows:

$$\mathbf{H} \triangleq \text{diag}(\mathbf{A}_G \mathbf{1})^{-1} \mathbf{A}_G, \quad (2)$$

where $\mathbf{1}$ denotes a properly sized column vector of ones.

Block Teleportation Matrix \mathbf{M} . Our teleportation matrix is created to provide different teleportation behavior of the random surfer depending on the origin partite set of the node he is currently in. In particular, the i^{th} row of matrix \mathbf{M} which denotes the teleportation probability vector of node v_i , distributes evenly its mass between the nodes of the origin partite set of v_i . Concretely, the ij^{th} element of matrix \mathbf{M} , is defined as

$$M_{ij} \triangleq \begin{cases} |\mathcal{M}_i|^{-1}, & \text{if } v_j \in \mathcal{M}_i, \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

where \mathcal{M}_i denotes the origin partite set of v_i . Note that while we have assumed for simplicity uniform teleportation within each block, in real scenarios other distributions could be used depending on the purpose of the model.

Ranking Vector. The ranking vector produced by our model is defined as the stationary distribution of the stochastic matrix \mathbf{S} that brings together the normalized adjacency matrix \mathbf{H} , and the block teleportation matrix \mathbf{M} :

$$\mathbf{S} \triangleq \eta \mathbf{H} + (1 - \eta) \mathbf{M} \quad (4)$$

$$\equiv \eta \mathbf{H} + \mu \mathbf{M} \quad (5)$$

with $\eta, \mu > 0$ and $\mu \triangleq 1 - \eta$.

For the rest of this work, we opted for using the greek letter μ to denote the teleportation probability $1 - \eta$, as in (5) in order to make the presentation more tight and clear.

B. Well-Defined Ranking

In the standard PageRank model the teleportation matrix serves the purpose of ensuring that the final stochastic matrix corresponds to an ergodic Markov chain. As a result, the ranking vector produced by PageRank becomes well-defined as it corresponds to its unique stationary distribution. In the following theorem, we show that the same thing holds for our block teleportation model also.

Theorem 1 *The ranking vector produced by our model is a well-defined vector that corresponds to the unique stationary distribution of the Markov chain with transition probability matrix \mathbf{S} .*

¹For simplicity we assume that \mathbf{A}_G does not contain zero rows. In practice isolated nodes of the original graph can be either excluded or “patched” by the addition of artificial edges to other teleportation blocks.

Proof: From the decomposition theorem of Markov chains we know that the state space \mathcal{S} of every chain, can be partitioned uniquely as $\mathcal{S} = \mathcal{T} \cup \mathcal{C}_1 \cup \mathcal{C}_2 \cup \dots$, where \mathcal{T} is the set of transient states, and the \mathcal{C}_i are irreducible closed sets of recurrent states [13]. Thus, it suffices to show that $\mathcal{T} = \emptyset$ and that there exists only one irreducible closed set of recurrent states which is also aperiodic.

Proof of Irreducibility. We choose a recurrent state, i , and we denote the set that contains it, \mathcal{C} . Note that since \mathcal{S} is finite there will always exist at least one recurrent state (and in particular it will be positive recurrent [13]). We will prove that starting from i , one can visit every other state of the Markov chain – and therefore, every state in the chain, belongs to \mathcal{C} . Assume, for the sake of contradiction that there exists a state j outside the set \mathcal{C} . This, by definition, means that there exists no path that starts in state i and ends in state j . Here, we will show that it is always possible to construct such a path.

Let v_i be the node corresponding to state i and v_j the node corresponding to state j . Also let \mathcal{H}_i denote the set that contains the adjacent nodes of v_i . We must have one of the following cases:

$v_j \in \mathcal{M}_i$: States i, j belong to the same teleportation block and thus they are directly connected through matrix \mathbf{M} . In particular, the conditional probability of visiting state j given that we are in i is $\Pr\{i \rightarrow j\} = S_{ij} = \mu M_{ij} > 0$.

$v_j \notin \mathcal{M}_i$: The states belong to different teleportation blocks and we distinguish the following cases:

- 1) $v_j \in \mathcal{H}_i$. Obviously the states i, j are directly connected and the conditional probability of visiting j given that we are in i is $\Pr\{i \rightarrow j\} = \eta H_{ij} > 0$,
- 2) $v_j \notin \mathcal{H}_i$ but $\mathcal{M}_j \cap \mathcal{H}_i \neq \emptyset$. Let j' be a state in $\mathcal{M}_j \cap \mathcal{H}_i$. In this case, there exists the path $i \rightarrow j' \rightarrow j$ with probability $\Pr\{i \rightarrow j' \rightarrow j\} = \eta H_{ij'} \mu M_{j'j} > 0$,
- 3) $v_j \notin \mathcal{H}_i$ and $\mathcal{M}_j \cap \mathcal{H}_i = \emptyset$. In this case j is neither connected to i nor it belongs to a neighboring set of states. However, from the definitions of our model, it holds that there exists a sequence of blocks $\mathcal{M}_i, \mathcal{M}_{t_1}, \mathcal{M}_{t_2}, \dots, \mathcal{M}_{t_m}, \mathcal{M}_j$ with the property for every pair of consecutive blocks in the sequence there exists at least one edge between nodes that belong to these blocks. Let the corresponding sequence of edges be the following:

$$e_{iv_1}, e_{v'_1 v_2}, e_{v'_2 v_3}, \dots, e_{v'_m j}$$

Without loss of generality, we assume $v_1 \neq v'_1, v_2 \neq v'_2, \dots, v_m \neq v'_m$ which presents the worst case scenario. Notice that the existence of the above sequence together with the definitions of matrices \mathbf{H} and \mathbf{M} imply that the corresponding consecutive states communicate, and as a result there exists a path of positive probability between states i and j :

$$i \xrightarrow[\mathbf{M}]{\mathbf{H}} v_1 \xrightarrow[\mathbf{M}]{\mathbf{H}} v'_1 \xrightarrow[\mathbf{M}]{\mathbf{H}} v_2 \xrightarrow[\mathbf{M}]{\mathbf{H}} v'_2 \dots \rightarrow v'_m \xrightarrow[\mathbf{M}]{\mathbf{H}} j$$

In conclusion, there will always be a positive probability path starting from state i and ending in state j . But because state i is recurrent and belongs to the irreducible closed set of states \mathcal{C} , state j belongs to the same irreducible closed set of states too. This contradicts our assumption. Thus, there exist no states outside the irreducible closed set \mathcal{C} , and the irreducibility part of our proof is complete.

Proof of Aperiodicity. It is known that the period of a state i is defined to be the greatest common divisor of the epochs at which a return to the state is possible [13]. When the period of every state is one, the chain is called aperiodic. Thus, taking into account the fact that aperiodicity is a class property (see [13]) it suffices to take any given state and show that it is possible to return to it in consecutive time epochs. In our model this can be seen readily because the diagonal elements of matrix \mathbf{S} are – thanks to the block teleportation matrix \mathbf{M} – all greater than zero; thus, for any state and for every return trajectory to that state of length m , there exists another possible trajectory of length $m + 1$, that follows a self loop at any of the intermediate states. In other words, every state in the Markov chain is aperiodic.

In conclusion, we have proved that the Markov chain that corresponds to matrix \mathbf{S} is irreducible and its states are positive recurrent and aperiodic. This makes the chain ergodic, and our proof is complete. ■

C. Decomposability

In this section we will show that for standard values of μ the final Markov chain arising from our model is *Nearly Completely Decomposable* (NCD) into aggregates that correspond to the connected components of the graph. Before we delve into the details, we discuss briefly NCD Markov chains, following the classic presentation of Courtois [7].

1) *NCD Markov Chains:* Let \mathbf{P} be the $n \times n$ irreducible stochastic matrix, representing the transition matrix of an ergodic Markov chain. Matrix \mathbf{P} may be written as: $\mathbf{P} = \mathbf{P}^* + \varepsilon \mathbf{C}$; where \mathbf{P}^* is a block-diagonal matrix of order n , given by $\mathbf{P}^* = \text{diag}(\mathbf{P}_{11}^*, \mathbf{P}_{22}^*, \dots, \mathbf{P}_{LL}^*)$; and matrices \mathbf{P}_{II}^* are irreducible stochastic matrices of order $n(I)$. Hence, $n = \sum_{I=1}^L n(I)$, and because both \mathbf{P} and \mathbf{P}^* are stochastic, the row-sums of \mathbf{C} are zero. Matrix \mathbf{C} and the non-negative real number ε are chosen such that for all rows it holds

$$\varepsilon \sum_{J \neq I} \sum_{l=1}^{n(J)} C_{m_l l_J} = \sum_{J \neq I} \sum_{l=1}^{n(J)} P_{m_l l_J} \quad (6)$$

and $\varepsilon = \max_{m_I} \sum_{J \neq I} \sum_{l=1}^{n(J)} P_{m_l l_J}$, where $P_{m_l l_J}$ denotes the element at the intersection of the m^{th} row and l^{th} column of the \mathbf{P}_{IJ} submatrix of \mathbf{P} . The parameter ε is referred to as the *maximum degree of coupling* between the subsystems \mathbf{P}_{II}^* . When ε is sufficiently small, \mathbf{P} is called *nearly completely decomposable*.

The analysis of decomposable systems has been pioneered by Simon and Ando who reported on state aggregation in linear models of economic systems [27]. The theory has

been used since, in many complex problems in diverse disciplines ranging from cognitive theory and social sciences, to stochastic modeling and performance evaluation [7], data mining [22], [23] and information retrieval [24].

2) *Decomposability of Matrix S*: In our case, if we define a decomposition of the nodes into aggregates that correspond to the connected components of the underlying graph, it becomes intuitively apparent that when the teleportation parameter is small, the resulting matrix will be nearly completely decomposable subject to the same decomposition, since the maximum degree of coupling between the aggregates will be upper bounded by parameter μ . The following theorem states exactly that.

Theorem 2 *When the value of the teleportation parameter μ is small enough, the Markov chain corresponding to matrix \mathbf{S} is NCD with respect to the partition of the nodes of the initial graph, into different connected components.*

Proof: Let us consider a general graph, in which there are L connected components, $\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_L\}$. Let us assume that the rows and columns of the corresponding matrix \mathbf{S} , are organized such that, nodes within the same connected component occupy consecutive rows and columns in \mathbf{S} . It suffices to show that the maximum degree of coupling ε , with respect to the proposed partition, is strictly less than μ . For simplicity we write matrix \mathbf{S} as $\mathbf{S} = \mathbf{A} + \mathbf{B}$, where \mathbf{A} contains the block diagonal elements of \mathbf{S} (with respect to the underlying decomposition) and \mathbf{B} the off-block diagonal elements. Using the above notation we see that the maximum degree of coupling is equal to

$$\varepsilon = \max_{m_I} \left(\sum_{J \neq I} \sum_{l=1}^{n(J)} S_{m_I l J} \right) = \|\mathbf{B}\|_{\infty}.$$

Notice that in our model ε denotes the maximum probability with which the random surfer leaves a connected component for another. Of course this can only happen through the teleportation matrix, which by definition is followed by the random surfer with probability μ . Thus, the maximum possible value for ε will always be strictly less than μ , as needed (see Figure 1 for a small example). Therefore, for small enough values of μ , the maximum degree of coupling between the aggregates will be small and the corresponding Markov chain will be nearly completely decomposable, which completes the proof. ■

3) *Stochasticity Adjustment of the Aggregates*: In order to get a well formulated NCD model, we must define rigorously the exact way the strictly substochastic diagonal block matrices of \mathbf{S} will be made stochastic. Let us note here that the stochasticity adjustment of the aggregates can be approached in many ways (e.g. see §15.5 in [19]). In our case, since the coupling between the different connected components is based solely on the teleportation model, the stochasticity adjustment of the diagonal blocks of \mathbf{S} can arise naturally and conveniently by the definition of a new block

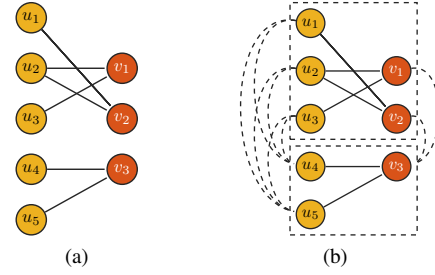


Figure 1. Example graph with two connected components. Dashed lines show the teleportation probabilities that connect the two aggregates. The maximum degree of coupling equals $\frac{2}{3}\mu$.

teleportation model from the scratch within each connected component. Concretely, we define a matrix \mathbf{X} , such that

$$\mathbf{S}^* \triangleq \mathbf{A} + \mathbf{X} = \text{diag}\{\mathbf{S}(\mathcal{G}_1), \mathbf{S}(\mathcal{G}_2), \dots, \mathbf{S}(\mathcal{G}_L)\}$$

where $\mathbf{S}(\mathcal{G}_K)$ is the final stochastic matrix of our model defined for the subgraph \mathcal{G}_K . Then, we can write

$$\mathbf{S} = \mathbf{A} + \mathbf{B} = \mathbf{A} + \mathbf{X} + \mathbf{B} - \mathbf{X} = \mathbf{S}^* + \varepsilon \mathbf{C} \quad (7)$$

which gives us an expression for matrix $\mathbf{C} \triangleq \frac{1}{\varepsilon}(\mathbf{B} - \mathbf{X})$, and concludes the definition of our NCD model.

We are now ready to show how this observation about our model enables a very nice time-scale analysis of the random surfing process's transient behavior towards equilibrium. Our analysis is based on the seminal results of Simon and Ando [27]. For our case in particular, the analysis that follows gives us information about the properties of the sought-after limiting surfing distribution as well as insights on how to compute it.

4) *Stochastic Dynamics*: Let us consider a matrix \mathbf{S} that can be brought to the form of equation (7) and the stochastic processes $\pi_{(t)}^\top = \pi_{(t-1)}^\top \mathbf{S}$ and $\pi_{(t)}^{*\top} = \pi_{(t-1)}^{*\top} \mathbf{S}^*$. The Simon-Ando theorems [27] predict that for sufficiently small ε the dynamic behavior of the stochastic process $\pi_{(t)}^\top$ may be dissociated into four stages, that can be traced into the eigencharacteristics of matrices \mathbf{S}^t and \mathbf{S}^{*t} . The interpretation of these stages in terms of our random surfing process is given below:

Short-term Dynamics. The evolution of \mathbf{S}^t and \mathbf{S}^{*t} with t , for small t , is mostly governed by the smallest eigenvalues of both matrices that are close to each other. Thus, $\pi_{(t)}^\top$ and $\pi_{(t)}^{*\top}$ evolve similarly.

Short-term Equilibrium. The small eigenvalues of \mathbf{S}^t and \mathbf{S}^{*t} have vanished while the L predominant eigenvalues of \mathbf{S}^t remain close to unity. A *similar equilibrium is being reached within each subgraph of \mathbf{S} and \mathbf{S}^** .

Long-term Dynamics. The preponderantly varying part of \mathbf{S}^t , now, involves its eigenvalues $\lambda_2, \dots, \lambda_L$. The whole graph moves towards equilibrium under the influence of the weak interactions among subgraphs, but *the relative values in the short-term equilibria within the subgraphs are approximately preserved*.

Long-term Equilibrium. The Perron eigenvalue of \mathbf{S}^t dominates all others. A global equilibrium is attained in the complete graph. The final ranking vector is reached.

Notice that when our model enjoys the short- and long-term equilibrium characteristics presented above, one can study each subgraph in isolation, knowing that the conditional ranking vector he finds will reveal pairwise relations within the subgraph *that will stand* over the complete graph – even if the absolute final ranking scores are different.

As we proved in Theorem 2, the decomposability of our model is ensured by parameter μ being sufficiently small. But *how small* should it be for the above four stages to be clearly distinguishable? As discussed by Courtois in [7], the aggregates must be able to reach equilibrium while the $\lambda_2^t, \dots, \lambda_L^t$ remain close to unity. In NCD systems the aggregates may be adequately studied using the block diagonal submatrices of \mathbf{S}^* , and since the convergence to equilibrium of the completely decomposable subsystems is controlled by their subdominant eigenvalues, a necessary condition for the above analysis to hold is the modulus of $\lambda_L(\mathbf{S})$ to be larger than the modulus of the maximum subdominant eigenvalue of the block diagonal submatrices of \mathbf{S}^* . Concretely,

$$|\lambda_L(\mathbf{S})| > \max_I |\lambda_2(\mathbf{S}_{\Pi}^*)|. \quad (8)$$

Thankfully, for the usual values of η, μ (including the canonical value for $\eta = 0.85$ proposed by the authors of PageRank [26]) the above condition is typically satisfied – as was the case for every dataset we experimented on. In Section IV we report some relevant experimental results for a number of publicly available datasets.

III. ALGORITHM AND COMPUTATIONAL ANALYSIS

One of the most convenient characteristics of PageRank’s teleportation model, is that it can be managed very efficiently from a computational point of view. Thankfully, the same thing holds for the new teleportation model we propose. In particular, matrix \mathbf{M} is by definition a rank- K matrix (where K is the number of partite sets in the graph) and as a result it can be factorized to a product of K -dimensional matrices. If we rearrange the rows and columns of \mathbf{M} such that the nodes belonging to the same part to be together, it becomes clear that matrix \mathbf{M} can be written as $\mathbf{R}\mathbf{\Delta}\mathbf{R}^\top$ with matrices $\mathbf{R} \in \mathfrak{R}^{n \times K}$ and $\mathbf{\Delta} \in \mathfrak{R}^{K \times K}$ be defined as follows:

$$\mathbf{R} \triangleq \text{diag}(\mathbf{1}_{|\mathcal{V}_1|}, \mathbf{1}_{|\mathcal{V}_2|}, \dots, \mathbf{1}_{|\mathcal{V}_K|}), \quad (9)$$

$$\mathbf{\Delta} \triangleq \text{diag}(1/|\mathcal{V}_1| \ 1/|\mathcal{V}_2| \ \dots \ 1/|\mathcal{V}_K|). \quad (10)$$

Notice that the number of non-zero elements that we need to store for matrix $\mathbf{\Delta}$ is only K , and for matrix \mathbf{R} , only n (all equal to 1). Therefore, given that matrix \mathbf{H} , is typically very sparse, the final stochastic matrix \mathbf{S} can be expressed as a sum of sparse and low-rank components, which makes the use of iterative methods ideal for the extraction of the stationary distribution. The final algorithm is as follows:

Algorithm 1 Block Teleportation Rank

Input: $\mathbf{H}, \mathbf{M} \in \mathfrak{R}^{n \times n}$, scalars $\eta, \mu > 0$ such that $\eta + \mu = 1$, and convergence tolerance ϵ .

Output: π^\top

- 1: Let the initial approximation be $\pi_{(0)}^\top$. Set $k = 0$.
- 2: Compute

$$\begin{aligned} \pi_{(k+1)}^\top &\leftarrow \pi_{(k)}^\top \mathbf{H} \\ \phi^\top &\leftarrow \pi_{(k)}^\top \mathbf{M} \\ \pi_{(k+1)}^\top &\leftarrow \eta \pi_{(k+1)}^\top + \mu \phi^\top \end{aligned}$$

- 3: Normalize $\pi_{(k+1)}^\top$ and compute $r = \|\pi_{(k+1)}^\top - \pi_{(k)}^\top\|_1$.
 - 4: If $r < \epsilon$, quit with $\pi_{(k+1)}^\top$, otherwise $k \leftarrow k + 1$ and go to step 2.
-

The extreme sparsity of the factors of \mathbf{M} , together with the fact that in our algorithm this matrix is accessed only through matrix-vector products, imply that the number of floating point operations per iteration will be dominated by the SpMV product with the normalized adjacency matrix \mathbf{H} . Thus, the number of operations per iteration arising from our algorithm is asymptotically the same with the corresponding number of operations involved in the application of the power method for the computation of the traditional PageRank. The number of iterations till convergence, on the other hand, is controlled by the spectral characteristics of the particular stochastic matrix. Concretely, each iteration of the algorithm effectively updates the estimation of the ranking vector as follows:

$$\pi_{(k+1)}^\top \leftarrow \pi_{(k)}^\top \mathbf{S},$$

where $\pi_{(0)}^\top$ an arbitrary initial stochastic vector. The final ranking vector is defined as the limiting distribution that arises if we let our random surfer jump from node to node following the transition probabilities of matrix \mathbf{S} , forever. That is, $\pi^\top = \lim_{k \rightarrow \infty} \pi_{(k)}^\top$. It can be proved that the rate at which $\pi_{(k)}^\top \rightarrow \pi^\top$ depends on the modulus of the subdominant eigenvalue of matrix \mathbf{S} [28]. In particular, the asymptotic rate of convergence to the limiting distribution, is the rate at which $|\lambda_2(\mathbf{S})|^k \rightarrow 0$. Therefore, the number of floating point operations needed to satisfy a tolerance criterion ϵ , may be obtained approximately from

$$\Omega = \frac{\Theta(\text{nnz}(\mathbf{H})) \log \epsilon}{\log |\lambda_2(\mathbf{S})|}. \quad (11)$$

Thus, the time complexity of our method is controlled by the spectral characteristics of matrix \mathbf{S} . But what can we say about the spectrum of matrix \mathbf{S} ?

From Theorem 1 we know that $|\lambda_2(\mathbf{S})| < 1$, for every $\eta, \mu > 0$ such that $\eta + \mu = 1$, holds. In the general case this is the furthest we can go without delving into specific details of the particular graph under consideration, which are generally not known in advance. Thankfully, for the significant and widely occurring class of K -partite graphs with chromatic number, $\chi(\mathcal{G}) = 2$, we can dig a little deeper. After reviewing relevant literature [11], [14],

[15], [20] and having experimented with a large number of realistic datasets publicly available on the Web [18], we find that in the vast majority of realistic modeling scenarios the corresponding multipartite graphs fall into this category, which includes the widely common bipartite graphs as well as multipartite graphs that have a tree-like block structure.

A. Convergence Analysis

In this section, we explore the spectral characteristics of the class of matrices \mathbf{S} that correspond to 2-chromatic graphs. Although, Algorithm 1 will converge for any block teleportation model yielding a well-defined ranking vector, in our subsequent analysis we assume that the graphs under consideration are connected. Let us note that this assumption does not harm the generality of our results, since, in case there exist more than one connected components, the implied decomposability of the model gives us the theoretical grounds for computing the ranking vectors for each connected component separately – and in parallel – using Algorithm 1. Then, if need be, we can combine the independent solutions using the steady state probabilities of the *coupling matrix* [7], [28] that accounts for the way these aggregates interact with each other. Let us also note that since we are interested in the relevant scores of the correspondent nodes in the final ranking vector, it is not necessary to resort to iterative aggregation/disaggregation methods [28]; a single aggregation step is enough.

The following theorem sheds a little more light to the spectral characteristics of matrix \mathbf{S} arising from graphs fulfilling the above assumptions. In particular, we predict analytically the existence of an eigenvalue, which for realistic values of parameters η, μ , is likely to denote the subdominant eigenvalue of the final stochastic matrix.

Theorem 3 *Assuming \mathcal{G} is a connected graph for which $\chi(\mathcal{G}) = 2$ holds, the spectrum of the stochastic matrix \mathbf{S} is such that $-\eta + \mu \in \lambda(\mathbf{S})$.*

Proof: When \mathcal{G} is connected, matrix \mathbf{H} is irreducible [19]. Furthermore, since the graph is 2-chromatic, it can be reduced graph-theoretically to a bipartite form [6], with the different teleportation blocks split into two partite sets of possibly heterogeneous nodes. The rows and columns of matrices \mathbf{H} and \mathbf{M} are considered arranged accordingly.

Notice that a random walk on this graph results in a periodic Markov chain with period $d = 2$. Therefore, from the Perron-Frobenius theorem [13] we know that

$$\lambda_1(\mathbf{H}) = 1, \quad \text{and} \quad \lambda_2(\mathbf{H}) = e^{2i\pi/d} = e^{i\pi} = -1.$$

The Perron eigenvalue, λ_1 , is associated with the right eigenvector $\mathbf{1}$ whereas the eigenvalue λ_2 , with a right eigenvector which we denote \mathbf{v} . Turning our attention to the eigenvector \mathbf{v} we get the following useful lemma.

Lemma 1 *Vector \mathbf{v}*

$$\mathbf{v} \triangleq \left[\overbrace{1 \ 1 \ 1 \ \dots \ 1}^{\text{\#nodes of 1}^{\text{st}} \text{ partite set}} \ \overbrace{-1 \ -1 \ \dots \ -1}^{\text{\#nodes of 2}^{\text{nd}} \text{ partite set}} \right]$$

is an eigenvector of both row-stochastic matrices \mathbf{H} and \mathbf{M} . In particular, $(-1, \mathbf{v})$ is an eigenpair of matrix \mathbf{H} , and $(1, \mathbf{v})$ is an eigenpair of matrix \mathbf{M} .

Proof: Let n' be the number of nodes and K' the number of blocks in the first partite set. The proof can be done by straightforward calculations. In particular,

$$\mathbf{H}\mathbf{v} = \begin{pmatrix} \mathbf{0} & \mathbf{H}_{12} \\ \mathbf{H}_{21} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{1}_{n'} \\ -\mathbf{1}_{n-n'} \end{pmatrix} = \begin{pmatrix} -\mathbf{1}_{n'} \\ \mathbf{1}_{n-n'} \end{pmatrix} = -\mathbf{v}$$

Similarly, for matrix \mathbf{M} we get

$$\mathbf{M}\mathbf{v} = \begin{pmatrix} \mathbf{M}_{11}\mathbf{1}_{n_1} \\ \vdots \\ \mathbf{M}_{K'K'}\mathbf{1}_{n_{K'}} \\ -\mathbf{M}_{K'+1K'+1}\mathbf{1}_{n_{K'+1}} \\ \vdots \\ -\mathbf{M}_{KK}\mathbf{1}_{n_K} \end{pmatrix} = \begin{pmatrix} \mathbf{1}_{n_1} \\ \vdots \\ \mathbf{1}_{n_{K'}} \\ -\mathbf{1}_{n_{K'+1}} \\ \vdots \\ -\mathbf{1}_{n_K} \end{pmatrix} = \mathbf{v}$$

and the proof is complete \blacksquare

Now let us define a nonsingular matrix, $\mathbf{Q} \triangleq (\mathbf{1} \ \mathbf{v} \ \mathbf{X})$, which contains in its first two columns the eigenvectors that correspond to the principal and the subdominant eigenvalues of matrix \mathbf{H} . Also let

$$\mathbf{Q}^{-1} \triangleq \begin{pmatrix} \mathbf{y}_1^\top \\ \mathbf{y}_2^\top \\ \mathbf{Y}^\top \end{pmatrix} \quad (12)$$

We then have $\mathbf{Q}^{-1}\mathbf{Q} = \mathbf{I}$, which can be usefully expressed

$$\begin{pmatrix} \mathbf{y}_1^\top \mathbf{1} & \mathbf{y}_1^\top \mathbf{v} & \mathbf{y}_1^\top \mathbf{X} \\ \mathbf{y}_2^\top \mathbf{1} & \mathbf{y}_2^\top \mathbf{v} & \mathbf{y}_2^\top \mathbf{X} \\ \mathbf{Y}^\top \mathbf{1} & \mathbf{Y}^\top \mathbf{v} & \mathbf{Y}^\top \mathbf{X} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \mathbf{0} \\ 0 & 1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{pmatrix} \quad (13)$$

Now, if we consider the similarity transformation of our stochastic matrix, $\mathbf{Q}^{-1}\mathbf{S}\mathbf{Q}$, and taking into consideration Lemma 1 and the identities (13), we have

$$\begin{aligned} \mathbf{Q}^{-1}\mathbf{S}\mathbf{Q} &= \mathbf{Q}^{-1}(\eta\mathbf{H} + \mu\mathbf{M})\mathbf{Q} = \dots = \\ &= \begin{pmatrix} 1 & 0 & \eta\mathbf{y}_1^\top\mathbf{H}\mathbf{X} + \mu\mathbf{y}_1^\top\mathbf{M}\mathbf{X} \\ 0 & -\eta + \mu & \eta\mathbf{y}_2^\top\mathbf{H}\mathbf{X} + \mu\mathbf{y}_2^\top\mathbf{M}\mathbf{X} \\ 0 & 0 & \eta\mathbf{Y}^\top\mathbf{H}\mathbf{X} + \mu\mathbf{Y}^\top\mathbf{M}\mathbf{X} \end{pmatrix} \end{aligned} \quad (14)$$

From which we directly get that, $-\eta + \mu$ is an eigenvalue of matrix \mathbf{S} , and our proof is complete. \blacksquare

Notice that for usual values of the parameters $\eta \in [0.8, 0.95]$ and $\mu \in [0.05, 0.2]$, the modulus of the eigenvalue predicted by Theorem 3, is big enough to very likely be the subdominant eigenvalue of matrix \mathbf{S} , i.e. the eigenvalue that controls the number of iterations till convergence to the stationary distribution. Indeed, as can be seen from our proof, the only realistic way for $-\eta + \mu$ to not represent the subdominant eigenvalue of \mathbf{S} is if $\lambda_3(\mathbf{H})$ is very close to unity. Notice that this denotes the mathematical “fingerprint” of an underlying NCD stochastic system, and can be dealt with, by further decomposition of the problematic graph.

Fortunately, this was not the case for any of the connected graphs we experimented on, where we find that the spectrum of matrix \mathbf{H} is such that $-\eta + \mu$ is the subdominant eigenvalue of \mathbf{S} .

Suppose we have a graph with more than one connected components. Notice that when the eigenvalue predicted by Theorem 3 denotes the subdominant eigenvalue of every connected aggregate, and assuming $\mu < 1/2$ holds, condition (8) is always fulfilled. This fact, arises directly from the observation that for $\varepsilon < 1/2$, $|\lambda_L(\mathbf{S})|$ will be bigger than $1 - 2\varepsilon$ (see Courtois [7]). Thus, if we take into account that in our case ε will always be strictly less than μ (see the proof of Theorem 2), we get $|\lambda_L| \geq 1 - 2\varepsilon > 1 - 2\mu = \eta - \mu$.

B. Choosing the Starting Vector

Regardless of the choice of the initial stochastic vector, Theorem 1 certifies that Algorithm 1 will eventually converge to π^\top . In this section, we will see that utilizing carefully the symmetries of the Markov chain that arise from our novel teleportation model, can result to even faster convergence. The key observation is that under the assumptions discussed above, our chain also enjoys the property of *lumpability*; and this, if exploited carefully, allows us to take a glimpse of the properties of the sought-after vector π^\top .

In particular, let us note that since the graph is 2-chromatic, it can be reduced graph-theoretically to a bipartite form [6], with the nodes split into two partite sets containing elements originated from possibly different teleportation blocks (see Figure 2 for a small example). This defines a partition of the nodes, which corresponds to a partition of the states of our Markov chain denoted $\mathcal{A} \triangleq \{\mathcal{A}_1, \mathcal{A}_2\}$.

Theorem 4 *The final Markov chain of our model that corresponds to a 2-chromatic graph, is lumpable with respect to the partition \mathcal{A} defined above.*

Proof: It suffices to show that the probability of moving from a state $i \in \mathcal{A}_1$ to the set \mathcal{A}_2 , i.e. $\Pr\{i \rightarrow \mathcal{A}_2\} = \sum_{j \in \mathcal{A}_2} S_{ij}$, has the same value for every $i \in \mathcal{A}_1$, and that the probability of moving from a state $i \in \mathcal{A}_2$ to the set \mathcal{A}_1 , i.e. $\Pr\{i \rightarrow \mathcal{A}_1\} = \sum_{j \in \mathcal{A}_1} S_{ij}$, has the same value for every $i \in \mathcal{A}_2$ (see Kemeny and Snell [16] for a proof). Let us consider the first part:

$$\Pr\{i \rightarrow \mathcal{A}_2\} = \eta \sum_{j \in \mathcal{A}_2} H_{ij} + \mu \sum_{j \in \mathcal{A}_2} M_{ij} \quad (15)$$

The second term of the RHS is equal to 0, since by the definition of our teleportation model the random surfer can only teleport to nodes belonging to the same block. The first term, on the other hand, equals η , as a result of the stochasticity of \mathbf{H} and the fact that the outgoing edges of our graph connect only vertices belonging to different partite sets. Thus, $\Pr\{i \rightarrow \mathcal{A}_2\} = \sum_{j \in \mathcal{A}_2} S_{ij} = \eta$ for all $i \in \mathcal{A}_1$. Following exactly the same path one can show that $\Pr\{i \rightarrow \mathcal{A}_1\}$, equals to η , also. Thus, the criterion of lumpability is verified, and the proof is complete. ■

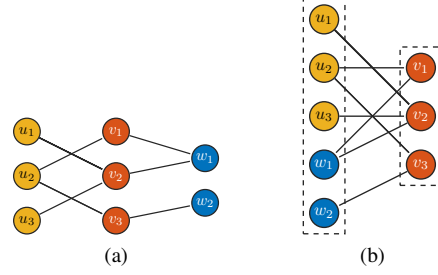


Figure 2. Example graph with 3 teleportation blocks. The final Markov chain defined over this graph is lumpable into 2 lumps highlighted in the second graph.

Besides being interesting from a purely theoretical point of view, the lumpability of our model carries the potential of letting us have a first look to the properties of our final ranking vector.

Theorem 5 *When the final Markov chain is lumpable with respect to partition \mathcal{A} , the left Perron-Frobenius eigenvector of the corresponding stochastic matrix \mathbf{S} has the property, the sum of the elements that correspond to nodes in the first lump of states to be equal to the sum of the elements that correspond to nodes in the second lump of states.*

Proof: For simplicity and without loss of generality, we assume that the rows and columns of the stochastic matrix \mathbf{S} are arranged such that all the nodes in the first lump correspond to the first $|\mathcal{A}_1|$ rows and columns of \mathbf{S} .

Let π^\top be the unique left eigenvector that corresponds to the Perron eigenvalue of \mathbf{S} . By definition it holds $\pi^\top \mathbf{S} = \pi^\top$. Now, taking into account the arrangement of matrix \mathbf{S} , if we multiply from the right with matrix $\begin{pmatrix} \mathbf{1}_{\mathcal{A}_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{\mathcal{A}_2} \end{pmatrix}$ we get

$$\begin{aligned} \pi^\top \mathbf{S} \begin{pmatrix} \mathbf{1}_{\mathcal{A}_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{\mathcal{A}_2} \end{pmatrix} &= \pi^\top \begin{pmatrix} \mathbf{1}_{\mathcal{A}_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{\mathcal{A}_2} \end{pmatrix} \\ \pi^\top \begin{pmatrix} \mu \mathbf{M}_{11} \mathbf{1}_{\mathcal{A}_1} & \eta \mathbf{H}_{12} \mathbf{1}_{\mathcal{A}_2} \\ \eta \mathbf{H}_{21} \mathbf{1}_{\mathcal{A}_1} & \mu \mathbf{M}_{22} \mathbf{1}_{\mathcal{A}_2} \end{pmatrix} &= (\pi_1^\top \mathbf{1}_{\mathcal{A}_1} \quad \pi_2^\top \mathbf{1}_{\mathcal{A}_2}) \\ (\pi_1^\top \quad \pi_2^\top) \begin{pmatrix} \mu \mathbf{1}_{\mathcal{A}_1} & \eta \mathbf{1}_{\mathcal{A}_1} \\ \eta \mathbf{1}_{\mathcal{A}_2} & \mu \mathbf{1}_{\mathcal{A}_2} \end{pmatrix} &= (\pi_1^\top \mathbf{1}_{\mathcal{A}_1} \quad \pi_2^\top \mathbf{1}_{\mathcal{A}_2}) \end{aligned}$$

Then solving the last system taking into consideration that π^\top denotes a probability vector and that $\eta + \mu = 1$, we get

$$\pi_1^\top \mathbf{1}_{\mathcal{A}_1} = \pi_2^\top \mathbf{1}_{\mathcal{A}_2} \quad (16)$$

and our proof is complete. ■

Translating this result to our random surfer model, implies that in the long run the percentage of time the random surfer spends being in the nodes of lump \mathcal{A}_1 equals the percentage of time he spends being in the nodes of lump \mathcal{A}_2 . Notice that one could readily utilize Theorem 5, by choosing an initial vector for Algorithm 1, that satisfies Equation (16), in the hope of decreasing the expected number of iteration till convergence. A simple selection, that makes no other assumptions about the graph is

$$\pi_{(0)}^\top = \left(\frac{1}{2|\mathcal{A}_1|} \mathbf{1}_{\mathcal{A}_1}^\top \quad \frac{1}{2|\mathcal{A}_2|} \mathbf{1}_{\mathcal{A}_2}^\top \right) \quad (17)$$

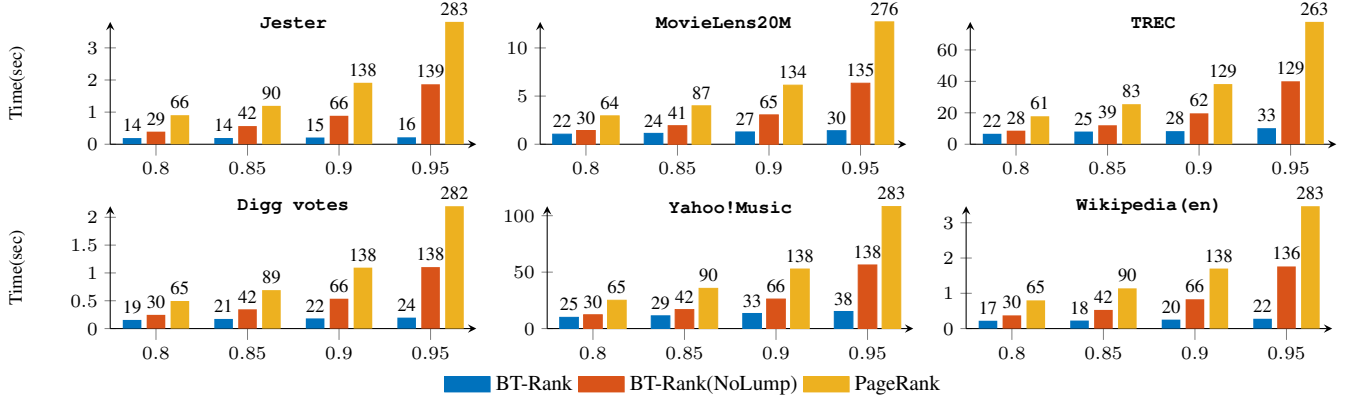


Figure 3. Execution times and number of iterations till convergence for different values of parameter η .

In our computational tests, presented in the following section, we will see that the resulting benefit from following this approach is, in fact, very significant.

IV. EXPERIMENTAL EVALUATION

The experimental evaluation of our method was done using a number of publicly available datasets, originated from many different application areas. Every dataset used throughout this paper can be downloaded from [1] (MovieLens), from [2] (Yahoo!Music) and from the collection [18] (the rest).

A. Experimental Verification of NCD

In order to confirm the properties predicted by the near complete decomposability of our model, we perform the following experiment. We take six disconnected bipartite graphs and we run our algorithm on the complete graph as well as the three largest aggregates (which account for the vast majority of the nodes in each dataset), using the canonical value for $\eta = 0.85$. Then, we take each of the ranking vectors and we calculate its correlation with the corresponding subvectors arising from the complete graph. The metrics used for this comparison are the Kendall's τ and Spearman's ρ correlation coefficients. Their value is 1 for perfect match and -1 for reversed ordering. Table I reports the results.

Table I
RANKING CORRELATION TESTS.

Dataset Name	τ			ρ		
	1st	2nd	3rd	1st	2nd	3rd
Amazon (Rat)	0.99	1	1	1	1	1
DBLP	0.99	1	1	1	1	1
Stack Overflow	0.98	1	1	1	1	1
BookCrossing (Rat)	0.98	1	1	1	1	1
Wikinews (en)	0.99	1	1	1	1	1
Youtube	0.99	1	1	1	1	1

We see that the rankings produced for each aggregate in isolation, imply (near) identical orderings with those arising from the complete graph. The results are in accordance with our theoretical analysis presented in Section II-C, and validate the dissociation of the random surfing process's dynamics into the four stages we predicted. Notice here, that the solution of the subgraphs in isolation (i.e. the solution of the diagonal blocks of \mathbf{S}^*) captures the short-term equilibria of the complete random surfing process which, as can be seen by our results, are indeed preserved as the complete system moves towards its steady state.

B. Computational Experiments

As we discussed in Section III, when we have disconnected graphs, Algorithm 1 can be applied for each aggregate in isolation. Then, the independent solutions can come together based on the interactions between the subgraphs. Of course, this reduces the dimensionality of the problem and lowers the computational burden for computing the final ranking vector; especially if we take into account the fact that the calculation of the rankings of the aggregates can be done in parallel. Here, in order to test the computational implications arising solely from the use of our novel teleportation model, we run our model against the standard PageRank with uniform teleportation on six datasets modeled by connected multipartite graphs. For fair comparisons we followed the exact same implementation approach for both methods based on the standard power method, which is commonly used for the computation of PageRank². We consider two versions of our method, de-

²In the literature there have been proposed more elaborate numerical algorithms for the extraction of the stationary distribution of the final Markov chain [28], or for the computation of PageRank in particular (see for example [19] or more recently [21]). Of course almost every such method can be straightforwardly applied to the computation of our ranking vector as well. However, since our goals in this work were mainly theoretical, we opted for using the most standard computational approach for both methods in order to highlight the intrinsic advantage arising from the spectral characteristics of our alternative teleportation component.

noted **BT-Rank** and **BT-Rank(NoLump)**. The first version exploits the lumpability of the Markov chain using the starting vector (17), whereas the second one is using the standard uniform starting vector. Figure 3 shows the time needed for the computation of the ranking vectors in each case and also the number of iterations (the labels over the bars) needed by the algorithms to converge – up to a difference in the L_1 -norm lower than 10^{-6} – for values of parameter η in the range $[0.80, 0.95]$.

The difference in the number of iterations is very large. Notice that even without exploiting lumpability, our method always converges in less than half iterations than PageRank. And this was true for every dataset and for every value of η tested. The exploitation of lumpability, however, makes the difference in convergence speed even larger (with our method exhibiting, at the same time, very modest increase of iteration steps with η). As expected, this difference in iterations is directly reflected to the actual wall-clock timing difference of the methods, since the factorization of our teleportation model (see Section III), makes the SpMV product with \mathbf{M} in each iteration cost significantly less than the SpMV product with \mathbf{H} , thereby allowing the positive spectral effect of our method to show.

C. Qualitative Evaluation

While our goals in this work were primarily theoretical, for completeness we also perform an experiment in order to assess the quality of our method in producing personalized ranking vectors. For our tests we used the MovieLens1M dataset which we modeled as a 3-partite graph (users-movies-genres). Similar to PageRank, personalization can be achieved through our block teleportation model, exploiting available information about the users. In our case, since we only have their ratings, we use the following very simple definition for matrix \mathbf{M} :

$$\mathbf{M} \triangleq \text{diag}(\mathbf{1}e_i^T, \mathbf{1}\omega_i^T, \mathbf{1}\varpi_i^T)$$

where ω_i is the normalized vector of the users' ratings over the set of movies, and ϖ_i , the normalized vector of his mean ratings per genre. The solution of the corresponding BT-Rank model gives us a personalized ranking with respect to user u_i . In terms of random surfing our model translates to the following: when the surfer teleports from a node of the user-set he goes to the node corresponding to user u_i ; when he teleports from a node of the movie-set he goes to some movie according to distribution ω_i ; and when he teleports from a node of the genre-set he goes to some genre according to distribution ϖ_i . Note that when one has more information about the user (e.g if we know that the user trusts the opinion of a given set of users, or that he has expressed preferences about particular genres etc.), our model has the flexibility to incorporate it to the teleportation model through careful definition of the corresponding block teleportation vectors.

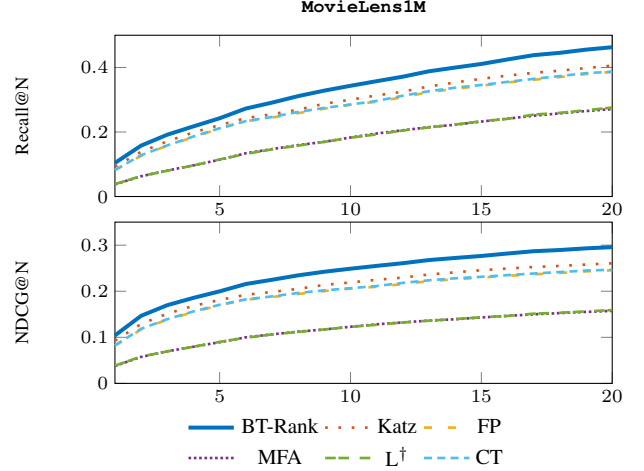


Figure 4. Top-N Quality using the Recall@N and NDCG@N metrics.



Figure 5. Top-N Quality using the MRR metric.

In order to evaluate the quality of our method in suggesting Top-N lists of movies, we adopt the methodology proposed in [8]. In particular, we randomly sample 1.4% of the ratings of the dataset in order to create a probe set \mathcal{P} , and we use each movie v_j , rated with 5-star by user u_i in \mathcal{P} to form the test set \mathcal{T} . Finally, for each movie in \mathcal{T} , we randomly select another 1000 unrated movies of the same user and we order the 1001 movie-lists according to the rankings produced by BT-Rank as well as five other state-of-the-art graph-based methods; namely, the node similarity based methods L^\dagger , and *Katz*; the random walk approaches First Passage Time (*FP*) and Commute Time (*CT*); and the Matrix Forest Algorithm (*MFA*) (for details about the competing methods see [10] and the references therein). For the evaluation of the recommendation quality, we use the standard *Recall@N* and *Normalized Discounted Cumulative Gain (NDCG@N)* metrics focusing on the range $N = [1, \dots, 20]$; and the *Mean Reciprocal Rank (MRR)* metric (due to space constraints for the definitions of these metrics we refer the reader to [23]). We report our results in Figures 4 and 5. As we can see our method achieves good results in every metric, managing to outperform all the other graph-based methods considered.

V. CONCLUSIONS AND FUTURE WORK

The vast majority of applications of PageRank in the literature adopt the traditional rank-one teleportation matrix

that is defined using, either the standard uniform vector proposed by Page *et al.* [26] or, in some cases, an application-specific teleportation vector [3], [29]. Recently, Gleich and Rossi [12] proposed a dynamical system reformulation of PageRank that incorporates a time-evolving teleportation vector. Nikolakopoulos and Garofalakis [24] proposed a generalization of PageRank that includes a more complex low-rank teleportation model designed to highlight the decomposable structure of the Web Graph, and they studied the conditions under which it obviates the need for uniform teleportation [25]. In this work, we revisit the traditional PageRank model and we modify it for random surfing on multipartite graphs. The simple alternative teleportation model we propose accounts for the heterogeneity of the nodes and results in an ergodic Markov chain yielding a well-defined ranking vector. Our novel teleportation matrix is low-rank and can be written as a product of extremely sparse matrices that can be handled very efficiently, making our method readily applicable to very large graphs. We explore analytically our model's implications and we uncover several nice theoretical properties, that affect the final stochastic matrix in a computationally redeemable way. This was verified by our experiments in several real-world datasets, where we found that our method consistently outperforms PageRank.

A very interesting path that remains to be explored involves the definition of a systematic framework for the creation of teleportation models that are more in-sync with the spectral properties of the underlying graphs. Notice, that the traditional rank-one teleportation matrices can not do the trick; however, allowing the teleportation model to be low-rank instead, gives the necessary room for combining easy handling, richer modeling, as well as computational efficiency. In this work we followed this path, focusing on the commonly occurring class of multipartite graphs. Our approach was primarily theoretical. However, we feel that both our analysis, and our promising experimental results suggest there remain more to be discovered, by revisiting and gaining a deeper understanding of the – many times overlooked – teleportation model.

REFERENCES

- [1] GroupLens. <http://grouplens.org/>.
- [2] Yahoo Webscope Program. <https://webscope.sandbox.yahoo.com/>.
- [3] R. Andersen, F. Chung, and K. Lang. Local graph partitioning using pagerank vectors. In *Foundations of Computer Science, 2006. FOCS '06. 47th Annual IEEE Symposium on*, pages 475–486, Oct 2006.
- [4] R. Baeza-Yates, P. Boldi, and C. Castillo. Generic damping functions for propagating importance in link-based ranking. *Internet Math.*, 3(4):445–478, 2006.
- [5] P. Boldi, M. Santini, and S. Vigna. Pagerank: Functional dependencies. *ACM Trans. Inf. Syst.*, 27(4):19:1–19:23, Nov. 2009.
- [6] A. Bondy and U. Murty. *Graph Theory*. Springer, 2008.
- [7] P.-J. Courtois. *Decomposability: Queueing and Computer System Applications*. ACM monograph series. Academic Press, 1977.
- [8] P. Cremonesi, Y. Koren, and R. Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems, RecSys '10*, pages 39–46. ACM, 2010.
- [9] N. Eiron, K. S. McCurley, and J. A. Tomlin. Ranking the web frontier. In *Proceedings of the 13th international conference on World Wide Web, WWW '04*, pages 309–318, New York, NY, USA, 2004. ACM.
- [10] F. Fouss, K. Francoise, L. Yen, A. Pirotte, and M. Saerens. An experimental investigation of kernels on graphs for collaborative recommendation and semisupervised classification. *Neural Netw.*, 31:53–72, July 2012.
- [11] S. Gerster, E. Qeli, C. H. Ahrens, and P. Bhlmann. Protein and gene model inference based on statistical modeling in k-partite graphs. *Proc. Natl. Acad. Sci. USA*, 107(27):12101–12106, 2010.
- [12] D. F. Gleich and R. A. Rossi. A dynamical system for pagerank with time-dependent teleportation. *Internet Math.*, 10(1-2):188–217, 2014.
- [13] G. Grimmett and D. Stirzaker. *Probability and random processes*. Oxford university press, 2001.
- [14] Z. Guan, C. Wang, J. Bu, C. Chen, K. Yang, D. Cai, and X. He. Document recommendation in social tagging services. In *Proceedings of the 19th International Conference on World Wide Web, WWW '10*, pages 391–400, New York, NY, USA, 2010. ACM.
- [15] A. Hotho, R. Jschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In *The Semantic Web: Research and Applications*, vol.4011, pages 411–426. Springer, 2006.
- [16] J. Kemeny and J. Snell. *Finite Markov Chains*. Springer, 1983.
- [17] G. Kollias, E. Gallopoulos, and A. Grama. Surfing the network for ranking by multidamping. *IEEE Trans. Knowl. Data Eng.*, 26(9):2323–2336, 2014.
- [18] J. Kunegis. Konect: The koblenz network collection. In *Proceedings of the 22Nd International Conference on World Wide Web, WWW '13 Companion*, pages 1343–1350.
- [19] A. N. Langville and C. D. Meyer. *Google's PageRank and beyond: The science of search engine rankings*. Princeton Univ. Press, 2011.
- [20] B. Long, X. Wu, Z. M. Zhang, and P. S. Yu. Unsupervised learning on k-partite graphs. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '06*, pages 317–326, New York, NY, USA, 2006. ACM.
- [21] T. Maehara, T. Akiba, Y. Iwata, and K.-i. Kawarabayashi. Computing personalized pagerank quickly by exploiting graph structures. *Proceedings of the VLDB Endowment*, 7(12):1023–1034, 2014.
- [22] C. D. Meyer and C. D. Wessell. Stochastic data clustering. *SIAM J. Matrix Anal. Appl.*, 33(4):1214–1236, 2012.
- [23] A. Nikolakopoulos and J. Garofalakis. NCDREC: A decomposability inspired framework for top-n recommendation. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on*, pages 183–190, Aug 2014.
- [24] A. N. Nikolakopoulos and J. D. Garofalakis. NCDawareRank: a novel ranking method that exploits the decomposable structure of the web. In *Proceedings of the sixth ACM international conference on Web search and data mining, WSDM '13*, pages 143–152. ACM, 2013.
- [25] A. N. Nikolakopoulos and J. D. Garofalakis. Random surfing without teleportation. In *Algorithms, Probability, Networks, and Games*, volume 9295, pages 344–357. Springer, 2015.
- [26] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999.
- [27] H. A. Simon and A. Ando. Aggregation of variables in dynamic systems. *Econometrica*, 29(2):pp. 111–138, 1961.
- [28] W. Stewart. *Introduction to the numerical solution of Markov chains*. Princeton University Press, 1994.
- [29] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast random walk with restart and its applications. In *Proceedings of the Sixth International Conference on Data Mining, ICDM '06*, pages 613–622, Washington, DC, USA, 2006. IEEE Computer Society.
- [30] G.-R. Xue, Q. Yang, H.-J. Zeng, Y. Yu, and Z. Chen. Exploiting the hierarchical structure for link analysis. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '05*, pages 186–193, NY, USA, 2005. ACM.