POLITECNICO DI TORINO
Repository ISTITUZIONALE

Ranking a set of objects: a graph based least-square approach

Original
Ranking a set of objects: a graph based least-square approach / Christoforou, E., Nordio, A., Tarable, A., Leonardi, E.. - In: IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING. - ISSN 2327-4697. - ELETTRONICO. - 8:1(2021), pp. 803-813. [10.1109/TNSE.2021.3053423]

Availability:
This version is available at: 11583/2862351 since: 2021-03-19T09:20:23Z

Publisher:
IEEE

Published
DOI:10.1109/TNSE.2021.3053423

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

29 November 2026

Ranking a set of objects: a graph based least-square approach

Evgenia Christoforou, *Member, IEEE*, Alessandro Nordio, *Member, IEEE*,
Alberto Tarable, *Member, IEEE*, Emilio Leonardi, *Senior Member, IEEE*

Abstract—We consider the problem of ranking N objects starting from a set of noisy pairwise comparisons provided by a crowd of equal workers. We assume that objects are endowed with intrinsic qualities and that the probability with which an object is preferred to another depends only on the difference between the qualities of the two competitors. We propose a class of non-adaptive ranking algorithms that rely on a least-squares optimization criterion for the estimation of qualities. Such algorithms are shown to be asymptotically optimal (i.e., they require $O(\frac{N}{\epsilon^2}\log\frac{N}{\delta})$ comparisons to be (ϵ,δ)-PAC). Numerical results show that our schemes are very efficient also in many non-asymptotic scenarios exhibiting a performance similar to the maximum-likelihood algorithm. Moreover, we show how they can be extended to adaptive schemes and test them on real-world datasets.

Index Terms—Ranking algorithms, noisy evaluation, applied graph theory, least-square estimation

1 Introduction

Ranking algorithms have many applications. For example they are used for ranking pages, user preferences against advertisements on the web, hotels, restaurants, or online games [1], [2]. In general a ranking algorithm infers an estimated order relation among objects starting from a set of evaluations or comparisons. Sometimes, such evaluations are performed by human "workers" in the framework of crowdsourcing applications. However, since the behavior of humans cannot be deterministically predicted, it is usually described through the adoption of a probabilistic model. Then, the challenge in designing algorithms is the ability to infer reliable estimates of the ranking, starting from "noisy" evaluations of the objects. Often ranking algorithms resort to pairwise comparisons of objects. In this work, we focus on such a class of ranking algorithms. Several stochastic models have been proposed in the literature to represent the outcome of comparisons and offer just examples of possible workers' behavior (see e.g. the Bradley-Terry-Luce and the Thurstone models [3], [4], [5], [6]). Most of them are based on the idea that objects to be compared have an intrinsic quality and that the probability that object i is preferred to object j depends on their qualities q_i and q_j. Up to now, however, we are not aware of any work supporting experimental evidence for their applicability to general contexts. In this scenario, we devise a class of efficient algorithms, which reconstruct object qualities from pairwise difference through a least-square (LS) approach. To do so, we establish a parallelism between the estimation process and the average cumulative reward of random walks on a weighted graph.

- *E. Cristoforou is with Research Centre on Interactive Media, Smart Systems and Emerging Technologies (RISE), Nicosia, Cyprus, email: evgenia.christoforou@gmail.com.*
- *A. Nordio and A. Tarable are with IEIIT-CNR (Institute of Electronics, Telecommunications and Information Engineering of the National Research Council of Italy), Italy, email: firstname.lastname@ieiit.cnr.it.*
- *E. Leonardi is with DET, Politecnico di Torino, Torino, Italy, email: emilio.leonardi@polito.it, and is associate researchers with IEIIT-CNR.*

1.1 System model

Let $\mathcal{Q} \subset \mathbb{R}$ be a compact set. We assume that N objects are available for ranking: object i is endowed with an intrinsic quality, $q_i \in \mathcal{Q}$, which is unknown to the system. Qualities induce a true ranking r among objects, in which $r(i) \prec r(j)$ iff $q_i > q_j$[1]. A ranking algorithm resorts to a set of observations (or answers) provided by workers, which compare pairs of objects and return the identity of the object they prefer. The comparison procedure implicitly contains some randomness reflecting the workers' behavior. Thus, in general, workers' answers can be modeled as a collection of binary random variables, whose distribution depends on the qualities of the objects to be evaluated.

Due to this randomness in the evaluation process, the inferred ranking for object i, $\widehat{r}(i)$, does not always coincide with the true ranking $r(i)$. The reliability of $\widehat{r}(i)$ depends on how the evaluation process is organized. In particular, it depends on (i) the workers' behavior, (ii) the choice of the set of object pairs to be compared, (iii) the number of workers assigned to each pair of objects, and (iv) the processing algorithm used to infer the ranking from workers' answers.

We assume that all workers behave similarly and that their behavioral model is known by the system. They provide independent answers. In particular, a worker comparing objects i and j, will express a preference for object i against j with probability:

$$p_{i,j} = 1 - p_{j,i} = F(q_i - q_j) \tag{1}$$

where the function $F(\cdot)$ is differentiable and strictly increasing in its argument (and therefore invertible) and such that $F(0) = \frac{1}{2}$. Moreover, we assume that $F'(q)$ is bounded away from zero for $q \in \bar{\mathcal{Q}}$ where $\bar{\mathcal{Q}} = \{q_i - q_j | q_i, q_j \in \mathcal{Q}\}$.

1. The symbol $\prec$ is a precedence operator. If $r(i) \prec r(j)$ then object i "precedes" or "is preferable to" object j.

When the object pair (i,j) is compared, the worker's answer is modeled as a binary random variable, $w_{i,j} \in \{0,1\}$, whose outcomes have probability

$$\mathbb{P}(w_{i,j}=1)=p_{i,j}; \qquad \mathbb{P}(w_{i,j}=0)=1-p_{i,j}\,. \tag{2}$$

The model in (1) is pretty general. For example, it encompasses:

- the Thurstone model [5], where the preferred object (in a pair) is chosen in accordance with the qualities *as perceived* by the worker and defined as

  $$\widetilde{q}_i = q_i + n_i, \qquad \widetilde{q}_j = q_j + n_j$$

  respectively, where n_i and n_j are zero-mean random variables that represent noise terms. In this case $F(\cdot)$ is the cumulative distribution function of the zero-mean random variable $\eta_{i,j}=n_i-n_j$, i.e.,

  $$p_{i,j}=\mathbb{P}\left(\eta_{i,j}<q_i-q_j\right) \tag{3}$$

- the Bradley-Terry-Luce (BTL) model [3], [4], where

  $$p_{i,j}=\frac{\mathrm{e}^{q_i-q_j}}{1+\mathrm{e}^{q_i-q_j}}\,. \tag{4}$$

Let $\mathcal{V}=\{1,\dots,N\}$ be the set of objects. We observe that an arbitrary choice of a set of object pairs to be compared, denoted by $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$, automatically induces an undirected graph $\mathcal{G}$, whose vertex and edge sets are, respectively, $\mathcal{V}$ and $\mathcal{E}$. Clearly, it is possible to infer a total ranking of the N objects only if the graph $\mathcal{G}$ is connected, hence in the following we will always assume that $\mathcal{G}$ is connected.

Each object pair $(i,j)\in\mathcal{E}$ is assigned to a number of workers W. In general, an increase of W leads to a more reliable estimate of the ranking. On the other hand, the overall complexity, C, of the ranking algorithm is proportional to the total number of workers employed in the process, i.e.,

$$C=|\mathcal{E}|W.$$

Then, an efficient ranking algorithm must find a good trade-off between the complexity C and the reliability of the inferred ranking, i.e., by returning an almost correct ranking of objects with a minimal number of pair comparisons.

About the reliability of the inferred ranking we say that an estimated ranking is ϵ-quality approximately correct (or, is an ϵ-quality ranking) if $\widehat{r}(i)\prec\widehat{r}(j)$ whenever $q_i\geq q_j+\epsilon$. Moreover a ranking algorithm is (ϵ,δ)-PAC [7], [8], [9] if it returns an ϵ-quality ranking with a probability larger than $1-\delta$.[2]

1.2 Paper contribution and related work

This paper contributes to a better understanding of the fundamental limits of ranking algorithms based on noisy pairwise comparisons. Our main results complement and extend previous findings about minimal complexity of ranking algorithms under different non-parametric preference models, recently derived in [8], [9]. As shown in [8], [9], the efficiency of ranking algorithms is crucially determined by the structure of the underlying preference model.

On the one hand, under a non-parametric preference model satisfying both Strong Stochastic Transitivity (SST) and Stochastic Triangle Inequality (STI) properties,[3] a provably asymptotically-optimal[4] *adaptive* algorithm has been proposed, under the restriction that $\delta>\frac{1}{N}$. In particular, the algorithm proposed in [9] is (ϵ,δ)-PAC provided that $O(\frac{N}{\epsilon^2}\log\frac{N}{\delta})$ comparisons are dynamically allocated on the basis of previous outcomes. On the other hand, in [8], [9], it is shown that $\Omega(N^2)$ comparisons are needed to obtain a reliable ranking as soon as either STI or SST are relaxed.

When considering parametric models, estimating a ranking is essentially related to estimating the underlying qualities. The authors in [10], [11] provide a characterization of the expected 2-norm distance between estimated and true qualities (later on referred to as mean square error (MSE)), in connection with the properties of a fixed graph $\mathcal{G}$. In particular [10], under the assumption that $F(\cdot)$ is log-concave, provides universal (i.e., applicable to optimal algorithms, such as the maximum-likelihood (ML) algorithm) order-optimal upper and lower bounds for the MSE, relating it to the spectral gap of a certain scaled version of the Laplacian of $\mathcal{G}$. The very recent paper [11], for the BTL model only, introduces an LS algorithm and provides upper and lower bounds for a variant of the MSE and the relative tail probabilities achievable by such algorithm, characterizing it in terms of the graph resistance.

Interesting works are also [12], [13], [14], [15], [16], [17], [18]. In [16], [17] an LS approach for ranking is first introduced, but no theoretical guarantees are given. In particular, [17] proposes Sync-Rank, a semi-definite programming algorithm based on the angular synchronization framework. In [12], [13], instead, an iterative algorithm that emulates a weighted random walk of graph $\mathcal{G}$ is proposed and its performance analyzed under the BTL model. The previous papers provide bounds on the achievable MSE (i.e. the 2-norm distance between true and estimated qualities) and the corresponding tail probabilities (i.e. the probability that the distance between true and estimated qualities exceeds a given threshold). A direct comparison between the performance of algorithms proposed in [11], [14], [15] is reported in [11] where the LS approach is shown to be, in general, asymptotically more efficient. Under the BTL model, [12], [13] propose and analyze algorithms able to identify the top-k objects. Finally, [18] describes a ranking algorithm based on the singular value decomposition approach by assuming that workers return unquantized noisy estimates of quality differences.

Regarding online ranking algorithms, in [7], for the BTL model, an online algorithm inspired by a finite-budget version of quick sort is described, able to obtain an (ϵ,δ)-PAC ranking with $O(\frac{N}{\epsilon^2}\log N\log\frac{N}{\delta})$ comparisons. In [19], it is shown that, for online ranking algorithms, parametric models help to reduce the complexity only by logarithmic factors, in order sense.

2. Our definition of (ϵ,δ)-PAC algorithm slightly differs from the original given in [7], [8], [9] since it applies to object qualities. However, it can be easily shown to be asymptotically equivalent to the original.

3. A preference model is said to be non-parametric if pairwise preference probabilities are not necessarily induced by object qualities. It satisfies the SST if $p_{i,k}\geq\max(p_{i,j},p_{j,k})$ whenever $r(i)\prec r(j)\prec r(k)$. It satisfies STI if $p_{i,k}+\frac{1}{2}<p_{i,j}+p_{j,k}$ whenever $r(i)\prec r(j)\prec r(k)$.

4. A (ϵ,δ)-PAC ranking algorithm is asymptotically-optimal if its complexity is $O(\frac{N}{\epsilon^2}\log\frac{N}{\delta})$.

In this work, unlike [11], we introduce a rather general parametric preference model according to which preference probabilities are determined by an arbitrary smooth monotonic function of object-quality differences. In this scenario, we show that order-optimal *non-adaptive algorithms* can be defined without the necessity of introducing any restriction to parameter δ. In particular, differently from [10], [11], we work with the PAC framework and show that our algorithms are (ϵ, δ)-PAC, provided that $O(\frac{N}{\epsilon^2}\log\frac{N}{\delta})$ comparisons are blindly allocated in a single round. Observe that our preference model does not necessarily satisfy STI, while it satisfies SST. Our ranking procedure is based on the reconstruction of object qualities from pairwise quality differences, by adopting a LS approach akin to the one in [11]. Notice however that the analysis in [11] only applies to the case where $\Omega(N\log^2\frac{N}{\delta})$ total comparisons are performed, and therefore it is not useful to answer questions about the existence of order-optimal algorithms. Our analysis establishes a parallelism between the quality estimation process and the cumulative reward of random walks on graphs. As an original contribution, we also introduce a *weighted* LS (WLS) algorithm with performance very close to the more complex ML algorithm. Finally, by simulation, we show that the performance of our algorithms is extremely good also in non-asymptotic scenarios.

The paper is organized as follows: in Section 2 we introduce a ranking algorithm based on the Maximum Likelihood (ML) approach, which is used as a performance reference. In Section 3 we describe our proposed LS estimation algorithm, whose asymptotic analysis is investigated in Section 4. The LS estimation algorithm is then tested in Sections 5 and 6 against synthetic and real-world datasets, respectively. Finally, in Section 7 we draw our conclusions.

1.3 Notation

Boldface uppercase and lowercase letters denote matrices and vectors, respectively. $\mathbf{I}$ is the identity matrix. The transpose of the matrix $\mathbf{A}$ is denoted by $\mathbf{A}^\mathsf{T}$, while $[\mathbf{A}]_{i,j}$ indicates its (i,j) entry. For the sake of notation compactness, we use the notation $\mathbf{A}=\{a_{i,j}\}$ to define a matrix $\mathbf{A}$ whose elements are $a_{i,j}$. Calligraphic letters denote sets or graphs. Finally, the symbol $\odot$ represents the Hadamard product.

2 ML QUALITY ESTIMATION

Consider a connected graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ with $|\mathcal{V}|=N$ vertices where each pair of objects $(i,j)\in\mathcal{E}$ is evaluated W times by independent workers [5]. Without loss of generality we assume that the indices of the objects connected by the generic edge $(i,j)\in\mathcal{E}$ are such that $i>j$. Moreover, we assume that the m-th worker evaluating the pair of objects (i,j) outputs the binary random variable $w_{i,j,m}$ whose distribution is given by (2).

In our proposed ML approach, the estimate of the ranking can be obtained by sorting the quality estimates $\widehat{\mathbf{q}}=[\widehat{q}_1,\ldots,\widehat{q}_N]^\mathsf{T}$ which are obtained as follows:

$$\widehat{\mathbf{q}}=\arg\max_{\mathbf{q}}\log\mathbb{P}\left(\{w_{i,j,m},(i,j)\in\mathcal{E},m=1,\ldots,W\}|\mathbf{q}\right). \tag{5}$$

5. The generalization to a number of evaluations that depends on the specific edge is straightforward.

When workers are independent of each other and behave similarly, the random variables $w_{i,j,m}$ can be modeled as independent and identically distributed. Therefore, the conditional probability in (5) factorizes as

$$\mathbb{P}\left(\{w_{i,j,m},(i,j)\in\mathcal{E},m=1,\ldots,W\}|\mathbf{q}\right)=\prod_{(i,j)\in\mathcal{E}}\prod_{m=1}^{W}\mathbb{P}\left(w_{i,j,m}|\mathbf{q}\right).$$

By using (2) we write

$$\mathbb{P}\left(w_{i,j,m}|\mathbf{q}\right)=p_{i,j}^{w_{i,j,m}}\left(1-p_{i,j}\right)^{1-w_{i,j,m}}.$$

where we recall that $p_{i,j}=F(q_i-q_j)$. By substituting the above result in (5), the ML estimate of the qualities $\mathbf{q}$ can be rewritten as

$$\begin{aligned}\widehat{\mathbf{q}}&=\arg\max_{\mathbf{q}}\log\mathbb{P}\left(\{w_{i,j,m},(i,j)\in\mathcal{E},m=1,\ldots,W\}|\mathbf{q}\right)\\&=\arg\max_{\mathbf{q}}\sum_{(i,j)\in\mathcal{E}}\sum_{m=1}^{W}\log p_{i,j}^{w_{i,j,m}}+\log\left[(1-p_{i,j})^{(1-w_{i,j,m})}\right]\\&=\arg\max_{\mathbf{q}}\Psi(\mathbf{q})\end{aligned} \tag{6}$$

where

$$\Psi(\mathbf{q})=\sum_{(i,j)\in\mathcal{E}}s_{i,j}\log p_{i,j}+(1-s_{i,j})\log(1-p_{i,j}).$$

and $s_{i,j}=\frac{1}{W}\sum_{m=1}^{W}w_{i,j,m}$. The function $\Psi(\mathbf{q})$ has a finite global maximum. Indeed, since $p_{i,j}\in[0,1]$, and $s_{i,j}\in[0,1]$, it is straightforward to show that $\Psi(\mathbf{q})\leq 0$. However, in general, $\Psi(\mathbf{q})$ is a non-convex function of $\mathbf{q}$ and its maximization is non trivial. Nevertheless, a local maximum can be found by using standard techniques such as, for example, the Newton-Raphson method, which works iteratively and requires the function $F(\cdot)$ to be twice differentiable.

Let $\widehat{\mathbf{q}}_t$ be the estimate of $\mathbf{q}$ at iteration $t=1,2,\ldots$ Then the estimate of $\mathbf{q}$ at iteration $t+1$ can be updated as follows:

$$\widehat{\mathbf{q}}_{t+1}=\widehat{\mathbf{q}}_t-[\mathbf{S}(\widehat{\mathbf{q}}_t)]^{-1}\nabla\Psi(\mathbf{q})_t$$

where $\nabla\Psi(\mathbf{q})$ and $\mathbf{S}(\mathbf{q})$ are, respectively, the gradient and the Hessian matrix of $\Psi(\mathbf{q})$. Specifically, $[\nabla\Psi(\mathbf{q})]_h\triangleq\frac{\partial\Psi(\mathbf{q})}{\partial q_h}$ and $[\mathbf{S}(\mathbf{q})]_{h,k}\triangleq\frac{\partial^2\Psi(\mathbf{q})}{\partial q_h\partial q_k}$. In order to compute $\nabla\Psi(\mathbf{q})$ and $\mathbf{S}(\mathbf{q})$ consider a generic node $h\in\mathcal{V}$ and the set $\mathcal{E}_h\subseteq\mathcal{E}$ of edges connecting node h to its neighbors. Then, the function $\Psi(\mathbf{q})$ can be rewritten as

$$\Psi(\mathbf{q})=c+\sum_{(i,j)\in\mathcal{E}_h}s_{i,j}\log p_{i,j}+(1-s_{i,j})\log(1-p_{i,j}) \tag{7}$$

where the term c does not depend on q_h. Since $p_{i,j}=F(q_i-q_j)$, we can write the partial derivatives of $p_{i,j}$ as follows:

$$\frac{\partial p_{i,j}}{\partial q_i}\triangleq p'_{i,j};\quad \frac{\partial p_{i,j}}{\partial q_j}\triangleq -p'_{i,j}$$

and, similarly

$$\frac{\partial^2 p_{i,j}}{\partial q_i^2}=\frac{\partial^2 p_{i,j}}{\partial q_j^2}\triangleq p''_{i,j};\qquad \frac{\partial^2 p_{i,j}}{\partial q_j\partial q_j}=-p''_{i,j}.$$

It immediately follows that

$$[\nabla\Psi(\mathbf{q})]_h = \sum_{(h,j)\in\mathcal{E}_h} p'_{h,j}\frac{s_{h,j}-p_{h,j}}{p_{h,j}(1-p_{h,j})} - \sum_{(i,h)\in\mathcal{E}_h} p'_{i,h}\frac{s_{i,h}-p_{i,h}}{p_{i,h}(1-p_{i,h})}$$

and

$$\begin{aligned}[\mathbf{S}(\mathbf{q})]_{h,h} = &\sum_{(h,j)\in\mathcal{E}_h}\left[p''_{h,j}\frac{s_{h,j}-p_{h,j}}{p_{h,j}(1-p_{h,j})}\right.\\ &\left.-(p'_{h,j})^2\,\frac{p_{h,j}^2+s_{h,j}(1-2p_{h,j})}{p_{h,j}^2(1-p_{h,j})^2}\right]\\ &+\sum_{(i,h)\in\mathcal{E}_h}\left[p''_{i,h}\frac{s_{i,h}-p_{i,h}}{p_{i,h}(1-p_{i,h})}\right.\\ &\left.-(p'_{i,h})^2\,\frac{p_{i,h}^2+s_{i,h}(1-2p_{i,h})}{p_{i,h}^2(1-p_{i,h})^2}\right].\end{aligned}$$

Moreover, for $h\neq k$

$$[\mathbf{S}(\mathbf{q})]_{h,k}=\begin{cases}0, & \text{if } (h,k)\notin\mathcal{E}_h \text{ or } (k,h)\notin\mathcal{E}_h\\ (p'_{h,k})^2\frac{p_{h,k}^2+s_{h,k}(1-2p_{h,k})}{p_{h,k}^2(1-p_{h,k})^2}-p''_{h,k}\frac{s_{h,k}-p_{h,k}}{p_{h,k}(1-p_{h,k})}, & \text{if } (h,k)\in\mathcal{E}_h \text{ or } (k,h)\in\mathcal{E}_h\end{cases}$$

The above equations can be specialized for both the Thurstone model as well as for the BTL model, by using the expressions for $p_{i,j}$ provided, respectively, in (3) and (4).

3 Least-squares quality estimation

In this section, we propose a simpler linear estimation algorithm, based on a least-square criterion, that can be applied on the connected graph $\mathcal{G}(\mathcal{V},\mathcal{E})$. Let the distance between objects i and j be

$$d_{i,j}=q_i-q_j$$

and let $\mathcal{W}_{i,j}$ be the set of binary answers, of cardinality W, provided by the workers comparing the pair (i,j). Also, let $K_{i,j}$ be the number of times object i is preferred to object j. Then, by construction, $K_{i,j}$ follows the binomial distribution $K_{i,j}\sim\mathrm{Bin}(W,p_{i,j})$, where $p_{i,j}=F(d_{i,j})$. Out of the evaluation results, an estimate $\widehat{d}_{i,j}$ of $d_{i,j}$ is formed as

$$\widehat{d}_{i,j}=F^{-1}\left(\widehat{p}_{i,j}\right)=F^{-1}(y_{i,j}+p_{i,j}), \tag{8}$$

where $\widehat{p}_{i,j}=K_{i,j}/W$ is the estimate of $p_{i,j}$, and $y_{i,j}=\widehat{p}_{i,j}-p_{i,j}$ represents the estimation error on the probability $p_{i,j}$. Note that $y_{i,j}$ has zero mean and variance $\mathbb{E}[y_{i,j}^2]=\frac{p_{i,j}(1-p_{i,j})}{W}$. As a consequence, $\widehat{d}_{i,j}=d_{i,j}+z_{i,j}$, where $z_{i,j}$ represents the error on the estimate of $d_{i,j}$ induced by the presence of $y_{i,j}$. From the set of noisy estimates $\{\widehat{d}_{i,j},(i,j)\in\mathcal{E}\}$, the estimate $\widehat{\mathbf{q}}=[\widehat{q}_1,\dots,\widehat{q}_N]^\mathsf{T}$ of $\mathbf{q}=[q_1,\dots,q_N]^\mathsf{T}$ can be obtained by solving the following LS optimization problem

$$\widehat{\mathbf{q}}=\arg\min_{\mathbf{x}}\sum_{(i,j)\in\mathcal{E}}\omega_{i,j}\left(x_i-x_j-\widehat{d}_{i,j}\right)^2 \tag{9}$$

where $\omega_{i,j}$ are arbitrary positive weights, whose setting is discussed in Section 3.1. The solution of (9) satisfies the following linear equations:

$$\widehat{q}_i=\sum_{j\in\mathcal{N}_i}\omega_{i,j}\frac{\widehat{q}_j+\widehat{d}_{i,j}}{\rho_i},\quad i=1,\dots,N \tag{10}$$

where $\mathcal{N}_i$ represents the neighborhood of node i (i.e., the set of nodes connected to i in $\mathcal{G}$), and ρ_i is its generalized degree, i.e., $\rho_i=\sum_{j\in\mathcal{N}_i}\omega_{i,j}$. We can compactly express the previous linear system in terms of the $N\times N$ matrix $\widetilde{\mathbf{H}}$ associated to the graph $\mathcal{G}$, whose elements are defined as

$$[\widetilde{\mathbf{H}}]_{i,j}=\begin{cases}\omega_{i,j}/\rho_i, & (i,j)\in\mathcal{E},\\ 0, & \text{otherwise.}\end{cases}$$

Let $\mathbf{I}$ be the identity matrix, $\widetilde{\mathbf{M}}=\mathbf{I}-\widetilde{\mathbf{H}}$, and $\mathbf{Z}=\{z_{i,j}\}$. Moreover let $\mathbf{D}=\{d_{i,j}\}$ and $\widehat{\mathbf{D}}=\{\widehat{d}_{i,j}\}$ be, respectively, the antisymmetric matrices of the true and estimated quality differences[6]. Thus, from (10) we can write:

$$\widetilde{\mathbf{M}}\widehat{\mathbf{q}}=(\widetilde{\mathbf{H}}\odot\widehat{\mathbf{D}})\mathbf{1}=(\widetilde{\mathbf{H}}\odot\mathbf{D})\mathbf{1}+(\widetilde{\mathbf{H}}\odot\mathbf{Z})\mathbf{1}\,. \tag{11}$$

where $\mathbf{1}=[1,\dots,1]^\mathsf{T}$ is a column vector of size N. We observe that, since graph $\mathcal{G}$ is connected, by construction $\mathrm{rank}(\widetilde{\mathbf{M}})=N-1$. Therefore $\widetilde{\mathbf{M}}$ is singular. Indeed $\widetilde{\mathbf{M}}\mathbf{1}=\mathbf{0}$, as it can be easily checked. This implies that the associated linear operator on $\mathbb{R}^N$ is not injective and that, given a solution $\widehat{\mathbf{q}}'$ of (10), also $\widehat{\mathbf{q}}''=\widehat{\mathbf{q}}'+\alpha\mathbf{1}$ is a solution of (10) for any $\alpha\in\mathbb{R}$. Note, however, that, for the purposes of object ranking, the actual value of α is irrelevant, since every solution of the form $\widehat{\mathbf{q}}''=\widehat{\mathbf{q}}'+\alpha\mathbf{1}$ induces the same object ranking. Therefore, we can arbitrarily fix the quality of, say, object N to 0 as a reference, i.e., $q_N=0$. To take into account this constraint, we define the new matrices $\mathbf{H}$ and $\mathbf{M}$ as follows:

$$[\mathbf{H}]_{i,j}=\begin{cases}[\widetilde{\mathbf{H}}]_{i,j} & i<N,\ \forall j\\ 0 & i=N,\ \forall j\end{cases} \tag{12}$$

and $\mathbf{M}=\mathbf{I}-\mathbf{H}$, respectively. We then replace $\widetilde{\mathbf{M}}$ and $\widetilde{\mathbf{H}}$ in (11) with, respectively, $\mathbf{M}$ and $\mathbf{H}$. Since $\mathbf{M}$ is full rank, solving for $\mathbf{q}$ we obtain

$$\widehat{\mathbf{q}}=\mathbf{M}^{-1}(\mathbf{H}\odot\widehat{\mathbf{D}})\mathbf{1}=\mathbf{q}+\mathbf{M}^{-1}(\mathbf{H}\odot\mathbf{Z})\mathbf{1} \tag{13}$$

where we have used the fact that $\mathbf{q}=\mathbf{M}^{-1}(\mathbf{H}\odot\mathbf{D})\mathbf{1}$.

3.1 Weight optimization

In the following, we will consider two possible choices for the weights $\omega_{i,j}$. The first, which will be studied in the next section for its simplicity, corresponds to $\omega_{i,j}=1$ for all i,j, and will be called unweighted LS or simply LS. The second, which will be called weighted LS (WLS), is dictated by the fact that the estimates $\widehat{d}_{i,j}$ do not have the same reliability. Indeed, by developing (8) at the first order for $W\to\infty$, we obtain

$$z_{i,j}=\widehat{d}_{i,j}-d_{i,j}=\left.\frac{dF^{-1}(p)}{dp}\right|_{p=p_{i,j}}y_{i,j}+O\left(y_{i,j}^2\right)\,.$$

6. Notice that the (i,j)-th entry of $\widehat{\mathbf{D}}$ is defined only for $(i,j)\in\mathcal{E}$. The same is true for matrix $\mathbf{Z}$.

so that, if we neglect the higher-order term, $z_{i,j}$ is a zero-mean random variable with variance

$$\sigma_{i,j}^2 = \left(\frac{dF^{-1}(p)}{dp}\bigg|_{p=p_{i,j}} \right)^2 \frac{p_{i,j}(1-p_{i,j})}{W}$$

Given the values of q_j, $j \in \mathcal{N}_i$, the optimal weights for $W \to \infty$ in (10) are then proportional to $\sigma_{i,j}^{-2}$. For our WLS algorithm, we will then set $\omega_{i,j} = \widehat{\sigma}_{i,j}^{-2}$, with

$$\widehat{\sigma}_{i,j}^2 = \left(\frac{dF^{-1}(p)}{dp}\bigg|_{p=\widetilde{p}_{i,j}} \right)^2 \frac{\widetilde{p}_{i,j}(1-\widetilde{p}_{i,j})}{W},$$

where $\widetilde{p}_{i,j} = \max(\min(\widehat{p}_{i,j}, 1-\xi), \xi)$, for a small positive parameter ξ such that $\frac{dF^{-1}(p)}{dp}\Big|_{p=\xi}$ exists finite. Note that, under this setting, $0 < \omega_{i,j} < \infty$.

4 Asymptotic analysis of the LS Estimator

All the theoretical results in this section are obtained by considering the unweighted LS estimator, for simplicity. However, they can be extended to the general weighted case as long as $\min_{i,j} \omega_{i,j} / \max_{i,j} \omega_{i,j}$ is bounded away from 0, as in the case described in Section 3.1.

The following propositions derive the conditions for the asymptotic convergence of the estimated qualities to their true values. We always assume in the following a connected graph $\mathcal{G}$ with matrix $\mathbf{H}$ defined as in (12) and $\mathbf{M} = \mathbf{I} - \mathbf{H}$. We start by presenting a preliminary asymptotic result on the mean square error.

Proposition 4.1. Consider the unweighted LS estimator in (10) applied to connected graph $\mathcal{G}$. Assume that the node degrees are upper-bounded and define $\rho_{\inf} \triangleq \inf_i \rho_i$. Then the MSE on the estimates $\widehat{\mathbf{q}}$ can be bounded by

$$\mathbb{E}[(\widehat{\mathbf{q}} - \mathbf{q})^\mathsf{T} (\widehat{\mathbf{q}} - \mathbf{q})] \leq O\left(\lambda_{\mathbf{C}}^{\max} \frac{N}{W \rho_{\inf}} \right) \qquad (14)$$

where $\lambda_{\mathbf{C}}^{\max}$ is the largest eigenvalue of $\mathbf{C} = \left(\mathbf{M}^{-1}\right)^\mathsf{T} \mathbf{M}^{-1}$ and $W \geq \beta \log N$ for a sufficiently large β.

The proof is provided in Section 1 of the Supplemental Material.

Even if an expression similar to (14) is reported in [10], we remind that the latter was derived for perfect ML estimators under the assumption that $F(\cdot)$ is log-concave; our results, instead, apply to LS algorithm for a general, strictly increasing $F(\cdot)$. Furthermore, (14) complements and extends results in [11] under more general settings (we recall that results in [11] apply to the BTL model only). It is also to be noted that the theoretical results in [11] only apply to the regime where W is large, i.e., $W = \Omega\left(\log^2 \frac{N}{\delta}\right)$. Under such constraint, for any connected graph, the total complexity of the algorithm in terms of number of comparisons is at least $\Omega\left(N \log^2 \frac{N}{\delta}\right)$.

From (14), we can deduce that, whenever $\lambda_{\mathbf{C}}^{\max}$ is bounded (as for example in the case of Ramanujan graphs), by symmetry, $\mathbb{E}[(\widehat{q}_i - q_i)^2] = O\left(\frac{1}{W\rho_{\inf}}\right)$, $i = 1, \ldots, N$. Thus, if $W \to \infty$ for $N \to \infty$, then $\widehat{\mathbf{q}}$ converges in probability to $\mathbf{q}$.

To find out the minimum number of comparisons under which the LS approach satisfies the (ϵ, δ)-PAC conditions, we need to evaluate $\mathbb{P}(\sup_i |\widehat{q}_i - q_i| > \epsilon)$ for $\epsilon > 0$. The following proposition gives sufficient conditions in order for the absolute error to converge to zero in the properly defined limiting regime.

Proposition 4.2. Consider the unweighted LS estimator in (10) applied to connected graph $\mathcal{G}$. For any $\epsilon > 0$, as N grows, $\mathbb{P}(\sup_i |\widehat{q}_i - q_i| > \epsilon) < \delta$, provided that

i) $\limsup_{N\to\infty} \|\mathbf{M}^{-1}\|_\infty < \infty$ (i.e., the ∞-norm of $\mathbf{M}^{-1}$ is bounded),

ii) the total number of edges of $\mathcal{G}$ is $O(N)$, and $W > \beta(\epsilon, \delta) \log N$ for some $\beta(\epsilon, \delta) = O\left(\frac{1}{\epsilon^2} \frac{\log \frac{N}{\delta}}{\log N}\right)$.

Assumption i) can be weakened by the following condition i'):

i') $\limsup_{N\to\infty} \sup_{\mathbf{A}:\|\mathbf{A}\|_\infty \leq 1} \|\mathbf{M}^{-1}(\mathbf{H} \odot \mathbf{A})\mathbf{1}\|_\infty < \infty$.

The proof is provided in Section 2 of the Supplemental Material.

Remark 4.1. Note that Proposition 4.2 provides sufficient conditions for the existence of an (ϵ, δ)-PAC ranking algorithm with complexity $O(\frac{N}{\epsilon^2} \log \frac{N}{\delta})$. In the following subsection, we characterize classes of graphs meeting condition (i) or (i') of Proposition 4.2.

4.1 Considerations on graph structure

Proposition 4.2 grants that the absolute error $\sup_i |\widehat{q}_i - q_i|$ can be well controlled as $N \to \infty$ under some conditions on the matrix $\mathbf{M}$ (condition (i) or (i')). Such conditions hold depending on the structure of the graph $\mathcal{G}$. In order to characterize the class of graphs for which condition (i) or condition (i') holds, we first observe that (13) computes the quality of object i as the average value of the sum of estimated quality differences along all paths joining node i to the reference node N. In other words, $\widehat{q}_i$ can be regarded as the average total reward earned by a standard random walk that starts from node i and stops as soon as it hits node N, when estimates $\widehat{d}_{i,j}$ are the elementary rewards associated to graph edges [20]. Then, in Section 4.1.1 we restrict our asymptotic analysis to directed and acyclic graphs. Finally, in Section 4.1.2 we extend it to the more general class of undirected graphs.

4.1.1 $\mathcal{G}$ is directed and acyclic

An explicit solution of (10) can be found when graph $\mathcal{G}$ is turned into a directed graph, i.e., by choosing for all edges one of the two possible directions. While this assumption is suboptimal, since it constrains the random walk to a subset of possible trajectories, it greatly simplifies the analysis. Indeed, first observe that, for directed graphs and unweighted LS estimation, (10) can be rewritten as:

$$\widehat{q}_i = \sum_{j \in \mathcal{N}_i^-} \frac{\widehat{q}_j + \widehat{d}_{i,j}}{\rho_i^-}.$$

where $\mathcal{N}_i^-$ represents the set of in-neighborhoods of i and $\rho_i^- = |\mathcal{N}_i^-|$. Then, when the graph is directed and acyclic,

and has the reference node N as a common ancestor, an explicit solution for $\widehat{q}_i$, $i = 1, \ldots, N-1$, is

$$\widehat{q}_i = \frac{1}{\rho_i^-} \sum_{j_1 \in \mathcal{N}_i^-} \left[\widehat{d}_{i,j_1} + \frac{1}{\rho_{j_1}^-} \sum_{j_2 \in \mathcal{N}_{j_1}^-} \gamma_{j_2} \right] \tag{15}$$

where

$$\gamma_{j_2} = \widehat{d}_{j_1,j_2} + \ldots + \frac{1}{\rho_{j_{\ell_i-1}}^-} \sum_{j_{\ell_i} \in \mathcal{N}_{j_{\ell_i-1}}^-} \widehat{d}_{j_{\ell_i-1} j_{\ell_i}}$$

and ℓ_i is the length of the longest (simple) path from node i to the reference node. Proposition 4.3 gives sufficient conditions for a directed acyclic graph to meet the requirements of Proposition 4.2. The proposition exploits the notion of proximality between nodes according to the following definition:

Definition 4.1. Given a family of graphs $\{\mathcal{G}_N\}_N$, we say that a node i is *proximal* to the reference node N, with parameters (τ, h), if a random walk starting from i reaches the reference node N within h hops with a probability that is bounded below by τ asymptotically with N.

Proposition 4.3. Given a family of directed and acyclic weakly connected graphs $\{\vec{\mathcal{G}}_N\}_N$ with bounded diameter, condition (i') of Proposition 4.2 is satisfied if one of the following three conditions is met: (i) all paths from any node to the reference have bounded length, (ii) $\sup_i \rho_i^- < \infty$, or (iii) a fraction bounded away from 0 of the in-neighbors of any node is (τ, h) proximal for some $\tau > 0$ and $h < \infty$.

The proof is provided in Section 3 of the Supplemental Material.

4.1.2 $\mathcal{G}$ is undirected

Now, let us go back to the original formulation (10) on the undirected graph. In the following, we will show that, considered from the point of view of a given node, the solution of (10) for an undirected graph can be obtained by defining an equivalent problem for a properly defined directed acyclic graph.

Consider the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ on N nodes and let $\mathbf{T}$ be the $(N-1) \times (N-1)$ matrix obtained from matrix $\mathbf{H}$ by removing the last row and column (i.e., those corresponding to the reference node N). Consider a given node i, $i = 1, \ldots, N-1$, and notice that $\left[(\mathbf{I} - \mathbf{T})^{-1}\right]_{i,j}$ gives the average number of times that node j is visited in the random walk starting from i, before ending in the reference node N [21]. Define

$$\theta_{j,i} = \begin{cases} \frac{\left[(\mathbf{I}-\mathbf{T})^{-1}\right]_{i,j}}{\rho_j}, & j < N \\ 0, & j = N \end{cases} \tag{16}$$

which can be seen to be the average number of times any edge incident to node j is traversed in the direction from j to its neighbors, in the standard random walk defined on $\mathcal{G}$.

Using the above definition, we build a directed acyclic graph and define a biased random walk on it that can be proved to be stochastically equivalent to the standard random walk on $\mathcal{G}$, from the point of view of node i. Let $\vec{\mathcal{G}}_i = (\mathcal{V}, \vec{\mathcal{E}}_i)$ be a directed acyclic graph, where $(j, \ell) \in \vec{\mathcal{E}}_i$ if and only if $(j, \ell) \in \mathcal{E}$ and $\theta_{j,i} > \theta_{\ell,i}$.[7] Let us also define a biased random walk on graph $\vec{\mathcal{G}}_i$, for which, given that the current node is j, the probability of taking outgoing edge (j, ℓ), $\ell \in \mathcal{N}_{ji}^-$ is given by

$$\eta_{j \to \ell, i} = \frac{\theta_{j,i} - \theta_{\ell,i}}{\sum_{\ell' \in \mathcal{N}_{ji}^-} (\theta_{j,i} - \theta_{\ell',i})} \tag{17}$$

where $\mathcal{N}_{ji}^-$ is the set of in-neighbors of j in $\vec{\mathcal{G}}_i$.

The usefulness of defining the above biased random walk on $\vec{\mathcal{G}}_i$ stems from the following proposition, which states that it is equivalent to the standard random walk on $\mathcal{G}$ in a precise sense.

Proposition 4.4. The estimate of q_i given by (13) on $\mathcal{G}$ can be obtained by solving

$$\check{q}_j = \sum_{\ell \in \mathcal{N}_{ji}^-} (\check{q}_\ell + \widehat{d}_{j,\ell}) \eta_{j \to \ell, i}, \;\; i = 1, \ldots, N \tag{18}$$

on $\vec{\mathcal{G}}_i$, and then setting $\widehat{q}_i = \check{q}_i$.

The proof is provided in Section 4 of the Supplemental Material.

According to Proposition 4.4, $\widehat{q}_i$ can be equivalently seen as the average total reward of the standard random walk on graph $\mathcal{G}$ or as the average total reward of the biased random walk on graph $\vec{\mathcal{G}}_i$. The following proposition gives sufficient conditions for a family of graphs to meet the conditions of Proposition 4.2.

Proposition 4.5. Given a family of connected graphs $\{\mathcal{G}_N\}_{N \in \mathbb{N}}$ with bounded diameter, condition i') of Proposition 4.2 is satisfied if, for each node i, $i = 1, \ldots, N-1$ one of the following conditions is satisfied: (i) all paths in $\vec{\mathcal{G}}_i$ from i to the reference have bounded length, (ii) in $\vec{\mathcal{G}}_i$, a fraction bounded away from 0 of the in-neighbors of any node is proximal.

The proof is provided in Section 5 of the Supplemental Material.

In the remaining part of this section, we substantiate our theoretical findings by giving a few examples of graph families that meet the conditions of Proposition 4.5 and are suitable for LS quality estimation.

Example 4.1. Consider the family of complete graphs on N nodes, i.e., $\mathcal{G}_N = \mathcal{K}_N$.[8] Because of symmetry, we can easily see that, after a proper permutation of the nodes, $(\vec{\mathcal{G}}_N)_i = (\vec{\mathcal{G}}_N)_1$ for every $i = 1, \ldots, N-1$. For the same reason, in $(\vec{\mathcal{G}}_N)_1$, $\theta_{j_1,1} = \theta_{j_2,1}$ for $j_1, j_2 = 2, \ldots, N-1$. Thus, the only surviving edges in $(\vec{\mathcal{G}}_N)_1$ are the edges connected either to node 1 or to the reference node N. Since the maximum path length from node 1 to the reference in $(\vec{\mathcal{G}}_N)_1$ is 2, this family of graphs meets

7. It is easy to prove that $\vec{\mathcal{G}}_i$ is acyclic. Indeed, suppose that the cycle $(j_1, j_2, \ldots, j_r, j_1)$ belongs to $\vec{\mathcal{G}}_i$. This implies that, by definition, $\theta_{j_1,i} > \theta_{j_2,i} > \cdots > \theta_{j_r,i} > \theta_{j_1,i}$, which is impossible.

8. Note that even if this class of graphs satisfies property (i'), it can not used to build efficient ranking algorithms, since it has $O(N^2)$ edges.

condition i) of Prop. 4.5. In particular, the estimate of q_i is given by

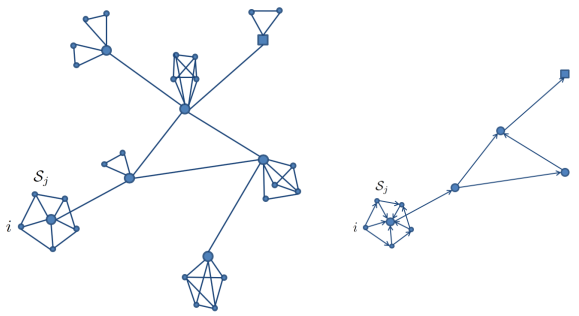
$$\widehat{q}_i = \frac{2}{N}\widehat{d}_{i,N} + \frac{1}{N}\sum_{\substack{j=1 \\ j\neq i}}^{N-1}\left(\widehat{d}_{i,j} + \widehat{d}_{j,N}\right)$$

Example 4.2. Let N' and Δ be any two positive numbers. Let us build the family of graphs $\{\mathcal{G}_N\}_{N\geq N'}$ as follows. Nodes $N-N'+1,\dots,N$ (a set that includes the reference) are "hubs" with potentially unbounded degree. The subgraph induced by the hub nodes is a connected arbitrary graph. The remaining nodes have maximum degree Δ and are divided into N' subsets $\mathcal{S}_1,\dots,\mathcal{S}_{N'}$. All nodes in subset $\mathcal{S}_j$, $j=1,\dots,N'$, are neighbors of hub node $N-j+1$, while their other neighbors all belong to $\mathcal{S}_j$. It is easy to see that, for this family of graphs, the diameter is bounded by $N'+1$.
Consider a node $i\in\mathcal{S}_j$. Since all paths that reach the reference must pass through the hub nodes, it is easy to see that, in $(\overrightarrow{\mathcal{G}}_N)_i$, node i is connected only to nodes belonging to $\mathcal{S}_j\cup\{N-N'+1,\dots,N\}$. Whenever the biased random walk on $(\overrightarrow{\mathcal{G}}_N)_i$ leaves $\mathcal{S}_j$ (by reaching hub node $N-j+1$) does not enter it any more. Thus, we can divide into two parts the biased random walk: the first on the subgraph of $(\overrightarrow{\mathcal{G}}_N)_i$ induced by $\mathcal{S}_j$, where hub node $N-j+1$ serves as reference, and the second on the hub nodes. Then, we can deduce the following facts.

- In the first part of the random walk, since hub node $N-j+1$ is the reference, the probability of reaching it in one step from any node in $\mathcal{S}_j$ is larger than $1/\Delta$. Thus, the probability of reaching it within D' steps is upper-bounded by $\tau = 1-\left(1-\frac{1}{\Delta}\right)^{D'}$.
- The second part of the random walk lasts for at most $N'-1$ steps.

Thus, every node is proximal with parameters $(\tau, D'+N'-1)$, and condition ii) of Prop. 4.5 is satisfied.
Figure 1 (left) shows an example of an undirected graph with $N=32$ nodes, $N'=7$ hubs and maximum degree of non-hub nodes $\Delta=4$. Figure 1 (right) shows the directed graph $\overrightarrow{\mathcal{G}}_i$. As it can be seen, if i belongs to $\mathcal{S}_j$, only the hub nodes and other nodes in $\mathcal{S}_j$ survive in $\overrightarrow{\mathcal{G}}_i$. The direction of edges follows the net flow of probability from node i to the reference, depicted as a square.

Example 4.3. Star graphs represent a particular sequence $\{\mathcal{G}_N\}_N$ of acyclic graphs with bounded-length paths. Therefore, they satisfy condition i) of Prop. 4.5. In such a case, object N is taken as pivot (i.e., center of the star) and qualities of all the other objects are estimated only through direct comparison with it. Observe, that, in such particular case, ranking among objects can be directly inferred from $\widehat{p}_{i,N}$ without the necessity of inverting function $F(\cdot)$, since objects can be ranked according to the following rules: $\widehat{r}(i)\prec\widehat{r}(j)$ iff $\widehat{p}_{i,N}>\widehat{p}_{j,N}$ and $\widehat{r}(i)\prec\widehat{r}(N)$ iff $\widehat{p}_{i,N}>1/2$. Therefore, star graphs are appealing when function $F(\cdot)$ (i.e. the precise worker model) is not known.



(a) (b)

Fig. 1. Left: example of undirected graph according to example 4.2, with $N=32$ nodes, $N'=7$ hubs and maximum degree of non-hub nodes $\Delta=4$. The hubs are represented by big circles, the reference by a square. Right: Directed graph $\overrightarrow{\mathcal{G}}_i$, with edge direction corresponding to the net probability flow from node i to the reference.

Remark 4.2. Although our unweighted LS estimator is akin to the one in [11], our analysis of its performance differs substantially, also because we consider the PAC approach. Consequently, our characterization of "good" graphs does not coincide with that of [11]. For instance, a particular case of Example 4.2 is the wheel graph, which corresponds to choosing $N'=1$ (the reference node as the only hub) and $\Delta=3$. From [11, Theorem 1], the wheel graph would require $W=O(N)$ comparisons per edge in order for the upper bound on the estimation error to hold. Instead, Prop. 4.5 allows to conclude that $W>\beta(\epsilon,\delta)\log N$ is enough to achieve the (ϵ,δ)-PAC. More in general, we remark that results in [11] can be applied only to algorithms employing $\Omega(N\log^2\frac{N}{\delta})$ comparisons.

5 Results with synthetic datasets

We present numerical results showing the performance of LS, WLS and ML algorithms for moderate values of N. As a performance reference we also consider the algorithms "MergeRank" proposed in [22], and "Sync-rank" proposed in [17]. We highlight that in order to adapt the latter to our setting, the entries of the matrix $\mathbf{C}$ in [17, eq.(21)] have been computed as follows

$$C_{i,j} = \widehat{p}_{i,j} = F^{-1}(\widehat{p}_{i,j})$$

for every edge $(i,j)\in\mathcal{E}$. In other words $C_{i,j}$ is a local estimate of the distance between node i and node j, based on the knowledge of the function $F(\cdot)$ and on the estimated probability $\hat{p}_{i,j}=2\frac{\sum_{m=1}^{W}w_{i,j,m}}{W}-1$.

In Figures 2–6 we compare the error probability achieved by several ranking algorithms versus the complexity per object C/N. The LS (which essentially coincides with the algorithms proposed in [11], [16]), WLS [9], ML, and "Sync-rank" algorithms have been applied to randomly generated regular graphs [23] whose nodes have degree ρ. Workers'

9. Reported WLS results have been obtained by setting $\xi=10^{-4}$.

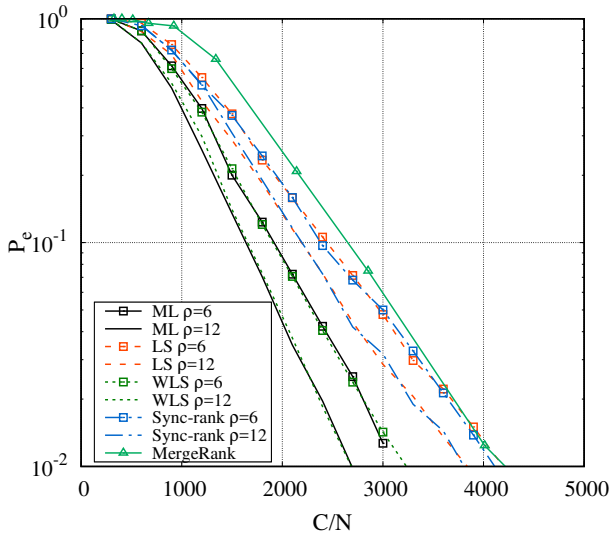


Fig. 2. Error probability achieved by several ranking algorithms plotted versus the complexity per object C/N, for $N = 50$ and $\epsilon = 0.04$. Object qualities are equally spaced in $[0, 1)$ and the workers behave according to the Thurstone model.

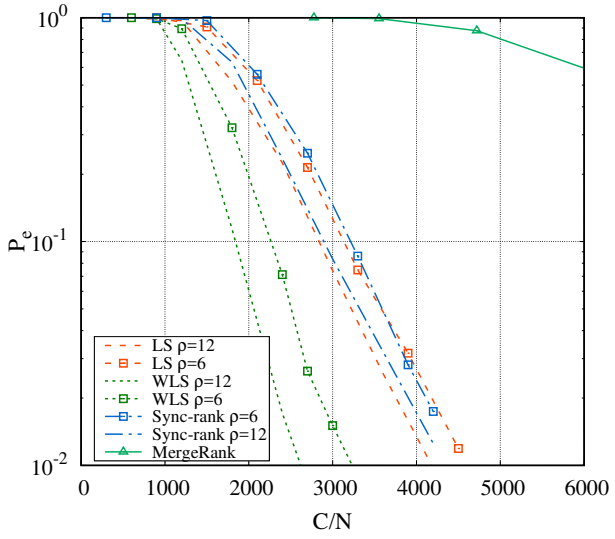


Fig. 3. Error probability achieved by several ranking algorithms plotted versus the complexity per object C/N, for $N = 500$ and $\epsilon = 0.04$. Object qualities are equally spaced in $[0, 1)$ and the workers behave according to the Thurstone model.

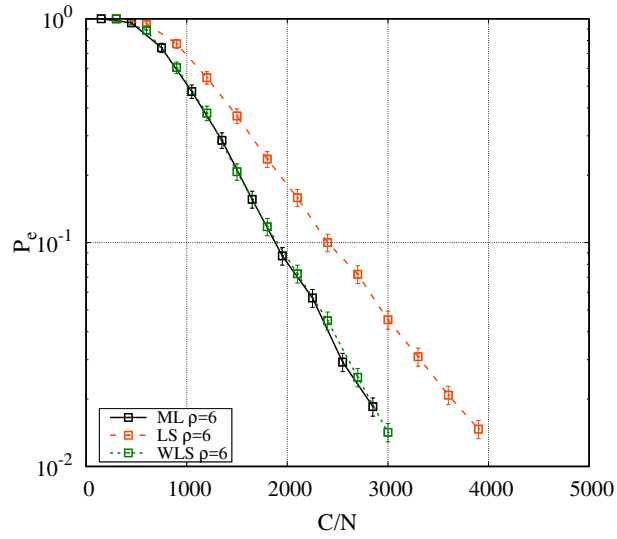


Fig. 4. Performance of the ML, LS and WLS algorithms and their associated confidence intervals, plotted versus the complexity per object C/N, for $N = 50$ and $\epsilon = 0.04$. Object qualities are equally spaced in $[0, 1)$ and the workers behave according to the Thurstone model.

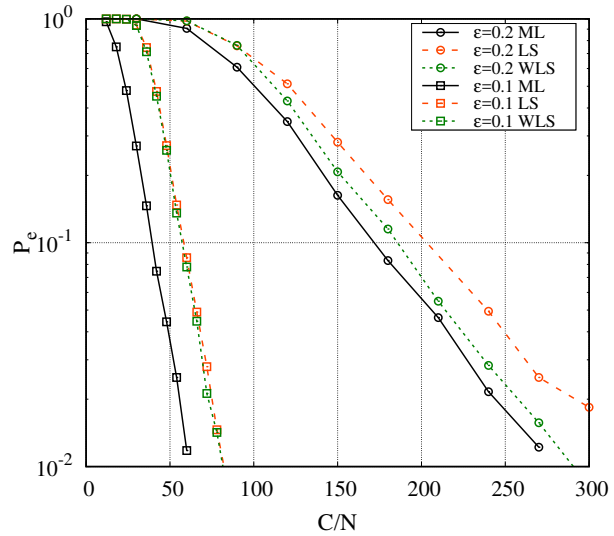


Fig. 5. Performance of the ML, LS and WLS algorithms, plotted versus the complexity per object C/N, for $N = 50$ and for $\epsilon = 0.1, 0.2$. Object qualities are equally spaced in $[0, 1)$ and the workers behave according to the Thurstone model.

behavior is described by the Thurstone model detailed in Section 1.1 where $p_{i,j} = F(q_i - q_j)$ and $F(\cdot)$ is the cdf of a Gaussian random variable with zero mean and standard deviation σ, i.e., $F(q_i - q_j) = \text{erf}\left(\frac{q_i - q_j}{\sqrt{2}\sigma}\right)$, with $\sigma = 0.4$. On the y-axis we display the empirical probability, P_e, of generating an output which is not an ϵ-quality ranking. Note that an error is counted whenever at least two objects, whose quality difference exceeds ϵ, appear swapped in the estimated ranking.

In Figures 2 and 3 we consider a system with $N = 50$ and $N = 500$ objects, respectively. We generate random regular graphs with degree $\rho = 6$ (lines with square markers) and $\rho = 12$ (lines without markers). Object qualities are equally spaced in $[0, 1)$ and $\epsilon = 0.04$. The figure shows that the "Sync-rank" and LS algorithms behave similarly, while the WLS algorithm shows superior performance. It is interesting to observe that the WLS algorithm provides significant enhancements with respect to the LS algorithm and almost perfectly matches the performance of the more (computationally) complex ML approach. For $N = 500$ the performance of the ML is not shown because of its high computational complexity.

The accuracy of the proposed results are shown in Figure 4 where confidence intervals are associated to each curve. Each simulation point was obtained by counting 1000 error events, and the confidence level was set to 3

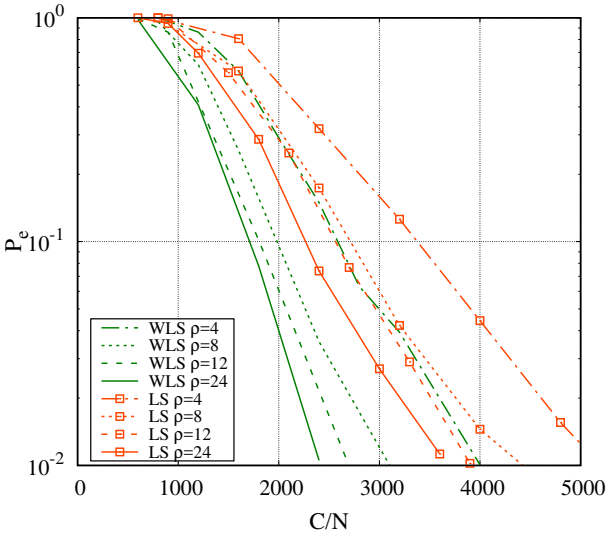


Fig. 6. Performance of the LS and WLS ranking algorithms plotted against the complexity per object C/N, for $N = 200$ objects. Object qualities are drawn from a uniform distribution in $[0, 1)$ and the workers behave according to the Thurstone model.

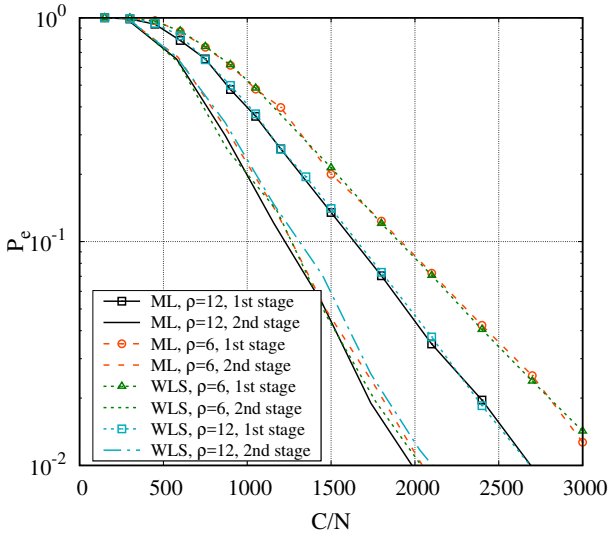


Fig. 7. Error probability provided by ML and WLS algorithms when a 2-stage adaptive approach is employed, for $\rho = \rho^{(1)} = \rho^{(2)} = 6, 12$, and $N = 50$.

standard deviations from the average of the dataset. As the figure shows, the proposed results are highly accurate, thus supporting the validity of our solutions.

In Figure 5 we focus on the case $N = 50$ and vary the parameter ϵ. As ϵ increases, less stringent criteria are required for declaring the correctness of the estimated ranking[10] and, therefore, fewer comparisons per edge, hence a smaller C/N is required to achieve the same error probability. Also, we observe that as W decreases, the weights employed in the WLS algorithm tend to be less reliable. This explains why for $\epsilon = 0.2$ the WLS perform as the LS algorithm while for smaller ϵ it approaches the ML performance.

Figure 6 compares the performance of the LS and of the WLS algorithms for $N = 200$ objects. Object qualities are randomly generated according to a uniform distribution in $[0, 1)$, and $\epsilon = 0.04$. The figure reports the empirical error probability for different values of the nodes' degree of the graph. We first observe that, given C/N, the number of tests per edge of the graph decreases as the degree ρ increases. Hence, as ρ increases, distances between pair of objects (corresponding to edges of the graph) are estimated with a decreasing accuracy. In spite of that, a larger number of neighbors for each node (i.e., a larger ρ) leads to a more reliable evaluation of object qualities. This effect is more evident when the WLS algorithm is employed. Indeed, because of the weights $\omega_{i,j}$, as ρ increases, WLS is able to well exploit the increasing number of highly reliable edges in the graph connecting objects with similar qualities; at the same time WLS is able to limit the impact of the greater number of scarcely reliable edges that connect objects with largely different qualities.

10. We recall that an error is counted whenever at least two objects, whose quality difference exceeds ϵ are swapped in the estimated ranking

5.1 Adaptive multistage approach

The performance of the proposed ranking algorithms can be improved by adopting a multistage approach where, at each stage, new edges are added to the graph, depending on the quality estimates obtained at the previous stage. The rationale of this approach stems from the fact that such algorithms provide approximate rankings, in which the probability of swapping the order of two objects increases as their distance (in terms of their qualities) decreases. Therefore, in order to mitigate this phenomenon and, thus, to improve the reliability of the estimate, it is convenient to (i) add to the graph extra edges connecting neighboring objects (in terms of their estimated qualities); (ii) assign additional workers to the already existing edges connecting the aforementioned neighboring objects. This procedure can be iterated until a desired performance level is achieved.

In our simulation setup, we have considered a 2-stage approach where we first apply the estimation algorithm to a random regular graph, $\mathcal{G}^{(1)}(\mathcal{V}, \mathcal{E}^{(1)})$, of degree $\rho^{(1)}$, obtaining the vector of estimates $\widehat{\mathbf{q}}^{(1)}$. In the second stage, we create a new regular graph, $\mathcal{G}^{(2)}(\mathcal{V}, \mathcal{E}^{(2)})$ of degree $\rho^{(2)}$, where each node is connected to its $\rho^{(2)}$ closest neighbors, according to the estimates $\widehat{\mathbf{q}}^{(1)}$. Finally, the estimation algorithm is applied to the graph $\mathcal{G}^{(1)} \cup \mathcal{G}^{(2)}$ obtaining the output $\widehat{\mathbf{q}}^{(2)}$ which is used to infer the ranking. In Figure 7 we show the performance of the ML and WLS algorithms when the proposed multistage approach is employed. For both algorithms we show the error probability versus the number of tests per object, C/N, for $\rho = \rho^{(1)} = \rho^{(2)} = 6, 12$, and $N = 50$. We observe that the second stage allows for a significant improvement of the performance and a reduction of about 60% of the required tests per object for $\rho = 6$ and of about 30% for $\rho = 12$. In both cases the performance of the WLS algorithm is very close to that provided by the ML algorithm.

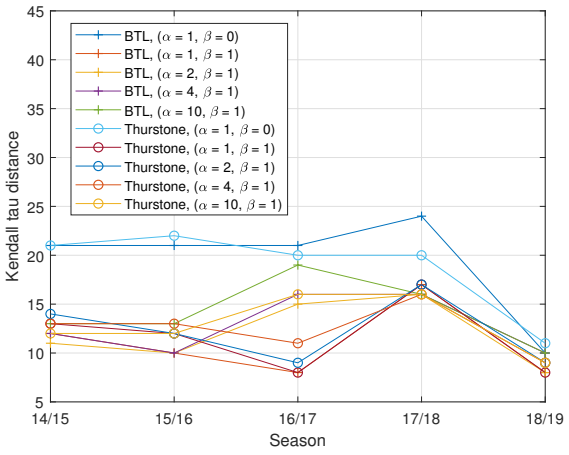


Fig. 8. Distance between true and estimated ranking for Premier League scores. The x-axis is the season. The y-axis is the Kendall tau distance (number of inversions) between the final season ranking and the output of the WLS algorithm, for different choices of the model and of the parameters.

6 Results with Real-World Datasets

In this section, we show that our algorithm works well even when considering a real-world scenario, where the "evaluations" are the outcome of experiments, and not synthetically generated by simulations. In particular, we consider five recent seasons of the English Premier League and build up a $N = 20$ complete graph, where nodes are the football teams and edges are the matches between each pair of them. The match between team i and team j is considered as lasting for 180 minutes, since it includes both the round when i is at home and the round where i is away. If team i has scored $x_{i,j}$ goals in the match against team j, we count $K_{i,j} = \alpha x_{i,j} + \beta$ evaluations in favor of i when compared to j, where $\alpha > 0$ and $\beta \geq 0$ are constant. The total number of comparisons between i and j is then simply $W_{i,j} = K_{i,j} + K_{j,i}$[11].

The WLS algorithm has been run with $\xi = 10^{-4}$ and both the Thurstone and BTL models, to see the influence of the underlying worker model. The true ranking is assumed to be the final season ranking. The results have been plotted in terms of the Kendall tau distance, which counts the number of inversions in the estimated ranking with respect to the true ranking, i.e. the number of pairs (i, j) for which i is ranked better than j in the true ranking and worse than j in the estimated one.

Results are shown in Fig. 8. First, we can observe that the performance is better with $\beta > 0$ than with $\beta = 0$, since in the latter case there might be some edges for which the estimated preference probability is very close to either 0 or 1. Such edges are automatically dropped by the WLS algorithm, while in the former case each object in each comparison receives at least β preferences, so that all edges are used for ranking computation. Second, the Thurstone model seems to be slightly better suited than the BTL model. Third, while in most cases, the influence of parameters is limited, there are cases (like season 16/17) that are more sensible to the choice of α and β. It is worth mentioning that, in [17], the Sync-Rank algorithm is applied to older Seasons of the Premier League. Comparatively, for $\beta > 0$, Kendall tau distance for our algorithm never goes beyond 20, giving rise to a Kendall correlation larger than 0.90, which is a better result than those shown in [17].

11. With this definition, the edge between i and j may be actually missing if $x_{i,j} = x_{j,i} = 0$ and $\beta = 0$.

7 Conclusions

In this work, we have focused on the problem of ranking N objects starting from a set of noisy pairwise comparisons. Objects are assumed to be endowed with intrinsic qualities. A general parametric model is introduced where the probability $p_{i,j}$ that object i is preferred to j is given by an arbitrary smooth monotonic function of the difference between the qualities of the two competitors. For such a scenario we developed a class of order-optimal ranking algorithms, i.e. algorithms that are provably (ϵ, δ)-PAC when $O(\frac{N}{\epsilon^2}\log(\frac{N}{\delta}))$ comparisons are blindly allocated in a single round. Our ranking procedure is based on the reconstruction of object qualities, from pairwise quality differences, by adopting a simple LS approach. The analysis establishes a parallelism between the quality estimation process and the cumulative reward accumulated by random walks on graphs. Regarding the choice of the graph, we first provide examples of graph families that are asymptotically suitable for our proposed LS estimation technique (see Examples 4.1- 4.3). Then, through simulation results we show that, even when the number of nodes is finite and the graph is randomly generated, our proposed WLS algorithm provides excellent performance and approaches that obtained by the ML algorithm. Moreover, in order to further improve the performance, we propose an effective "multistage" strategy in which we aim at concentrating the comparisons between objects whose qualities are not too different.

Our results complement and extend previous studies [7], [8], [9] on the minimal complexity of ranking algorithms under different non-parametric preference models. We remark that the results of this work, as summarized above, have wide applicability as they apply to a general parametric model. Thus, they are independent of a specific worker behavioral model and of the considered context.

References

[1] M. Richardson, E. Dominowska, and R. Ragno, "Predicting clicks: Estimating the click-through rate for new ads," in *Proceedings of the 16th International Conference on World Wide Web*, ser. WWW '07, 2007, pp. 521–530.

[2] R. Herbrich, T. Minka, and T. Graepel, "Trueskill™: A bayesian skill rating system," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. C. Platt, and T. Hoffman, Eds. MIT Press, 2007, pp. 569–576.

[3] R. L. Plackett, "The analysis of permutations," *Applied Statistics*, pp. 193–202, 1975.

[4] R. D. Luce, *Individual choice behavior: A theoretical analysis*. Courier Corporation, 2012.

[5] L. L. Thurstone, "The method of paired comparisons for social values." *The Journal of Abnormal and Social Psychology*, vol. 21, no. 4, p. 384, 1927.

[6] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: I. the method of paired comparisons," *Biometrika*, vol. 39, no. 3/4, pp. 324–345, 1952.

[7] B. Szörényi, R. Busa-Fekete, A. Paul, and E. Hüllermeier, "Online rank elicitation for plackett-luce: A dueling bandits approach," in *Advances in Neural Information Processing Systems*, 2015, pp. 604–612.

[8] M. Falahatgar, A. Orlitsky, V. Pichapati, and A. T. Suresh, "Maximum selection and ranking under noisy comparisons," *arXiv preprint arXiv:1705.05366*, 2017.

[9] M. Falahatgar, A. Jain, A. Orlitsky, V. Pichapati, and V. Ravindrakumar, "The limits of maxing, ranking, and preference learning," in *International Conference on Machine Learning*, 2018, pp. 1426–1435.

[10] N. B. Shah, S. Balakrishnan, J. Bradley, A. Parekh, K. Ramchandran, and M. J. Wainwright, "Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2049–2095, 2016.

[11] J. Hendrickx, A. Olshevsky, and V. Saligrama, "Graph resistance and learning from pairwise comparisons," in *Proceedings of the 36th International Conference on Machine Learning*. International Machine Learning Society, 2019.

[12] M. Jang, S. Kim, C. Suh, and S. Oh, "Top-k ranking from pairwise comparisons: When spectral ranking is optimal," *arXiv preprint arXiv:1603.04153*, 2016.

[13] Y. Chen and C. Suh, "Spectral mle: Top-k rank aggregation from pairwise comparisons," in *International Conference on Machine Learning*, 2015, pp. 371–380.

[14] S. Negahban, S. Oh, and D. Shah, "Iterative ranking from pair-wise comparisons," in *Advances in neural information processing systems*, 2012, pp. 2474–2482.

[15] ——, "Rank centrality: Ranking from pairwise comparisons," *Operations Research*, vol. 65, no. 1, pp. 266–287, 2016.

[16] A. N. Hirani, K. Kalyanaraman, and S. Watts, "Least squares ranking on graphs, hodge laplacians, time optimality, and iterative methods," *CoRR*, vol. abs/1011.1716, 2010. [Online]. Available: http://arxiv.org/abs/1011.1716

[17] M. Cucuringu, "Sync-rank: Robust ranking, constrained ranking and rank aggregation via eigenvector and sdp synchronization," *IEEE Transactions on Network Science and Engineering*, vol. 3, no. 1, pp. 58–79, 2016.

[18] A. d'Aspremont, M. Cucuringu, and H. Tyagi, "Ranking and synchronization from pairwise measurements via svd," *arXiv preprint arXiv:1906.02746*, 2019.

[19] K. R. M. J. W. Reinhard Heckel, Nihar B. Shah, "Active ranking from pairwise comparisons and when parametric assumptions don't help," *arXiv preprint arXiv:1606.08842*, 2016.

[20] B. Ewald, J. Humpherys, and J. West, "Computing expected transition events in reducible markov chains," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 3, pp. 1040–1054, 2010.

[21] J. G. Kemeny and J. L. Snell, *Markov Chains*. Springer-Verlag, New York, 1976.

[22] M. Falahatgar, Y. Hao, A. Orlitsky, V. Pichapati, and V. Ravindrakumar, "Maxing and ranking with few assumptions," in *Advances in Neural Information Processing Systems*, 2017, pp. 7063–7073.

[23] B. Bollobás, *Random graphs*. Cambridge university press, 2001.



Evgenia Christoforou is a Research Associate at the TAG MRG at CYENS (formerly known as RISEup). Previously, she held a postdoc position at Politecnico di Torino, Italy. In 2017 she received her PhD in Mathematical Engineering from Universidad Carlos III de Madrid, were she also received an award for the best thesis of the department. During her PhD studies she was employed by IMDEA Networks Institute while she was also awarded a highly competitive FPU scholarship from the Spanish Ministry of Education, Culture and Sports (MECD) to pursue her studies. Her main research focuses on improving quality and increasing crowdsourcing reliability as well as looking into data bias generated through crowdsourcing.



Alessandro Nordio (S'00-M'03) is a Researcher with the Institute of Electronics, Computer and Telecommunication Engineering of the Italian National Research Council. In 2002 he received the Ph.D. in Telecommunications from EPFL, Lausanne, Switzerland. From 2002 to 2009 he was a post-doc researcher with the Electronic Department of Politecnico di Torino, Italy. His research interests are in the field of signal processing, wireless sensor networks, theory of random matrices, and crowdsourcing systems.



Alberto Tarable (S'00-M'02) received the Laurea degree (summa cum laude) in 1998 and the Ph.D. degree in Electronic Engineering in February 2002, both from Politecnico di Torino. From 2002 to 2012, he worked as a researcher in the Department of Electronics and Telecommunications of Politecnico di Torino. From 2012, he holds a research position in the Institute of Electronics, Computer and Telecommunication Engineering of the Italian National Research Council. His research interests include MIMO systems and space-time coding, anytime coding and coding schemes for relay channels.



Emilio Leonardi (S'94–M'99–SM'09) received the Dr. Ing. degree in electronics engineering and Ph.D. degree in telecommunications engineering from the Politecnico di Torino, Turin, Italy, in 1991 and 1995, respectively. He is currently an Associate Professor with the Department of Electronics, Politecnico di Torino, Turin, Italy. His research interests include performance evaluation of computer networks and distributed systems, dynamics over social networks, and human centric computation