Key Node Identification Method Integrating Information Transmission Probability and Path Diversity in Complex Network

Article *in* The Computer Journal · November 2022

DOI: 10.1093/comjnl/bxac162

CITATIONS
0

READS
30

4 authors, including:

Luyuan Gao
Chongqing University of Technology
1 PUBLICATION **0** CITATIONS

SEE PROFILE

Giacomo Fiumara
Università degli Studi di Messina
110 PUBLICATIONS **2,666** CITATIONS

SEE PROFILE

Pasquale De Meo
Università degli Studi di Messina
161 PUBLICATIONS **3,596** CITATIONS

SEE PROFILE

https://doi.org/10.1093/comjnl/bxac162

Key Node Identification Method Integrating Information Transmission Probability and Path Diversity in Complex Network

Xiaoyang Liu[1,*], Luyuan Gao[1], Giacomo Fiumara[2] and Pasquale De Meo[3]

[1]*School of Computer Science and Engineering, Chongqing University of Technology, Chongqing, 400054, China*
[2]*Department of Mathematical and Computer Science, University of Messina, V.le F. Stagno D'Alcontres, 31, 98166 Messina, Italy*
[3]*Department of Ancient and Modern Civilizations, University of Messina, V.le G. Palatucci, 25, 98166 Messina, Italy*
* *Corresponding author: lxy3103@cqut.edu.cn*

Previous key node identification approaches assume that the transmission of information on a path always ends positively, which is not necessarily true. In this paper, we propose a new centrality index called *Information Rank* (IR for short) that associates each path with a score specifying the probability that such path successfully conveys a message. The IR method generates all the shortest paths of any arbitrary length coming out from a node u and defines the centrality of u as the sum of the scores of all the shortest paths exiting u. The IR algorithm is more robust than other centrality indexes based on shortest paths because it uses alternative paths in its computation, and it is computationally efficient because it relies on a Beadth First Search-BFS to generate all shortest paths. We validated the IR algorithm on nine real networks and compared its ability to identify super-spreaders (i.e. nodes capable of spreading an infection in a real network better than others) with five popular centrality indices such as Degree, Betweenness, K-Shell, DynamicRank and PageRank. Experimental results highlight the clear superiority of IR over all considered competitors.

Keywords: complex network; key node identification; information dissemination; path diversity

Received 21 May 2022; Revised 9 September 2022; Editorial Decision 12 October 2022
Handling editor: Fairouz Kamareddine

1. INTRODUCTION

1.1. Background

Complex networks are a powerful mathematical tool to describe the structure and functioning of many (both natural and man-made) real-world systems and, thus, we are not surprised that Complex Networks have been widely applied to solve challenging problems in a broad range of disciplines [1–5]. As an example sociologists use complex networks as a metaphor to abstract large (both online and offline) social networks and to define strategies capable of suppressing the spread of misinformation in human communities [1]. Researchers from Life Sciences make use of complex networks to better elucidate interactions between proteins [2]. Complex networks are also relevant to design transportation networks and find the most efficient routes to deliver goods [3,4] as well to better describe the structure of financial markets [5].

In its simplest form, a complex network is modeled as a graph $G = (N, E)$, being N the set of nodes and E the set of edges. A node corresponds to a basic component of the complex network: e.g. a node could coincide with an individual [6] (if the complex network is associated with a social network), a node could correspond to a railway station/airport (if the complex network represents a public transport system) and, finally, a

node could correspond to an atom (if the complex network is used to represent a molecule).

Edges in complex networks are used to model associations between nodes: in the wake of the examples we introduced above, we have that an edge could identify a friendship relationship between two individuals in a social network; in a transport network, an edge could identify a railway connection between two stations or a flight connecting two airports and, finally, in the case of molecules, edges specify bonds between pairs of atoms.

We can easily realize that *not all the nodes in a complex network have the same importance*: in general, the structure of a complex network depends on few nodes (which we will call *key nodes*) that are able to dominate the evolution of the network itself (and, consequently, key nodes control the evolution of the system that the complex network represents) [7].

A fundamental research area in the field of complex networks focus on the design of algorithms to spot key nodes. Intuitively, key nodes are understood as those nodes which, if removed from a complex network (or, at least, if we make them inactive), contribute more than all others nodes to the performance decay of the system corresponding to that network [8]. In some cases, if we were able to make key nodes inactive, we would also be able to disintegrate the network itself.

The identification of key nodes clearly depends on the domain of interest: e.g. in a transportation network [9], key nodes correspond to airports/railway stations which, if temporarily closed, cause the biggest slowdowns in the movement of goods and people [10]. In online social networks (e.g. TikTok, which is probably one of the most popular social networks in the world), the key nodes correspond to celebrities or, more generally, to people capable of catalyzing consensus [8]. In addition, large-scale social platforms (such as the aforementioned TikTok or Weibo) make use algorithms to detect key nodes to rank bloggers in their field of interest who will be subsequently recommended to affiliated users (and such a feature is particularly relevant for new users; [11–13]); the identification of the most influential bloggers in an online social networks can help business analysts to quickly understand the mainstream content in a particular field and, thus, it is of the utmost relevance to design appropriate marketing and advertising strategies. In epidemiological studies (e.g. think of the COVID19 pandemic), key nodes correspond to super-spreaders, i.e. individuals who contribute the most to the diffusion of an infection. Thus, the ability of identifying super-spreaders and formulating corresponding isolation measures can effectively delay the spread of an epidemic [14].

The identification of key nodes plays a key role in the understanding of multiple phenomena of interest of complex networks such as information cascades and synchronization processes.

1.2. Motivation

Many algorithms to detect key nodes rely on the topological structure of the graph G associated with a complex network.

In particular, some approaches rely on the assumption that key nodes should be able to quickly interact with as many nodes as possible in G. Two nodes in G can interact if there is at least one *path* connecting them: we define a path in G connecting a *source* node i with a *target* node j as a sequence of distinct nodes such that consecutive nodes in the sequence are tied by an edge. Given a pair of nodes, there are, in principle, several paths connecting them and we are interested in the shortest of these paths (i.e. the path containing the fewest number of edges).

Some approaches do not require that nodes to visit must be distinct but they allow to visit a node more than once. In the language of graph theory, we say that these approaches replace paths with *walks*: here, a walk is understood as a sequence of nodes such that consecutive nodes in the sequence are connected by an edge but, unlike paths, a walk can contain repeated nodes. Approaches based on walks assume that a node is important if it is connected by many walks to all other nodes; consequently, these approaches count the number of walks converging in a node and, in some cases (see the popular *Katz coefficient* [15]) they weight the contribution provided by a walk by a factor, which decreases with walk length. Approaches based on walks generally include cycles in G.

It is worth observing that some approaches run multiple random walks on G to detect key nodes.

Methods above associate each node in G with a numerical score, called centrality, and they assume the higher the centrality, the more influential a node.

The approaches cited above suffer from some relevant drawbacks, namely:

(i) *Shortest path missing*. There are some nodes that are not on any shortest path, so methods based on shortest path cannot calculate the centrality of these special nodes or they will wrongly associate a centrality equal to zero to these nodes. Methods based on shortest-paths methods assume that information between nodes is transmitted only through the shortest path, but such an assumption is sometimes unrealistic.

(ii) *Double counting*. Methods based on walks are demanding from a computational standpoint. For instance, the calculation of the Katz coefficient would require to calculate the inverse of a matrix with $|N|$ rows and $|N|$ columns and such a computation is unfeasible in moderately large graphs.

1.3. Main contributions

The main contributions of this paper are listed below:

(i) We introduce a new centrality index for unweighted graphs called *Information Rank* (IR, in short) and we

provide an algorithm to calculate it. Our approach manages all shortest paths exiting from a source node u of arbitrary length l between 1 and D (here D is the network diameter). Shortest paths do not contribute equally to the computation of the IR associated with u, but each path is associated with a score, which quantifies the ability of such a path to successfully transfer information from u to a target node. We have therefore considered a simple model to estimate the probability of successful transfer of information along a path. The IR of a node is obtained by summing the weights associated with each of the paths the node u emanates. Observe that a node has IR score equal to zero if and only if it does not belong to any of the shortest paths associated with u and, thus, this solves the shortest path missing problem mentioned above.

(ii) We have shown that calculating the IR of a node u is equivalent to performing a Breadth First Search-BFS from that node. Since the computational complexity of the BFS is proportional to the number of edges, it follows that the calculation of the IR is efficient even in large but sparse networks. Since the algorithm does not traverse the same node more than once, we have that the IR algorithm is exempt from the double counting problem.

(iii) We experimentally compared IR with 5 centrality indices (Degree, Beetweeness, K-Shell, DynamicRank and PageRank) on 9 real datasets. In our tests, we simulated the spread of an influence in each dataset according to the Susceptible-Infected-Recovered (SIR) model; the simulations allowed us to rank node according to their ability to infect other nodes. The ranking obtained from the SIR model acts as a golden truth useful to evaluate the effectiveness of each centrality index. Our experimental results show that the IR method obtains a consistently higher level of correlation (calculated by means of *Kendall's correlation coefficient*) than the rankings generated by all the other methods (in some cases with an improvement of more than 100% over the second best-performing centrality index).

2. RELATED WORK

Key nodes identification can be regarded as the task of sorting the nodes by some centrality index. A number of centrality indices have been proposed so far in the literature. Some of the existing methods to compute node centrality rely only on local information, whereas other methods make use of the whole network topology in the computation of node centralities.

In the following we describe both local and global methods to calculate node centrality.

Degree centrality [16] assumes that the greater the degree of a node is, the more important that node. The degree centrality index is easy to use and understand, with a minimal computational cost. However, because it ignores higher-order neighbors, it is often inaccurate. Kitsak *et al.* presented the K-shell decomposition technique [17], which takes into account the node's network location as well as its degree. The K-shell decomposition technique extends the degree by taking into account high-order neighbors of a node: it has a low computational complexity and it solves some of the drawbacks of the degree centrality but it seems to be not effective in many practical scenarios. Community structure [8]—i.e. the fact a network of divide itself into subgroups of densely connected nodes—is also used to describe the characteristics of complex networks. Bonacich *et al.* [16] proposed a method to sort nodes by centrality which is based on the k-shell decomposition, and considers community structure. Experiments on SIR model show that this method is superior to Kitsak *et al.*'s method. A further parameter to measure node centrality is *eccentricity* [18], which is the maximum length of the shortest path between a node and other nodes in the network. The maximum value of eccentricity of all nodes is the network diameter, and the minimum value of eccentricity is the network radius. The closer the eccentricity of a nodes is to the network radius, the more important the node. However, this method is susceptible to the influence of special values. Closeness centrality [19] is a centrality index defined as the average value of the distance between nodes. The Katz centrality [20] considers walks in a graph to compute node centrality.

In addition to the topological structure of the network, there are feature based methods [21–23] that consider the influence of the quality of vertex neighbors on the centrality of nodes based on the topological structure of nodes, with the most well-known algorithms being PageRank [24] and LeaderRank [25].

The probability of information transmission also plays a key role in the identification of important nodes in complex networks [26]. Bao *et al*. pointed out that the probability of information transmission between nodes is roughly equal to the product of the number of shortest paths between them and the probability of transmission. Based on this, the SP algorithm was proposed to calculate the centrality value of nodes [27]. Chen *et al.* claimed that information transmission dynamics is a key factor in identifying important nodes in real complex networks, and they proposed a node sorting algorithm named *DynamicRank* [28].

Table 1 summarizes the main methods to compute node centrality in complex networks.

In this paper we introduce a new centrality model, called *Information Rank*—IR in short, which relies on shortest paths, but it discriminates shortest paths based on their ability to successfully transfer information from one node to another one in a network.

In our view, in fact, the centrality of a node quantifies the node's ability to propagate information and, in line with previous research work, we believe that a node that appears to be 'sufficiently close' to all other nodes in the network can exploit such a proximity to efficiently propagate the messages it generates. The distance (and, hence, proximity) of an arbitrary

TABLE 1. Basic overview of key nodes sorting algorithms

Category	Method	Equation	Variables	Topology
Neighbor node	Degree centricity	$DC(u) = k_u/(n-1)$	k_u: degree of node u. n: network nodes number.	Local
	ClusterRank	$s_i = f(c_i)\sum_{j\in\Gamma_i}(k_j^{out}+1)$	$f(c_i)$: function of the clustering coefficient c_i of the node i. k_j^{out}: out degree of node j. Γ_j: first-order neighbor nodes of j.	Global
The path	Betweenness centrality	$BC(u) = \sum_{u\neq s, u\neq t, s\neq t}\frac{g_{st}^u}{g_{st}}$	g_{st}: number of all shortest paths from node s to t. g_{st}^u: number of shortest paths from node s to t through u in g_{st}.	Global
	Closeness centrality	$CC(u) = 1/(\sum_{v-1}^{n-1} d(u,v))$	$d(u,v)$: the shortest distance between nodes u and v.	Global
The feature vectors	PageRank	$PR_u(t) = \sum_{v=1}^{n} a_{vu}\frac{PR_v(t-1)}{k_v^{out}}$	PR_v: the PR value of the moment node v. a_{vu}: value at row u and column v in the adjacency matrix.	Global
	LeaderRank	$S_i = s_i(t_c) + s_g(t_c)/N$	$s_g(t_c)$represents the centrality score calculated by LeaderRank when the background node g reaches the steady state time t_c, $s_i(t_c)$represents the centrality score calculated by the LeaderRank when the node i reaches the steady state time t_c, N is the number of points the $s_g(t_c)$ will give to other nodes.	Global

pair of nodes is understood as the length of the shortest path connecting the two nodes. This reasoning leads one to hypothesize (and this is the purpose of centrality measures such as the closeness centrality and the betweenness centrality) that a node with a 'short' distance to all other nodes is necessarily a node with high centrality.

Such reasoning has, however, an important limitation: in fact, the propagation of information in a network may be affected by errors and a message sent from a source node to a target node may fail. For example, two nodes u and v could be connected via a path p of length l but the transfer of a message on p could end up in a failure.

The presence of multiple alternative paths connecting u with v would enable us to compute the centrality of u in a more robust fashion. In fact, if there were multiple paths of length l between u and v besides p (or, possibly, paths of length $l+1$ or bigger), then u could benefit of one of these alternative paths or it could leverage a sub-optimal path of length $l+1$ (or bigger) to cope with the failure of p.

Our intuition is, therefore, to consider the totality of shortest paths coming out from a node u. For each of these paths, we suggest computing a score specifying the probability such a path can successfully transfer a message. The centrality of u is defined as the sum of the scores associated with all paths exiting u.

In what follows, we formalize our intuitions, and we show that the calculation of the Information Rank is computationally equivalent to running a Breadth First Search—BFS on an undirected graph, which makes IR competitive even in large but sparse networks.

3. METHODOLOGY

3.1. Definition

Graphs are a powerful mathematical abstraction to describe complex networks. In this section we provide some basic definitions on graphs, which will be extensively used throughout the paper. A graph $\mathcal{G}$ is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, v_2, \cdots, v_n\}$ is the *node set* and $\mathcal{E} = \{e_1, e_2, \cdots, e_m\}$ is the *edge se*t. Some definitions are described below:

(i) Paths. A *path* in $\mathcal{G}$ starting from a node u (called *source node*) to a node w (called *target node*) is a sequence of nodes $\mathcal{P} = \{u, i_0, i_1, \cdots, i_n, w\}$such that all elements in $\mathcal{P}$ are distinct (that is, no repeated nodes are allowed) and pair of consecutive nodes in $\mathcal{P}$ are connected by an edge.

The length of a path is defined as the number of edges in the path. For a fixed pair of nodes u and w, there are, in general, multiple paths connecting u to w. In the following, we use the symbol $\mathcal{P}(u,w)^l$ to denote the set of paths from u to w of length l. For example in Fig. 1, the paths between a and e of length $l=2$ are respectively $\mathcal{P} = \{a, d, e\}$ and $\mathcal{P} = \{a, c, e\}$.

(ii) Paths Reachable Nodes Set

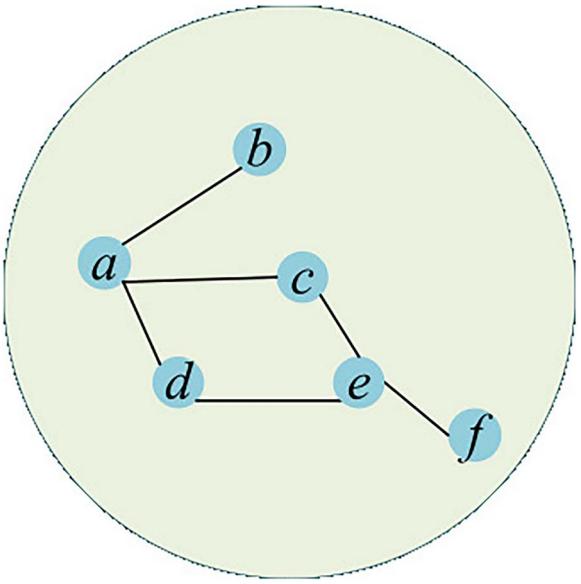

FIGURE 1. Explains the two definitions Paths and Paths Reachable Nodes Set above.

Given a starting node u and a positive integer we define a set $\mathcal{N}(u)^l$, which consists of all nodes in $\mathcal{G}$ which includes all nodes w in $\mathcal{V}$such that there exists at least one path of length l connecting u to w.

$$\mathcal{N}(u)^L = \left\{\mathcal{N}(u)^1 \cup \mathcal{N}(u)^2 \cdots \cup \mathcal{N}(u)^L\right\} \tag{1}$$

We observe that the set $\mathcal{N}(u)^L$ can be easily constructed by running a Breadth First Search in $\mathcal{G}$, which starts from the node u. We provide some examples to illustrate the notion of Path Reachable Node Set. Let us consider Fig. 1 and observe that $\mathcal{N}(a)^1$ is the set of nodes that can be reached from the node a by a path of length one, i.e. $\mathcal{N}(a)^1$ is the set of neighbors of a. From nodes c and d we can reach the node e, which means that e can be reached from a by means of a path of length $l = 2$. As a consequence, $\mathcal{N}(a)^2$ will contain nodes b, c, d and e. From the node e we can reach nodes c and d (which have been already visited) and the node f (which has not yet been visited). Therefor we have that $\mathcal{N}(a)^3$ will contain nodes b, c, d, e and f.

3.2. Proposed method

Given an undirected $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we assume that any node in $\mathcal{N}(u)^L$ can transmit a piece of information to its neighbors at rate equal to. If the node u transmits information to$w$and there is a path of length l joining u and w, the probability of successful transmission is μ^l, and the probability of failure is $1 - \mu^l$.

Suppose that multiple paths between nodes p and q exist: transmission failure is possible if and only if all paths of any arbitrary lengths running from p to q fail to transmit information. If we assume that the propagation of information along different paths can be treated as independent events, we have that the probability $UN\_PROPA(p, q)$ the node p will never send a piece of information to the node q can be expressed as:

$$UN\_PROPA\,(p,q) = \prod_{l=1}^{L}\left(1-\mu^l\right)^{|\mathcal{P}(p,q)^l|} \tag{2}$$

where $|\ \mathcal{P}(u,w)^l\ |$ represents the number of paths of length l between p and q (there is no self-loop in the path), and the longest propagation path L is a parameter to limit the length of the longest path involved in the calculation. If the shortest path length between p and q is greater than L, we set $UN\_PROPA(p, q) = 1$.

The probability $PROPA(p, q)$ of successful transmission of information from node p to node q, propagation value ($PROPA$) can be expressed as:

$$PROPA\,(p,q) = 1 - \prod_{l=1}^{L}\left(1-\mu^l\right)^{|\mathcal{P}(p,q)^l|} \tag{3}$$

We propose a novel node scoring scheme and, more precisely, the score of an arbitrary node u depends on its ability of successfully transmitting a piece of information to any node w that can be reached by u through a shortest path whose length is bounded above by L. In other words, we compute the score of a node u as follows:

$$Score(u) = \sum_{w\in\mathcal{N}(u)^L}(PROPA\,(u,w)) \tag{4}$$

where $\mathcal{N}(u)^L$ is the set of nodes that can be reached by a path with length L and starting node u.

Here, a toy-network is used to explain the score of a node, as shown in Fig. 2. The toy-network has eight nodes and eight edges. We wish to compute the score of the Node 1 and, to this end, we set $L = 3$, and$\mu = 0.2$.

Observe that $\mathcal{N}(1)^3 = \{2, 3, 4, 5, 6, 7, 8\}$, we determine the paths from the Node 1 to the nodes in $\mathcal{N}(1)^3$. Table 2 gives the paths between Nodes 1 to $\mathcal{N}(1)^3$, and the $PROPA$ value. Here are two examples to illustrate the diversity of paths. For example, from Nodes 1 to 3, there are two paths of Lengths 1 and 3, $\mathcal{P}(1, 3)^1 = \{1, 3\}$and $\mathcal{P}(1, 3)^3 = \{1, 4, 6, 3\}$. Using Equation (3) to get the $PROPA$ value of Nodes 1–3 as: $PROPA(1, 3) = 1 - \prod_{l=1}^{3}(1 - 0.2^l)^{|\mathcal{P}(1,3)^l|} = 1 - (1 - 0.2^1)^1 \times (1 - 0.2^2)^0 \times (1 - 0.2^3)^1 = 0.2064$.

Another example, we have that $\mathcal{P}(1, 6)^2 = \{1, 3, 6\}$ and $\mathcal{P}(1, 6)^2 = \{1, 4, 6\}$ are two paths with the same length $l = 2$ from Nodes 1 to 6. Then we have $|\ \mathcal{P}(1, 6)^2\ | = 2$. Again, using Equation (3), we obtain$PROPA(1, 6) = 1 -$

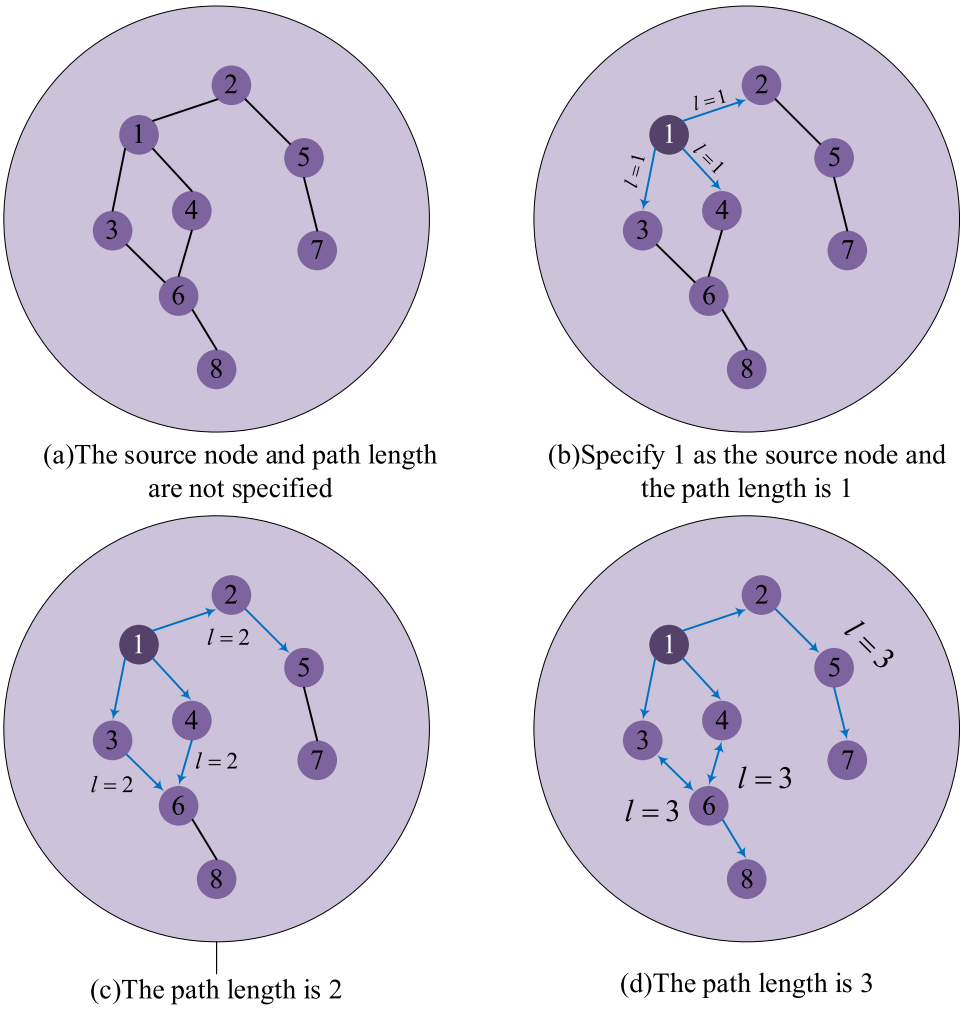

FIGURE 2. Changes of nodes that can be reached when toy-network and path length $l = 1, 2, 3$, respectively.

$\prod_{l=1}^{3}(1-0.2^l)^{|\mathcal{P}(1,3)^l|} = 1-(1-0.2^1)^0 \times (1-0.2^2)^2 \times (1-0.2^3)^0 = 0.0080$.

Finally, the centrality score of Node 1 is calculated using Equation (4) and we obtain $Score(1) = 0.7551$.

3.3. Computational analysis

We conclude this section by analyzing the computational complexity of our approach.

The most computationally intensive part of our approach is given by the computation of the Path Reachable Node Sets. To this end, we run a Breadth First Search (BFS) from a starting node u. The BFS algorithm explores the input graph, and it generates a tree rooted at u; if we assume that the input network is unweighted then the BFS algorithm returns the shortest paths from the root node u to any other node in the graph that can be reached by u (i.e. for which there exists a path running from u to that node). Specifically, nodes in the level l of the tree constructed by the BFS correspond to nodes, which can be reached from u through the shortest path of length equal to l. Consequently, the Path Reachable Node Set $N(u)^l$ is formed by the nodes in the subtree of the BFS tree, which is rooted in the node u and ends in the lth level.

Observe that no union operation between sets must be performed in practice: in fact, if a node x occurs at the level l, then this means that we cannot reach x from u in less than l-hops (otherwise, the length of the shortest path connecting x to u would be less than l and then x would appear in one of the previous levels of the BFS tree). Because of such an observation, we have that the computational cost of constructing Path Reachable Node Sets is equivalent to the cost of running a BFS search on a graph with n nodes and e edges.

The time required for constructing the BFS tree is proportional to the graph size and, thus, it amounts to $O(e)$. Because of we need to find all Path Reachable Node Sets for all nodes u in the graph, we arrive at an overall computational complexity

ALGORITHM 1. Proposed IR core algorithm.

Input: Network $\mathcal{G}$; information transmission probability μ; path length L
Output: The *SSP* records a node to the specified path length of the nodes successfully pass the information probability
1 Initialize the *SSP* dictionary
2 Initialize the *Paths* dictionary to record the number of Paths from node u to $\mathcal{N}(u)^L$
3 Initialize the temporary path set *tempPath* where each entry is a path of length L
4 *for* $l = 1;\ l \leq L; l++$ do
5 $npthas = []$
6 *for path* in *tempPath* do
7 *for* neighbor in neighbors of the last node in *path* do
8 if neighbor not in *path* then
9 add neighbor to *npaths*
10 if neighbor not in *Paths.keys*() then
11 $Paths[neighbor] = [0]_{1\times L}$
12 $Paths[neighbor][l]+ = 1$
13 *for* key in *Paths.keys*()do
14 Equation (2)
15 SSP[Key] = Equation (3)
16 PROPA = Equation (4)

TABLE 2. Toy-network selects Node 1 as the source node, path length of IR $L = 3$, and probability of information transmission success can be adopted by $\mu = 0.2$

Node	Paths node set	*PROPA* value
2	(1,2)	0.2000
3	(1,3), (1,4,6,3)	0.2064
4	(1,4), (1,3,6,4)	0.2064
5	(1,2,5)	0.0400
6	(1,3,6), (1,4,6)	0.0784
7	(1,2,5,7)	0.0080
8	(1,3,6,8), (1,4,6,8)	0.0159

Propagation (*PROPA*) value that Node 1 can reach at path length.

equal to $O(ne)$. If we assume our input graph is sparse, then e is in the same order of magnitude of n and, thus, the complexity is quadratic in the number of nodes.

4. EXPERIMENTS ANALYSIS

This section details the experiments we carried out to validate our new centrality index.

In Section 4.1, we illustrate the procedures/metrics we adopted to measure the effectiveness of our approach, whereas Section 4.2 describes some benchmark methods. In Section 4.3, we describe the datasets used in our evaluation. In Section 4.4, we investigate whether changing the parameters of our method has an effect on the correlation between the resulting rankings and those generated by infection propagation in simulated networks (modeled by the SIR model), whereas in Section 4.5 we compare our method with benchmarks.

4.1. Evaluation index

(i) SIR spreading model

A first procedure to assess the effectiveness of a centrality index consists of simulating the spread of an infection across a network. In this way, nodes can be ranked on the basis of the number of the network nodes they are actually able to infect.

The most commonly used model in epidemiological studies is the *SIR model*, where 'S' represents susceptible node, 'I' denotes infected node and 'R' denotes a recovered node. In the process of virus transmission, an infected node in complex network infects the susceptible node in the neighbor node with probability and the infected node itself becomes the recovered node with probability β_r. A recovered node will not be infected again. If the SIR model is used to measure the importance of nodes, one node is taken as infected and the other nodes are regarded as susceptible during initialization. When there are no infected in the complex network, the number of removed nodes can be used to describe the infection capacity of the node, and, thus, its importance.

In this paper, the SIR model is simulated 1000 times, and N_r represents the number of recovered nodes. When there are no more infectors in the network, the *infection capacity* of a node u is defined as the ratio of the number of recovered nodes N_r by the total number of nodes N in the network.

In the following we interpret the infection capacity as a measure of importance of u.

(ii) Kendall τ coefficient

The Kendall τ coefficient is used to evaluate the correlation between the ranking generated by an arbitrary node ranking method and the node ranking obtained from the SIR model we introduced above. More formally, the Kendall τ coefficient is defined as follows:

$$\tau\,(X, Y) = \frac{n_c - n_d}{\sqrt{(n_0 - n_1)\,(n_0 - n_2)}} \tag{5}$$

where X and Y are two different sorting sequences respectively, and elements are taken out from$X$ and Y, respectively in order to combine them into a joint observation value in the form of $(x_1, y_1), \cdots, (x_i, y_i), \cdots, (x_n, y_n)$. Here, n_c represents the concordant number of pairs in the combined observations. (x_i, y_i) and (x_j, y_j) are a pair of joint observations, when the order between the elements in the same position is the same, it is called concordant. Otherwise, it is discordant, and at the same time, it is concordant and discordant.

The closer the Kendall τ coefficient is to 1, the stronger the correlation between the two rankings is, and in this paper, the better the ranking method is.

4.2. Benchmark methods

We compared our centrality index with the following competitors:

(i) *Degree Centrality (DC)* describes the direct influence of a nodes. The degree centrality of a node is the number of edges incident on that node normalized by the total number of network nodes, and it is defined as follows:

$$DC(u) = \frac{k_u}{n-1} \tag{6}$$

where u is the degree of nodek_u.

(ii) *Betweenness Centrality (BC)* describes the ability of nodes to control-network traffic along the shortest path in a network. The betweenness of a node u is defined as:

$$BC(u) = \sum_{u \neq s, u \neq t, s \neq t} \frac{g_{st}^u}{g_{st}} \tag{7}$$

where g_{st} represents the number of all shortest paths from node s to t, and g_{st}^u denotes the number of shortest paths from node s to t through u in g_{st}.

(iii) K-shell, a centrality index based on the notion of K-core in a network. Specifically, The K-core is the largest subgraph with nodes of degree K or more. The number of cores of a node is K, the maximum number of k-cores containing the node.

(iv) *DynamicRank,* which is defined as:

$$Rank(u) = \sum_{i=1}^{3} \sum_{v \in \Gamma_i(u)} score\,(v, i) \tag{8}$$

Here node v represents any one of the third-order neighbor nodes, $score(v, i)$ is calculated by calculating the sum of the infection probability between node v and nodes in its third-order neighborhood.

(v) *PageRank (PR)* is a very popular method to rank Web pages. In *PR*, each node (web page) initially receives the same *PR* value; such a *PR* value is updated in an iterative fashion by assigning to each node the *PR* of the nodes to which such a node points to until convergence is achieved. The Equation to compute the *PR* of a node is as follows:

$$PR_u(t) = \sum_{v=1}^{n} a_{vu} \frac{PR_v\,(t-1)}{k_v^{out}} \tag{9}$$

where the output degree of node v is denoted by k_v^{out}, when the *PR* value of each node does not change, the computing is stopped.

4.3. Datasets

(i) Oz [29]: Such a dataset describes friendship relationships among 217 students in a dormitory in a university campus in Australia (ii) Figeys [30]: Complex networks of protein interactions in human cells are presented using mass spectrometry-based methodologies for large-scale studies of protein interactions in human cells. (iii) Vidal [31]: In a study of human binary protein interactions, a proteomic scale map is drawn and a complex network was abstracted based on its initial version. (iv) Stelzl [32]:Networks represent interacting protein pairs in humans (*Homo sapiens*). (v) Jazz [33]: A collaborative network of jazz musicians. (vi) Arenas [34]:Metabolic network of *Caenorhabditis elegans*. (vii) Faa [35]: The route network is built from the preferred route database of the FAA's (Federal Aviation Administration) National Flight Data Center (NFDC). (viii) Email [36]: A complex web of E-mail communications from the Rovira I Virgili University in Tarragona, southern Catalonia, Spain. (ix) Facebook [37]: The social friendship network obtained by selecting certain tags in Facebook.

Table 3 presents some attribute numerical indicators of the nine datasets, which are commonly used in complex networks to describe the network topology and degree distribution.

Specifically, we reported the number n of nodes, the number e of edges, the average degree $< k >$, the largest degree k_{max}, the clustering coefficient c and the degree heterogeneity H.

We recall that the clustering coefficient measures the tendency of nodes in a network to form triangles: a triangle is a 3-tuple (a, b, c) such that there is an edge from a to b, from b to c and from c to a.

The degree heterogeneity H is defined as the ratio of the average of the square of node degrees (i.e. $< d^2 >$) to the square of the average degree $< d >^2$. The degree heterogeneity measures the un-balancement in node degree distribution: specifically, larger values of H imply the existence of few nodes collecting most of the network edges.

4.4. Parameter analysis

Our node ranking method depends on two parameters namely path length L and probability of propagation μ. In our experiments we considered also different values of the β_i, which specifies the recovery rate in the in SIR model.

Firstly, the infection rate of SIR model set as$\beta_i = 1.5$, the path length L varied between 1 and 5, and the information transmission probabilityμincreased from 0.05 to 0.5 each time. Figure 3 shows the experimental results of parameter analysis.

TABLE 3. Nine basic properties of real-world networks

Data	n	e	$< k >$	k_{max}	c	H
Oz	217	1839	16.9	56	0.3628	1.2165
Figeys	2239	6432	5.7	314	0.04	9.9034
Vidal	3023	6149	4.1	129	0.0658	3.7373
Stelzl	1702	3155	3.7	95	0.006	4.5557
Jazz	198	2742	27.7	100	0.6175	1.3948
Arenas	453	2025	8.9	237	0.6465	4.5258
Faa	1226	2408	3.9	34	0.0675	1.9
Email	1133	5451	9.6	71	0.2202	1.9511
Facebook	2888	2981	2.1	769	0.0272	119.7577

The total number of nodes is denoted by n. The total number of edges is denoted by$e$. $< k >$ represents the average degree of network. k_{max} represents the maximum degree of network. The average clustering coefficient of network is denoted by $< c >$. $H$is degree heterogeneity, $H =< k^2 > / < k >^2$.

In Fig. 3 we report the Kendall τ coefficient between the nodes ranking generated by our model and the one obtained in SIR simulations with. The optimal values of L and μ, i.e. the values allowing to achieve the highest level of agreement between our method and the SIR model, vary from dataset to dataset.

Figure 3 indicates that the path length L and information transmission probability μ of IR are different when it achieves the best performance on different datasets.

In Oz dataset, IR has the best performance when the path length of IR is 3, 4 or 5 and the probability of success of information transmission μ is between 0.05 and 0.15. In Fiegys dataset, we get the best results if the path length is 3 and μ is between 0.05 and 0.15. It is desirable that the Vidal data set has a wider range of information transmission probability. When the path length is 4 or 5, the value range of μ is within $0.1 \sim 0.3$, and the Kendall τ coefficient of IR is the best.

Stelzl, Jazz and Arenas datasets all had the best performance when the path length is 3, 4 or 5. However, the three data sets had different probability ranges of success of information transmission, which were 0.1–0.25, 0.05–0.1 and 0.05–0.2, respectively. Faa dataset has the best IR performance when IR with fixed path length $L = 5$ and probability of success of information transmission μ is between 025 and 0.5. The range of IR performance in the Email data set is the best when the path length is 3, 4 or 5, and the value μ is 0.1–0.35. The Kendall τ coefficient of IR on the other eight data sets is better than that of Facebook data set. From our experiments we concluded that the best option for the path length in Facebook dataset is 2 or 3. Any value of μ between 0.05 and 0.5 guarantees the largest correlation when L is 2, and μ between 0.05 and 0.15 is the best performance when L is 3.

From the experiment above we observe that, in general, the best combination of parameters was $L = 3, \mu = 0.15$.

Then, the IR parameter is fixed at $L = 3, \mu = 0.15$, and compared with benchmarks; in addition, the infection rateβ_i varied between 1.0 and 2.0 with 0.1 step. The experimental results are shown in Table 4.

As can be seen from Table 4, the Kendall τ coefficients of IR and the other five methods in Oz, Fiegys, Vidal, Stelzl, Jazz, Arenas and Email datasets do not change significantly if the infection rate changes too. In the Faa data set, IR and the other five methods are observed to decrease between 1.0 and 1.3, and then the performance gradually improved. In the Facebook data set, IR and PR kept getting better with the increase of infection rate β_i, whereas the other four methods do not change much.

4.5. Comparison with the benchmark methods

In Section 4.4, we showed that the IR method is not sensitive to the change of infection rate β_i of SIR model. Therefore, in this Section, we set the infection rate $\beta_i = 1.5$ of the fixed SIR model to rank the benchmark facts, so as to compare the rankings obtained by IR and the other five benchmark methods with their Kendall τ correlations to show the performance of IR.

Table 5 shows results that with choices of best parameters L and μ, we compute the Kendall's τ correlation coefficient between the ranking generated by our method and the node ranking generated by each of the five benchmark methods introduced in Section 4.2. The highest level of agreement was observed in the Jazz dataset, whereas the lowest level of agreement occurred in the Facebook dataset.

In Fig. 4, we plotted how the Kendall's correlation coefficient varies as function of the infection rate β_i. In this experiment, we assumed that the infection rate varied from 1.0 to 2.0 and we computed the corresponding Kendall τ coefficient between the ranking generated by one of the methods under inquiry and the ranking obtained by running SIR simulation. Such an experiment was repeated for all nine input datasets. Figure 4 clearly reports our IR method (associated with the red line) is consistently above all the methods under inquiry and, thus, this indicates the superiority of our approach against its competitors.

We observe (see Table 6) that the IR and Dynamic Rank (DR) methods showcase the highest values of correlation on

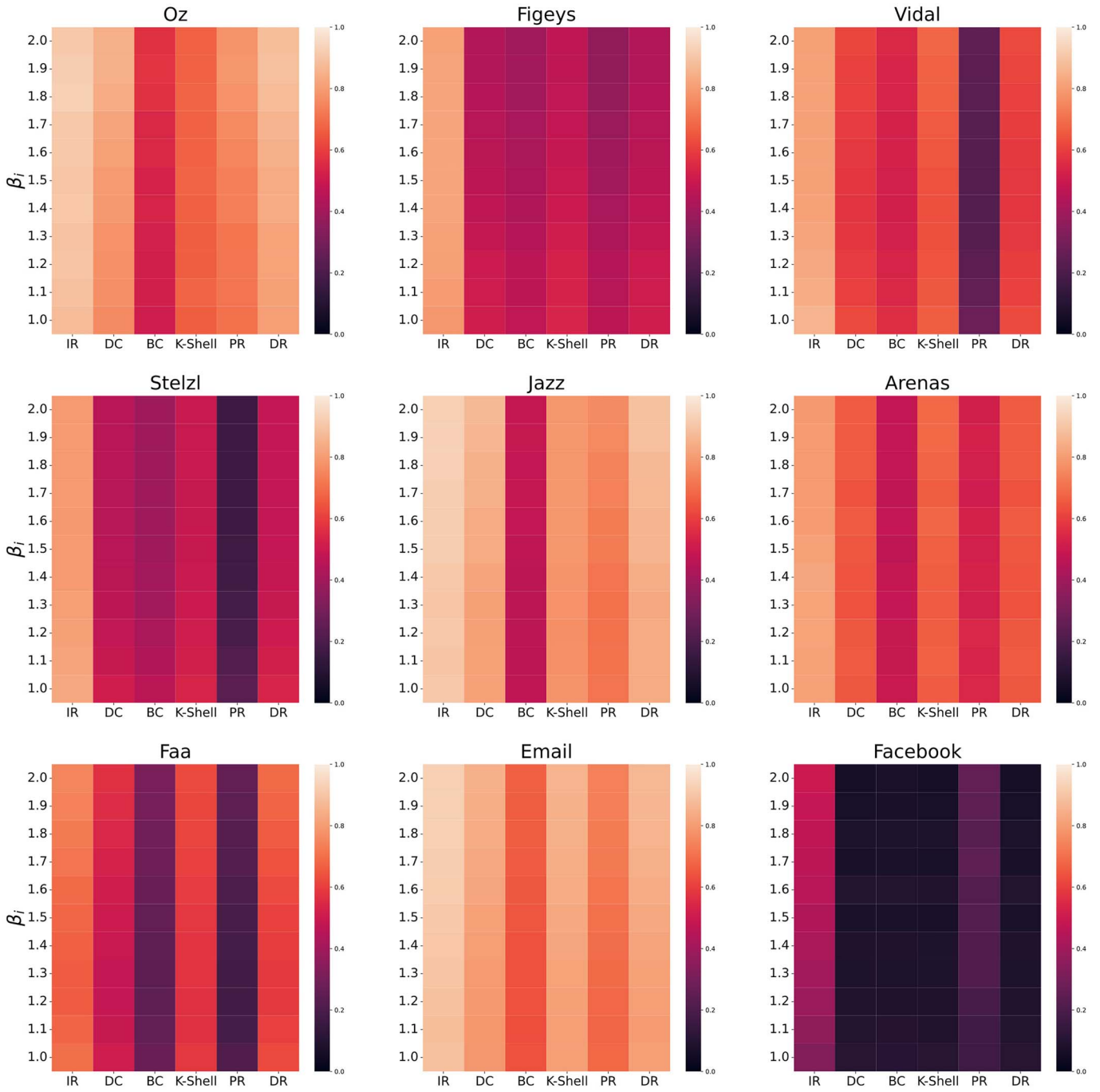

FIGURE 3. The heatmap shows the Kendall τ correlation between the ranking obtained by varying the two parameters L and μ of the IR method and the ranking obtained by the SIR simulation with fixed infection rate $\beta_i = 1.5$.

Oz, Jazz and Email dataset (here the Kendall τ coefficient ranges from 0.81 and 0.84).

In Figeys, Vidal, Stelzl and Arenas the IR method displays the highest level of correlation with the K-shell method and reported values oscillate between 0.508 and 0.663.

The lowest level of the Kendall τ coefficient occurs in the Facebook dataset: here, the method displaying the highest correlation with IR is the PageRank (PR) but the observed Kendall τ coefficient is only equal to 0.41.

Results arising from Table 6 are consistent with experimental data reported in Table 7. Here, we reported the Kendall τ coefficient between the ranking generated by each of the methods under inquiry and the node ranking deriving from the SIR simulations. The IR algorithm always displays the highest

TABLE 4. Kendall τ coefficient between the ranking obtained by changing the SIR model infection rate and the ranking obtained by IR and its five methods

Datasets	β_i Methods	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
Oz	IR	0.877	0.884	0.897	0.893	0.903	0.901	0.909	0.909	0.926	0.918	0.908
	DC	0.760	0.767	0.771	0.774	0.792	0.794	0.811	0.818	0.837	0.847	0.844
	BC	0.512	0.517	0.516	0.523	0.535	0.531	0.544	0.547	0.568	0.572	0.567
	K-Shell	0.662	0.675	0.662	0.662	0.665	0.674	0.670	0.672	0.675	0.669	0.671
	PR	0.701	0.707	0.708	0.713	0.732	0.731	0.749	0.757	0.774	0.783	0.780
	DR	0.803	0.807	0.814	0.816	0.834	0.834	0.851	0.857	0.875	0.885	0.883
Figeys	IR	0.789	0.798	0.805	0.810	0.818	0.817	0.820	0.820	0.819	0.817	0.811
	DC	0.515	0.507	0.501	0.486	0.472	0.467	0.458	0.451	0.444	0.439	0.438
	BC	0.474	0.467	0.461	0.448	0.433	0.428	0.420	0.415	0.407	0.403	0.402
	K-Shell	0.537	0.530	0.524	0.512	0.499	0.495	0.487	0.480	0.475	0.470	0.469
	PR	0.468	0.450	0.444	0.426	0.414	0.405	0.394	0.386	0.374	0.367	0.364
	DR	0.515	0.507	0.501	0.486	0.472	0.467	0.458	0.451	0.444	0.439	0.438
Vidal	IR	0.853	0.839	0.827	0.817	0.812	0.809	0.808	0.808	0.809	0.810	0.808
	DC	0.621	0.603	0.591	0.579	0.576	0.580	0.585	0.590	0.601	0.606	0.615
	BC	0.560	0.544	0.534	0.523	0.518	0.521	0.524	0.526	0.534	0.537	0.545
	K-Shell	0.661	0.651	0.643	0.635	0.635	0.640	0.646	0.653	0.664	0.670	0.679
	PR	0.274	0.252	0.238	0.226	0.223	0.223	0.225	0.228	0.237	0.240	0.248
	DR	0.624	0.607	0.594	0.583	0.580	0.584	0.589	0.595	0.605	0.611	0.620
Stelzl	IR	0.831	0.819	0.807	0.802	0.795	0.793	0.792	0.795	0.793	0.794	0.793
	DC	0.513	0.491	0.476	0.465	0.459	0.455	0.451	0.452	0.455	0.455	0.455
	BC	0.457	0.435	0.421	0.410	0.403	0.398	0.394	0.395	0.397	0.397	0.397
	K-Shell	0.534	0.518	0.507	0.497	0.494	0.491	0.490	0.491	0.495	0.496	0.496
	PR	0.238	0.213	0.194	0.182	0.171	0.166	0.160	0.160	0.162	0.162	0.160
	DR	0.539	0.517	0.501	0.492	0.485	0.481	0.478	0.479	0.482	0.482	0.482
Jazz	IR	0.907	0.897	0.909	0.901	0.909	0.919	0.919	0.922	0.929	0.929	0.923
	DC	0.812	0.809	0.807	0.805	0.816	0.833	0.835	0.845	0.848	0.863	0.868
	BC	0.469	0.468	0.464	0.463	0.468	0.472	0.477	0.486	0.478	0.489	0.487
	K-Shell	0.777	0.772	0.769	0.774	0.778	0.782	0.782	0.782	0.785	0.785	0.785
	PR	0.707	0.704	0.703	0.700	0.710	0.725	0.725	0.737	0.740	0.756	0.759
	DR	0.834	0.834	0.833	0.830	0.840	0.857	0.857	0.867	0.869	0.885	0.889
Are-	IR	0.812	0.811	0.816	0.816	0.820	0.812	0.798	0.796	0.794	0.795	0.790
nas	DC	0.649	0.654	0.650	0.640	0.644	0.645	0.653	0.640	0.654	0.656	0.656
	BC	0.496	0.492	0.491	0.481	0.484	0.473	0.482	0.470	0.479	0.484	0.480
	K-Shell	0.657	0.670	0.663	0.659	0.660	0.666	0.676	0.664	0.676	0.685	0.687
	PR	0.542	0.538	0.542	0.522	0.529	0.521	0.526	0.511	0.522	0.524	0.519
	DR	0.649	0.654	0.650	0.641	0.644	0.645	0.653	0.640	0.654	0.656	0.656
Faa	IR	0.697	0.672	0.659	0.657	0.668	0.679	0.690	0.708	0.721	0.736	0.748
	DC	0.511	0.491	0.484	0.484	0.492	0.503	0.513	0.528	0.539	0.556	0.568
	BC	0.272	0.252	0.244	0.241	0.249	0.257	0.266	0.278	0.286	0.301	0.310
	K-Shell	0.584	0.570	0.567	0.571	0.578	0.585	0.592	0.602	0.610	0.616	0.623
	PR	0.209	0.187	0.179	0.177	0.184	0.192	0.201	0.216	0.226	0.243	0.255
	DR	0.622	0.598	0.586	0.586	0.597	0.609	0.621	0.640	0.654	0.674	0.688
Email	IR	0.888	0.881	0.890	0.896	0.904	0.909	0.915	0.920	0.923	0.926	0.924
	DC	0.789	0.784	0.789	0.796	0.804	0.811	0.822	0.831	0.838	0.843	0.849
	BC	0.639	0.632	0.637	0.638	0.641	0.645	0.651	0.656	0.661	0.663	0.664
	K-Shell	0.804	0.803	0.807	0.816	0.824	0.828	0.837	0.843	0.847	0.851	0.854
	PR	0.685	0.679	0.683	0.688	0.695	0.701	0.711	0.720	0.727	0.731	0.737
	DR	0.804	0.798	0.804	0.811	0.819	0.827	0.838	0.847	0.854	0.860	0.866

(*Continued*)

TABLE 4. Continued

Datasets	β_i Methods	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
Face-	IR	0.332	0.362	0.386	0.404	0.422	0.435	0.454	0.464	0.471	0.481	0.500
book	DC	0.105	0.090	0.087	0.081	0.084	0.071	0.090	0.067	0.078	0.066	0.062
	BC	0.098	0.094	0.083	0.078	0.085	0.073	0.093	0.071	0.080	0.069	0.071
	K-Shell	0.103	0.089	0.085	0.079	0.082	0.069	0.089	0.066	0.076	0.065	0.060
	PR	0.165	0.171	0.188	0.211	0.210	0.222	0.214	0.246	0.232	0.243	0.254
	DR	0.105	0.090	0.087	0.081	0.084	0.071	0.090	0.067	0.078	0.066	0.062

TABLE 5. The combination of parameters when IR achieves the best performance in Kendallτcoefficients

Dataset	L	μ	Kendall τ coefficient
Oz	5	0.05	0.9293
Figeys	3	0.05	0.8749
Vidal	5	0.1	0.9132
Stelzl	5	0.1	0.9085
Jazz	4	0.05	**0.9444**
Arenas	4	0.05	0.8688
Faa	5	0.45	0.9093
Email	4	0.1	0.9359
Facebook	2	0.05	**0.3889**

TABLE 6. Proposed IR method and ranking obtained respectively by five other benchmark methods on nine networks

Data	IR-DC	IR-BC	IR-K-Shell	IR-PR	IR-DR
Oz	0.7938	0.5315	0.6684	0.7312	**0.8403**
Figeys	0.4796	0.4394	**0.5080**	0.4221	0.4796
Vidal	0.6095	0.5510	**0.6600**	0.2478	0.6128
Stelzl	0.4811	0.4255	**0.5108**	0.1903	0.5072
Jazz	0.8218	0.4673	0.7744	0.7127	**0.8477**
Arenas	0.6409	0.4761	**0.6634**	0.5171	0.6410
Faa	0.4717	0.2517	**0.5551**	0.1716	**0.5899**
Email	0.7994	0.6451	0.8153	0.6915	**0.8154**
Facebook	0.2529	0.2903	0.2519	**0.4121**	0.2529

correlation. In some cases, we highlight a relevant gap between the IR algorithm and the second best-performing method: for instance, in Fygeys the IR method has a correlation equal to 0.873, whereas the second best-performing method is the K-shell with a correlation equal to 0.498. Therefore, in Fygeys, the IR method is 1.75 times better than the second best-performing method.

All the methods under inquiry achieve their lowest correlation on Facebook but the IR method is still the best-performing one; for instance, its Kendall τ coefficient with the SIR method is 101.4% times higher than that displayed by the PR (which is the second best-performing method).

5. CONCLUSIONS

Key node identification is an important problem in network science area. A few key nodes, in fact, control the propagation dynamics in the whole network. Unlike previous literature, we propose a novel node ranking method called *Information Rank* (IR), which considers shortest paths along with their ability of

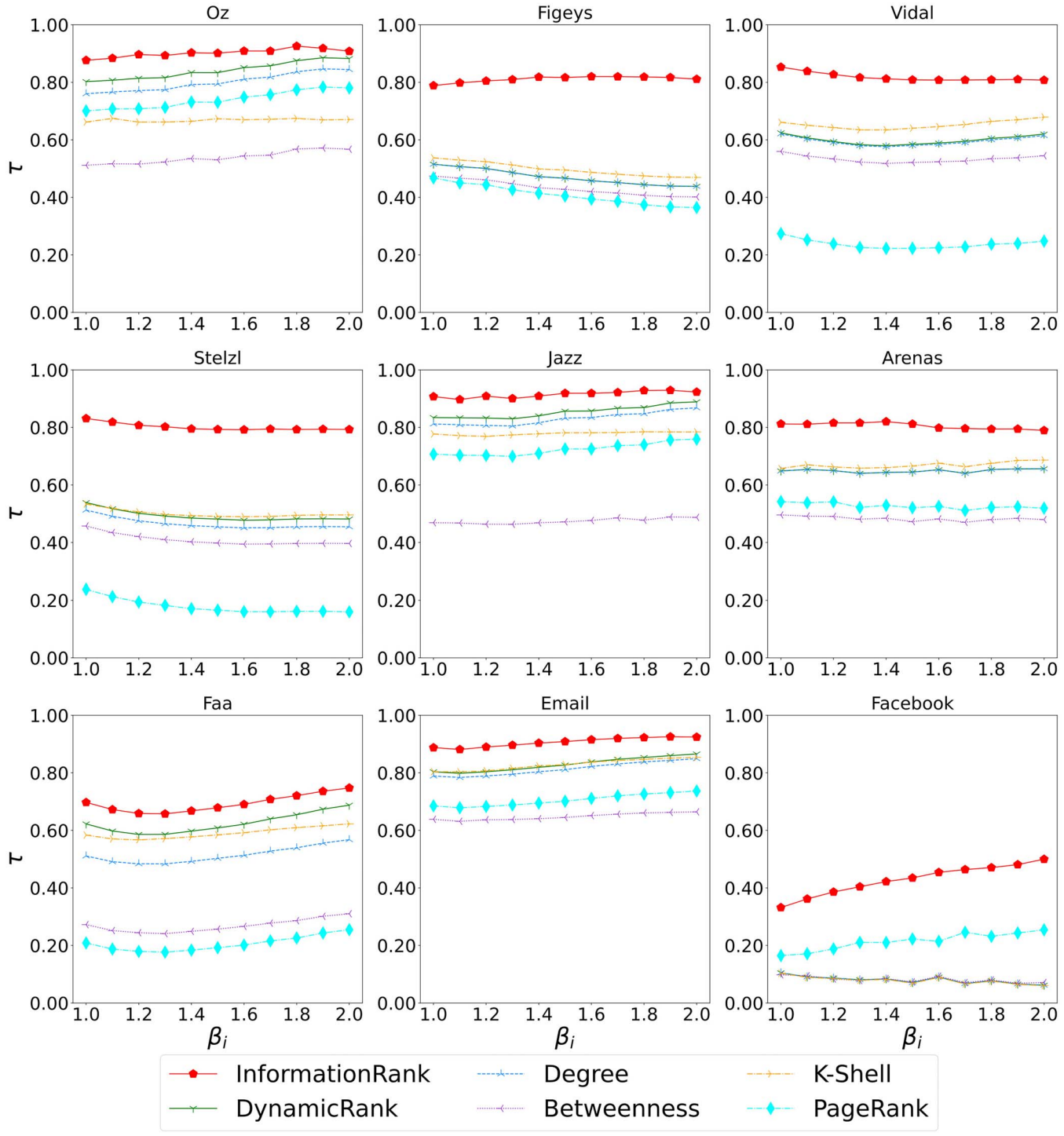

FIGURE 4. Comparison of Kendallτcoefficients between fixed IR parameters and five reference methods in different infection rates.

actually propagation information across a network to determine the centrality of a node. The rationality and validity of the IR method have been verified by comparative analysis on different datasets and with different benchmark methods.

We plan to extend our approach in several ways.

Firstly, the IR algorithm has been designed to operate on complex networks consisting of a single layer, but we are aware that many real-world systems are effectively modeled by means of multiplex/multilayer networks. Specifically, dynamic processes on networks such as the propagation of information

TABLE 7. The corresponding parameters in Section 4.4 were taken from IR in nine data sets, and the Kendall τ coefficients between the benchmark ranking of SIR model infection rate β_i varying from 1.0 to 2.0 with 0.1 and the ranking of IR and five benchmark methods were averaged

Data	IR	DC	BC	K-Shell	PR	DR
Oz	**0.9200**	0.8014	0.5391	0.6686	0.7395	**0.8418**
Figeys	**0.8731**	0.4707	0.4325	**0.4980**	0.4085	0.4707
Vidal	**0.9033**	0.5952	0.5333	**0.6525**	0.2376	0.5992
Stelzl	**0.9044**	0.4660	0.4094	**0.5007**	0.1788	0.4926
Jazz	**0.9402**	0.8310	0.4745	0.7791	0.7241	**0.8541**
Arenas	**0.8771**	0.6491	0.4829	**0.6693**	0.5270	0.6492
Faa	**0.8840**	0.5154	0.2688	**0.5906**	0.2062	**0.6251**
Email	**0.9293**	0.8143	0.6478	0.8287	0.7053	**0.8298**
Facebook	**0.4314**	0.0799	0.0813	0.0783	**0.2142**	0.0799

For each dataset, we highlighted in bold, the highest and the second highest Kendall's τ coefficient.

as well as the spread of an infection in human societies cannot be restricted to a single mode of transmission: it is well known, for instance, that most viruses spread between people who are at conversational distance, or it can be transmitted to individuals who touched their eyes/noses after touching surfaces, which have been contaminated by the virus. Therefore, multiple type of interactions is to be considered and each of these interactions corresponds to a specific layer of a multiplex/multilayer network. There is a broad range of centrality metrics defined for multiplex/multilayer networks (think of Novel Multiplex PageRank in Multilayer Networks [38,39], Random Walk Centrality in interconnected multilayer networks [40], Multiplex PageRank, Random Walks on Multiplex Networks [41], the Core Link Community Detection (CLCD) [42] and Giant Congenital Melanocytic Nevus (GCMN) [43] methods to cite a few) and we wish to extend our approach to multiplex/multilayer networks as well as to compare it against the methods we cited above.

As a further extension, we plan to design approximate versions of the IR method capable of computing centralities in very large networks. To this end, we observe that an intense research work has been done to find shortest paths in graphs and we wish to consider the most recent findings to get significant computational savings [44].

SUPPLEMENTARY MATERIAL

Supplementary material is available at www.comjnl.oxfordjournals.org.

FUNDING

Science and Technology Research Project of Chongqing Municipal Education Commission (KJZD-K202001101), Graduate Innovation Fund of Chongqing University of Technology under Grant (gzlcx20222054), General Project of Chongqing Natural Science Foundation (cstc2021jcyj-msxmX0162), 2021 National Education Examination Research Project (GJK2021028) and 2020 Chongqing Municipal Human Resources and Social Security Bureau of Innovation Project for Returned Overseas Person (cx2020031).

AUTHORS' CONTRIBUTIONS

Xiaoyang Liu was involved in methodology, formal analysis and data curation; Luyuan Gao took the responsibility of conceptualization, software, methodology, validation, data curation and writing—review & editing; Giacomo Fiumara and Pasquale De Meo were involved in methodology and writing—review & editing.

DATA AVAILABILITY

The data underlying this article will be shared on reasonable request to the corresponding author.

CONFLICT OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] Yao, X., Gu, Y., Gu, C. and Huang, H. (2022) Fast controlling of rumors with limited cost in social networks. *Comput. Commun.*, 182, 41–51.
[2] Paci, P. and Fiscon, G. (2022) SWIMmeR An R-based software to unveiling crucial nodes in complex biological networks. *Bioinformatics*, 38, 586–588.
[3] Yang, F., Yan, F., Zhang, C., Tang, X., Li, J., Zhang, X. and Gan, Y. (2021) Applying the virtual input-output method to the identification of key nodes in busy traffic network. *Complexity*, 2021, 1–7.
[4] Xu, M. *et al.* (2018) Discovery of critical nodes in road networks through mining from vehicle trajectories. *IEEE Trans. Intell. Transp. Syst.*, 20, 583–593.

[5] Amini, H., Filipovic, D. and Minca, A. (2020) Systemic risk in networks with a central node. *SIAM J. Financ. Math.*, 11, 60–98.
[6] Ding, J. *et al.* (2019) Key nodes selection in controlling complex networks via convex optimization. *IEEE Trans. Cybern.*, 51, 52–63.
[7] Wang, N., Gao, Y., He, J. and Yang, J. (2022) Robustness evaluation of the air cargo network considering node importance and attack cost. *Reliab. Eng. Syst. Saf.*, 217, 108026.
[8] Zou, Q., Li, Y., Yang, X. and Zhou, Z. (2021) Identification of key nodes in directed network based on overlapping community structure. *Autom. Control Comput. Sci.*, 55, 167–176.
[9] Liu, B. *et al.* (2017) Recognition and vulnerability analysis of key nodes in power grid based on complex network centrality. *IEEE Trans. Circuits Syst. II*, 65, 346–350.
[10] Gupta, M. and Mishra, R. (2021) Spreading the information in complex networks: identifying a set of top-N influential nodes using network structure. *Decis. Supp. Syst.*, 149, 113608.
[11] Centola, D. (2010) The spread of behavior in an online social network experiment. *Science*, 329, 1194–1197.
[12] Nuñez, S., Inthamoussou, F.A., Valenciaga, F., de Battista, H. and Garelli, F. (2021) Potentials of constrained sliding mode control as an intervention guide to manage COVID19 spread. *Biomed. Signal Process. Control*, 67, 102557.
[13] Ullah, A., Sheng, J., Long, J., Long, J. and Khan, N. (2021) Identification of influential nodes via effective distance-based centrality mechanism in complex networks. *Complexity*, 2021, 1–16.
[14] Newman, M. (2018) *Networks: An Introduction*. Oxford University Press.
[15] Katz, L. (1953) A new status index derived from sociometric analysis. *Psychometrika*, 18, 39–43.
[16] Bonacich, P. (1972) Factoring and weighting approaches to status scores and clique identification. *J. Math. Sociol.*, 2, 113–120.
[17] Kitsak, M., Gallos, L.K., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H.E. and Makse, H.A. (2010) Identification of influential spreaders in complex networks. *Nat. Phys.*, 6, 888–893.
[18] Cavallaro, L., Costantini, S., De Meo, P., Liotta, A. and Stilo, G. (2022) Network connectivity under a probabilistic node failure model. *IEEE Trans. Network Sci. Eng.*, 9, 2463–2480.
[19] De Meo, P., Musial-Gabrys, K., Rosaci, D., Sarnè, G.M.L. and Aroyo, L. (2017) Using centrality measures to predict helpfulness-based reputation in trust networks. *ACM Trans. Internet Technol.*, 17, 1–20.
[20] Carmi, S., Havlin, S., Kirkpatrick, S., Shavitt, Y., Shir, E. and E. (2007) A model of Internet topology using k-shell decomposition. *Proc. Natl. Acad. Sci.*, 104, 11150–11154.
[21] Hage, P. and Harary, F. (1995) Eccentricity and centrality in networks. *Soc. Networks*, 17, 57–63.
[22] Freeman, L.C. (1978) Centrality in social networks conceptual clarification. *Soc. Networks*, 1, 215–239.
[23] Sabidussi, G. (1966) The centrality index of a graph. *Psychometrika*, 31, 581–603.
[24] Page, L., Brin, S., Motwani, R. and Winograd, T. (1999) *The PageRank Citation Ranking: Bringing Order to the Web*. Stanford InfoLab.
[25] Lü, L., Zhang, Y.C., Yeung, C.H. and Zhou, T. (2011) Leaders in social networks, the delicious case. *PloS one*, 6, e21202.
[26] Brin, S. and Page, L. (1998) The anatomy of a large-scale hypertextual web search engine. *Comput. Networks ISDN Syst.*, 30, 107–117.
[27] Bao, Z.K., Ma, C., Xiang, B.B. and Zhang, H.F. (2017) Identification of influential nodes in complex networks: method from spreading probability viewpoint. *Physica A*, 468, 391–397.
[28] Chen, D.B., Sun, H.L., Tang, Q., Tian, S.Z. and Xie, M. (2019) Identifying influential spreaders in complex networks by propagation probability dynamics. Chaos: an interdisciplinary. *J. Nonlinear Sci.*, 29, 033120.
[29] Freeman, L.C., Webster, C.M. and Kirke, D.M. (1998) Exploring social structure using dynamic three-dimensional color images. *Social Networks*, 20, 109–118.
[30] Ewing, R.M. *et al.* (2007) Large-scale mapping of human protein–protein interactions by mass spectrometry. *Mol. Syst. Biol.*, 3, 89–17.
[31] Rual, J.F. *et al.* (2005) Towards a proteome-scale map of the human protein–protein interaction network. *Nature*, 437, 1173–1178.
[32] Stelzl, U. *et al.* (2005) A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122, 957–968.
[33] Gleiser, P.M. and Danon, L. (2003) (2003) Community structure in jazz. *Adv Complex Syst.*, 6, 565–573.
[34] Duch, J. and Arenas, A. (2005) Community detection in complex networks using extremal optimization. *Phys. Rev. E*, 72, 027104.
[35] *Air Traffic Control-Network Analysis of Air Traffic Control-KONECT*. http://konect.uni- koblenz.de/networks/maayan-faa.
[36] Guimerà, R., Danon, L., Díaz-Guilera, A., Giralt, F. and Arenas, A. (2003) Self-similar community structure in a network of human interactions. *Phys. Rev. E*, 68, 065103.
[37] Traud, A.L., Kelsic, E.D., Mucha, P.J. and Porter, M.A. (2011) Comparing community structure to characteristics in online collegiate social networks. *SIAM Rev.*, 53, 526–543.
[38] Tu, X., Jiang, G.P., Song, Y. and Zhang, X. (2018) Novel multiplex PageRank in multilayer networks. *IEEE Access*, 6, 12530–12538.
[39] Halu, A., Mondragón, R.J., Panzarasa, P. and Bianconi, G. (2013) Multiplex pagerank. *PloS one*, 8, e78293.
[40] Solé-Ribalta, A., De Domenico, M., Gómez, S. and Arenas, A. (2016) Random walk centrality in interconnected multilayer networks. *Phys. D*, 323-324, 73–79.
[41] Interdonato, R., Tagarelli, A., Ienco, D., Sallaberry, A. and Poncelet, P. (2017) Local community detection in multilayer networks. *Data Min Knowl Discovery*, 31, 1444–1479.
[42] De Figueirêdo, B.C.B., Nakamura, F.G. and Nakamura, E.F. (2021) (2021) A group-based centrality for undirected multiplex networks: a case study of the Brazilian Car Wash Operation. *IEEE Access*, 9, 81946–81956.
[43] Tortosa, L., Vicent, J.F. and Yeghikyan, G. (2021) (2021) An algorithm for ranking the nodes of multiplex networks with data based on the PageRank concept. *Appl. Math Comput.*, 392, 125676.
[44] Sommer, C. (2014) Shortest-path queries in static networks. *ACM Comput. Surv. (CSUR)*, 46, 1–31.