Applied Network Science

RESEARCH Open Access

Compressive closeness in networks

Hamidreza Mahyar[1,3*†], Rouzbeh Hasheminezhad[2†] and H Eugene Stanley[1]

*Correspondence:
hmahyar@bu.edu
†Hamidreza Mahyar and Rouzbeh Hasheminezhad contributed equally to this work.
[1]Boston University, Boston, USA
[3]TU Wien, Vienna, Austria
Full list of author information is available at the end of the article

Abstract

Distributed algorithms for network science applications are of great importance due to today's large real-world networks. In such algorithms, a node is allowed only to have local interactions with its immediate neighbors; because the whole network topological structure is often unknown to each node. Recently, distributed detection of central nodes, concerning different notions of importance, within a network has received much attention. Closeness centrality is a prominent measure to evaluate the importance (influence) of nodes, based on their accessibility, in a given network. In this paper, first, we introduce a local (ego-centric) metric that correlates well with the global closeness centrality; however, it has very low computational complexity. Second, we propose a compressive sensing (CS)-based framework to accurately recover high closeness centrality nodes in the network utilizing the proposed local metric. Both ego-centric metric computation and its aggregation via CS are efficient and distributed, using only local interactions between neighboring nodes. Finally, we evaluate the performance of the proposed method through extensive experiments on various synthetic and real-world networks. The results show that the proposed local metric correlates with the global closeness centrality, better than the current local metrics. Moreover, the results demonstrate that the proposed CS-based method outperforms state-of-the-art methods with notable improvement.

Keywords: Compressive sensing, Closeness centrality, Social networks

Introduction

Many real-world systems can be modeled by a network $G = (V, E)$ of interacting actors. The actors are demonstrated by a set of nodes V with cardinality $|V|$ that are connected via the set of edges (links) E with cardinality $|E|$. The edges can be directed or undirected, depending on the type of interactions. Some well-known examples of such real-world systems include technological and transportation infrastructures, communication systems, biological networks, and social interactions. Centrality measures are means of quantifying the importance of a node within the given network. Some notions of centrality only consider local properties of the network; however, some of them reflect global properties. Proper quantification of importance should be done given the application context. To address applications in which reachability of a node to the entire network is of importance, researchers have introduced the *closeness centrality* measure. For an arbitrary node u, its closeness centrality $C(u)$ is defined as the inverse of its average distance to the other nodes in the network. More formally:

$$C(u) = \frac{|V| - 1}{\sum_{v \neq u \in V} d(u, v)} \tag{1}$$

© The Author(s). 2019 **Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

where $d(u, v)$ is the shortest distance between u and v. Locating public facilities over a transportation network such that they are easily accessible to everyone or identifying people with ideal social network location for information dissemination or network influence can be mentioned as scenarios in which identifying high closeness centralities is of great interest (Saxena et al. 2017; Taheri et al. 2017a, 2017b). In these scenarios, we are mainly interested in efficiently and accurately detecting top-k high closeness centrality nodes in the network, while their exact relative order compared to each other, as well as the actual closeness centrality values, are not so important.

A trivial approach to identify top-k closeness centrality nodes consists of the following steps: (1) Utilizing breadth-first search (BFS) for calculating closeness centrality for each node in $O(|V| + |E|)$ with a total computational cost of $O\left(|V||E| + |V|^2\right)$; (2) Sorting the computed values via a sorting algorithm in $O\left(|V| \log(|V|)\right)$, then report the top-$k$ nodes. The high computational cost of $O(|V||E| + |V|^2)$ and the requirement of full knowledge of the network topology may prevent such a method from being applied on large real-world networks (Wehmuth and Ziviani 2012). To address this issue, developing scalable distributed algorithms is of great importance, where each node is only interacting with its immediate neighbors (You et al. 2017).

To the best of our knowledge, there is no distributed and decentralized algorithm for the task of detecting top-k high closeness centrality nodes that operates while requiring each node only to have local interactions with its immediate neighbors. However, several algorithms are satisfying these properties and compute exact or approximated closeness centrality of each node in the network. Approximation approaches compute an alternative centrality score that highly correlates with the global closeness centrality. An efficient sorting algorithm can then be utilized on top of these methods to identify top-k high closeness centrality nodes. There are two major shortcomings with such approaches: (1) Not exploiting the fact that the vector consisting of closeness centrality values has a few large coefficients (k) and many small coefficients so that it can be well approximated by a k-sparse vector (signal). In general, a centrality measure (*e.g.* closeness centrality) must have a right-skewed probability distribution to be useful in selecting important nodes. (2) Requiring *direct measurement* (query) from each node, which is not always possible due to log-in requirements, API query limits, and treating user data as proprietary.

To address these issues, we transform the problem of detecting top-k closeness central nodes to the problem of sparse recovery in networks. The breakthrough of the sparse recovery problem is compressive sensing (aka compressive sampling) which performs a few indirect end-to-end measurements on a signal x and recovers a good sparse approximation of that signal. However, two additional requirements must be taken into account when these measurements are performed over a graph, rather than an arbitrary signal. Creating feasible measurements that satisfy these constraints (will be discussed in "Compressive Sensing over Networks" section) has initiated the field of compressive sampling over graphs.

Our contributions in this paper are two-fold: (1) We propose a local (ego-centric) metric which can be computed in a distributed manner at each node. The computation can be carried out requiring each node to have only local knowledge of its immediate neighborhood. In "Experimental Evaluation" section, we experimentally show that the suggested local metric is highly correlated with the global closeness centrality on many real-world and synthetic networks. (2) We propose a general compressive sensing framework for distributed identification of central nodes in networks based on the introduced local metric using indirect end-to-end (aggregated) measurements. We experimentally show the superiority of our approach in terms of accuracy for the prediction of high closeness central nodes compared to the best existing competing methods.

The rest of this paper is organized as follows. In "Preliminaries" section, we briefly explain the preliminary notations and definitions. We review the related works on distributed detection of central nodes requiring only local interactions with the neighbors from each node, in "Related Work" section. In "Proposed Method" section, we introduce our novel approach in detail and analyze its time and space complexity. Later in "Experimental Evaluation" section, the settings and results of our experimental evaluations are presented. We conclude the paper in "Conclusion" section.

A preliminary version of this paper has appeared in (Mahyar et al. 2018). Here, we explain the backgrounds and the intuitions behind the idea in more details. Also, we comprehensively review the related work and describe their limitations with our corresponding solutions. Moreover, we add three different types of real datasets and several test scenarios to our extensive experimental evaluations to show the generalization of the proposed method.

Preliminaries

Compressive Sampling

As an alternative to direct measurements, one can utilize sampling-based approaches. Based on the Nyquist-Shannon theorem, a general signal x can be completely recovered by sampling it with the Nyquist rate. However, sampling with the Nyquist rate can be costly or impossible due to a massive scale in many real-world networks we are facing today. If the underlying signal is sparse in a suitable basis, sampling with the Nyquist rate only to recover a relatively small fraction of non-zero elements results in loss of system resources and induces two sources of error, sampling (collection) error and identification (compression) error.

The state-of-the-art approach for recovery of sparse signals is Compressive Sensing/Sampling (CS) which addresses these drawbacks. In compressive sampling, one can simultaneously sample and compress a signal $x_{n\times 1}$ through a measurement matrix $\mathcal{A}_{m\times n}$ where $m \ll n$ to acquire the following linear system:

$$y_{m\times 1} = \mathcal{A}_{m\times n}\, x_{n\times 1} \tag{2}$$

The resulting system is under-determined and does not have a unique solution in general. $\mathcal{A}$ is said to satisfy the $2k$-restricted isometry property (RIP) if there exists $0 < \delta_{2k} < 1$, such that for all $2k$-sparse signals x', it holds:

$$(1-\delta_{2k})||x'||_2 \leq ||Ax'||_2 \leq (1+\delta_{2k})||x'||_2 \tag{3}$$

In case the measurement matrix satisfies the $2k$-RIP one can prove uniqueness of a k-sparse solution to the above linear system ($y = \mathcal{A}x$). To see this, assume x_1 and x_2 are both k-sparse signals and $\mathcal{A}x_1 = \mathcal{A}x_2$, so vector $x' = x_1 - x_2$ is a $2k$-sparse signal (has at most $2k$ non-zero entries). Since $\mathcal{A}$ satisfies the $2k$-RIP, Eq. (3) can be rewritten for some $0 < \delta'_{2k} < 1$ which ensures $x_1 = x_2$, as:

$$(1-\delta'_{2k})||x_1 - x_2||_2 \leq 0 \leq (1+\delta'_{2k})||x_1 - x_2||_2 \tag{4}$$

Let x^* be any arbitrary k-sparse vector, and $\mathcal{A}$ be an arbitrary measurement matrix that satisfies the $2k$-RIP property. Then given what we have discussed so far, it is easy to see that x^* can be recovered by solving:

$$\min_{x} \|x\|_0 \quad \text{s.t.} \quad y = \mathcal{A}x \tag{5}$$

where $\|x\|_0$ indicates the number of non-zero entries in x. Unfortunately, solving this optimization problem is NP-hard. Thus the following relaxation is considered which utilizes the sparsity inducing ℓ_1-norm and is referred to as Basis Pursuit (BP):

$$\min_{x} \|x\|_1 \quad \text{s.t.} \quad y = \mathcal{A}x \tag{6}$$

It has been shown when the $2k$-restricted isometry is satisfied for $\mathcal{A}$, the solution of BP is x^*. In this case, by utilizing the convexity of BP, the recovery is very efficient and computationally fast. Note that the strict condition $y = \mathcal{A}x$ within the Basis Pursuit formulation is very sensitive to imperfect sparsity or noise. The following formulation, known as LASSO, addresses this by removing the exact constraint and penalizing its violation:

$$\min_{x} \|x\|_1 + \|\mathcal{A}x - y\|_2^2 \tag{7}$$

This objective has extremely fast distributed numerical solvers and will be utilized for the optimization step in this paper.

Compressive Sensing over Networks

In case the signal to be recovered is defined over a graph (network), three additional constraints must be taken into account (Xu et al. 2011; Mahyar et al. 2013a) in CS problems: (1) Each element $\mathcal{A}_{i,j}$ would be 1 if the node j is visited by measurement i and 0 otherwise; (2) The nodes visited by a measurement must correspond to a connected induced sub-graph (Ghalebi et al. 2017; Mahyar et al. 2015b, 2018a, 2017); (3) The signal x which contains a graph property, defined for each node, is almost always non-negative ($x \geq 0$).

Based on the compressive sensing framework, we would like to efficiently recover k highest closeness centrality nodes from m indirect end-to-end measurements, in a way that $m \ll n$. In the linear system $y_{m\times 1} = \mathcal{A}_{m\times n}\, x_{n\times 1}$, let $\mathcal{A}$ be an $m \times n$ measurement matrix, where its i-th row corresponds to the i-th feasible measurement. For $i = 1, \ldots, m$ and $j = 1, \ldots, n$, $\mathcal{A}_{ij} = 1$ if and only if node j is visited by the i-th measurement, otherwise $\mathcal{A}_{ij} = 0$. Let x be an $n \times 1$ non-negative vector whose j-th entry is the value of a certain type of network characteristic (*e.g.* a global/local centrality metric) over node $j \in V$, and $y \in \mathcal{R}^m$ denotes the measurements vector whose i-th entry represents the additive aggregation values of network nodes in the i-th row of the measurement matrix $\mathcal{A}$ that induces a *connected sub-graph* over G. Note that this way of measurements construction already satisfies the network topological constraints of the feasibility conditions mentioned in the beginning of this section.

For the example network shown in Fig. 1 with $n = 10$ nodes and $|E| = 11$ links, each of two measurements m_1 and m_2 includes a different subset of connected nodes. The corresponding feasible measurement matrix $\mathcal{A}$ with these measurements is:

$$\mathcal{A} = \begin{array}{c} \\ m_1 \\ m_2 \end{array} \begin{array}{c} \begin{array}{cccccccccc} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 & v_9 & v_{10} \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array} \tag{8}$$

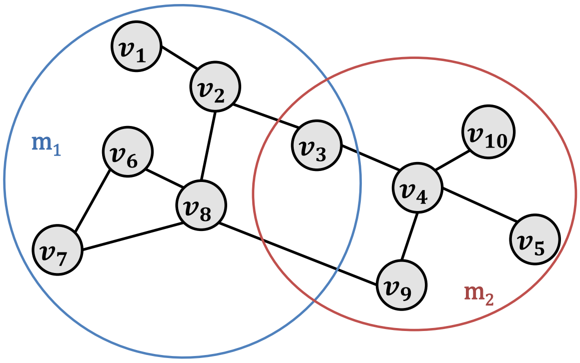


Fig. 1 A network with 10 nodes and 11 links. The measurements m_1 and m_2 are feasible considering the network topological constraints (each of them induces a connected sub-graph over the network)

To understand how the additive aggregation over connected induced sub-graphs is motivated for each measurement in practice, we mention an example from (Wang et al. 2012). Consider a network where the nodes represent sensors, and the links represent communications between sensors. For the set T of active nodes within an arbitrary feasible measurement that induce a connected sub-graph, a node $u \in T$ monitors the total values corresponding to nodes in T. Every node in T obtains values from its children, if any, and aggregates them with its value on the spanning tree rooted at u, then sends the sum to its parent. After that, the fusion center can obtain the sum of values corresponding to all the nodes in T by only communicating with u. The explained paradigm in data acquisition and aggregation is highly utilized within the wireless sensor network literature for applications such as air quality monitoring, volcanic activity detection, and object localization (Middya et al. 2017). Some recent work has applied a similar acquisition and aggregation paradigm in network tomography (Mahyar et al. 2013a), community detection (Mahyar et al. 2015b) and finding key actors in social networks (Mahyar 2015; Mahyar et al. 2015a; Grosu et al. 2018).

Based on the above idea, a straight forward approach utilized in practice to construct measurement matrices satisfying these properties, is to create a correspondence between every single measurement and a *random walk* on the graph. Each random walk additively aggregates values computed by the nodes during the walk. The random walk strategy and the values computed by the nodes are what separate a method from the others. Performance of these methods and RIP satisfaction can then be verified theoretically or experimentally (Mahyar et al. 2018; Mahyar 2015; Mahyar et al. 2018a; Xu et al. 2011). An alternative approach (Mahyar et al. 2018b) employs a well-known randomized method in compressive sensing literature which satisfies the restricted isometry property with very high probability and makes deriving theoretical recovery guarantees straightforward. Also, it is possible to show that each constructed measurement will almost surely correspond to an induced connected sub-graph.

Related Work

In this section, we first review local metrics that highly correlate with the global closeness centrality and can be computed in a distributed manner relying only on interactions of neighboring nodes. After that, we review compressive sensing (CS)-based methods that can be utilized to recover top-k central nodes, using the mentioned local metrics by constructing a feasible measurement matrix.

Local Closeness Metrics

Dist-Exact (You et al. 2017): They proposed a distributed method to compute and collect the set of nodes with an exact distance of h from an arbitrary node u. The parameter h varies from 1 to $\mathcal{D}$, where $\mathcal{D}$ denotes the diameter of the network. The collected sets can then be utilized to compute the closeness centrality at each node.

Dist-Est (Wang and Tang 2015): They derived a set of affine constraints which are distributed in nature and characterize closeness centrality according to its original definition. The derived constraints are used to develop an algorithm, which enables nodes in a network to cooperatively estimate their closeness centrality.

DACCER (Wehmuth and Ziviani 2012): Let $\text{vol}_h(u)$ denote the sum of degrees for all nodes in the h-hop neighborhood of u. In this work, the authors showed a high correlation between $\text{vol}_h(u), \forall u \in V$ and the closeness centrality distribution for $h > 0$. The correlation is shown to become stronger as h grows.

Weight-Vol (Kim and Yoneki 2012): This work was an extension to the metric in DACCER, based on two simple observations. First, closer nodes to a node have more contributions than farther nodes in the dissemination of the node's information. Second, the nodes with low clustering coefficients are hubs linking neighboring network parts.

CS-based Methods for Data Aggregation

RW (Xu et al. 2011): This work is one of the state-of-the-art methods in compressive sensing over graphs that constructs random-walk based measurements. Each measurement in the measurement matrix can be used to aggregate a metric of choice additively.

TopCent (Mahyar 2015): This method constructs a measurement matrix to recover top-k degree central nodes in networks. Since degree centrality is highly correlated with the closeness centrality in some real-world networks, this method is expected to perform well for the task of detecting closeness centralities, as well.

DICeNod (Mahyar et al. 2018b): This approach does not perform walks to create a measurement matrix; instead, it utilizes a well-known randomized matrix construction technique in compressive sensing. They showed that the constructed measurements correspond to induced connected sub-graphs in networks with high probability.

Proposed Method

In this section, we introduce the proposed framework in the following steps: (1) defining a new ego-centric centrality measure; (2) introducing a subroutine, called CS-HiClose-ScoreCompute, which calculates the proposed ego-centric centrality metric in a distributed and decentralized manner; (3) introducing a subroutine, called CS-HiClose-Aggregate, which aggregates the local scores via decentralized measurements construction in compressive sensing. This will be executed only after the execution of the previous subroutine; and (4) analyzing the overall time and space complexity of the proposed approach. The pseudo-code of the proposed approach, CS-HiClose, is in Algorithm 1, which mainly calls the two subroutines mentioned in steps (2) and (3).

```
Algorithm 1 The Proposed Method: CS-HiClose
Input: $V, m, l, h$
    $V$: set of network nodes
    $m$: number of required measurements
    $l$: measurements length
    $h$: neighbourhood radius size at each node
    CS-HiClose-ScoreCompute($V, h$)
    $\hat{x}$ = CS-HiClose-Aggregate($V, m, l$)
Output: sparse approximation $\hat{x}$
```

Proposed Local Metric

We introduce the h-hop ego-centric (local) closeness centrality of node v as:

$$egoC_h(v) = \sum_{\tau=1}^{h} |B_\tau(v)|/\tau \tag{9}$$

where $B_\tau(v)$ indicates the set of nodes that have an exact shortest distance of length τ from node v. The intuition behind this metric is that, the farther nodes from v have lower effect in dissemination of goods (*e.g.* information) emerged from it.

Score Computation Subroutine

The computation of the sets $B_\tau(v)$ for $\tau \leq h, \forall v \in V$ can be done by executing a breadth-first search (BFS) process at each node in parallel, with exploration radius of h. This will require computational cost of at most $O(\Delta^h)$ where Δ is the maximum degree of the network. The required memory storage at each node is also $O(\Delta^h)$. The computed sets can be utilized to evaluate ego closeness centrality at each node in a distributed and decentralized manner, with $O(1)$ computational and storage cost per node. Thus we will have the following steps for ego-closeness computation:

(i) For each node $v \in V$ in the network, run $BFS_h(v)$ to calculate the number of nodes in its i-hop neighborhood denoted as $B_i(v)$ where i ranges from 1 to h. This step can be executed in a decentralized manner for each node independently from the others.

(ii) Once $B_i(v)$ is available for each node $v \in V$, i ranging from 1 to h, one can easily compute the ego-closeness centrality metric based on Eq. (9). This step can be also executed in a decentralized fashion for each node independently. The pseudo-code for this subroutine is in Algorithm 2.

Score Aggregation Subroutine

The proposed compressive sensing-based method for aggregating the computed ego-centric metric is depicted in Algorithm 3, which contains fours steps:

(i) The first node v_{first} is added to the visited set S and all of its neighbors are added to the neighbor set $\mathcal{N}(S)$.

(ii) The next node is selected relative to $egoC_h(v_{next})$ from the nodes in $\mathcal{N}(S)$, which are already computed in the previous subroutine.

Algorithm 2 CS-HICLOSE-SCORECOMPUTE(V, h)

```
Input: $V, h$
  $V$: set of network nodes
  $h$: neighborhood radius size at each node
  Foreach $v \in V$ do                                    ▷ In a distributed manner
      Calculate $BFS_h(v)$ to initialize $B_i(v)$ for $i = 1 \ldots h$
      $egoC_h(v) = \sum_{\tau=1}^{h} |B_\tau(v)|/\tau$
  end for
Output: For each node $v \in V$, its $h$-hop ego-centric measure $egoC_h(v)$ is computed
```

Algorithm 3 CS-HICLOSE-AGGREGATE(V, m, l)

```
Input: $V, m, l$
  $V$: set of network nodes
  $m$: number of required measurements
  $l$: measurements length
  $\mathcal{A} = \mathbf{0}_{m \times n}$
  $y = \mathbf{0}_{m \times 1}$
  for $i = 1 \to m$ do                                    ▷ In a distributed manner
      Choose $v_{first}$ uniformly at random from $V$
      $S = \{v_{first}\}$
      $\mathcal{N}(S) = \mathcal{N}(v_{first})$
      $\mathcal{A}[i, v_{first}] = 1$
      $y[i] = egoC_h(v_{first})$
      for $j = 1 \to l$ do
          Choose $v_{next}$ relative to $egoC_h(v_{next})$ from $\mathcal{N}(S)$
          $S = S \bigcup \{v_{next}\}$
          $\mathcal{N}(S) = \mathcal{N}(S) \setminus \{v_{next}\}$
          $\mathcal{N}(S) = \mathcal{N}(S) \bigcup \mathcal{N}(v_{next})$
          $\mathcal{A}[i, v_{next}] = 1$
          $y[i] = y[i] + egoC_h(v_{next})$
      end for
  end for
  $\hat{x} = \min_x \|x\|_1 + \|\mathcal{A}x - y\|_2^2$           ▷ See Eq. (7)
Output: sparse approximation $\hat{x}$
```

(iii) The selected next node is added to the visited set S and it is removed from the neighbor set $\mathcal{N}(S)$, then its neighbors are added to the neighbor set $\mathcal{N}(S)$.

(iv) The steps (i) $-$ (iii) are fulfilled 'l' times which is the length of a measurement, to generate a new row for the matrix $\mathcal{A}$ and the vector y.

(v) Step (iv) is repeated 'm' times (in parallel) to construct a feasible measurement matrix $\mathcal{A}$ with 'm' measurements and the corresponding measurement vector y.

(vi) To find the sparse approximation $\hat{x}$ of x, we optimize the LASSO objective function subject to the linear sketch of $y = \mathcal{A}x$, based on Eq. (7).

In this algorithm, we have m parallel aggregation processes, where each is to be started from a node selected uniformly at random from V. The random seeds to choose the starting point of each aggregating process can be fixed in time $O(m)$. A measurement corresponding to a process with a starting node will keep track of two sets S and $\mathcal{N}(S)$. The set S is initialized with v_{first} and the set $\mathcal{N}(S)$ is initialized by its immediate neighbors, denoted by $\mathcal{N}(v_{first})$. Within l sequential iterations, a candidate v_{next} from $\mathcal{N}(S)$ will be selected relative to $egoC_h(v_{next})$, removed from $\mathcal{N}(S)$ and added to S. Moreover the neighbors of v_{next} that are not already present in $\mathcal{N}(S)$ will be added to $\mathcal{N}(S)$. In other words S is the set of visited nodes and $\mathcal{N}(S)$ is the set of candidate nodes that are not in S but are connected to some node(s) in S. This ensures that the set of visited nodes S at each single iteration corresponds to an induced connected sub-graph from the network. At iteration i of total l iterations, the maximum size of $\mathcal{N}(S)$ is $\min(i\Delta - i, |V|)$, thus selection of a member from $\mathcal{N}(S)$ relative to ego-closeness centralities using a binary search will be possible with computational cost of $\log(\min(i\Delta - i, |V|))$. The total cost of applying this binary search method is $O(|V|\log(|V|))$ in total. To show this, we consider two different cases. If $l \le \left\lfloor \frac{|V|}{\Delta-1} \right\rfloor$, then:

$$\begin{aligned}\sum_{i=1}^{l} \log(\min(i\Delta, |V|)) &= \sum_{i=1}^{l} \log\left(i\Delta - i\right) = \log\left(l!\right) + l\log\left(\Delta - 1\right)\\ &\le l\log(l) + l\log(\Delta) \le 2|V|\log(|V|)\end{aligned}$$

Otherwise, if $l > \left\lfloor \frac{|V|}{\Delta-1} \right\rfloor$, then:

$$\begin{aligned}&\sum_{i=1}^{l} \log(\min(i\Delta, |V|)) = \sum_{i=1}^{\left\lfloor \frac{|V|}{\Delta-1} \right\rfloor} \log\left(i\Delta - i\right) + \left(|V| - \left\lfloor \frac{|V|}{\Delta - 1} \right\rfloor\right)\log(|V|)\\ &= \log\left(\left\lfloor \frac{|V|}{\Delta - 1} \right\rfloor!\right) + \left\lfloor \frac{|V|}{\Delta - 1} \right\rfloor \log\left(\Delta - 1\right) + \left(|V| - \left\lfloor \frac{|V|}{\Delta - 1} \right\rfloor\right)\log(|V|)\\ &\le \left\lfloor \frac{|V|}{\Delta - 1} \right\rfloor \left(\log\left(\frac{|V|}{\Delta - 1}\right) + \log\left(\Delta - 1\right)\right) + \left(|V| - \left\lfloor \frac{|V|}{\Delta - 1} \right\rfloor\right)\log(|V|)\\ &= |V|\log(|V|)\end{aligned}$$

Moreover, the number of deletions from and additions to $\mathcal{N}(S)$ are at most $|V|$. Each addition/deletion operation can be done efficiently in $O(1)$, using an array structure. Thus, the total time complexity for the aggregating stage is $O(m + |V|\log(|V|) + |V|) = O(|V|\log(|V|))$, where we have assumed $m \ll |V|$ aggregating processes (measurements). The required space for each aggregating process is $O(l)$ to save the visited nodes, and $O(1)$ for saving the aggregated values of the visited nodes. Also, a space of at most $O(|V|)$ is required for keeping track of the lists S and $\mathcal{N}(S)$. Finally, global space storage of size $O(m)$ is needed to save the initial measurements seeds.

The Complexity Analysis of CS-HiClose

Overall, our approach requires a running time of $O(|V|\log(|V|) + \Delta^h)$, local storage of $O(\Delta^h)$ at each node and global storage of size $O(m)$ for the seeds. Besides, a local storage space of $O(l)$ is required for each aggregating process (measurement). In the next section, we will show a high correlation between the proposed ego-centric centrality with

$h = 2$ and the global closeness centrality of the nodes in various networks. The experiments indicate that one does not gain much more correlation by increasing h to some number greater than two, although one will endure Δ times higher computational and storage cost to do so, in the worst case. Thus, we suggest $h = 2$ for satisfactory yet efficient utilization of our algorithm. It is worth noting that in most real-world networks, in particular social networks, nodes are connected to a tiny portion of the whole network's nodes, which means Δ (and in turn Δ^2) is very small. For example, the maximum number of connections allowed on Twitter and Facebook is about 5000, that is much smaller than their network size (Mahyar et al. 2018a). This shows that our approach is practically efficient and scalable on real-world networks.

Experimental Evaluation

In this section, we experimentally evaluate the performance of the proposed method in various scenarios over both synthetic and real-world networks. We first introduce the networks used for the evaluation. Then, we explain the settings of the experiments. Finally, the achieved results for each test scenario and their analyses are presented.

Datasets

For the evaluations of the proposed method, we considered both synthetic and real networks. We summarize the properties of the real-world networks used in experiments in Table 1. The four notations $\langle deg \rangle$, $\langle \mathcal{C} \rangle$, $\mathcal{D}$, and $\delta_{0.9}$ represent the "average degree", "average clustering coefficient", "network diameter", and "90-percentile effective diameter", respectively. In the case of a disconnected network, we extracted the largest (strongly) connected component.

We also considered three well-known models (i.e. Barabási-Albert (BA), Erdős-Rényi (ER), and Watts-Strogatz (SW)) for generating synthetic networks. We have summarized these networks in Table 2. In ER network, the link existence probability $p = 0.01$ ensures that the generated network is connected as $p > \frac{\ln |V|}{|V|}$ is a sharp threshold for the connectedness of ER networks with $|V|$ vertices.

Settings

To evaluate the accuracy of the proposed method (CS-HiClose) compared to the competing methods in identifying top-k closeness centrality nodes, we measured the *precision* and *recall* of the algorithms. Precision quantifies the number of correctly detected nodes in the list of k highest closeness centrality nodes divided by the total number of detected

Table 1 Real-world networks

Network	$\lvert V \rvert$	$\lvert E \rvert$	$\langle deg \rangle$	$\langle \mathcal{C} \rangle$	$\mathcal{D}$	$\delta_{0.9}$
Facebook (Opsahl and Panzarasa 2009)	1893	6917	7.31	0.06	8	3.65
Twitter (Leskovec et al. 2007)	3656	94356	51.62	0.3	6	2.89
ca-AstroPh (Leskovec et al. 2007)	17903	197001	22.01	0.32	14	5.01
ca-CondMat (Leskovec et al. 2007)	21363	91314	8.55	0.26	15	6.52
ca-HepPh (Leskovec et al. 2007)	11204	117634	21	0.66	13	5.79
ca-HepTh (Leskovec et al. 2007)	8638	24816	5.75	0.28	18	7.42
email-Enron (Leskovec et al. 2009)	33696	180811	10.73	0.09	13	4.79
DBLP (Yang and Leskovec 2015)	317080	524933	3.31	0.31	23	8.16
wiki-Vote (Leskovec et al. 2010)	7066	51831	14.67	0.13	7	3.78

Table 2 Synthetic network models

Network model	$\lvert V\rvert$	$\lvert E\rvert$	Parameter	$\langle deg\rangle$
Barabási-Albert (BA) (Barabasi and Albert 1999)	500	2979	5	11.92
Erdős-Rényi (ER) (Erdos and Renyi 1960)	500	4000	0.01	16
Watts-Strogatz (SW) (Watts and Strogatz 1998)	500	4466	[0.2 ; 9]	17.86

nodes. Recall quantifies the number of correctly identified nodes divided by the total number of nodes in the network. The relevancy of the detected nodes (precision) and the portion of relevant nodes that are detected (recall) are both of importance. To take both into account, we utilized the popular F-measure metric, a harmonic mean of precision and recall, which is defined as:

$$\text{F-measure} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{10}$$

Since CS-HiClose, RW, TopCent, and DICeNod have a source of randomness, the experiments were repeated ten times to reduce the variance. The denoted points in the figures represent the mean value of these repetitions along with their asymmetric standard deviations, which quantifies the amount of variations of F-measure at each point in each figure. Implementation codes in Python can be found at `https://github.com/hamidreza-mahyar/CS-HiClose`. We used POGS (POGS 2018), a fast and parallel optimization solver, for the optimization phase of CS-HiClose. POGS tries to minimize LASSO (Eq. (7)) as an objective function and is extremely quick by leveraging the power of GPUs. For example, (Parikh and Boyd 2014), it can solve the LASSO objective on a graph of 100,000 nodes with 10,000 measurements in only 21s on a single Nvidia K40 GPU. For computations of the global closeness centrality in Eq. (1), we used available tools in *Python-iGraph* package.

Evaluation Results

Correlation between Our ego-Closeness and the Global Closeness

We experimentally analyzed the correlation between the proposed ego-centric (local) centrality metric and the global closeness centrality over several synthetic and real-world networks. To compare these two centrality metrics, we used Pearson product moment correlation coefficient (ρ), which in fact measures the strength of a linear association between two variables and is defined as (Benesty et al. 2009):

$$\rho = \frac{\sum_{i=1}^{|V|}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{|V|}(x_i - \overline{x})^2 \sum_i (y_i - \overline{y})^2}} \tag{11}$$

where $|V|$ is the number of network nodes and x_i, y_i correspond to the local and global centrality measures of node i, respectively. $\overline{x}$ and $\overline{y}$ are mean of these variables. The Pearson coefficient ρ can take a value in range $[-1, +1]$. A value of 0 shows that there is not any association, a value greater than 0 indicates a positive association, and a value less than 0 indicates a negative association.

Table 3 illustrates the correlation coefficients between the proposed ego-closeness and the global closeness centrality. As mentioned in "Proposed Local Metric" section, the computational and storage cost of the ego-closeness centrality is directly impacted by choice of h. Thus, we aim to yield good results in distributively assessing top-k network

Table 3 Pearson correlation coefficients between the proposed ego-centric closeness centrality with small exploration radius (i.e. $h = \{2, 3\}$) and the global closeness centrality on synthetic and real-world networks

		Facebook	Twitter	ca-AstroPh	ca-CondMat	ca-HepPh	ca-HepTh	email-Enron	DBLP	wikiVote	BA	ER	SW
h	$k/\|V\|$												
2	0.1	1.00	1.00	0.99	0.91	0.98	0.93	0.97	0.90	0.99	1.00	1.00	0.96
	0.2	1.00	1.00	0.98	0.91	0.98	0.93	0.98	0.87	0.99	1.00	1.00	0.96
	0.3	1.00	1.00	0.98	0.91	0.97	0.93	0.98	0.84	0.99	1.00	1.00	0.96
	0.4	1.00	1.00	0.98	0.90	0.96	0.91	0.98	0.82	0.99	1.00	1.00	0.96
	0.5	1.00	1.00	0.96	0.87	0.95	0.89	0.92	0.76	0.99	0.99	1.00	0.96
	0.6	1.00	1.00	0.94	0.86	0.93	0.88	0.91	0.74	0.99	0.99	1.00	0.97
	0.7	1.00	1.00	0.93	0.84	0.91	0.86	0.89	0.72	0.99	0.99	1.00	0.97
	0.8	0.99	1.00	0.90	0.82	0.89	0.84	0.83	0.69	0.99	0.99	1.00	0.97
	0.9	0.99	1.00	0.87	0.79	0.86	0.81	0.79	0.66	0.98	0.99	1.00	0.97
	1.0	0.96	0.99	0.80	0.73	0.80	0.75	0.72	0.60	0.97	1.00	1.00	0.97
3	0.1	0.99	0.97	1.00	0.98	1.00	0.98	0.99	0.95	1.00	1.00	0.99	0.97
	0.2	0.99	0.96	1.00	0.98	1.00	0.98	0.99	0.92	1.00	1.00	0.99	0.97
	0.3	1.00	0.96	1.00	0.98	1.00	0.98	0.99	0.90	0.99	1.00	1.00	0.98
	0.4	1.00	0.96	1.00	0.98	1.00	0.97	0.99	0.88	0.99	1.00	1.00	0.98
	0.5	0.99	0.96	1.00	0.97	1.00	0.96	0.99	0.85	0.99	1.00	1.00	0.98
	0.6	0.99	0.96	1.00	0.96	1.00	0.95	0.99	0.83	0.99	1.00	1.00	0.99
	0.7	0.99	0.97	1.00	0.95	0.99	0.94	0.99	0.81	0.99	1.00	1.00	0.99
	0.8	0.99	0.98	1.00	0.93	0.99	0.92	0.99	0.78	0.99	1.00	1.00	0.99
	0.9	0.99	0.99	0.99	0.90	0.98	0.89	0.98	0.74	0.99	1.00	1.00	0.99
	1.0	0.98	0.96	0.95	0.84	0.94	0.83	0.95	0.67	0.99	0.99	1.00	0.99

centralities with a small value of h. We calculated the correlation for various sparsity levels k and small values of h (i.e. 2 and 3) for different networks. It is worth noting that for $h = 1$, any local metric would be the same as the degree centrality. Overall, the results show that our proposed local metric and the global closeness centrality highly correlate on various types of networks. According to the problem addressed in this paper, we want to identify top-k central nodes for $k \ll |V|$, so the results show that in this case choosing $h = 2$ is sufficient yet efficient, in terms of having a good trade-off between computational complexity and accuracy.

Table 4 shows the Pearson correlation coefficients between the existing local metrics reviewed in "Local Closeness Metrics" section (i.e. Dist-Exact, DACCER, and Weight-Vol) and our proposed ego-centric centrality measure, all with $h = 2$, and the global closeness centrality on synthetic and real-world networks. In this experiment, we mainly focus on high sparsity levels $k = \{0.1|V|, 0.2|V|, 0.3|V|, 0.4|V|\}$. After implementing DistEst (Wang and Tang 2015), we found that the computed values for this metric critically depend on parameters' initialization (*e.g.* each node should have an estimation about its closeness value which is an unrealistic assumption). Moreover, this metric needs a huge number of iterations for message passing to converge. To have a fair comparison, we set the same number of iterations as our metric, but its correlation coefficients were around 0, so the results for this metric were excluded.

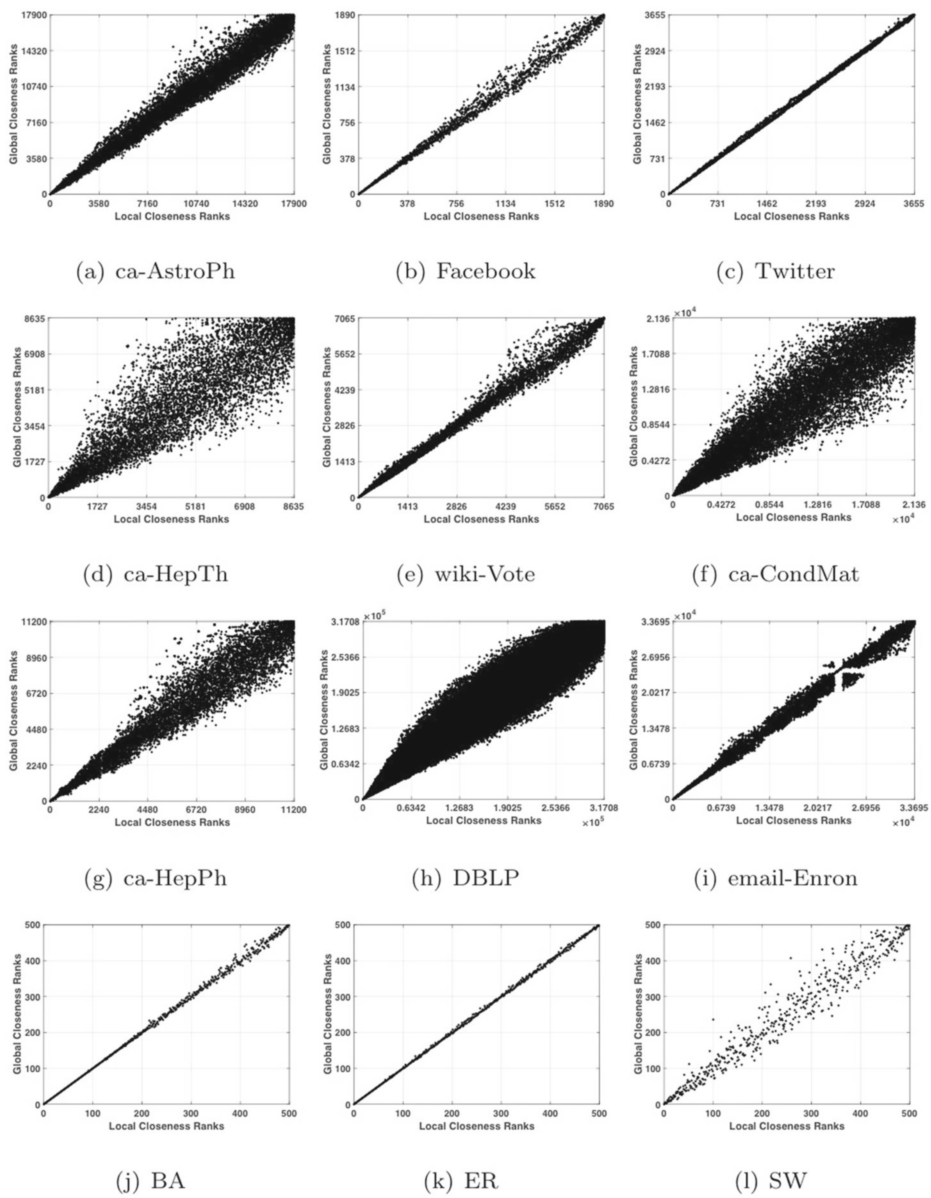
The results show that Dist-Exact for $h = 2$ has a linear correlation, but a negative association with the closeness centrality in networks with various levels of sparsity. One can observe that our proposed metric has almost always the best correlation coefficient compared to the other metrics. Another interesting observation in Tables 3 and 4 is that our ego-centric metric has lower correlation coefficient with the global closeness centrality on the networks (i.e. ca-CondMat, ca-HepTh, and DBLP) with a small average degree, relative to their network size.

To have more analysis of the correlation between the proposed ego-centric (local) metric and the global closeness centrality, Fig. 2, shows the scatter plots of all nodes' ranks provided by one versus the other, on various networks. Each point in the figure corresponds to a node's rank using these two metrics. Based on the results of the previous test cases, we calculated our local measure for $h = 2$ to have low computational complexity, yet high accuracy. One can easily observe the linear correlation and positive association (as the rank with respect to the local metric increases, so does the rank with respect to the global metric), especially for the top-k nodes' ranks which is the target of this paper. One can easily see a similar observation, as in Tables 3 and 4, that our metric has relatively lower correlation with the global closeness centrality on ca-CondMat, ca-HepTh, and DBLP networks. These networks share the property that their average network degree is lower relative to their network size.

Although the Pearson product-moment correlation coefficient is the most common and almost exclusively used measure for correlation studies of centrality indices, non-linear dependencies are not adequately captured by it. Moreover, assuming only a linear correlation between the two scores is very strong and maybe not realistic. A common workaround to depict some of the existing non-linear dependencies is to employ the Pearson correlation on the logarithm of the original scores, and it is mainly used for illustrative purposes (Schoch 2015). Table 5 is similar to Table 3; However, it shows the Pearson

Table 4 Pearson correlation coefficients between the existing local metrics (Dist-Exact, DACCER, Weighted-Vol, and our proposed ego-centric centrality) with $h = 2$ and the global closeness centrality on synthetic and real-world networks for varying percentage of sparsity

		Facebook	Twitter	ca-AstroPh	ca-CondMat	ca-HepPh	ca-HepTh	email-Enron	DBLP	wikiVote	BA	ER	SW
$k/\|V\|$	Local Metric												
0.1	Dist-Exact	-0.93	-0.41	-0.94	-0.83	-0.89	-0.87	-0.75	-0.73	-0.94	-0.94	-0.99	-0.93
	DACCER	0.91	0.47	0.96	0.87	0.77	0.91	0.96	0.87	0.83	0.97	0.96	0.95
	Weight-Vol	0.97	0.80	0.97	0.89	0.93	0.90	**0.97**	**0.90**	0.97	0.99	0.97	0.93
	Our Metric	**1.00**	**1.00**	**0.99**	**0.91**	**0.98**	**0.93**	**0.97**	**0.90**	**0.99**	**1.00**	**1.00**	**0.96**
0.2	Dist-Exact	-0.93	-0.61	-0.93	-0.80	-0.84	-0.84	-0.57	-0.70	-0.94	-0.93	-0.99	-0.91
	DACCER	0.92	0.58	0.97	0.89	0.79	0.92	0.97	0.84	0.89	0.97	0.98	0.94
	Weight-Vol	0.97	0.77	0.97	**0.91**	0.92	0.92	**0.98**	**0.87**	0.98	0.99	0.99	0.94
	Our Metric	**1.00**	**1.00**	**0.98**	**0.91**	**0.98**	**0.93**	**0.98**	**0.87**	**0.99**	**1.00**	**1.00**	**0.96**
0.3	Dist-Exact	-0.93	-0.73	-0.91	-0.78	-0.82	-0.81	-0.61	-0.67	-0.94	-0.92	-0.99	-0.93
	DACCER	0.93	0.63	0.97	0.90	0,81	0.92	**0.98**	0.81	0.92	0.97	0.99	0.95
	Weight-Vol	0.97	0.75	**0.98**	**0.91**	0.92	0.92	**0.98**	**0.84**	0.98	0.99	0.99	**0.96**
	Our Metric	**1.00**	**1.00**	**0.98**	**0.91**	**0.97**	**0.93**	**0.98**	**0.84**	**0.99**	**1.00**	**1.00**	**0.96**
0.4	Dist-Exact	-0.92	-0.79	-0.90	-0.74	-0.76	-0.75	-0.59	-0.61	-0.94	-0.91	-0.99	-0.94
	DACCER	0.95	0.67	0.97	0.89	0.86	**0.91**	**0.98**	0.79	0.95	0.97	0.99	0.95
	Weight-Vol	0.98	0.75	**0.98**	**0.90**	0.93	**0.91**	**0.98**	**0.82**	0.98	0.99	0.99	**0.96**
	Our Metric	**1.00**	**1.00**	**0.98**	**0.90**	**0.96**	**0.91**	**0.98**	**0.82**	**0.99**	**1.00**	**1.00**	**0.96**



(a) ca-AstroPh (b) Facebook (c) Twitter

(d) ca-HepTh (e) wiki-Vote (f) ca-CondMat

(g) ca-HepPh (h) DBLP (i) email-Enron

(j) BA (k) ER (l) SW

Fig. 2 Correlations between the nodes' ranks provided by the proposed local metric and the global closeness centrality on synthetic and real-world networks. These two metrics correlate very well

correlation on the logarithms of the proposed ego-closeness (with $h = 2$) and the global closeness scores. The result suggests that our proposed ego-centric metric not only has a high positive linear association (as inferred by Table 3) but also demonstrates a very high positive non-linear association with the global closeness centrality.

Running Time Comparison

In Table 6, we empirically compare the running time for computation of the local metrics reviewed in "Local Closeness Metrics" section (i.e. Dist-Exact, DACCER, Weight-Vol, and our proposed ego-centric measure) over the synthetic networks. The running time of these metrics measured in a simulated distributed environment on a 2.5 GHz Intel Core i7 Apple MacBook Pro laptop. We set the radius of the local neighborhood

Table 5 Pearson correlation coefficients between the **logarithms** of the proposed ego-centric score with small exploration radius (i.e. $h = \{2, 3\}$) and the global closeness score on synthetic and real-world networks

		Facebook	Twitter	ca-AstroPh	ca-CondMat	ca-HepPh	ca-HepTh	email-Enron	DBLP	wikiVote	BA	ER	SW
h	$k/\|V\|$												
2	0.1	0.99	1.00	0.98	0.91	0.96	0.93	0.94	0.88	0.98	0.99	1.00	0.96
	0.2	0.99	0.99	0.98	0.92	0.96	0.93	0.92	0.87	0.98	0.99	1.00	0.95
	0.3	0.99	0.99	0.98	0.92	0.96	0.92	0.91	0.86	0.98	0.98	1.00	0.95
	0.4	0.98	0.99	0.98	0.93	0.96	0.91	0.90	0.86	0.99	0.98	1.00	0.96
	0.5	0.98	0.99	0.98	0.92	0.96	0.91	0.90	0.85	0.98	0.98	1.00	0.96
	0.6	0.97	0.99	0.98	0.92	0.96	0.90	0.91	0.85	0.98	0.97	1.00	0.96
	0.7	0.97	0.99	0.98	0.93	0.96	0.90	0.92	0.85	0.97	0.97	1.00	0.97
	0.8	0.97	0.98	0.98	0.93	0.96	0.90	0.83	0.86	0.96	0.96	1.00	0.97
	0.9	0.97	0.97	0.98	0.93	0.96	0.89	0.88	0.86	0.96	0.96	1.00	0.97
	1.0	0.97	0.92	0.96	0.92	0.94	0.86	0.90	0.84	0.96	0.96	0.99	0.97
3	0.1	0.99	0.97	0.99	0.97	1.00	0.97	0.99	0.98	1.00	1.00	0.99	0.97
	0.2	0.99	0.96	0.99	0.97	0.99	0.98	0.98	0.98	0.99	1.00	0.99	0.97
	0.3	0.99	0.96	0.99	0.97	0.99	0.98	0.98	0.97	0.99	1.00	1.00	0.98
	0.4	1.00	0.95	0.99	0.98	0.98	0.98	0.98	0.97	0.99	1.00	1.00	0.98
	0.5	0.99	0.96	0.98	0.98	0.97	0.98	0.97	0.97	0.98	1.00	1.00	0.98
	0.6	0.99	0.96	0.98	0.98	0.97	0.98	0.96	0.97	0.98	1.00	1.00	0.99
	0.7	0.99	0.97	0.97	0.98	0.97	0.98	0.94	0.97	0.98	1.00	1.00	0.99
	0.8	0.98	0.98	0.97	0.98	0.97	0.98	0.93	0.97	0.97	1.00	1.00	0.99
	0.9	0.97	0.99	0.97	0.98	0.97	0.98	0.93	0.97	0.97	1.00	1.00	0.99
	1.0	0.89	0.85	0.97	0.98	0.97	0.96	0.95	0.95	0.92	0.99	1.00	0.99

Table 6 Running time (in *milliseconds*) comparison for different local metrics on synthetic networks in a simulated distributed environment

Network	Dist-Exact	DACCER	Weight-Vol	Our metric
Barabási-Albert (BA)	3.77	4.26	13.78	3.74
Erdős-Rényi (ER)	1.20	1.45	8.16	1.18
Watts-Strogatz (SW)	1.10	1.02	6.65	0.80

for each node to $h = 2$, similar to the other experiments and due to the same reasons.

Note that in the distributed and decentralized setting that we considered here, each node in the network begins executing a process to compute its corresponding local metric based on its visible neighborhood radius. Each node's process runs independently of the other nodes' processes. The distributed running time that we report for a metric on a network is equal to the longest execution time among all network nodes' processes for computation of the desired local metric. Table 6 shows that our proposed metric is the fastest local measure to be calculated locally in a decentralized manner over all of the synthetic networks.

Effect of Sparsity Level k on Accuracy:

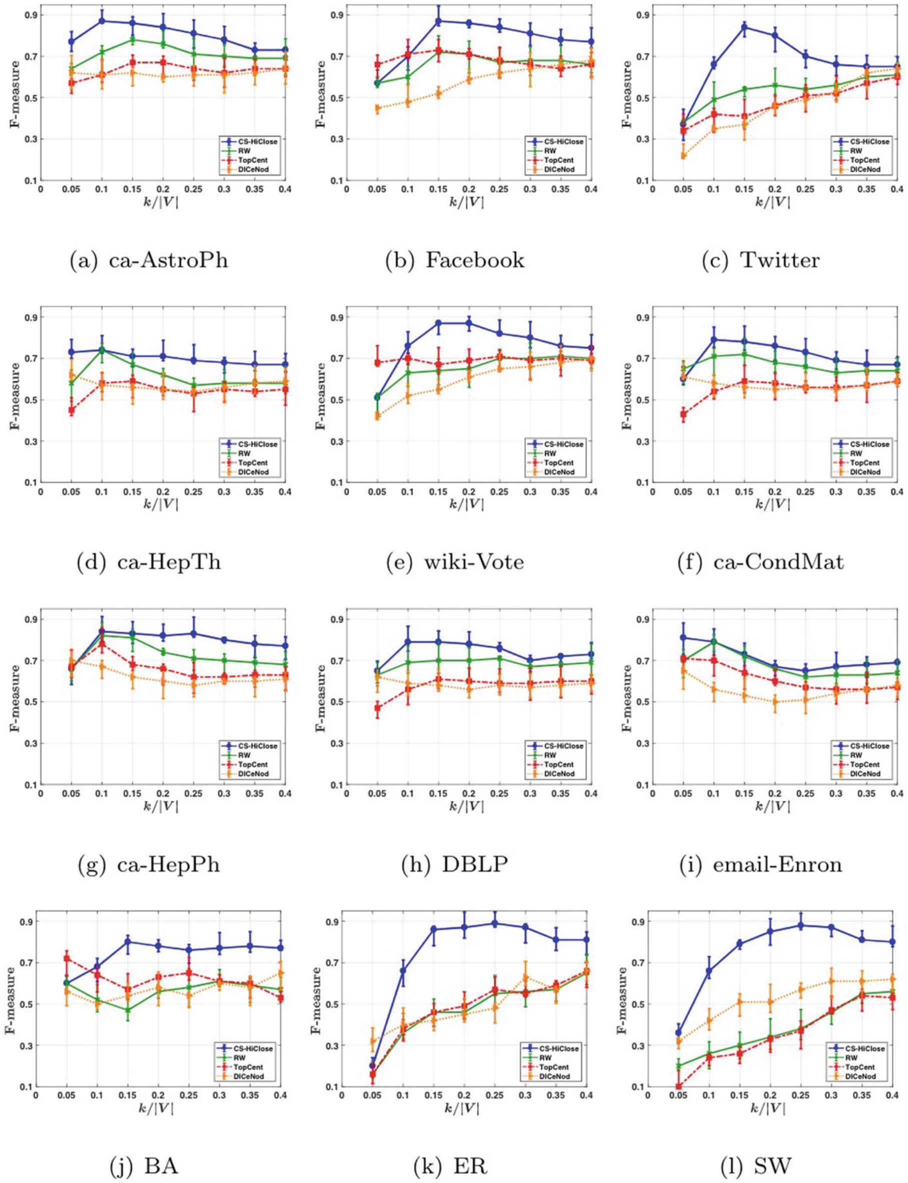
Figure 3 shows the effect of sparsity level k on the accuracy of CS-HiClose in comparison with the CS-based competing methods in the case where the number of measurements set to $0.4|V|$ and the measurements length set to $0.25|V|$. The measurements length in DICeNod is defined according to another parameter $d = \frac{\varepsilon}{Ck}m$, where $\varepsilon \in (0, \frac{1}{6})$ and $C > 1$. To have a fair comparison, we chose ε and C in a way that the average measurement length in this method and the other methods are the same. The higher the value of F-measure is, the more correlation between the top-k nodes identified by a method and the global closeness centrality will be.

Effect of Number of Measurements m on Accuracy:

The accuracy of CS-HiClose is compared to the existing CS-based methods in terms of F-measure for varying number of measurements, while the measurements length (l) set to $0.25|V|$ and the sparsity (k) set to $0.15|V|$ in a network with $|V|$ nodes. For DICeNod, l is determined based on m and k. In Fig. 4, it is clearly depicted that CS-HiClose outperforms the competing methods in terms of having higher F-measure for almost all number of measurements. Moreover, our method has better accuracy even in small number of measurements. This improvement can be very important in the situations where performing measurements has a high computational cost (Mahyar et al. 2015a; Mahyar et al. 2013b).

Effect of Measurement Length l on Accuracy

Figure 5 illustrates that CS-HiClose has higher F-measure for the most measurement lengths in all test cases, in comparison with the CS-based methods RW, TopCent and DICeNod. Since the concept of measurement length is again irrelevant to the other competing methods, we only compared our accuracy with the CS-based approaches. The horizontal axis in Fig. 5 shows the measurement length l divided by the total number of network nodes $|V|$. This experiment is performed over the network with $|V|$ nodes where the number of measurements sets to $m = 0.4|V|$ and the sparsity level sets to $k = 0.2|V|$



(a) ca-AstroPh (b) Facebook (c) Twitter

(d) ca-HepTh (e) wiki-Vote (f) ca-CondMat

(g) ca-HepPh (h) DBLP (i) email-Enron

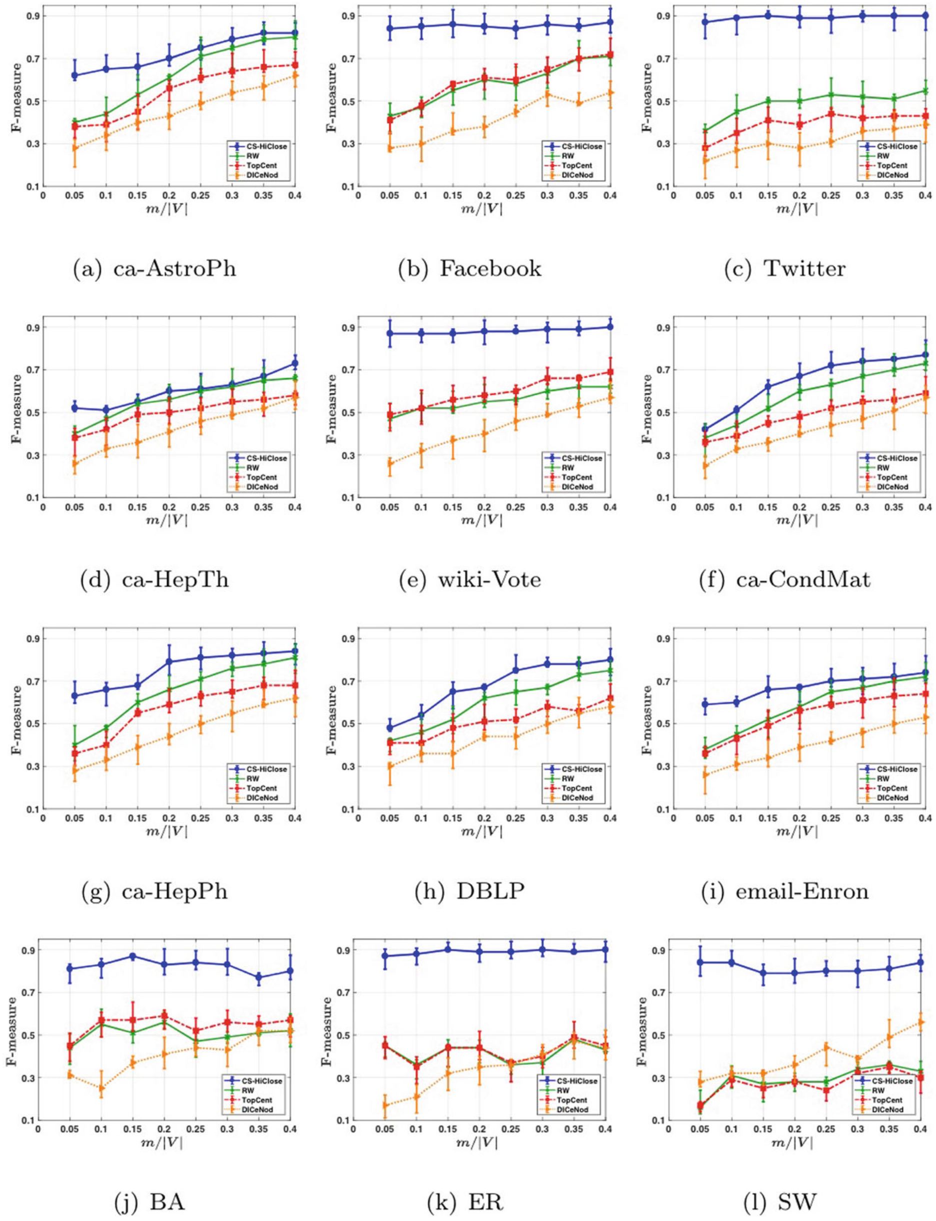
(j) BA (k) ER (l) SW

Fig. 3 Effect of sparsity level k on the accuracy of CS-HiClose and the competing methods for the number of correctly detected top-k closeness centrality nodes. For all methods, we set the number of measurements to $0.4|V|$ and the measurements length to $0.25|V|$. The higher the value of F-measure is, the more correlation between the top-k nodes list identified by a method and the global closeness centrality will be

for all methods. We repeated each test 10 times to reduce the methods' randomness, and the points in the figures show the mean value of these repetitions. In Fig. 5, we can easily observe an increasing trend for F-measure in CS-HiClose when we increase the length of the measurements.

Conclusion

Closeness centrality has been utilized as a primary metric to measure the relative importance/influence of nodes in a given network. In this paper, we introduced a new ego-centric metric which has little computational cost and correlates well with the global



(a) ca-AstroPh (b) Facebook (c) Twitter

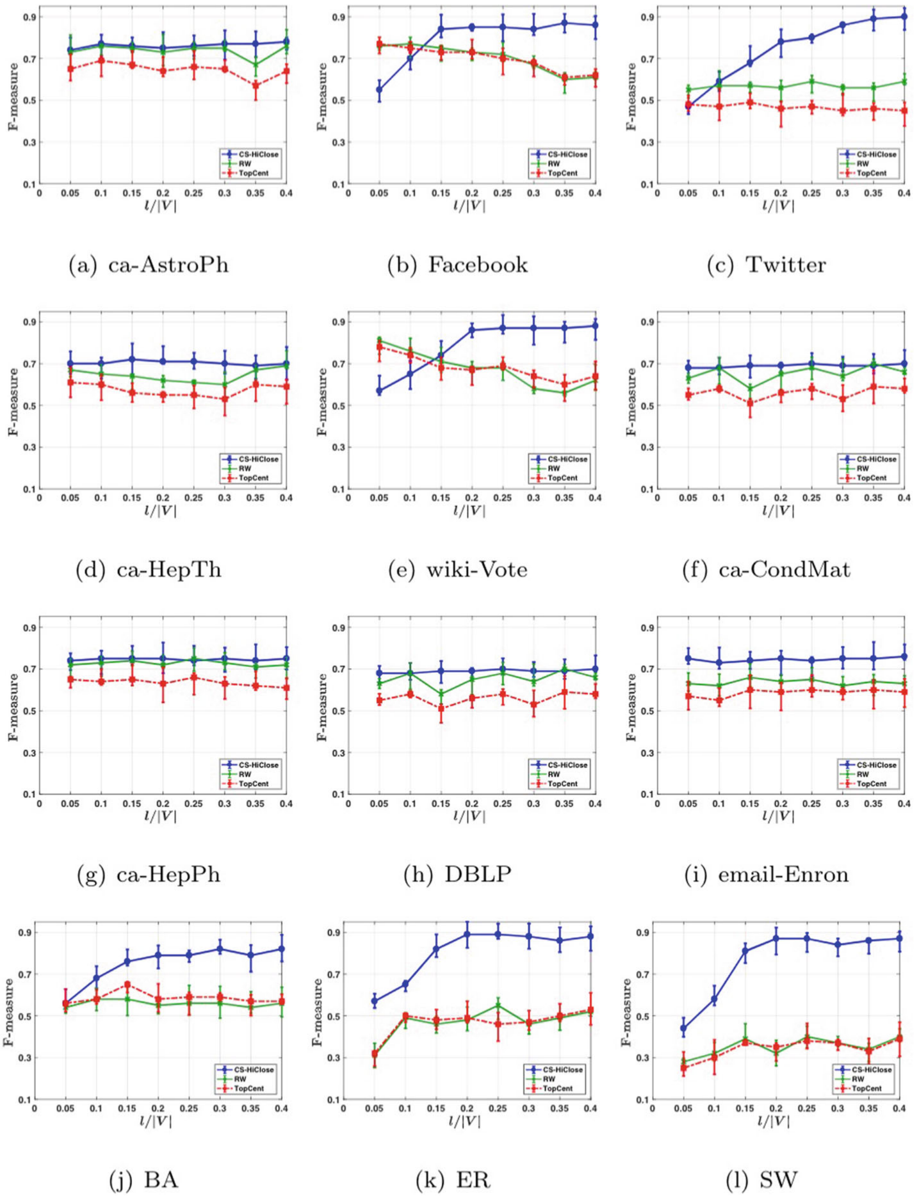
(d) ca-HepTh (e) wiki-Vote (f) ca-CondMat

(g) ca-HepPh (h) DBLP (i) email-Enron

(j) BA (k) ER (l) SW

Fig. 4 Effect of the required number of measurements *m* on the accuracy of CS-HiClose in terms of F-measure, compared to RW, TopCent, and DICeNod. For each method, we set the measurements length to 0.25|*V*| and the sparsity to 0.15|*V*| in a network with |*V*| nodes

closeness centrality. Then, we proposed a compressive sensing framework for distributed detection of top-*k* central nodes based on the ego-closeness metric using only indirect measurements. Extensive experimental evaluations on both synthetic and real networks demonstrated that the proposed method outperforms the best existing methods to efficiently detect high closeness centrality nodes, in terms of having high F-measure and low complexity. The experimental results indicated that our proposed ego-centric metric depicts a lower correlation with the global closeness network centrality on networks with low average degree relative to their network's size. Generalization of our ego-centric metric to address this limitation will be of interest for future work.



(a) ca-AstroPh (b) Facebook (c) Twitter

(d) ca-HepTh (e) wiki-Vote (f) ca-CondMat

(g) ca-HepPh (h) DBLP (i) email-Enron

(j) BA (k) ER (l) SW

Fig. 5 Effect of measurement length *l* on the accuracy of CS-HiClose in terms of F-measure, compared to RW, TopCent, and DICeNod. For each method, we set the number of measurements to 0.4|*V*| and the sparsity level to 0.2|*V*| in a network with |*V*| nodes

Acknowledgements

The author is grateful to the anonymous reviewers for their helpful comments to improve the manuscript. The authors acknowledge the TU Wien Bibliothek for financial support through its Open Access Funding Program.

Authors' contributions

HM implemented and measured most of the algorithms and contributed to the experimentation and generated most of the figures. RH contributed to the theoretical analysis and algorithm description. HES contributed with ideas. All authors read and approved the final manuscript.

Availability of data and materials

The datasets analysed during the current study are available in the SNAP repository, http://snap.stanford.edu/data/index.html.

Competing interests

The authors declare that they have no competing interests.

Author details
[1]Boston University, Boston, USA. [2]ETH, Zurich, Switzerland. [3]TU Wien, Vienna, Austria.

Received: 28 February 2019 Accepted: 20 September 2019
Published online: 06 November 2019

References

Barabasi AL, Albert R (1999) Emregence of scaling in random networks. Science 286(5439):509–512
Benesty J, Chen J, Huang Y, Cohen I (2009) Noise reduction in speech processing. Springer Science & Business Media Vol. 2. pp 1–4
Erdos P, Renyi A (1960) On the evolution of random graphs. In: Publication of the Mathematical Institute of the Hungarian Academy of Science. pp 17–61
Ghalebi E, Mahyar H, Grosu R, Rabiee HR (2017) Compressive sampling for sparse recovery in networks. In: Proc of the 23rd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 13th International Workshop on Mining and Learning with Graphs, Halifax, Nova Scotia, Canada. pp 1–8
Grosu R, Ghalebi E, Movaghar A, Mahyar H (2018) Compressed sensing in cyber physical social systems. In: Principles of Modeling. pp 287–305
Kim H, Yoneki E (2012) Influential neighbours selection for information diffusion in online social networks. In: ICCCN. pp 1–7
Leskovec J, Huttenlocher D, Kleinberg J (2010) Predicting positive and negative links in online social networks. In: WWW. pp 641–650
Leskovec J, Kleinberg J, Faloutsos C (2007) Graph evolution: Densification and shrinking diameters. ACM TKDD 1(1):2
Leskovec J, Lang KJ, Dasgupta A, Mahoney MW (2009) Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. Internet Math 6(1):29–123
Mahyar H (2015) Detection of top-k central nodes in social networks: A compressive sensing approach. In: IEEE/ACM ASONAM, Paris, France. pp 902–909
Mahyar H, Ghalebi E, Rabiee H, Grosu R (2017) The bottlenecks in biological networks. In: Proc of the 34th International Conference on Machine Learning (ICML), Computational Biology Workshop, Sydney, Australia. pp 1–5
Mahyar H, Hasheminezhad R, Ghalebi E, Grosu R, Stanley HE (2018) A compressive sensing framework for distributed detection of high closeness centrality nodes in networks. In: International Conference on Complex Networks and Their Applications. pp 91–103
Mahyar H, Hasheminezhad R, Ghalebi E, Nazemian A, Grosu R, Movaghar A, Rabiee HR (2018) Compressive sensing of high betweenness centrality nodes in networks. Physica A Stat Mech Appl 497:166–184
Mahyar H, Hasheminezhad R, Ghalebi E, Nazemian A, Grosu R, Movaghar A, Rabiee HR (2018) Identifying central nodes for information flow in social networks using compressive sensing. Soc Netw Anal Min 8(1):33
Mahyar H, Rabiee HR, Hashemifar ZS (2013a) UCS-NT: An Unbiased Compressive Sensing Framework for Network Tomography. In: IEEE ICASSP, Canada. pp 4534–4538
Mahyar H, Rabiee HR, Hashemifar ZS, Siyari P (2013b) UCS-WN: An Unbiased Compressive Sensing Framework for Weighted Networks. In: CISS, USA
Mahyar H, Rabiee HR, Movaghar A, Hasheminezhad R, Ghalebi E, Nazemian A (2015a) A low-cost sparse recovery framework for weighted networks under compressive sensing. In: IEEE SocialCom, Chengdu, China. pp 183–190
Mahyar H, Rabiee HR, Movaghar A, Ghalebi E, Nazemian A (2015b) CS-ComDet: A compressive sensing approach for inter-community detection in social networks. In: IEEE/ACM ASONAM, France. pp 89–96
Middya R, Chakravarty N, Naskar MK (2017) Compressive sensing in wireless sensor networks–a survey. IETE Tech Rev 34(6):642–654
Opsahl T, Panzarasa P (2009) Clustering in weighted networks. Soc Net 31(2):155–163
Parikh N, Boyd S (2014) Block splitting for distributed optimization. Math Program Comput 6(1):77–102
POGS (2018) Proximal operator graph solver. In: http://foges.github.io/pogs/. Accessed Feb 2019
Saxena A, Gera R, Iyengar S (2017) Fast estimation of closeness centrality ranking. In: Proceedings of the 2017 IEEE/ACM ASONAM. pp 80–85
Schoch D (2015) A positional approach for network centrality. PhD thesis. Universität Konstanz, Konstanz
Taheri SM, Mahyar H, Firouzi M, Ghalebi E, Grosu R, Movaghar A (2017a) HellRank: a hellinger-based centrality measure for bipartite social networks. Soc Netw Anal Min 7(1):22
Taheri SM, Mahyar H, Firouzi M, Ghalebi K E, Grosu R, Movaghar A (2017b) Extracting implicit social relation for social recommendation techniques in user rating prediction. In: Proceedings of the 26th International Conference on World Wide Web Companion. pp 1343–1351
Wang W, Tang CY (2015) Distributed estimation of closeness centrality. In: Decision and Control (CDC), 2015 IEEE 54th Annual Conference On. pp 4860–4865
Wang M, Xu W, Mallada E, Tang Ak (2012) Sparse recovery with graph constraints: Fundamental limits and measurement construction. In: IEEE INFOCOM. pp 1871–1879
Watts DJ, Strogatz SH (1998) Collective dynamics of small-world networks. Nature 393(6684):440–442
Wehmuth K, Ziviani A (2012) Distributed assessment of the closeness centrality ranking in complex networks. In: Simp. Comp. Net. for Pract
Yang J, Leskovec J (2015) Defining and evaluating network communities based on ground-truth. Knowl Inf Syst 42(1):181–213
You K, Tempo R, Qiu L (2017) Distributed algorithms for computation of centrality measures in complex networks. IEEE TAC 62(5):2080–2094
Xu W, Mallada E, Tang A (2011) Compressive sensing over graphs. In: IEEE INFOCOM. pp 2087–2095

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.