Centrality Measures Based on Current Flow*

Ulrik Brandes and Daniel Fleischer**

Department of Computer & Information Science, University of Konstanz
Daniel.Fleischer@uni-konstanz.de

Abstract. We consider variations of two well-known centrality measures, betweenness and closeness, with a different model of information spread. Rather than along shortest paths only, it is assumed that information spreads efficiently like an electrical current. We prove that the current-flow variant of closeness centrality is identical with another known measure, information centrality, and give improved algorithms for computing both measures exactly. Since running times and space requirements are prohibitive for large networks, we also present a randomized approximation scheme for current-flow betweenness.

1 Introduction

Centrality measures are an important tool in network analysis [6]. In social, biological, communication, and transportation networks alike, it is important to know the relative structural prominence of nodes or links to identify key elements in the network. The structure of a network is represented by a graph, so we will speak of vertices and edges in the following.

In social network analysis [22], the two most frequently used measures are vertex betweenness and vertex closeness centrality. They are based on the assumption that information (or whatever else is the content of linkages) is transmitted along shortest paths. While betweenness centrality measures the degree to which a vertex is between pairs of other vertices, i.e. on shortest paths connecting them, closeness is just the inverse of the average distance to other vertices.

A common criticism for shortest-paths based measures is that they do not take into account spread along non-shortest paths, and are thus not appropriate in cases where link content distribution is governed by other rules [4]. A betweenness measure based on network flow has been proposed in [10], and recently a variation of betweenness based on the flow of electrical current has raised considerable attention [18].

We here generalize closeness in the latter spirit and proof that the resulting measure is exactly what is already known under the name of information centrality. Despite its wide recognition, information centrality is not frequently utilized because its foundations are not very intuitive and therefore hard to understand

* Research partially supported by DFG under grant Br$\sim$2158/1-2.
** Corresponding author.

V. Diekert and B. Durand (Eds.): STACS 2005, LNCS 3404, pp. 533–544, 2005.

by substantively oriented social scientists. Our new derivation thus provides an intuition that builds on well-known concepts and should therefore find easier reception.

Moreover, we give improved algorithms for computing current-flow based measures and describe a probabilistic approach for approximating them in large networks. The performance of the latter algorithm is evaluated on real-world and random instances.

2 Preliminaries

In this section, we recall basic definitions and facts about electrical networks (see, e.g., [3]). Throughout the paper, we only consider graphs $G = (V, E)$ that are simple, undirected, connected and have $n \geq 3$ vertices. An *electrical network* $N = (G; c)$ is such a graph together with positive edge weights $c : E \rightarrow \mathbb{R}_{>0}$ indicating the *conductance* or *strength* of an edge. Equivalently, the network can be defined in terms of positive edge weights $r : E \rightarrow \mathbb{R}_{>0}$ indicating the *resistance* or *length* of an edge, where conductance and resistance are related by $c(e) = 1/r(e)$ for all $e \in E$.

We are interested in how current flows through an electrical network. A vector $b : V \rightarrow \mathbb{R}$ called *supply* defines where current externally enters and leaves the network. A vertex $v \in V$ with $b(v) \neq 0$ is called an *outlet*; it is called a *source*, if $b(v) > 0$, and a *sink* otherwise. Since there should be as much current entering the network as leaving it, $\sum_{v \in V} b(v) = 0$ is required. Actually, we will only consider the case in which a unit current enters the network at a single source $s \in V$ and leaves it at a single sink $t \in V \setminus \{s\}$, i.e. we consider unit *st-supplies*

$$b_{st}(v) = \begin{cases} 1 & v = s, \\ -1 & v = t, \\ 0 & \text{otherwise} . \end{cases}$$

To account for the directionality of flow, each edge is given an arbitrary orientation. While the actual choice of orientation is of no importance, we denote by $\overrightarrow{e}$ the directed edge corresponding to the orientation of $e \in E$, and by $\overrightarrow{E}$ the set of all oriented edges.

Definition 1. *Let $N = (G; c)$ be an electrical network with supply b. A vector $x : \overrightarrow{E} \rightarrow \mathbb{R}$ is called* (electrical) current*, if it satisfies*

1. Kirchhoff's Current Law (KCL)

$$\sum_{(v,w) \in \overrightarrow{E}} x(v,w) - \sum_{(u,v) \in \overrightarrow{E}} x(u,v) = b(v) \qquad \textit{for all } v \in V \ ,$$

2. Kirchhoff's Potential Law (KPL)

$$\sum_{i=1}^{k} x(\overrightarrow{e_i}) = 0 \qquad \textit{for every cycle } e_1, \ldots, e_k \textit{ in } G \ .$$

Lemma 1. *For an electrical network $N = (G; c)$ and any supply b, there is a unique current $x : \overrightarrow{E} \to \mathbb{R}$.*

A value $x(\overrightarrow{e}) > 0$ is interpreted as current flowing in the direction of $\overrightarrow{e}$, whereas $x(\overrightarrow{e}) < 0$ denotes current flowing against the direction of $\overrightarrow{e}$. For an st-supply, the corresponding current is called an st-current and denoted by x_{st}.

Currents are related to *potential differences* (or *voltages*) $\hat{p} : \overrightarrow{E} \to \mathbb{R}$ by *Ohm's Law*, $\hat{p}(\overrightarrow{e}) = x(\overrightarrow{e})/c(e)$ for all $e \in E$. A vector $p : V \to \mathbb{R}$ is said to assign *absolute potentials* if $\hat{p}(v, w) = p(v) - p(w)$ for all $(v, w) \in \overrightarrow{E}$.

Lemma 2. *Let $N = (G; c)$ be an electrical network with supply b. For any fixed vertex $v_1 \in V$ and constant $p_1 \in \mathbb{R}$, there are unique absolute potentials $p : V \to \mathbb{R}$ with $p(v_1) = p_1$.*

Again, we use $\hat{p}_{st}$ and p_{st} to indicate that the potential differences and absolute potentials are based on an st-supply. Potentials are easily computed from a given current and vice versa.

Absolute potentials can be computed directly using the *Laplacian* matrix $L = L(N)$ of $N = (G; c)$ defined by

$$L_{vw} = \begin{cases} \sum_{e\,:\,v \in e} c(e) & \text{if } v = w \\ -c(e) & \text{if } e = \{v, w\} \\ 0 & \text{otherwise} \end{cases}$$

for all $v, w \in V$. Note that the rows of L correspond to the left-hand side of KCL.

Lemma 3. *The absolute potentials of an electrical network $N = (G; c)$ with supply b are exactly the solutions of $Lp = b$.*

Since G is connected, the rank of L is $n - 1$ with a kernel spanned by $\mathbf{1} = (1, \ldots, 1)^{\mathrm{T}}$. This implies that any two assignments of absolute potentials differ only by an additive constant. Let there be a fixed vertex ordering $v_1, \ldots, v_n$ defining matrices and vectors. For brevity, we sometimes use i as an index instead of v_i. A way to choose an absolute potential is to fix, say, $p(v_1) = 0$, so that we obtain a restricted system

$$\widetilde{L}\widetilde{p} = \widetilde{b} \;,$$

where $\widetilde{L} \in \mathbb{R}^{n-1 \times n-1}$ is the matrix obtained from L by omitting the row and column of v_1, and $\widetilde{p}$ and $\widetilde{b}$ are obtained from p and b by omitting the entry of v_1. Since $\widetilde{L}$ is positive definite, and in particular regular, we get

$$p = \underbrace{\begin{pmatrix} 0 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & \widetilde{L}^{-1} \end{pmatrix}}_{=:\, C} \cdot b \;. \tag{1}$$

Matrix C will play a crucial role in computing centralities.

3 Current-Flow Measures of Centrality

Two of the most widely used centrality measures are based on a model of non-splitting information transmission along shortest paths. Note that in the following, distances may well be defined in terms of an edge length (or resistance) $r : V \to \mathbb{R}_{>0}$.

(Shortest-path) betweenness centrality [1, 9] $c_B : V \to \mathbb{R}_{\geq 0}$ is defined by

$$c_B(v) = \frac{1}{n_B} \sum_{s,t \in V} \frac{\sigma_{st}(v)}{\sigma_{s,t}}$$

where $\sigma_{s,t}$ denotes the number of shortest paths from s to t, $\sigma_{st}(v)$ denotes the number of shortest paths from s to t with v as an *inner vertex*, and $n_B = (n-1)(n-2)$ is a normalizing constant ($n_B = n(n-1)$ if v may also be a start or end vertex). It thus measures the degree to which a vertex is participating in the communication between pairs of other vertices.

(Shortest-path) closeness centrality [2] $c_C : V \to \mathbb{R}_{>0}$ is defined by

$$c_C(v) = \frac{n_C}{\sum\limits_{t \neq v} d_G(v,t)}$$

where $d_G(v,w)$ denotes the length of a shortest path between v and w and $n_C = n - 1$ is a normalizing constant. It thus measures the degree to which a vertex is close to other vertices (on average).

Both measures assume that information (or whatever else is being modeled) flows along shortest paths, and does not split. We next describe two alternative measures that build on the same intuition, but let information flow and split like current in an electrical network.

3.1 Current-Flow Betweenness Centrality

In electrical networks, the analog of the fraction of shortest st-paths passing through a vertex (or an edge) is the fraction of a unit st-current flowing through that vertex (or edge). Given a supply b, we therefore define the *throughput* of a vertex $v \in V$ to be

$$\tau(v) = \frac{1}{2}\left(-|b(v)| + \sum_{e : v \in e} |x(\overrightarrow{e})|\right) ,$$

where the term $-|b(v)|$ accounts for the fact that only inner vertices are considered in the definition of shortest-path betweenness centrality. To include start and end vertex, it should be replaced by $+|b(v)|$. Accordingly, the throughput of an edge $e \in E$ is defined as

$$\tau(e) = |x(\overrightarrow{e})| .$$

Let τ_{st} denote the throughput in case of an st-current.

Definition 2 ([18]). *Let* $N = (G; c)$ *be an electrical network.* Current-flow betweenness centrality $c_{CB} : V \rightarrow \mathbb{R}_{\geq 0}$ *is defined by*

$$c_{CB}(v) = \frac{1}{n_B} \sum_{s,t \in V} \tau_{st}(v) \qquad \textit{for all } v \in V \ ,$$

where $n_B = (n-1)(n-2)$.

Current-flow betweenness is well-defined because of Lemma 1. For the following reason, it is also called *random-walk betweenness.* A simple random st-walk is a random walk that starts at s, ends in t and continues at vertex $v \neq t$ by picking an incident edge $e \in E$ with probability $c(e) / \sum_{e' : v \in e'} c(e')$. Then, given an st-current, the amount of current flowing through a particular edge $\overrightarrow{e}$ equals the expected difference of the number of times that the simple random st-walk passes edge $\overrightarrow{e}$ along and against its orientation (see, e.g., [3]).

3.2 Current-Flow Closeness Centrality

Similar to the above variation of betweenness centrality, we utilize the analog of shortest-path distance in electrical networks to introduce a variant of closeness centrality.

Definition 3. *Let* $N = (G; c)$ *be an electrical network.* Current-flow closeness centrality $c_{CC} : V \rightarrow \mathbb{R}_{>0}$ *is defined by*

$$c_{CC}(s) = \frac{n_C}{\sum_{t \neq s} p_{st}(s) - p_{st}(t)} \qquad \textit{for all } s \in V \ .$$

Current-flow closeness centrality is well-defined, because by Lemma 2 any two absolute potentials differ only by an additive constant. Since we only consider unit st-currents, the term $p_{st}(s) - p_{st}(t)$ corresponds to the *effective resistance*, which can be interpreted as an alternative measure of distance between s and t.

Though not derived in the same fashion, it turns out that current-flow closeness has actually been considered before. *Information centrality* $c_I : V \rightarrow \mathbb{R}_{>0}$ is defined by

$$c_I(s)^{-1} = nC^I_{ss} + \mathrm{trace}(C^I) - \frac{2}{n} \ , \tag{2}$$

where $C^I = (L+J)^{-1}$ with Laplacian L and $J = \mathbf{1}\mathbf{1}^{\mathrm{T}}$ [20]. Information centrality is often referred to, but not frequently used; most likely because its underlying intuition is not widely understood.

Theorem 1. *Current-flow closeness centrality equals information centrality.*

Proof. We first note that Eq. (2) can be rewritten in terms of matrix elements only,

$$c_I(s)^{-1} = \sum_{t \in V} C^I_{ss} + C^I_{tt} - C^I_{st} - C^I_{ts} \ . \tag{3}$$

On the other hand, current-flow closeness can be rewritten into the same form, though with matrix C introduced in Eq. (1),

$$\begin{aligned} c_{CC}(s)^{-1} &= \sum_{t \neq s} p_{st}(s) - p_{st}(t) = \sum_{t \neq s} C_{ss} - C_{st} - (C_{ts} - C_{tt}) \\ &= \sum_{t \in V} C_{ss} + C_{tt} - C_{st} - C_{ts} \ . \end{aligned}$$

We show that these terms are actually equal using a matrix D with $C^I = C + D$ that contributes zero to the common summation scheme.

For $i = 1, \dots, n$ let $\boldsymbol{e_i} = (0, \dots, 0, 1, 0, \dots, 0)^{\mathrm{T}}$, with 1 in the ith position, and let $C_{\bullet i}$ and $C^I_{\bullet i}$ denote the ith column of the corresponding matrix. The columns of C are uniquely determined by

$$LC_{\bullet i} = \boldsymbol{e_i} - \boldsymbol{e_1} \quad \text{and} \quad C_{1i} = 0$$

and those of C^I satisfy

$$(L + J)\, C^I_{\bullet i} = \boldsymbol{e_i} \tag{4}$$

by definition. Projecting Eq. (4) onto the kernel of L, i.e. multiplying both sides with $\frac{1}{n}\mathbf{1}\mathbf{1}^{\mathrm{T}}$ from the left, yields

$$JC^I_{\bullet i} = (\mathbf{1}^{\mathrm{T}} C^I_{\bullet i})\, \mathbf{1} = \frac{1}{n}\mathbf{1} \ .$$

Eq. (4) is therefore equivalent to

$$LC^I_{\bullet i} = \boldsymbol{e_i} - \frac{1}{n}\mathbf{1} \quad \text{and} \quad \mathbf{1}^{\mathrm{T}} C^I_{\bullet i} = \frac{1}{n} \ .$$

Now let q be a vector with $Lq = L(C^I_{\bullet i} - C_{\bullet i}) = \boldsymbol{e_1} - \frac{1}{n}\mathbf{1}$. Then we have $C^I_{\bullet i} = C_{\bullet i} + q + d_i \mathbf{1}$ for some constants d_i (choosing q such that $q_1 = 0$ yields $d_i = C^I_{1i}$). In matrix notation we thus obtain $C^I = C + D$ with

$$D = \begin{pmatrix} q_1 + d_1 & q_1 + d_2 & \cdots \\ q_2 + d_1 & q_2 + d_2 & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix} \ .$$

It is easily verified that D contributes zero when subjected to the summation scheme of (3). $\square$

4 Improved Exact Computation

For comparison note that shortest-path betweenness and closeness can be computed in $\mathcal{O}(nm + n^2 \log n)$ time and $\mathcal{O}(n + m)$ space using an efficient implementation of Dijkstra's algorithm [5].

For current-flow betweenness centrality, matrix C defined in Eq. (1) is determined by inverting the reduced Laplacian. Since $p_{st} = Cb_{st}$ and $x(v,w) = (p(v) - p(w)) \cdot c(\{v,w\})$, we can use the incidence matrix $B = B(N) \in \mathbb{R}^{n \times m}$, defined by

$$B_{ve} = \begin{cases} c(e) & \text{if } \overrightarrow{e} = (v,w) \text{ for some } w \\ -c(e) & \text{if } \overrightarrow{e} = (u,v) \text{ for some } u \\ 0 & \text{otherwise ,} \end{cases}$$

to compute st-currents $x_{st} = BCb_{st}$. From the entries of *current-flow matrix* $F = BC$ the centrality scores are then determined via

$$\begin{aligned} c_{CB}(v) &= \frac{1}{n_B} \sum_{s,t \in V} \tau_{st}(v) \\ &= \frac{1}{n_B} \sum_{s,t \in V} \frac{1}{2} \left(-|b_{st}(v)| + \sum_{e \,:\, v \in e} |x_{st}(\overrightarrow{e})| \right) \\ &= \frac{1}{2-n} + \frac{1}{n_B} \sum_{s,t \in V} \sum_{e \,:\, v \in e} \frac{1}{2} |F_{es} - F_{et}| \\ &= \frac{1}{2-n} + \frac{1}{n_B} \sum_{s<t \in V} \sum_{e \,:\, v \in e} |F_{es} - F_{et}| \ , \end{aligned}$$

where $v_i < v_j$ if and only if $i < j$ (recall that we assume a fixed vertex ordering). The total time to compute current-flow betweenness is thus in $\mathcal{O}(I(n-1) + mn^2)$ [18], where $I(n) \in \mathcal{O}(n^3)$ is the time required to compute the inverse of an $n \times n$-matrix. Note that $I(n) \in \Omega(n^2 \log n)$ for arbitrary real matrices.

This can be improved as follows (see Alg. 1).

Theorem 2. *Current-flow betweenness can be computed in $\mathcal{O}(I(n-1) + mn \log n)$ time.*

Proof. We refer to Alg. 1. We can compute $c_{CB}(v)$ by summing up only the *inflows*, i.e. positive current on an edge directed to v or negative current on an edge leaving v, as follows. Note that for every non-outlet the inflow is equal to the outflow by KCL. We will later take care of the outlets. The *total inflow* τ_{in} into v equals

$$\begin{aligned} \tau_{\text{in}}(v) = \frac{1}{n_B} \sum_{(v,w) \in \overrightarrow{E}} \sum_{\substack{s<t: \\ F_{es} < F_{et}}} |F_{es} - F_{et}| + \frac{1}{n_B} \sum_{(w,v) \in \overrightarrow{E}} \sum_{\substack{s<t: \\ F_{es} > F_{et}}} |F_{es} - F_{et}| \\ = \frac{1}{n_B} \sum_{(v,w) \in \overrightarrow{E}} \sum_{i=1}^{n} (i - \text{pos}(\{v,w\}, v_i)) \cdot F_{ev_i} \\ + \frac{1}{n_B} \sum_{(w,v) \in \overrightarrow{E}} \sum_{i=1}^{n} (n + 1 - i - \text{pos}(\{w,v\}, v_i)) \cdot F_{ev_i} \ . \end{aligned}$$

Algorithm 1: Current-flow betweenness

Input: electrical network $N = (G; c)$ with vertices $v_1, \ldots, v_n$
Output: current-flow betweenness $c_{CB} : V \to \mathbb{R}_{\geq 0}$

begin
  $c_{CB} \leftarrow \mathbf{0}$
  $C \leftarrow \left(\begin{smallmatrix} 0 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & \widetilde{L}^{-1} \end{smallmatrix}\right)$
  for $e \in E$ **do**
    $F_{e\bullet} \leftarrow (BC)_{e\bullet}$
1.1   sort row $F_{e\bullet}$ in non-increasing order
    $\mathrm{pos}(e, v) \leftarrow$ rank of F_{ev} in sorted row $F_{e\bullet}$
    for $i = 1, \ldots, n$ **do**
      increase $c_{CB}(\mathrm{source}(\overrightarrow{e}))$ by $(i - \mathrm{pos}(e, v_i)) \cdot F_{ev_i}$
      increase $c_{CB}(\mathrm{target}(\overrightarrow{e}))$ by $(n + 1 - i - \mathrm{pos}(e, v_i)) \cdot F_{ev_i}$
  for $i = 1, \ldots, n$ **do**
1.2 $c_{CB}(v_i) \leftarrow (c_{CB}(v_i) - i + 1) \cdot 2/n_B$
end

Inflows include the vanishing unit current whenever v is the sink. In the summation over all pairs $s < t$ this will be the case $i-1$ times, namely when $v = v_i$. Note that the inflow of the source is always zero. Subtracting the vanishing currents from the total inflow yields half of the current-flow betweenness. The relation

$$c_{CB}(v_i) = 2(\tau_{\mathrm{in}}(v_i) - i + 1) ,$$

is accounted for in Line 1.2 of the algorithm.

The computational bottleneck after determining C by matrix inversion is the sorting of rows in Line 1.1, which takes $\mathcal{O}(mn \log n)$ time. Note that F is computed by multiplying C with an incidence matrix, so that it takes only $\mathcal{O}(mn)$ time. □

Information centrality can be computed by determining matrix C^I defined in the previous section and evaluating Eq. (2) [20]. The total running time is thus $\mathcal{O}(I(n) + n)$.

Using the new interpretation as current-flow closeness centrality, we see that it can also be determined from C rather than C^I (see Alg. 2). Thus sparseness is preserved and only one matrix inversion is required to compute both closeness and betweenness, which corresponds nicely to the fact that shortest-path betweenness and closeness can be computed during the same traversals.

A straightforward approach for matrix inversion uses Gaussian elimination, leading to a computation time of $\mathcal{O}(n^3)$. For networks from many application areas, however, sparse matrix techniques are appropriate. Since $\widetilde{L}$ is symmetric and positive definite, the conjugate gradient method (CGM) can be used with an incomplete LU-decomposition as a preconditioner. This yields a running time of

$\mathcal{O}(mn\sqrt{\kappa})$, where κ is the condition number of $\widetilde{L}$ times its approximate inverse obtained by applying the preconditioner. A rough estimate for the condition number is $\kappa \in \Theta(n)$ [12], leading to a running time of $\mathcal{O}(mn^{1.5})$ which is faster than the subsequent summation before its improvement to $\mathcal{O}(mn \log n)$.

The inverse of $\widetilde{L}$ can be computed column-by-column as needed in Line 2.1 of the algorithm for closeness centrality. Its memory requirement is in $\mathcal{O}(m)$.

For betweenness centrality, $\mathcal{O}(n^2)$ memory is required in the worst case. Here it is the current-flow matrix F that is processed row-by-row, implying that columns of $\widetilde{L}^{-1}$ corresponding to vertices u and w with $\{u, w\} \in E$ are needed simultaneously. Therefore, the column $v \in V$ needs to be determined only when the first row $F_{e\bullet}$ with $v \in e$ is encountered, and it can be dropped from memory when the last such row has been processed.

To reduce the memory requirements of Alg. 1, we therefore seek an ordering that minimizes the maximal number of columns that have to be kept in memory at the same time. That is, we would like to determine a one-to-one mapping $\pi : V \to \{1, \dots, n\}$ where

$$\delta(\pi) = \max_{1 \leq i \leq n} |\{u \in V \,:\, \exists\, w \in V, \{uw\} \in E \;\text{ with } \pi(u) \leq i < \pi(w)\}| \leq n$$

is minimum. Unfortunately, this is an $\mathcal{NP}$-hard problem known as *vertex separation* [17], or, equivalently [15], *minimum pathwidth.*

Heuristically, we can find a good ordering π^* by using algorithms for bandwidth- and envelope-reduction of matrices, since the bandwidth (of the Laplacian matrix of N ordered by π^*) is an upper bound for $\delta(\pi)$. Algorithm 1 is easily modified to use any precomputed ordering. The proven reverse Cuthill-McKee heuristic [7]) does not increase the asymptotic running time, while it reduces the memory requirement to $\mathcal{O}(\delta(\pi^*)n)$. Note that it can also be employed in the inversion of $\widetilde{L}$.

Algorithm 2: Current-flow closeness

Input: electrical network $N = (G; c)$
Output: current-flow closeness $c_{CC} : V \to \mathbb{R}_{>0}$

begin
    $c_{CC} \leftarrow \mathbf{0}$
    for $v \in V$ **do**
2.1 $C_{\bullet v} \leftarrow \begin{pmatrix} 0 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & \widetilde{L}^{-1} \end{pmatrix}_{\bullet v}$
        for $w \in V$ **do**
            increase $c_{CC}(v)$ by $C_{vv} - 2C_{wv}$
            increase $c_{CC}(w)$ by C_{vv}
    for $v \in V$ **do**
        $c_{CC}(v) \leftarrow 1/c_{CC}(v)$
end

5 Probabilistic Approximation

In large networks, both running time and space requirements of the algorithm for current-flow betweenness are prohibitive. Note that shortest-path closeness can be approximated quickly [21].

We show that a similar approach can be used to reduce not only the running time, but also the space requirements of (approximate) current-flow betweenness computations. For large data sets, this is often even more important.

The basic idea is that the betweenness of a vertex, i.e. the throughput over all st-currents, can be approximated using a small fraction of all pairs $s \neq t \in V$. A fully polynomial randomized approximation scheme is given in Alg. 3.

Theorem 3. *There is a randomized algorithm that, in $\mathcal{O}(\frac{1}{\varepsilon^2} m \sqrt{\kappa} \log n)$ time and $\mathcal{O}(m)$ space, approximates current-flow betweenness to within an absolute error of ε with high probability.*

Proof. Let $X_v^{(1)}, \ldots, X_v^{(k)}$ be independent random variables that return $\tau_{st}(v)$, for a pair $s \neq t \in V$, picked uniformly at random. With $c^* = n(n-1)/n_B$,

$$E\left(\frac{c^*}{k}\sum_{i=1}^{k} X_v^{(i)}\right) = c^* E(X_v^{(1)}) = \frac{1}{n_B}\sum_{s\in V}\sum_{t\neq s}\tau_{st}(v) = c_{CB}(v) \ ,$$

i.e. the scaled expected throughput of k st-currents is equal to the current-flow betweenness. Since $0 \leq \tau_{st}(v) \leq 1$, Hoeffding's bound [14] gives

$$\mathbb{P}\left(\left|\frac{c^*}{k}\sum_{i=1}^{k} X_i^v - c_{CB}(v)\right| \geq \varepsilon\right) \leq 2\exp\left(-2(\varepsilon/c^*)^2 k\right) \leq \frac{2}{n^{2\ell}}$$

when choosing $k = \ell \cdot \lceil (c^*/\varepsilon)^2 \log n \rceil$ pairs for arbitrary ℓ.

For each selected pair $s \neq t \in V$, the restricted system in Line 3.1 of Alg. 3 can be solved in $\mathcal{O}(m\sqrt{\kappa})$ time and $\mathcal{O}(m)$ space using CGM. □

Algorithm 3: Randomized approximation scheme for current-flow betweenness

---

Input: electrical network $N = (G; c)$, threshold $\varepsilon > 0$, constant ℓ
Output: current-flow betweenness approximation $c'_{CB} : V \to \mathbb{R}_{\geq 0}$

begin
  $c'_{CB} \leftarrow \mathbf{0}$ and $k \leftarrow \ell \cdot \lceil (c^*/\varepsilon)^2 \log n \rceil$
  for *i=1,... k* **do**
    select $s \neq t \in V$ uniformly at random and solve $\widetilde{L}\widetilde{p} = \widetilde{b}_{st}$
3.1   **for** $v \in V \setminus \{s, t\}$ **do**
      for $e = \{v, w\} \in E$ **do** increase $c'_{CB}(v)$ by $c(e) \cdot |\widetilde{p}(v) - \widetilde{p}(w)| \cdot c^*/2k$
end

---

6 Discussion

Current-flow betweenness and closeness are variants of (shortest-path) betweenness and closeness centrality for an alternative model of information spreading. In particular, we introduced current-flow closeness and proved that it is equal to information centrality, the original definition of which is rather unintuitive.

There is one and only one path between each pair of vertices in a tree, and the length of this path equals its resistance. We thus have the following result.

Theorem 4. *The two shortest-path and current-flow measures agree on trees.*

Corollary 1. *Betweenness and closeness can be computed in* $\mathcal{O}(n)$ *time and space on trees.*

Proof. A bottom-up followed by a top-down traversal similar to [19]. □

Finally, we want to remark that there is a straightforward extension of shortest-path betweenness to edges (simply replace the numerators by the number of shortest st-paths that use the edge) [1]. A similar extension of current-flow betweenness, that can be computed by slight modification of Alg. 1, is given by

$$c_{CB}(e) = \frac{1}{n_B} \sum_{s \neq t \in V} \tau_{st}(e) \qquad \text{for all } e \in E \ .$$

References

1. Anthonisse, J.M.: The rush in a directed graph. Technical Report BN 9/71, Stichting Mathematisch Centrum, Amsterdam (1971)
2. Beauchamp, M.A.: An improved index of centrality. Behavioral Science **10** (1965) 161–163
3. Bollobás, B.: Modern Graph Theory. Springer (1998)
4. Borgatti, S.P.: Centrality and Network Flow. Social Networks (to appear)
5. Brandes, U.: A faster algorithm for betweenness centrality. Journal of Mathematical Sociology **25** (2001) 163–177
6. Brandes, U., Erlebach, T., Eds.: Network Analysis. Springer LNCS (to appear)
7. Cuthill, E.H., McKee, J.: Reducing the bandwidth of sparse symmetric matrices. Proceedings of the 24th ACM National Conference (1969) 157–172
8. Diaz, J., Petit, J., Serna, M.: A Survey of Graph Layout Problems. ACM Comput. Surv. **34** (2002) 313–356
9. Freeman, L.C.: A set of measures of centrality based on betweenness. Sociometry **40** (1977) 35–41
10. Freeman, L.C., Borgatti, S.P., White, D.R.: Centrality in valued graphs: A measure of betweenness based on network flow. Social Networks **13** (1991) 141–154
11. Godsil, C., Royle, G.: Algebraic Graph Theory. Springer (2001)
12. Golub, G.H., van Loan, C.F.: Matrix Computations. Johns Hopkins University Press (1983)
13. Hackbusch, W.: Iterative Solution of Large Sparse Systems of Equations. Springer (1994)

14. Hoeffding, W.: Probability inequalities for sums of bounded random variables. Journal of the American Statistical Association **58** (1963) 13-30
15. Kinnersley, N.G.: The vertex separation number of a graph equals its path width. Information Processing Letters **42** (1992) 345–350
16. Kirchhoff, G.: Über die Auflösung der Gleichungen, auf welche man bei der Untersuchung der Linearen Vertheilung galvanischer Ströme geführt wird. Ann. Phys. Chem. **72** (1847) 497–508
17. Lengauer, T.: Black-white pebbles and graph separation. Acta Informatica **16** (1981) 465–475
18. Newman, M.E.J.: A Measure of betweenness centrality based on random walks. http://arxiv.org/abs/cond-mat/0309045 (2003)
19. Rosenthal, A., Pino, J.A.: A generalized algorithm for centrality problems on trees. Journal of the ACM **36** (1989) 349–381
20. Stephenson, K.A., Zelen, M.: Rethinking centrality: methods and examples. Social Networks **11** (1989) 1–37
21. Wang, J., Eppstein, D.: Fast approximation of centrality. Proceedings of the 12th ACM-SIAM Symposium on Discrete Algorithms (2001) 228–229
22. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Cambridge University Press (1994)

A Experimental Evaluation

We provide empirical evidence that the proposed algorithms for (approximate) computation of current-flow betweenness is practical. It has been implemented in Java using the yFiles[1] graph data structure and the JMP[2] linear algebra package. All experiments were performed on a regular PC with 2.4 GHz clock-speed and 3 GB main memory. Constant $\ell = 1$ for the approximation. See Fig. 1.

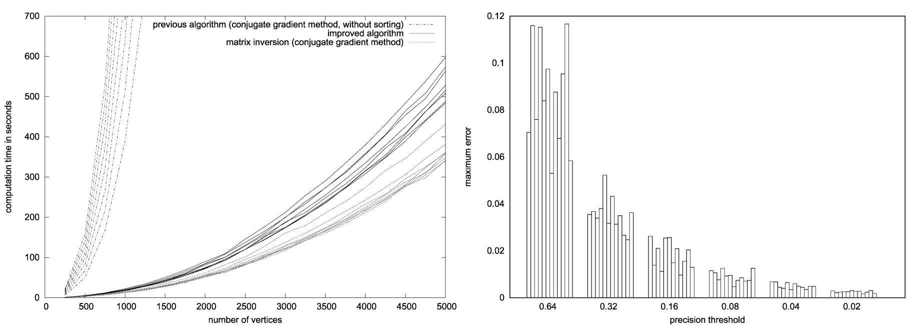

Fig. 1. Comparison of total running time for current-flow betweenness on random graphs with average degree $6, 8, \ldots, 20$ and maximum error of approximation on 6 random and 6 AS graphs with approximately 9000 vertices and 20000 edges each

[1] www.yworks.de
[2] www.math.uib.no/~bjornoh/jmp/index2.html