Cycle Breaking in Proxy Voting

Ben Chesnut, Jeremy Fox, Yixin Lin [*]

May 1, 2016

1   Introduction

Abstract

Classical voting schemes are an important part of our society but are fraught with problems, including problematic voter turnout, ill-informed voters, and illiquid representation. Proxy voting is a voting scheme that aims to alleviate such problems by providing a way for voters to delegate their vote to another voter. However, proxy voting comes with its own set of problems, especially when cycles are formed in voting delegation. We formalize proxy voting using graph theory and explore the issues of cycle-breaking, deriving an impossibility result and an approach to cycle-breaking using concepts social network theory.

1.1   Motivation

1.1.1   Problems with current voting schemes

Current voting schemes underly the foundations of government and society, but often come with systemic issues.

For example, voter turnout may be problematic due to a variety of issues. This is empirically true in the US voting system[1], which has lower voter turnout than most established democracies.

Rational ignorance[2], or the concept that it may be rational to forgo acquiring information depending on the costs incurred during acquisition, may be a factor. Regardless of the rationality behind it, ill-informed voting is a possibility difficult to avoid without raising the specter of literacy tests and their associated history.

Finally, the current voting scheme is has illiquid representation. There is arbitrary pooling and condensing of voting from systems like the US electoral college and easily-gerrymandered district boundaries, where external factors magnify and dilute the voting power of certain constituents over others.

---

[*]Vince Conitzer, Department of Computer Science, Duke University, Durham, NC 27708-0129, USA;

1.1.2 Introduction to Proxy Voting

Proxy voting is the voting scheme that one voter can delegate their voting power to another. This simple idea allows the sidestepping of many of the previous issues. The main reason is that it lowers the barrier to becoming a responsible voter drastically; to become a fully informed active voter is much harder than delegating to simply someone who is more informed. This recursive process has the natural tendency to shift voting power to more informed voters. In this sense, it is an extension of the idea of a republic with elected representatives, as opposed to a direct democracy.

2 Previous Work

Though proxy voting is a relatively new notion, inspired by the advancement of social networking theory and large social networks such as Facebook, Twitter, and LinkedIn, a significant amount of research from a wide range of academic angles has been put into this idea. Different works include exploring ways of implementing proxy voting systems on a network using cryptography [7], axiomatic analysis of proxy voting and desired properties of a delegative voting mechanism [8], and mathematical and algorithmic analysis of a proxy voting system in the form of graph theory [9]. In our paper, we bridge these last two ideas by defining a set of desired properties of a delegative voting scheme, showing that it is impossible to devise a proxy voting mechanism that meets all of the proposed criteria, and developing a mechanism that most optimally satisfies these properties while remaining computationally efficient.

3 Our Work

We start with some notation. Let $a \to b$ denote that voter a proxies to voter b. Let $G = (V, E)$ denote the graph of proxy votes, where V is the set of all voters, and E is the set of all edges.

3.1 Formalization as Graph Theory

To visualize and analyze a proxy voting system, we can represent the system as a graph, where each voter is a node having weight 1 and each proxy vote is a directed edge pointing from the proxying voter to the recipient of the proxy. As such, each node is a **source** of voting power, and voting power flows between nodes along edge paths, accumulating at nodes with no outgoing edge, or **sinks**, which represent voters who cast a vote instead of proxy. Each voter has voting power equal to its own weight of 1, plus the sum of the voting power of the nodes pointing to it. We can formally define the voting power $p(v)$ of voter v recursively:

Definition 3.1. Voting Power of a voter: For a voter v, $\quad p(v) = 1 + \sum_{\{i \in V \mid (i,v) \in E\}} p(i)$

Furthermore, we can define the voting power $P(Q)$ of a set Q as follows:

Definition 3.2. Voting Power of a set: For $Q \subseteq V, P(Q)$ is the sum of the voting power of the sinks in Q.

Proxy voting graphs also have a few special properties. Particularly, each node in the graph can have at most one outgoing edge, or outdegree at most one. Graphs with this property are called **directed pseudoforests**. As a consequence, we know that proxy voting graphs are made up of connected components each of which have at most one cycle.

3.2 Existence of cycles

There are cases where we can guarantee no cycles, even reasonably common cases.

Let us define a competence measure $C_i \in \mathbb{R}$ for $i \in V$, which is the objective measure of voter i's competence in voting. Let the perceived competence measure of voter j perceived by voter i be c_{ij} for $i, j \in V$.

We assume throughout this paper that $i \to j \implies c_{ii} < c_{ij}$, i.e., voter i proxies to voter j only if i perceives j to have greater competence than itself.

At the ideal limit where actual competence equals perceived competence, we have the following monotonicity theorem:

Theorem 3.1. *If $C_i = c_{ji} \; \forall j, i \in V$, then G contains no cycles.*

Proof. We prove by contradiction

Suppose there exists C_i for $i \in \{1, 2, ..., n\}$ for which nodes i form a cycle $1 \to 2 \to ...n \to 1$. Then $c_{12} < c_{23} < ... < c_{(n-1)n} < c_{n1} \implies C_1 < C_2 < ... < C_n < C_1 \implies C_1 < C_1. \Rightarrow\Leftarrow$ $\qquad \square$

3.3 Cycle Formation

Let $\mathbb{S}$ denote the set of all possible proxy voting systems. Let $S \in \mathbb{S}$ denote an arbitrary proxy voting system.

Definition 3.3. Choice of Role: S exhibits Choice of Role if all voters in S can choose whether they vote or proxy.

Definition 3.4. Enfranchisement: S exhibits Enfranchisement if $\forall a \in V, P(V) > P(V \setminus \{a\})$

Definition 3.5. Determinism: S exhibits determinism if repeated application of S results in the same candidate being chosen each time.

Definition 3.6. Representativeness: For a voter v, if v voted, let A_v denote the set of candidates for which v voted ($A_v = \emptyset$ if v did not vote). If v did not vote, let H_v denote the set of candidates for which v would vote if v were to vote ($H_v = \emptyset$ if v did vote. Let c denote the candidate chosen by the voting rule in S. S exhibits Representativeness if $c \in \bigcup_{v \in V} A_v \cup H_v$

Lemma 3.2 (Cycles Form). *If no voter votes, then there must exist some cycle.*

Proof. We prove by contradiction: Suppose that G does not contain a cycle. Then G must consist only of trees, which implies $|E| \leq |V| - 1$. However, since nobody voted, everybody proxied, which implies that $|E| = |V|$. $\Rightarrow\Leftarrow$ $\qquad\square$

Theorem 3.3. *There exists no system S for which the properties Choice of Role, Enfranchisement, and Determinism, and Representativeness simultaneously hold.*

Proof. We prove by counterexample:

Let us assume that all voters proxy, which is allowed under Choice of Role. We have from Lemma 3.2 that G must contain at least one cycle and no voters. By enfranchisement, this cycle must have some voting power, and by determinism and representativeness, this voting power may not go to an arbitrary candidate.

However, let $A = \bigcup_{v \in V} A_v$ as defined in Representativeness. S must select a candidate in A, but by determinism and representativeness, it cannot pick one arbitrarily and there is no information to pick a candidate without forcing a voter to vote instead of proxy. Therefore, these properties are incompatible even with arbitrary S.

$\qquad\square$

3.4   Centrality

As we mentioned in the previous section, one way to deal with cycles in proxy voting is through **preference elicitation** – that is, querying voters to find out what their true preferences are, and changing their proxy into an actual vote. This leads us to an important question: How do we decide who to query? Clearly, we seek to minimize the number of users queried (this adheres most closely to Choice of Role, as well as minimizing operating costs), which leads us to assume that for each pseudotree in the graph, we should query only one voter — there vote will be allocated to the rest of the pseudotree. Thus, our task is as follows: for one pseudotree, determine the voter best suited to represent the whole tree. We assume that the only information we have is to whom someone proxied – thus, we must determine who to query based solely on the structure of the graph. In order to do this, we make one more assumption: A proxy from A to B represents a vote from A in B's competence. Our task now is to choose the most competent voter in each connected competent — a task for which centrality measures from social network theory are uniquely suited.

A centrality measure intuitively captures the idea of the most important vertex within a graph – in a social network, centrality measures the most influential nodes in the network.[4][5] Centrality is a well established notion, having been used in various network analyses ranging from which people are most likely to spread disease to analyzing dominance in primate communities. Furthermore, centrality has already been successfully used to break cycles in a pseudoforest.[6]

We draw upon all of this work to propose the following: Centrality serves as an ideal tool with which to break cycles.

Now, we can describe a simple cycle breaking algorithm: After all votes are cast, compute the centrality of every node in a cycle. For every pseudotree, query the node in the pseudo tree who the centrality measure ranks the highest. Use the vote of this queried node to represent the vote of every other node in the pseudotree.

Although any centrality measure could be used to rank nodes in a pseudotree, here we propose the use of Katz Centrality as we feel it is best suited to the nature of proxy voting. Katz Centrality ranks nodes proportional to the number of nodes in their subtree, while penalizing the contribution of distant nodes. Another way of thinking of the Katz Centrality of a node i is that it counts the number of walks of size k through G that end at i, penalizing each step by a factor of α, and then summing over all k.

Let A be the adjacency matrix of G, and let $\alpha \in (0,1)$ be some discounting factor. If $C_{Katz}(i)$ denotes the Katz Centrality of a node i, then $C_{Katz}(i)$ is defined as follows:

Definition 3.7. Katz Centrality: $C_{Katz}(i) = \sum\limits_{k=1}^{\infty} \sum\limits_{j=1}^{n} \alpha^k (A^k)_{ji}$

A vector of the Katz Centrality for all nodes in the graph is expressed as follows:

Theorem 3.4. $\overrightarrow{C_{Katz}} = ((I - \alpha A^T)^{-1} - I)\vec{I}$

Theorem 3.5. $\overrightarrow{C_{Katz}}$ can be computed directly in time $O(n^{2.373})$

Proof. The fastest known algorithm for matrix inverse runs in time $O(n^{2.373})$. Matrix addition can be done in linear time – thus, the inverse takes the longest of all operations in this equation, so the runtime is equal to that of the matrix inverse. $\square$

We can, however, exploit the structure of our graph to greatly improve this runtime bound:

Theorem 3.6. *For nodes in a pseudoforest,* $\overrightarrow{C_{Katz}}$ *can be computed in* $O(n \log n \log \log n \log \log \log n)$ *time.*

Proof. Let T be a pseudotree in G with a cycle C of length k inside T. Let c_i be a node in C, and let t_i be the tree rooted at c_i that has no nodes in C other than c_i. Clearly we can calculate the centrality of this tree in linear time by simply traversing the tree — thus, we can compress the tree into a single node connected to c_i that stores the centrality of the entire tree. We can call this node d_i. Since we will eventually calculate the centrality of c_i by multiplying d_i by α and adding it to c_i, we can go ahead and store the value of d_i in c_i. Thus, we can transform T into an equivalent cycle C', where c'_i represents a node in C', as follows: $\forall c_i \in C, c'_i = 1 + \alpha C_{Katz} d_i$. We have now reduced our problem to that of calculating the centrality of a cycle, which can be done as follows. Let C

be the following cycle: $c_0 \to c_1 \to c_2 \to c_3 \to ... \to c_{k-1} \to c_k \to c_0$. Normally, when calculating centrality, each node implicitly stores a 1. Now each node in the cycle stores the value c_i'. From here, we can exploit the cycle, reducing the infinite sum to a convergent geometric series as follows:

$$
\begin{aligned}
C_{Katz}(c_0) &= \sum_{k-1}^{\infty} \alpha^k (A^k)_{ji} \\
&= \sum_{j=0}^{\infty} (\alpha^k)^j \sum_{j=0}^{k-1} \alpha^i c'_{(k-1) \mod k} \\
&= \frac{\sqrt{\sum_{j=0}^{k-1} \alpha^i c'_{(k-1) \mod k}}}{1 - \alpha^k}
\end{aligned}
$$

Using this, we can directly compute the centrality of c_0. Then, we can relate the centrality of c_i to c_{i-1} as follows:

$$
\forall 1 < i \le k, \quad C_{Katz}(c_i) = c_i' + \alpha C_{Katz}(C_{i-1})
$$

Thus, using this equation, we can traverse the cycle and compute all the centralities. Multiplication can be done in $O(N \log N \log \log N)$[10], where N is the number of bits of the number being multiplied. In our case, the numbers being multiplied are the centralities of the trees feeding into the cycle – clearly, these numbers are upper bounded by n. Therefore, we have that $N = \log n$. Since we have to do a linear number of multiplications, we have that the entire algorithm runs in time $O(n \log n \log \log n \log \log \log n)$.

$\square$

4 Conclusion

4.1 Future Work

4.1.1 Confidence Networks

In this paper, we explored the idea of using centrality measures to break cycles, our reasoning being that a proxy to someone is an implicit vote of confidence in that person's expertise. If we view the graph of proxy votes as a graph where directed edges represent votes of confidence, then centrality is the natural choice to choose the experts in the graph. However, centrality measures do not require that nodes only "proxy" to one other node — having this as a requirement in proxy voting limits the information we can obtain from voters, as it forces the graph to be unnecessarily sparse. One potential way to get around this problem is through the use of *confidence networks*, where voters are allowed to list not

just one but multiple other voters in which they feel confident. In this case, centrality measures would be used, not as a way to deal with the unfortunate case of cycles, but rather as way to calculate the relative expertise of voters in a system. There are many different ways this could be used — for instance, votes could be weighted by the calculated expertise of the voters who cast them.

4.1.2 Behavioral Model of Voting

What factors lead people to proxy? Quantify various aspects such as confidence, trust, bias, and knowledge and generate functions that say to whom each voter should proxy.

4.2 Impact

In this paper, we explained the motivation behind proxy voting and explored the aspect of cycles — why they form and how to break them. Ultimately, in order to implement a proxy voting system, we need a way to handle cycles. Until now, the only method proposed to deal with cycles involved granting them zero voting-power. By proposing a cycle-breaking method that does not disenfranchise voters in a cycle, we provide allow for the use of proxy voting in situations where voters would be upset if their votes were discarded, such as modern political elections.

References

[1] Jackman RW. *Political institutions and voter turnout in the industrial democracies.* American Political Science Review. 1987 Jun 1;81(02):405-23.

[2] Downs. (1957), *An Economic Theory of Democracy*; New York: Harper & Brothers, 1957; p. 244-46, 266-71

[3] Jan Behrens and Bjorn Swierczek *Preferential Delegation and the Problem of Negative Voting Weight.* Berlin, January 23, 2015.
http://www.liquid-democracy-journal.org/

[4] Linton C. Freeman *Centrality in Social Networks: Conceptual Clarification* The Netherlands, 2878.
http://moreno.ss.uci.edu/27.pdf

[5] M.E.J. Newman *Networks: An Introduction* Oxford, UK: Oxford University Press, 2010.

[6] Paolo Boldi and Corrado Monti *Cleansing Wikipedia Categories using Centrality* IW3C2, Montreal, Quebec, Canada, April 15, 2016
http://snap.stanford.edu/wikiworkshop2016/papers/

[7] Yefim I. Leifman *Secret and Verifiable Delegated Voting for Wide Representation*
http://eprint.iacr.org/2014/351.pdf

[8] James Green-Armytage *Direct Voting and Proxy Voting* Department of Economics, Bard College, Annandale-on-Hudson, NY, 2014l
http://inside.bard.edu/ armytage/proxy.pdf

[9] Paolo Boldi and Francesco Bonchi and Carlos Castillo and Sebastiano Vigna *Voting in Social Networks* CIKM, Hong Kong, China, 2009
http://www-kdd.isti.cnr.it/ bonchi/votingSN.pdf

[10] A. Schnhage and V. Strassen *Schnelle Multiplikation groer Zahlen* Computing 7, 281-292, 1971