Optimized Graph-Based Trust Mechanisms using Hitting Times

Alejandro Buendia†
New York, NY 10027
alb2281@columbia.edu

Daniel Boley‡
Computer Science and Engineering
Minneapolis, MN 55455
boley@umn.edu

†Columbia University
‡University of Minnesota

Abstract

Trust mechanisms can be computed by modeling a set of agents in a trust graph, a directed weighted graph in which agents are represented as vertices and relative trust as edges. Liu, Parkes, and Seuken [LPS16] present trust mechanisms that are robust against sybil attacks, including the proposed personalized hitting time (PHT) mechanism. In this paper, we propose a scalable algorithm for producing trust scores. In particular, we improve on the theoretical bound quoted by [LPS16] for computing PHT scores and other hitting time-based trust mechanisms.

1 Introduction

Trust mechanisms aggregate a set of reports from agents in an attempt to rank the trustworthiness of individuals within a network. A *trust graph* represents agents by vertices and relative trust by edges, encapsulating the relationships between interacting agents. Examples are readily seen in real applications, including the modeling of friends on social networks like Facebook[1], citation networks in which vertices represent papers and edges represent citations, and commerce networks in which vertices represent agents transacting with each other in online systems like Amazon[2]. Examples of the use of directed graphs to model these and other networks can be found in the Stanford Large Network Dataset Collection[3]. One can extend the data presented here by assigning relative trust to each edge; for example, transacting users on Amazon may evaluate each other in terms of the success of previous transactions, producing a so-called trust graph.

1.1 Related Work

The use of networks to model relative trust between agents has produced the need for scalable methods to calculate trust scores. A trust mechanism aggregates these trust scores, and by distributing it to the agents, allows for agents to determine trustworthy users with which to transact in the future. Liu et al. [LPS16] consider various trust mechanisms under sybil attacks, in which an agent controls the reports of a number of fake agents called sybils. For example, one such sybil attack is the “two-loop attack” in which the agent assigns a high

In: J. Zhang, M. Sensoy, and R. Falcone (eds.): Proceedings of the 19th International Workshop on Trust in Agent Societies, Sao Paulo, Brazil on May 8-9, 2017, published at http://ceur-ws.org

[1]http://www.facebook.com/
[2]http://www.amazon.com/
[3]http://snap.stanford.edu/data/

trust score to its sybils, and the sybils assign a high trust score to its agent. The corruption of such trust mechanisms as PageRank under the two-loop attack has spurred the need for more robust mechanisms. Liu et al. [LPS16] consider various trust mechanisms in turn by evaluating their robustness against sybil attacks. Some trust mechanisms are *global mechanisms* in which a single trust score is assigned to each agent that reflects the aggregate reports about that agent from every other reporting agent in the network. *Personalized mechanisms* are also considered in which the same agent may have different scores from the perspective of multiple agents. For instance, agent v_i may assign a certain trust score to agent v_k that differs from the trust score assigned to v_k by v_j.

Liu et al. [LPS16] focus on the proposed *personalized hitting time* (PHT) mechanism after concluding that it is most robust against sybil attacks compared to three other mechanisms: global hitting time (GHT) [HS06], PageRank, and personalized versions of PageRank (PPR). An algorithm for computing exact PHT scores is introduced with a time complexity of $\mathcal{O}(n^4)$. To circumvent this costly computation, the authors present a Monte Carlo algorithm for approximating PHT scores. Such work as [HS10] alludes to similar difficulties in theoretical computation of trust propagation and trust-based recommendation because of complexities that grow quickly with the number of agents.

1.2 Outline

In this work, we examine the four trust mechanisms considered in [LPS16]. We present an $\mathcal{O}(n^3)$ algorithm that computes hitting time-based trust scores, improving on the theoretical bound quoted in [LPS16]. Such an algorithm allows for computation of hitting time-based trust mechanisms that are not only robust to sybil attacks but also run with reasonable complexity as the number of agents increases. We proceed in this paper as follows. In Section 2, we present notation and graph theoretic preliminaries used in later sections. In Section 3, we more explicitly define the problem and various hitting time-based trust scores. In Section 4, we present the new algorithm and show its application to the trust scores under consideration. In Section 5, we run the algorithm on a simple example network. We conclude in Section 6 and suggest a possible avenue for future work.

2 Preliminaries

We denote matrices by non-bold uppercase letters, vectors by bold lowercase letters, and scalars by non-bold lowercase letters.

2.1 Graph Theoretic Preliminaries

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ be a digraph with a collection of n vertices $i \in \mathcal{V}$ and set of directed edges $(i \to j) \in \mathcal{E}$. We use an $n \times n$ adjacency matrix $A = [a_{ij}]$ to represent the weights on the edges of the digraph such that $a_{ij} > 0$ is the weight on edge $(i \to j) \in \mathcal{E}$, and $a_{ij} = 0$ if $(i \to j) \notin \mathcal{E}$. We restrict our consideration to strongly connected digraphs. A strongly connected digraph is a digraph $\mathcal{G}$ such that there exists a path $i = l_0 \to l_1 \to \cdots \to l_{\kappa-1} \to l_\kappa = j$ for all pairs of nodes (i, j), where each link $(l_{\iota-1} \to l_\iota) \in \mathcal{E}$ for $\iota = 1, 2, ..., \kappa$. $|\mathcal{V}| = n$ is called the order of the digraph.

A random walk over a strongly connected digraph can be modeled by a first-order Markov chain [GS12]. Let $\mathbf{1}$ denote the vector of all 1s, and let $D = \text{Diag}(\mathbf{d}) = \text{Diag}(A \cdot \mathbf{1})$ denote the diagonal matrix of vertex outdegrees. The Markov chain is represented by a matrix of transition probabilities $P = [p_{ij}]$ given by $D^{-1}A$.

If the underlying digraph is strongly connected, it can be shown that the transition probability matrix is irreducible and the Markov chain is ergodic, meaning that it is possible to move from every state to every other state [BP94]. It can also be shown by Perron-Frobenius theory that there exists a unique stationary probability vector $\boldsymbol{\pi}$ with all positive entries [BP94, Sen81]. This stationary probability vector satisfies $\boldsymbol{\pi}^T P = \boldsymbol{\pi}^T$ and $\boldsymbol{\pi}^T \mathbf{1} = 1$.

2.2 Trust Graph Model

We model the problem with a trust graph, a directed graph which represents a set of agents and relative trust between agents. We borrow notation from [LPS16] in specifying the trust graph model. Let V denote the set of agents and $v_i \in V$ denote an individual agent. Consider two agents v_i and v_j in which agent v_i initiates a transaction with provider agent v_j. We denote the set of agents with which v_i has transacted by V_i and the set of agents about which v_i makes a report following a transaction by $\hat{V}_i$. Let $\hat{w}_{ij} \in [0, 1]$ be the report that v_i

makes about v_j following a transaction. Let W_i denote the set of trust reports that v_i makes about the other agents with which it has transacted.

A trust mechanism takes the set of reports in W_i for all agents $v_i \in V$, and returns a trust score $x_{ij} \in \mathbb{R}_{\geq 0}$ for all pairs of agents $v_i, v_j \in V$ with $v_i \neq v_j$. This trust score quantifies the trustworthiness of agent v_j as perceived by agent v_i. A global trust mechanism is one in which $x_{ij} = x_{kj}$ for all agents $v_i, v_j, v_k \in V$; an aggregate common trust score about v_j is reported by all agents. In a personalized trust mechanism, we may have that $x_{ij} \neq x_{kj}$ when v_i and v_k report different trust scores for v_j.

The trust graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ is constructed so that each agent in V corresponds to a vertex in $\mathcal{V}$. We assign weights $a_{ij} = \hat{w}_{ij} > 0$ to each edge $(i \to j)$ in $\mathcal{E}$, setting $a_{ij} = 0$ when v_i has given no report on v_j. Then the trust scores are defined in terms of a random walk over $\mathcal{G}$. Let $(X_0, X_1, ...)$ denote the sequence of random variables that form the random walk, with each $X_i \in \mathcal{V}$. This random walk is defined by a transition probability matrix $P = D^{-1}A$. The probability of transition from v_i to v_j is given by the ij-th entry of P, namely

$$P(X_{t+1} = v_j \mid X_t = v_i) = \frac{a_{ij}}{\sum\limits_k a_{ik}}.$$

In the manner of [LPS16], we impose additional conditions on the random walk. The vertex X_0 is distributed according to some chosen distribution F_q. The random walk under consideration is α-terminating, meaning that the random walk terminates with probability α at each step. We denote the sequence of vertices visited by the α-terminating random walk by $(X_t)_{t=0}^{\tau}$. The random walk length is thus a random variable distributed by $\tau \sim \text{Geom}(1-\alpha)$. The quantity of interest is the hitting time of this random walk, or mean first passage time [GS12], defined as the expected number of steps to first reach a specified agent when starting a random walk from a vertex sampled from F_q. If v_i is chosen as the starting vertex in the random walk, then we denote the hitting time to reach v_j as $H(v_i, v_j) = \min\{t : X_t = v_j \mid X_0 = v_i\}$.

3 Defining Trust Scores

3.1 Problem Statement

Given a weighted digraph $\mathcal{G}$ that represents a trust graph, we seek an efficient algorithm that calculates trust scores under various hitting time-based trust mechanisms, for all pairs of agents $v_i, v_j \in V$ with $v_i \neq v_j$. In particular, we seek an algorithm that improves on a previously quoted $\mathcal{O}(n^4)$ bound that still remains effective against sybil attacks.

3.2 Trust Scores Defined

In this paper, we focus on four trust mechanisms: the personalized hitting time (PHT) mechanism, global hitting time (GHT) mechanism, personalized PageRank (PPR) mechanism, and PageRank mechanism. We reprise definitions of each trust mechanism from [LPS16]. Assume an α-terminating random walk $(X_t)_{t=0}^{\tau}$ with $\alpha \in [0, 1]$.

Definition 1. *Personalized Hitting Time (PHT) Mechanism [LPS16]. The personalized hitting time score $x_{PHT,ij}$ of agent v_j as viewed from agent v_i is the probability that an α-terminating random walk that starts from v_i visits v_j before restarting. That is,*

$$x_{PHT,ij} = P(v_j \in (X_t)_{t=0}^{\tau} \mid X_0 = v_i) = P(H(v_i, v_j) \leq \tau).$$

Definition 2. *Global Hitting Time (GHT) Mechanism [HS06]. The global hitting time score $x_{GHT,j}$ of agent v_j is the probability that an α-terminating random walk starting at a vertex sampled from F_q visits v_j before restarting. That is,*

$$x_{GHT,j} = P(v_j \in (X_t)_{t=0}^{\tau} \mid X_0 \sim F_q) = P(H(X_0, v_j) \leq \tau).$$

Definition 3. *Personalized PageRank (PPR) Mechanism. The personalized PageRank score $x_{PPR,ij}$ of agent v_j as viewed from agent v_i is the steady-state probability that an α-terminating random walk that starts and restarts from v_i spends at v_j.*

Definition 4. *PageRank Mechanism. The global PageRank score $x_{PR,j}$ of agent v_j is the steady-state probability that an α-terminating random walk that starts at a vertex sampled from F_q spends at v_j.*

Note that the GHT and PageRank mechanisms are global mechanisms, in which the first vertex X_0 is sampled from a restart distribution F_q, giving an aggregate trust score for a single agent. The PHT and PPR mechanisms are personalized mechanisms, in which the first vertex X_0 is assigned to a fixed v_i, giving differing trust scores for a single agent depending on the viewpoint taken. Note that we use the definition of the personalized PageRank mechanism in [LPS16], although various similar alternatives exist in the literature (for example, [FRCS05, JW03, Lof15]).

In [LPS16], Liu et al. evaluate the robustness of these trust mechanisms against sybils by quantifying the effect of manipulation in terms of *influence*, defined in [HS06]. The trust mechanisms are evaluated in relation to optimal manipulations that an agent may pursue. In all cases, a strategic agent will optimally drop all trust reports about other agents, equivalent to cutting its outlinks in the trust graph. The authors conclude that the optimal strategy for an agent v_j with access to sybils is to add sybils for all trust mechanisms except for the PHT mechanism, in which case adding one or more sybils does not increase the trust score of the agent. Furthermore, both the PHT and GHT mechanisms are in effect more robust to sybil attacks because they are resistant to the two-loop attack previously discussed, while the PPR and PageRank mechanisms remain susceptible. Given the superiority of these hitting time-based metrics, we focus on computation of PHT and GHT scores.

4 The Algorithm: Computing Trust Scores

We propose an $\mathcal{O}(n^3)$ algorithm to compute PHT and GHT scores, and in general, any hitting time-based trust mechanism. We will also show how the algorithm efficiently computes PageRank and PPR scores. Consider a strongly connected trust graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ of order n with transition probability matrix P. P governs the transitions for the underlying Markov chain modeling the α-terminating random walk $(X_t)_{t=0}^{\tau}$, in which $\alpha \in [0, 1]$.

We first modify trust graph $\mathcal{G}$ by adding an "augmented" $(n+1)$-th vertex v_{n+1} to account for probability of termination α. Let there be an edge from every vertex in $\mathcal{G}$ to v_{n+1} with transition probability α. Add a self-loop to v_{n+1} with transition probability 1. Construct a modified $(n+1) \times (n+1)$ matrix of transition probabilities P', representing transitions in the augmented graph $\mathcal{G}'$. Note that the original probabilities in P are scaled by $(1-\alpha)$ so that P' remains stochastic. P' can be partitioned as follows:

$$P' = \begin{bmatrix} (1-\alpha)P & \boldsymbol{\alpha} \\ \mathbf{0}^T & 1 \end{bmatrix}, \tag{1}$$

in which $\boldsymbol{\alpha}$ denotes an n-vector with each entry α and $\mathbf{0}$ denotes the n-vector of all 0s.

Using the prescription in [GS12], consider the $n \times n$ submatrix P. We can compute fundamental matrix N of the Markov chain for $\mathcal{G}'$ with absorbing state v_{n+1} by taking inverse $N = (I - (1-\alpha)P)^{-1}$. The ij-th entry of N gives the expected number of times that a random walk starting from vertex i will pass through vertex j when absorbed by implicit state v_{n+1}. We find inverse $(I - (1-\alpha)P)^{-1}$ using Gaussian elimination, an $\mathcal{O}(n^3)$ process. Note that a vector of hitting times is given by $\mathbf{h} = N \cdot \mathbf{1}$ in which $\mathbf{1}$ is the vector of all 1s. It is evident that the i-th entry of $\mathbf{h}$ is equivalent to the previously defined $H(v_i, v_{n+1})$. We now have the quantities used to compute the trust mechanisms in [LPS16] and consider each one in turn.

4.1 Personalized Hitting Time (PHT) Mechanism

Consider the absorbing Markov chain for augmented graph $\mathcal{G}'$ with matrix of transition probabilities P'. We seek to compute $x_{PHT,ij} = P(H(v_i, v_j) \leq \tau)$, in which τ is the length of the α-terminating random walk. We use the following theorem in the computation.

Theorem 1. *Let P be the transition matrix for a random walk with $n+1$ nodes. Let N with ij-th entry $N(i,j)$ be the fundamental matrix calculated for a single absorbing state, v_{n+1}. Then the probability that the random walk absorbed by v_{n+1} visits v_j starting from v_i is given by*

$$\textit{Pr(visiting } v_j \textit{ before } v_{n+1} \textit{ starting from } v_i\textit{)} = \frac{N(i,j)}{N(j,j)}. \tag{2}$$

Proof. Renumber the vertices so that v_j is ordered as the n-th vertex in transition matrix P to simplify the notation. Partition P as

$$P = \left[\begin{array}{cc|c} Q & \mathbf{r_1} & \mathbf{r_2} \\ \mathbf{s_1}^T & t_{11} & t_{12} \\ \hline \mathbf{s_2}^T & t_{21} & t_{22} \end{array}\right],$$

where Q is $(n-1)\times(n-1)$, $\mathbf{r_1}, \mathbf{r_2}$ are column $(n-1)$-vectors, and $\mathbf{s_1}^T, \mathbf{s_2}^T$ are row $(n-1)$-vectors.

To retrieve the count $N(i,j)$, let vertex $n+1$ be the single absorbing state and follow the prescription in [GS12] to calculate the matrix N, whose ij-th entry is the expected number of passages through v_j when starting a random walk from v_i. Then

$$N = \begin{bmatrix} (I-Q) & -\mathbf{r_1} \\ -\mathbf{s_1}^T & 1-t_{11} \end{bmatrix}^{-1} = \begin{pmatrix} W & \mathbf{x} \\ \mathbf{y}^T & z \end{pmatrix}, \tag{3}$$

partitioned conformally (so that z is a scalar). In particular, $x_i = N(i,n)$ and $z = N(n,n)$. From (3) we have $(I-Q)\mathbf{x} - z\mathbf{r_1} = \mathbf{0}$, or

$$\mathbf{x} = z(I-Q)^{-1}\mathbf{r_1}. \tag{4}$$

Now let v_n, v_{n+1} both be absorbing states. Follow [GS12] to find the vector of probabilities of reaching n before $n+1$ (starting from any vertex $i \in \{1,\ldots,n-1\}$), obtaining $\mathbf{b_1} = (I-Q)^{-1}\mathbf{r_1}$. From (4) this is just

$$\mathbf{b_1} = \frac{\mathbf{x}}{z} = \frac{N(:,n)}{N(n,n)},$$

in which $N(:,n)$ denotes the column vector of entries in N with second index n. Taking the i-th entry of $\mathbf{b_1}$ gives (2). □

Remark 1. *An alternative formulation of the above proof is as follows. By the Markov property, the behavior of the random walk at each discrete step is independent of the process up to that point. Hence, the average number of passages through v_j starting from v_i is equal to the probability that the random walk will reach v_j before absorption starting from v_i times the average number of passages through v_j once reached. That is,*

$$N(i,j) = Pr(\text{visiting } v_j \text{ before absorption in } v_{n+1} \text{ starting from } v_i) \times N(j,j).$$

Using Theorem 1, we see that the PHT score is given by

$$x_{PHT,ij} = \frac{N(i,j)}{N(j,j)}. \tag{5}$$

We vary the choice of v_i, v_j for all agents in V to compute all $\mathcal{O}(n^2)$ PHT scores.

4.2 Global Hitting Time (GHT) Mechanism

The GHT score is defined as $x_{GHT,j} = P(H(X_0, v_j) \le \tau)$, in which initial vertex X_0 is sampled from some restart distribution F_q. Again consider augmented graph $\mathcal{G}'$ with fundamental matrix N, in which v_{n+1} has been made the sole absorbing state. For the purposes of an example construction, we consider vertex X_0 to be sampled uniformly from the original vertex set $\mathcal{V}$, excluding v_{n+1}. Then F_q is a uniform distribution, and the GHT score is given by

$$x_{GHT,j} = \frac{1}{n-1}\sum_{i\neq j}\frac{N(i,j)}{N(j,j)}, \tag{6}$$

an average of $n-1$ PHT scores for which $i \neq j$. Varying the choice of v_j for all agents in V allows one to compute all $\mathcal{O}(n)$ GHT scores.

4.3 Personalized PageRank (PPR) Mechanism

Recall that the PPR score $x_{PPR,ij}$ of v_j from the perspective of v_i is the steady-state probability that an α-terminating random walk that starts and restarts from v_i spends at v_j. We compute this directly from the fundamental matrix N calculated above.

To compute the proportion of time spent at v_j when starting from v_i, consider that $N(i,j)$ gives the expected number of passages through v_j when starting the random walk from v_i. Then we can form the PPR score as

$$x_{PPR,ij} = \frac{N(i,j)}{\sum_{j=1}^{n} N(i,j)} = \frac{N(i,j)}{h_i}, \tag{7}$$

in which $\mathbf{h}$ is defined as above. Varying the choice of v_i, v_j for all agents in V allows one to compute all $\mathcal{O}(n^2)$ PPR scores.

4.4 PageRank Mechanism

Recall that the global PageRank score $x_{PR,j}$ of agent v_j is the steady-state probability that an α-terminating random walk starting at a vertex sampled from F_q spends at v_j. Again we consider the case in which the initial vertex is sampled uniformly from the original vertex set $\mathcal{V}$. Then, analogous to our formulation of GHT, we can simply take an average of $n-1$ PPR scores over all $i \neq j$. That is,

$$x_{PR,j} = \frac{1}{n-1} \sum_{i \neq j} \frac{N(i,j)}{\sum\limits_{j=1}^{n} N(i,j)} = \frac{1}{n-1} \sum_{i \neq j} \frac{N(i,j)}{h_i}. \tag{8}$$

Varying the choice of v_j for all agents in V allows one to compute all $\mathcal{O}(n)$ PageRank scores.

4.5 Formalization

We present formalized algorithms for computing PHT, GHT, PPR, and PageRank scores. We first summarize the process for computing fundamental matrix N used in the calculation.

Algorithm 1: Fundamental Matrix

Input: trust graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$
Output: fundamental matrix N for the augmented Markov chain

Compute $P = D^{-1}A$, with $D = \mathrm{Diag}(A \cdot \mathbf{1})$.
Augment $\mathcal{G}$ with $(n+1)$-th vertex and a new edge from every vertex to v_{n+1} to form $\mathcal{G}'$.
Scale entries in P according to (1) to form P'.
Compute inverse N of $n \times n$ submatrix $I - (1-\alpha)P$ via Gaussian elimination.

We present additional algorithms for computing the four trust scores for all pairs of agents v_i, v_j in the trust graph. In the cases of the GHT and PageRank mechanisms, let F_q be a uniform distribution over the original vertex set $\mathcal{V}$.

Algorithm 2: Personalized Hitting Time (PHT) Mechanism

Input: previously computed fundamental matrix N
Output: matrix of PHT scores X_{PHT}

$X_{PHT} \leftarrow 0$
for $i = 1...n$ **do**
  for $j = 1...n$ **do**
    if $i \neq j$ **then**
      $X_{PHT}(i,j) \leftarrow \frac{N(i,j)}{N(j,j)}$
    end
  end
end

Algorithm 3: Global Hitting Time (GHT) Mechanism

Input: previously computed fundamental matrix N
Output: vector of GHT scores $\mathbf{x}_{GHT}$

$\mathbf{x}_{GHT} \leftarrow \mathbf{0}$
for $j = 1...n$ **do**
  for $i = 1...n$ **do**
    if $i \neq j$ **then**
      $x_{GHT,j} \leftarrow x_{GHT,j} + \frac{N(i,j)}{N(j,j)}$
    end
  end
  $x_{GHT,j} \leftarrow \frac{1}{n-1} x_{GHT,j}$
end

Algorithm 4: Personalized PageRank (PPR) Mechanism

Input: previously computed fundamental matrix N
Output: matrix of PPR scores X_{PPR}

$\mathbf{h} \leftarrow \mathbf{0}$
$X_{PPR} \leftarrow 0$
for $i = 1...n$ **do**
  for $j = 1...n$ **do**
    $h_i \leftarrow h_i + N(i, j)$
  end
  for $j = 1...n$ **do**
    if $i \neq j$ **then**
      $X_{PPR}(i, j) \leftarrow \frac{N(i,j)}{h_i}$
    end
  end
end

Algorithm 5: PageRank Mechanism

Input: previously computed fundamental matrix N
Output: vector of PageRank scores $\mathbf{x}_{PR}$

$\mathbf{h} \leftarrow \mathbf{0}$
$\mathbf{x}_{PR} \leftarrow \mathbf{0}$
for $i = 1...n$ **do**
  for $j = 1...n$ **do**
    $h_i \leftarrow h_i + N(i, j)$
  end
end
for $j = 1...n$ **do**
  for $i = 1...n$ **do**
    if $i \neq j$ **then**
      $x_{PR,j} \leftarrow x_{PR,j} + \frac{N(i,j)}{h_i}$
    end
  end
  $x_{PR,j} \leftarrow \frac{1}{n-1} x_{PR,j}$
end

5 An Example Computation

We illustrate the results of this work with a simple example. Consider the trust graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ of order $n = 5$ in Figure 1 with agents in $V = \{v_1, ..., v_5\}$ and transition probability matrix P.

Construct the augmented graph $\mathcal{G}'$ in Figure 2 by adding vertex v_6. Then the transition probability from each vertex in $\mathcal{V}$ to augmented vertex v_6 is α. In this example, let $\alpha = 0.5$. Construct augmented transition probability matrix P'.

$$P = \begin{bmatrix} 0 & 0.4 & 0 & 0.6 & 0 \\ 0.2 & 0 & 0.5 & 0.3 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0.5 & 0 & 0 & 0 & 0.5 \\ 0.2 & 0 & 0.8 & 0 & 0 \end{bmatrix} \qquad P' = \begin{bmatrix} 0 & 0.2 & 0 & 0.3 & 0 & 0.5 \\ 0.1 & 0 & 0.25 & 0.15 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 0.25 & 0 & 0 & 0 & 0.25 & 0.5 \\ 0.1 & 0 & 0.4 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 1.0 \end{bmatrix}$$

Compute fundamental matrix N by taking $(I - (1 - \alpha)P)^{-1}$, in which $(1 - \alpha)P$ is the top left $n \times n$ block of

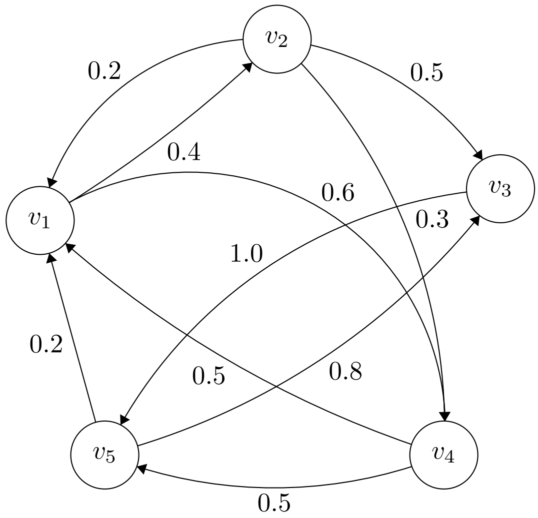

Figure 1: $\mathcal{G}$

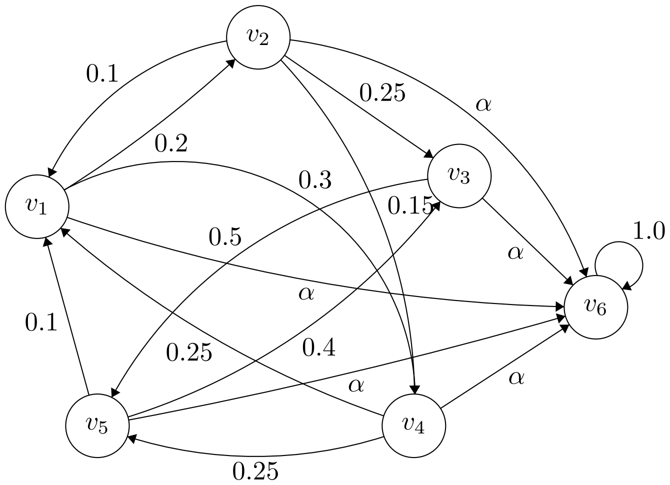

Figure 2: $\mathcal{G}'$

P'. We also calculate the vector of hitting times $\mathbf{h} = N \cdot \mathbf{1}$ for convenience.

$$N = \begin{bmatrix} 1.131 & 0.226 & 0.117 & 0.373 & 0.152 \\ 0.179 & 1.036 & 0.350 & 0.209 & 0.227 \\ 0.071 & 0.014 & 1.257 & 0.023 & 0.635 \\ 0.318 & 0.064 & 0.158 & 1.105 & 0.355 \\ 0.141 & 0.028 & 0.515 & 0.047 & 1.269 \end{bmatrix} \qquad \mathbf{h} = \begin{bmatrix} 2.0 \\ 2.0 \\ 2.0 \\ 2.0 \\ 2.0 \end{bmatrix}$$

We show the computation of example trust scores from the perspective of agent v_1. For the global hitting time and PageRank mechanisms, let F_q be a uniform distribution over the original vertex set $\mathcal{V}$ as before. Applying (5), (6), (7), and (8) for PHT, GHT, PPR, and PageRank scores, respectively, we find the following trust scores from the perspective of v_1:

PHT	
$x_{PHT,12}$	0.218
$x_{PHT,13}$	0.093
$x_{PHT,14}$	0.338
$x_{PHT,15}$	0.120

GHT	
$x_{GHT,2}$	0.080
$x_{GHT,3}$	0.227
$x_{GHT,4}$	0.148
$x_{GHT,5}$	0.270

PPR	
$x_{PPR,12}$	0.113
$x_{PPR,13}$	0.059
$x_{PPR,14}$	0.187
$x_{PPR,15}$	0.076

PageRank	
$x_{PR,2}$	0.042
$x_{PR,3}$	0.143
$x_{PR,4}$	0.082
$x_{PR,5}$	0.171

Comparing the scores allows one to create a ranking of the agents based on relative trustworthiness from the perspective of v_1. The orderings returned by the global mechanisms are consistent with each other, and the orderings returned by the personalized mechanisms are consistent with each other. However, the two mechanism types do not agree when compared. For instance, while agent v_1 considers agent v_4 the most trustworthy under the PHT and PPR mechanisms, v_4 is considered relatively untrustworthy globally. These results suggest that the personalized mechanisms, and particularly the PHT mechanism, give a more refined calculation of trust scores using the perspective of single agents.

The value of α acts as a general indication of trust within the network. A large value of α indicates a generally higher distrust between all interacting agents since the random walk is likely to terminate more quickly. At larger values of α, as in the example, one sees that the personalized mechanisms can differ greatly from the global ones. In such cases, the opinions of agents in the personalized mechanisms are dominated by the trust reports on incident edges. In contrast, the global mechanisms can remain relatively invariant under different values of α. Users of the algorithm may compute the mechanisms under different values of α to explore the variation in the personalized mechanisms. Since computing the scores for each value of α requires an $\mathcal{O}(n^3)$ computation, the importance of producing trust scores efficiently becomes even more pertinent.

6 Conclusion and Future Work

In this paper, we proposed an optimized algorithm for computing various graph-based trust mechanisms introduced in [LPS16]. In particular, we showed how the personalized hitting time (PHT) mechanism, most robust against sybil attacks compared to the global hitting time (GHT), personalized PageRank (PPR), and PageRank mechanisms, can be computed quickly using a probabilistic formulation. We expressed each of the trust mechanisms succinctly in terms of the fundamental matrix of an absorbing Markov chain modeling a random walk over a trust graph. Finally, we showed a simple calculation of the four trust mechanisms on an example trust graph, noting that the personalized mechanisms differed from the global mechanisms in providing a ranking of relative trustworthiness. In agreement with [LPS16], we found that personalized mechanisms appeared to give a more informative view of trustworthiness in the calculation. Future work should use real data taken from sources such as the Stanford Large Network Dataset Collection to explore the proposed algorithm on large, real-life networks.

References

[BP94] Abraham Berman and Robert J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences.* Society for Industrial and Applied Mathematics, 1994.

[FRCS05] Dániel Fogaras, Balázs Rácz, Károly Csalogány, and Tamás Sarlós. Towards scaling fully Personalized PageRank: Algorithms, lower bounds, and experiments. *Internet Mathematics*, 2(3):333–358, 2005.

[GS12] Charles M. Grinstead and J. Laurie Snell. *Introduction to Probability.* American Mathematical Society, 2012.

[HS06] John Hopcroft and Daniel Sheldon. Manipulation-resistant reputations using hitting time. In *Algorithms and Models for the Web-Graph.* WAW, 2006.

[HS10] Chung-Wei Hang and Munindar P. Singh. Trust-based recommendation based on graph similarity. In *Proceedings of the 13th AAMAS Workshop on Trust in Agent Societies (Trust).* AAMAS, 2010.

[JW03] Glen Jeh and Jennifer Widom. Scaling personalized web search. In *Proceedings of the 12th International Conference on World Wide Web*, pages 271–279. WWW, 2003.

[Lof15] Peter Lofgren. *Efficient Algorithms for Personalized PageRank.* PhD thesis, Stanford University, 2015.

[LPS16] Brandon Liu, David Parkes, and Sven Seuken. Personalized hitting time for informative trust mechanisms despite sybils. In *International Conference on Autonomous Agents & Multiagent Systems.* AAMAS, 2016.

[Sen81] Eugene Seneta. *Non-negative Matrices and Markov Chains.* Springer Science+Business Media, 1981.