Compressive Sensing over Strongly Connected Digraph and Its Application in Traffic Monitoring

Xiao Qi, Yongcai Wang, Yuexuan Wang, Liwen Xu

Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China

{qixiao0113, kyoi.cn}@gmail.com, {wangyuexuan, wangyc}@tsinghua.edu.cn

Abstract—Compressive sensing over graphs has recently attracted great research attentions, which takes limited number end-to-end measurements along paths (walks) to recover sparse vectors representing link/node properties. Unlike traditional compressive sensing, the along-path measurements rule out the freedom of random sampling, which introduces path constraints to the measurement matrix. The constraint makes explicit analysis of recovery performance difficult. Only for undirected graphs, early results showed that $O(klog(n))$ end-to-end measurements taken by random walks are sufficient to recover k-sparse edge vector. However, the problem becomes more difficult when directed graphs are considered, because of the easy state absorbing and the difficulty of evaluating the stationary distribution. But digraphs inherently model many network systems. In this paper, particularly for strongly connected digraphs with low node degrees, we presents bounds for the stationary distribution of random walks, and present deliberative proofs which put forward that $O(klog(n))$ path measurements are sufficient to recover k-sparse edge vectors. Further more, because urban road networks are exactly strongly connected, low degree digraphs, we designed efficient recovery methods to estimate road delays by a small number of probing cars. Although the road delay vector is actually not sparse, we leverage the empirical non-congested road delays as references and develop an algorithm which divide the problem to iteratively recover several k-sparse vectors. Simulation results show that when less than 10% edges are congested, more than 90% congestion states can be recovered correctly by 10% measurements.

I. INTRODUCTION

When networks become large in scale, it will be costly and operationally difficult for users to monitor the properties of all links. For inefficiency of monitoring each link, users generally rely on external end-to-end measurements, such as measuring path delays from source nodes to destination nodes to indirectly infer the properties of the internal links. This falls in the area of *network tomography*, which is important for locating fault links or analyze performance whose application areas range from internet to transportation networks. Because each end-to-end external measurement may incur considerable amount of costs, such as the measurement taken by probing cars equipped with GPS in the transportation networks. How to minimize the cost of external end-to-end measurements while accurately recovering the properties of all the internal links has not only theoretical significance but also application values.

A fact which softens the problem is that the link properties in a large graph generally constitute a sparse vector, because it is generally true that only a small portion of links incur obvious delays or packet loss rates than others. This fact

has inspired the application of compressive sensing (CS) to recover the link property vector by limited amount of end-to-end measurements. Compressive sensing provides fundamental theories and efficient recovery algorithms to perfectly recover signals that are sparse in an transformed domain. However, previous studies in [1–3] showed that the along-path measurements in graphs pose new *path restrictions* to measurement matrix, which makes measurements no longer be free at random. This constraint makes explicitly analysis to the recovery performance difficult. Only in undirected graphs, [1–3] proved that $O(klog(n))$ path measurements are sufficient to identify k-sparse link vector and proposed l_1 minimization based recovery algorithms. [4].

The problem becomes more difficult when it comes to digraphs, although digraphs are more pertinent models of many kinds of network systems, such as transportation networks and sensor networks. The reason is that the random walks (path measurement) on directed graphs, when considered as Markov chain, converge towards absorbing states (such as the nodes with 0 out-degree). In other words, the flows in the networks tends to be absorbed by several nodes. The stationary visiting probabilities of transient nodes (edges) are zero, which set obstacles for explicit analysis.

However, in this paper, we turn our attention to a special kind of digraphs, i.e., *strongly connected digraph*, whose stationary distributions do exist for all nodes (or edges)[5]. Although the stationary distribution cannot be formulated explicitly as in the undirected graph[6], we propose bounds for the stationary distribution and based on which we prove that $m = O\left(c^4 K T^2(n)\log\left(\frac{n}{K}\right)\right)$ end-to-end measurements are sufficient to recover k-sparse link vectors in strongly connected digraphs, where c is a constant, n is the number of the links and $k \leq \frac{K+1}{2}$; $T(n)$ is the *mixing time*, which will be introduced in Section III. The proof progress introduce an assumption that the node/edge degree is $O(1)$, which means degree is not large.

The strongly connected digraph with low degree is a perfect model for the transportation networks, where a car at any road can find a path to any other road, and the road crosses are naturally in low-degree when the road network is modeled by a graph. But the road delay vector is unfortunately not sparse. To address this difficulty, we leverage the empirical non-congest delays of the roads as a reference. It helps to estimate the congestion factors of the roads, and based on which, we develop algorithms to recover road delay vectors.

It decomposes the road delay vector to a set of sparse vectors and recovers them iteratively, which accomplish the task of recovering the non-sparse delay vector. Efficiency of the proposed algorithms is evaluated by intensive simulations.

The rest of the paper is organized as follows. Section II introduces the problem model. Section III presents the compressive sensing methodology in strongly connected digraphs. Section IV shows how traffic delay vector is recovered by limited number of probing cars. Section V presents the simulation results. Section VI reviews the related works. Section VII concludes the paper with future works.

II. PROBLEM FORMULATION

Consider a directed graph $G = (V, E)$, where V and E are node and edge sets, and let $|V| = N$ and $|E| = n$ be the numbers of nodes and edges respectively. Let $deg(*)^-$ and $deg(*)^+$ be functions to return the out-degree and in-degree of $*$, where $*$ can be an edge or a vertex. In this paper, we use edge degrees by default unless node degrees are specified. Consider an edge e, $deg(e)^-$ indicates the number of edges who end at e's start point and $deg(e)^+$ indicates the number of edges who start from e's end point. We assume the digraph is (D, c)-uniform digraph which satisfies $D \leq deg(e)^{-,+} \leq cD$, where c is a constant. We also consider G is strongly connected, i.e., there must be a path between any two vertexes.

Each edge $e = (u, v)$ has a value x_e which is the delay from u to v in the network context. We denote $\mathbf{x} = [x_1, \cdots, x_n]^T$ as the delay vector. An end-to-end measurement along a path measures the sum value of the edges that participate in the path. Mathematically speaking, if m end-to-end measurements are taken, the measurement vector $\mathbf{y}$ is a $m \times 1$ vector, where:

$$\mathbf{y} = M\mathbf{x}, \tag{1}$$

where M is an $m \times n$ binary matrix, whose element $M_{ij} = 1$ if the ith path routes through the jth edge and 0 otherwise. Note that value of edge can also be instantiated to link loss rate, and formula (1) can still be obtained easily via calculating logarithms of link loss rates[7].

We consider measurements are taken by random walks on the digraph. We define a edge transition probability matrix $\mathbf{P}_E = \{p_{ij}\}, e_i, e_j \in E$ where $p_{ij} = \frac{1}{deg(e_i)^+}$ if e_i's end point equal to e_j's start point, and 0 otherwise. The stationary distribution, π, of random walks on G, is defined formally as a distribution that is invariant to the transition probability, which is $\pi = \pi \mathbf{P}_E$. Unlike the stationary distribution of undirected graph, which is characterized in a closed form, the stationary distribution in digraphs has no closed-form expression[6]. Even so, we can give a bound for π under the strongly connected digraph (details in Section III).

The question now is whether we can estimate the delay vector $\mathbf{x}$ via as less the end-to-end measurements as possible, i.e., $m < n$. However, difference from the traditional CS, the measurement matrix M is constructed by the end-to-end, along-path measurement in digraph other than the freely random sampling, which may contradicts the incoherent requirement of CS's observation matrix. Another problem is $\mathbf{x}$

may be not sparse. But at this stage, we assume we can find a method which can convert $\mathbf{x}$ to be sparse. Therefore, in what is following, we consider $\mathbf{x}$ is k-sparse, i.e., having at most k non-zero values, and concentrate on whether CS can work, and if yes, how many end-to-end measurements are sufficient to recover $\mathbf{x}$ accurately.

III. HOW TO CONSTRUCT THE MEASUREMENT MATRIX

Some previous studies [1, 3] have proved that in undirected graphs, $O(k \log(n))$ path measurements are sufficient to recover k-sparse link vector. In this section, we will show whether compressive sensing over strongly connected digraph can also have theoretical performance guarantees. If it is possible, how many end-to-end measurements will suffice to recover the delay vector.

Before solving above two problems, we firstly introduce two useful notations for concisely presenting following sections.

- *The null space of matrix M :*

$$\mathcal{N}(M) = \{\mathbf{z} : M\mathbf{z} = 0\}.$$

- The mathematical form of *k-sparse vector*:

$$\Gamma_k = \{\mathbf{x} : \|\mathbf{x}\|_0 \leq k\},$$

where $\|\mathbf{x}\|_0$ indicates the l_0 norm of $\mathbf{x}$.

A. Null Space Condition for Vector Recovery

In this sub-section, we will give a theorem (Theorem 1) to characterize what kind condition the measurement matrix should satisfy such that the k-sparse vector can be recovered under the digraph constraints. Before introducing this theorem, we give two definitions for easily presenting the theorem, and a lemma which is used to prove it.

Definition 1 (K-disjunct matrix). *An $m \times n$ binary matrix M is called **K-disjunct**, if for any no more than $K+1$ columns (indexed by set $\mathbb{S} \subseteq \{1, 2, ..., n\}$), we select these $K+1$ columns to construct a sub-matrix $M_{\mathbb{S}}$, there is at least one row of $M_{\mathbb{S}}$, which has only a single "1" in that row.*

We can understand this definition from the digraph perspective, suppose the K-disjunct matrix is $m \times n$, each column represents a directed edge $(e_j, 1 \leq j \leq n)$ and each row indicates a path $(W_i = \{e_0, e_1, ..., e_t\})$ on digraph $G = (V, E)$, then take m paths and denote them by $\Lambda = \{W_i, 1 \leq i \leq m\}$, for $\forall E^* \subset E, |E^*| = K+1$, we say Λ can construct a K-disjunct matrix if the following **condition** is satisfied

$$\exists W_i \in \Lambda, \ s.t. \ |W_i \bigcap E^*| = 1. \tag{2}$$

Definition 2 (Matrix Spark [8]). *The spark of a given matrix $M_{m \times n}$ ($m \leq n$) is the smallest number of columns of M that are **linearly dependent**. If $m = n$ and $rank(M) = n$, then $spark(M) = n + 1$.*

Based on above definitions, we give a lemma depicting the relationship between them, which will be used in the proof of Theorem 1.

Lemma 1. *If the matrix $M_{m \times n}$ $(m \leq n)$ is K-disjunct, then $spark(M) > K + 1$*

Proof. $spark(M) > K+1$ means any $K'(\leq K+1)$ columns of M are linear independent, thus for prove this lemma, we only need to prove that any vector of $\mathcal{N}(M)$ has at least $(K + 2)$ non-zeros elements.

We prove it by contradiction. Assume $\exists \mathbf{z} \in \mathcal{N}(M), \mathbf{z} \neq 0$ and $||\mathbf{z}||_0 \leq K+1$. Suppose its support set is $\mathbb{S}$ (indexes set of non-zero elements). Select $K + 1$ columns from M following $\mathbb{S}$ and construct a new matrix $M_{\mathbb{S}}$. Since there exists one row in $M_{\mathbb{S}}$ with a single "1", thus $M_{\mathbb{S}}\mathbf{z} \neq 0$ must be satisfied, which contradicts the assumption that $\mathbf{z} \in \mathcal{N}(M)$. So each vector $\mathbf{z} \in \mathcal{N}(M)$ has at least $(K + 2)$ non-zeros elements. So we can conclude that any K' columns of M are linear independent, and which infers that $spark(M) > K+1$. $\square$

Now we give the first important theorem in this paper:

Theorem 1. *Let $M_{m \times n}$ be a K-disjunct matrix, and*

$$\mathbf{y} = M\mathbf{x}.$$

If $\mathbf{x}$ is a k-sparse vector, where $k \leq \frac{K+1}{2}$, then $\mathbf{x}$ is the unique $k-$sparsest vector satisfying $\mathbf{y} = M\mathbf{x}$.

Proof. We prove this theorem via contradiction. Assume $M\mathbf{x}_1 = M\mathbf{x}_2 = \mathbf{y}$, where $\mathbf{x}_1, \mathbf{x}_2 \in \Gamma_k$ and $\mathbf{x}_1 \neq \mathbf{x}_2$. Denote S_1 and S_2 are the support sets of $\mathbf{x}_1$ and $\mathbf{x}_2$ respectively. Since $M\mathbf{x}_1 = M\mathbf{x}_2$, we can get that $M(\mathbf{x}_1 - \mathbf{x}_2) = 0$. According to the assumption, $\mathbf{x}_1 - \mathbf{x}_2 \neq 0$ and $||\mathbf{x}_1 - \mathbf{x}_2||_0 \leq 2k \leq K+1$, we can infer that there exists one non-zero vector $\mathbf{x} = \mathbf{x}_1 - \mathbf{x}_2$ such that $M\mathbf{x} = 0$, where $||\mathbf{x}||_0 \leq K + 1$. However, we know that there does not exist such a non-zero vector from Lemma 1, which contradicts the assumption. $\square$

From Theorem 1, we know that if the measurement matrix is K-disjunct, then we can recovery the k-sparse vector precisely via compressive sensing. So in the next sub-section, we study the problem that how many path measurements, i.e., random walks on digraph, are needed to construct a K-disjunct matrix.

B. How Many Measurements are Needed?

In this paper, measurement matrix is generated by m independent random walks over the digraph. Our goal is to get the upper bound of the number of random walks, which is suffice to construct the K-disjunct measurement matrix. In other words we need a theoretical guarantee that how many random walks are suffice to construct Λ which satisfies the condition (2).

In [1, 3], a upper bound of the number of random walks over the (D, c)-uniform undirected graph has been proved, where $D \geq D_0, D_0 = O(c^2 KT(n))$. $T(n)$ is the δ-mixing time of a random walk which will be introduced soon. In this paper, we extend the method of [3] to give an upper bound in the (D, c)-uniform *strongly connected digraphs*, where $D \leq \lfloor \frac{e^2}{c} \rfloor$. For distinguish the edge e, we use e as **the base of natural logarithm**.

Theorem 2. *Suppose G is a (D, c)-uniform strongly connected digraph with δ-mixing time $T(n)$ (where $\delta := (1/2cn)^2$). For $D < \frac{e^2}{c}$, $K = O\left(\frac{n}{\sqrt{e}D^2+1}\right)$ and $t = O(n/(c^3 KT(n)))$ (random walk length), the measurement matrix M is K-disjunct with probability $1-o(1)$ after m measurements, where*

$$m = O\left(c^4 KT^2(n)\log\left(\frac{n}{K}\right)\right).$$

We can understand this theorem from the opposite aspect: if the conditions in Theorem 2 are all satisfied, then after m random walks on G, the probability that the measurement matrix M is not K-disjunct matrix is $o(1)$. In other words, to prove this theorem, we need to lower bound the probability that after m measurements, no random walk satisfies condition (2). So the critical step is determining the probability that a random walk satisfies condition (2) (i.e., $\pi_{e,E'}$, will be introduced soon), and in the following part of this sub-section, we will focus on this problem.

For statement convenience, we firstly introduce a concept of δ-mixing time of random walk and give two useful notations. According to [9] we know that the random walk on a digraph G has a unique stationary distribution if G is strongly connected and is not periodic, which is obviously in our context. Based on this fact, we give the definition of δ-mixing time under strongly connected digraph.

Definition 3 (δ-mixing time[3]). *Let $G = (V, E)$ be a (D, c)-uniform strongly connected digraph and denote by π its stationary distribution. For $e \in E$ and an integer τ, denote by π_e^τ the distribution that a random walk of length τ starting at e. Then the δ-mixing time of G is the smallest integer t such that $||\pi_e^\tau - \pi||_\infty \leq \delta$, for $\forall \tau \geq t$ and $\forall e \in E$. For concreteness, we define the quantity $T(n)$ as the δ-mixing time of G for $\delta := (1/2cn)^2$.*

Consider a random walk $W := (w_0, w_1, ..., w_t)$ of length t on a digraph $G = (V, E)$, where the random variables $w_i \in E$ denote the edge visited by the walk. We give the following quantities related to the walk W:

π_e The probability that W passes $e \in E$

$\pi_{e,E'}$ The probability that W passes $e \in (E \backslash E')$ but none of the edges in $E', E' \subseteq E$.

Lower bounding $\pi_{e,E'}$ is exactly the critical step in the proof of Theorem 2. We give a lemma about the lower bound of $\pi_{e,E'}$:

Lemma 2. *There are two scales $K_0 := O\left(\frac{n}{\sqrt{e}D^2+1}\right)$ and $t_0 := O\left(\frac{n}{c^3 KT(n)}\right)$ such that whenever $K < K_0$, by setting the path length $t := t_0$ the following holds. let E' be a set of at most K edges in the graph G, and let $e \notin E'$, then*

$$\pi_{e,E'} = \Omega\left(\frac{1}{c^4 KT^2(n)}\right).$$

For proving Lemma 2, we introduce some basic propositions that will be used in the proof. The first proposition gives a bound of stationary distribution of strongly connected digraph.

Proposition 1. *Let $G = (V, E)$ be a (D, c)-uniform strongly connected digraph, and denote by π the stationary distribution of G. Then for each $e \in E$, $1/cn \leq \pi(e) \leq c/n$*

The proof can be found in Section VIII-A. Actually, in [6], the authors proposed an approximated method to measure the stationary distribution under strongly connected digraph via $\mathbf{P}_E$. But in our paper, we don't need to get the exact stationary distribution, bounding it is enough.

Proposition 2. *Suppose a random walk $W := (w_0, w_1, ..., w_t)$ on digraph G starting from an arbitrary edge and set $j \geq i + T(n)$. Let ε denote any event that only depends on the first i edges visited by W. Then for every $\mu, \nu \in E$,*

$$|Pr[w_i = \mu | w_j = \nu, \varepsilon] - Pr[w_i = \mu | \varepsilon]| \leq \frac{2}{3cn}.$$

The proof details of this proposition are in Section VIII-B. The following proposition gives a bound of π_e, which is proved in Section VIII-C.

Proposition 3. *For the probability that W passes $e \in E$, i.e., π_e, we have*

$$\pi_e = \Omega\left(\frac{t}{cnT(n)}\right),$$

where t is the walk length.

Now we begin to prove Lemma 2, which is the technical core of this paper.

Proof of Lemma 2. Let $\mathcal{B}$ denote the event that $W = \{w_0, ..., w_t\}$ visits some edges in E'. Now

$$\begin{aligned}\pi_{e,E'} &= Pr[\neg\mathcal{B}, e \in W] \\ &= Pr[e \in W]Pr[\neg\mathcal{B}|e \in W] \\ &= \pi_e(1 - Pr[\mathcal{B}|e \in W]).\end{aligned}$$

Next we need to upper bound $Pr[\mathcal{B}|e \in W]$. Firstly we fix $i > 2T(n)$, and assume that $w_i = e$. Then fix some edge $e_0 \notin E'$ and assume that $w_0 = e_0$. Now we try to upper bound $Pr[\mathcal{B}|w_i = e, w_0 = e_0]$.

Let $\beta := i - T(n)$ and $\gamma := i + T(n)$, and assume that $T(n) + 1 < \beta < \gamma < t$. Partition W into four segments [3]:

$$\begin{aligned} W_1 &:= (w_0, w_1, ..., w_{T(n)}), \\ W_2 &:= (w_{T(n)+1}, w_{T(n)+2}, ..., w_{\beta-1}), \\ W_3 &:= (w_\beta, w_{\beta+1}, ..., w_\gamma), \\ W_4 &:= (w_{\gamma+1}, w_{\gamma+2}, ..., w_t). \end{aligned}$$

For $j = 1, 2, 3, 4$, define

$$\rho_j := Pr[W_j \text{ passes } E' | w_0 = e_0, w_i = e].$$

Now we upper bound each ρ_j. In some degenerate situation, W_j may be empty, thus the corresponding ρ_j will be 0.

W_2 and W_4 are "oblivious" of the conditioning on w_i and w_0 since they are sufficiently far from both. In particular, the distribution of each edge on W_4 is point-wise close to stationary distribution π. Therefore, under our conditioning the probability that each such edge belongs to E' is at

most $|E'|(c/n + \delta) < 2cK/n$. Similarly, each edge on W_2 has an almost-stationary distribution without the conditioning on w_i. However, by Proposition 2, the conditioning on w_i changes this distribution by up to $\delta' := 2/(3cn)$ at each edge. Altogether, for each $j \in \{T(n) + 1, ..., \beta - 1\}$, we have

$$Pr[e_j \in E' | w_i = e, w_0 = e_0] \leq |E'|(c/n + \delta + \delta') \leq 2cK/n.$$

Using a union bound on the number of steps, we conclude

$$\rho_2 + \rho_4 \leq tK(c/n + \delta + \delta') \leq 2ctK/n.$$

For bounding ρ_3, we need to divide W_3/w_i into two segments and bound them respectively. The details are shown in Section VIII-D, here we just list the results: for $\forall j = i + 1, ..., \gamma$,

$$Pr[w_j \in E' | w_i = e, w_0 = e_0] \leq \frac{2K}{n - K}. \tag{3}$$

for $\forall j = \beta, ..., i - 1$,

$$Pr[w_j \in E' | w_i = e, w_0 = e_0] \leq \frac{2K}{n - K} + K\delta'. \tag{4}$$

Altogether, using a union bound and by combining (3) and (4), we get that

$$\rho_3 \leq \frac{4T(n)K}{n - K} + KT(n)\delta' \leq \frac{5KT(n)}{n - K}.$$

Using the same reasoning, ρ_1 can be bounded as

$$\rho_1 \leq \frac{3KT(n)}{n - K}.$$

Finally, we obtain

$$Pr[\mathcal{B}|w_i = e, w_0 = e_0] \leq \sum_{j=1}^{4} \rho_j \leq \frac{8KT(n)}{n - K} + \frac{2ctK}{n}.$$

The next step is to relax the conditioning on the starting edge of the walk. The probability that the initial edge is in E' is at most $\frac{K}{n}$, and by Proposition 2, conditioning on w_i changes this probability by at most $K\delta' \leq \frac{K}{n}$. Now we write

$$\begin{aligned} Pr[\mathcal{B}|w_i = e] &\leq Pr[w_0 \in E'] + Pr[\mathcal{B}|w_i = e, w_0 \notin E'] \\ &\leq \frac{2K}{n} + \sum_{j=1}^{4} \rho_j \leq \frac{8KT(n)}{n - K} + \frac{2K(ct+1)}{n}. \end{aligned}$$

Now by taking $t = O(n/c^3KT(n))$, we know that

$$\begin{aligned} Pr[\mathcal{B}|w_i = e] &\leq \frac{8KT(n)}{n - K} + \frac{2K(cO(n/c^3KT(n)) + 1)}{n} \\ &= O\left(\frac{T(n)K}{n - K}\right), \end{aligned}$$

and using Proposition 3 we can get the conclusion:

$$\begin{aligned} \pi_{e,E'} &\geq \pi_e\left(1 - O\left(\frac{T(n)K}{n - K}\right)\right) \\ &= \Omega\left(\frac{t}{cnT(n)}\right) = \Omega\left(\frac{1}{c^4KT^2(n)}\right). \end{aligned}$$

$\square$

Lemma 2 give a lower bound of $\pi_{e,E'}$ which exactly satisfies our requirement. Now we are ready to prove Theorem 2.

The Proof of Theorem 2. Without loss of generality, take an arbitrary set of edges $E^* \subseteq E$ with cardinality $|E^*| = K + 1$. Denote by π_{E^*} the probability that a random walk visits one and only one element from E^*. In fact,

$$\pi_{E^*} = (K+1)\pi_{e,E^*/e} = \Omega\left(\frac{K+1}{c^4 K T^2(n)}\right).$$

Denote by P_f the failure probability, namely that the resulting matrix M is not K-disjunct. By a union bound we get

$$P_f \leq \binom{n}{K+1}(1 - \pi_{E^*})^m$$

$$\leq \left(\frac{en}{K+1}\right)^{K+1}\left(1 - \Omega\left(\frac{K+1}{c^4 K T^2(n)}\right)\right)^m$$

$$\leq exp\left((K+1)\log\left(\frac{en}{K+1}\right)\right)\left(1 - \Omega\left(\frac{1}{c^4 T^2(n)}\right)\right)^m.$$

If

$$\left((K+1)\log\left(\frac{en}{K+1}\right)\right) + m\log\left(1 - \Omega\left(\frac{1}{c^4 T^2(n)}\right)\right) < 0.$$

Thus by choosing

$$m = O\left(c^4 K T^2(n)\log\left(\frac{n}{K}\right)\right)$$

we can ensure that $P_f = o(1)$, and hence, M is K-disjunct with overwhelming probability. □

IV. COMPRESSIVE TRANSPORTATION STATE MONITORING

Compressive sensing over digraphs can benefit many applications. As an instance studied in this paper, we show that the abstract models of transportation networks are exactly strongly connected digraphs with low node/edge degrees. Therefore applying compressive sensing on transportation networks to monitor road delays becomes promising, which reduce the cost of taking measurements by probing cars. However the delays on roads, i.e, the edge delay vector are no longer sparse. Even though, in this section we present a new method to recover the edge delay vector even it is not sparse. Particularly, we utilize the empirical non-congested delays of roads to help to estimate the congestion factors of edges and paths at first, which ultimately leads to an efficient algorithm to recover the edge delays. To avoid ambiguity, in this section "road" and "edge" are both used to represent road segments and "path" means the route traveled by cars.

A. City Road Network Model

Given a digraph $G = (V, E)$, where $|V| = N$ and $|E| = n$. Each vertex $v \in V$ indicates a road crossing, and each directed edge $(u, v) \in E$ represents a road from crossing u to v. Fig.1 shows an example, where Fig.1(a) is a portion of the city road networks and Fig.1(b) is the corresponding digraph model. We notice that there is one apparent and important fact that the digraph model of road networks is strongly connected, because
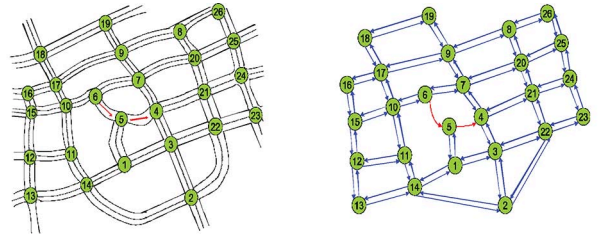
(a) A Portion of City Road Network (b) Corresponding Directed Graph

Fig. 1. Example of City Roads Networks

any car can reach any position from any starting point on the graph. The digraph model also has small node and edge degree, because a cross is generally intersection of two or three roads. So that in following part of this section, we study how to recover the delays of all roads by m probing cars walking randomly on the digraph to measure end-to-end path delays.

B. Problem formulation for transportation monitoring

For convenience, TABLE I lists some notations which will be used in the following parts. The real-time road delay

TABLE I
SUMMARY OF KEY NOTATIONS

Notation	Meaning
$\mathbf{x}$	The real-time road delay vector, x_i indicates the current delay (travel time) on road i
$\mathbf{x}^{nr}$	The empirical, non-congested road delays vector
$\widehat{\mathbf{x}}$	The estimated (recovered) road delays vector
$\mathbf{r}$	The road congestion rank vector
α_1, α_2	The factor of semi-congestion and congestion, where $0 < \alpha_1 < \alpha_2 < 1$
$\widetilde{\mathbf{p}}, \overline{\mathbf{p}}$	The path and road congestion factor vectors
$\overline{\mathbf{x}}$	The guessed approximate traffic condition vector, where $\overline{\mathbf{x}} = \mathbf{x}^{nr} + \langle\overline{\mathbf{p}} \cdot \mathbf{x}^{nr}\rangle$
M_D, M_D'	Two binary diagonal matrixes, the former controls which roads will be recovered; the latter indicates which roads determined the road congestion factor.

vector is represented by $\mathbf{x}$, whose element indicates the current delay of each road. Vector $\mathbf{x}^{nr}$ represents the empirical, non-congested road delays, whose elements represent the normal travel time on the corresponding roads without presence of congestion. Let vector $\mathbf{y}$ represent end-to-end delays measured by m probing cars in the city, and M be an $m \times n$ binary matrix, whose element $M_{i,j} = 1$ if the ith car passes through the jth road and 0 otherwise.

Apparently, $\mathbf{y} = M\mathbf{x}$ provides a appropriate formulation about the transportation monitoring. Unfortunately, $\mathbf{x}$, i.e., the road delay vector is not sparse. Nevertheless, with knowledge of the empirical non-congestion delay $\mathbf{x}^{nr}$, the difference vector $\mathbf{x}' = \mathbf{x} - \mathbf{x}^{nr}$ can rule out some non-congested roads because their differences from empirical non-congested delays are almost zero. Therefore, we consider the transformed problem as following:

$$\mathbf{y}' = \mathbf{y} - M\mathbf{x}^{nr} = M\mathbf{x}'. \tag{5}$$

We hope $\mathbf{x}'$ is sparse, but most times, it can not meet our expectation. The reason is that the ranges of $\mathbf{x}^{nr}$ are different, which means that even some road with big normal delay is unimpeded, the difference value may still be greater than some congested road with small normal delay values.

Although we can not recover $\mathbf{x}'$ directly most times, we still can infer some roads condition via Formula (5). For example, if $\mathbf{y}'(i) = 0$, then the real-time traffic condition of roads visited by the ith measurement path are equal to the corresponding elements values in $\mathbf{x}^{nr}$ and we can make sure that these roads are unimpeded. Further more, if we can infer which roads are most possible congested, then we can recover these roads with high priority. Suppose we select k most possible congested roads by our speculation, and index them by the set $S = \{s_i, 1 \le i \le k\}$. Given a binary diagonal matrix $M_D = \{d_{ij}\}$, where $d_{s_j s_j} = 0, s_j \in S$ or 1 otherwise. Then the vector $\mathbf{x}'' = \mathbf{x} - M_D\mathbf{x}$ must be k-sparse. The formulation becomes a new form:

$$\mathbf{y}'' = \mathbf{y} - MM_D\mathbf{x} = M\mathbf{x}''. \tag{6}$$

Actually, $\mathbf{y}''$ can not be gotten since $\mathbf{x}$ is unknown which means that $MM_D\mathbf{x}$ can not be calculated.

Nevertheless, we can try to construct a vector $\overline{\mathbf{x}}$, a substitute of $\mathbf{x}$ such that $MM_D\overline{\mathbf{x}} \approx MM_D\mathbf{x}$. Assume $\mathbf{x} \le 2\mathbf{x}^{nr}$, we need to find a factor vector $0 \le \overline{\mathbf{p}} \le 1$, and based on this factor vector, we can construct a vector $\overline{\mathbf{x}} = \mathbf{x}^{nr} + \langle \overline{\mathbf{p}} \cdot \mathbf{x}^{nr} \rangle$. Algorithm 1 shows a heuristic iteration method to get $\overline{\mathbf{p}}$. The fundamental idea is that firstly utilizing M_D' we delete the roads whose congestion state have been determined and minus their contributions from $\mathbf{y}'$, then we consider the left roads average the contribution to new $\mathbf{y}'$. We will give priority to the rows where only one or two roads are left when calculate $\overline{\mathbf{p}}$. Then the final version of the measurement formula become as the following:

$$\overline{\mathbf{y}} = \mathbf{y} - MM_D\overline{\mathbf{x}} \approx M\mathbf{x}''. \tag{7}$$

Since every time only k elements are recovered, thus the recovery algorithm should be executed until $\mathbf{x}$ totally be recovered.

C. The Recovery Algorithm

The algorithm can roughly divided into two steps:

step 1 Calculate road congestion factor vector.

step 2 Recover $\mathbf{x}$ by the descending order of the road congestion factor until $\mathbf{x}$ totally be recovered.

For step 2, we can utilize some common CS recovery algorithm, for example, l_1-norm minimization. The critical step is calculating the road congestion factor. So prior to introducing the algorithm details, we will show how to estimate the road congestion factor vector $\overline{\mathbf{p}}$. Specifically we need firstly calculate the path congestion factor vector $\widetilde{\mathbf{p}}$, then based on $\widetilde{\mathbf{p}}$, we can guess some road congestion factors, and these estimated factors will in turn influence the path congestion factors. This process will continue until all road congestion factor are determined. Algorithm 1 shows the details about

calculation of $\widetilde{\mathbf{p}}$ and $\overline{\mathbf{p}}$. Note that $M(i\cdot)$ indicate the ith row and M^T represents the transform of M.

Algorithm 1 Calculate Path Congestion Factors

Input:
  Measurement matrix $M_{m \times n} = \{m_{ij}\}$, $\mathbf{x}^{nr}$, $\mathbf{y}$.
Output:
  The estimated road congestion factors vector $\overline{\mathbf{p}}$.
1: $M_D' = I_{n \times n}$;// $I_{n \times n}$ is the identity matrix.
2: $\widetilde{\mathbf{p}} = 0, \overline{\mathbf{p}} = 0$;
3: $\mathbf{y}' = \mathbf{y} - MM_D'\mathbf{x}^{nr}$;
4: Choose the edges where $\mathbf{y}'(i) = 0$ and $m_{ij} = 1$. Index them by $S = \{s_i, 1 \le i \le t\}$ and set $d_{s_i s_i} = 0, 1 \le i \le t$;
5: $\widetilde{p}_i = \frac{\mathbf{y}'(i)}{\sum_{j=1}^n m_{ij} x_j^{nr} d_{jj}}$, for all $1 \le i \le n$;
6: **while** # of undetermined roads > 0 **do**
7:    $M' = MM_D'$;
8:    $[minV, minI] = \min(M'(i\cdot))$ for all $1 \le i \le m$;
9:    Choose the edges where $m_{minI,j} = 1$ and index them by $S = \{s_i, 1 \le i \le t\}$, and set $d_{s_i s_i} = 0$ for all $1 \le i \le t$;
10:   $\overline{p}_{s_i} = \frac{\mathbf{y}'(minI)}{M'(minI\cdot)\mathbf{x}^{nr}}$, for all $1 \le i \le t$;
11:   $\mathbf{y}' = \mathbf{y}' - M \langle \langle M'(i\cdot)^T \cdot \overline{\mathbf{p}} \rangle \cdot \mathbf{x}^{nr} \rangle$.
12:   $\widetilde{p}_i = \frac{\mathbf{y}'(minI)}{\sum_{j=1}^n m_{ij} x_j^{nr} d_{jj}}$, for all $1 \le i \le m$;
13: **end while**

Actually, there are still some roads can not be assigned congestion factor since they are never be visited by any path. In fact, we do not need to calculate these roads congestion factors and just need estimate their road condition when we finished the recovery algorithm. The details will be introduced in the following part.

Algorithm 2 Traffic Condition Vector Recovery Algorithm

Input:
  Measurement matrix $M_{m \times n} = \{m_{ij}\}$, $\mathbf{x}^{nr}$, $\mathbf{y}$.
  Sparsity level k.
Output:
  The estimated traffic condition vector $\widehat{\mathbf{x}}$ for $\mathbf{x}$.
1: $\widehat{\mathbf{x}} = \mathbf{x}^{nr}, M_D = I_{n \times n}$;
2: Calculate $\overline{\mathbf{p}}$ via Algorithm 1;
3: $\overline{\mathbf{y}} = \mathbf{y} - MM_D\overline{\mathbf{x}} = \mathbf{y} - MM_D(\mathbf{x}^{nr} + \langle \overline{\mathbf{p}} \cdot \mathbf{x}^{nr} \rangle)$;
4: **while** # of unrecovered roads > 0 **do**
5:   Select k roads following descending order of $\overline{\mathbf{p}}$'s elements, which are indexed by $S = \{s_i, 1 \le i \le k\}$;
6:   $d_{s_i s_i}' = 0$, for all $1 \le i \le k$;
7:   $\widehat{\mathbf{x}} = M_D'\widehat{\mathbf{x}} + (I - M_D') * l_1\_min(\overline{\mathbf{y}}, M)$;
8:   $M_D' = I$;
9: **end while**

The details of recovery algorithm is show in Algorithm 2. There are two matters need attention:

- In line 7, $l_1\_min(\overline{\mathbf{y}}, M)$ indicates the l_1 norm minimization function, which is a common algorithm in compressive sensing, so we omit the details.
- At last iteration, the number of the roads waiting for

recovery may be less than k. But for concise statement, we assume the value of k can change at last iteration.

We have mentioned that there may be some roads never be visited by any measurement path. So we need find a method to estimate these roads traffic condition. we propose a simple method based on the assumption: one road and its neighbour roads has highly similar traffic condition. For example road $e_i = (u, v)$ is never be visited in measurement process, denote by $E_i^- = \{e_j = (u', u), u' \in V/v\}$ and $E_i^+ = \{e_j = (v, v'), v' \in V/u\}$ the backward and forward neighbor roads set respectively, then the calculation formula is like following:

$$x_i = x_i^{nr} + \sum_{e_j \in E_i^-} \frac{\widehat{x}_{e_j} - x_{e_j}^{nr}}{deg(e_j)^+} + \sum_{e_j \in E_i^+} \frac{\widehat{x}_{e_j} - x_{e_j}^{nr}}{deg(e_j)^-}. \quad (8)$$

Based on the recovered traffic condition vector $\widehat{x}$, we can easily rank the roads congestion levels. As the widely used "red", "yellow", "green" representation of road conditions, by comparing the real-time road delay with the empirical non-congestion delay, we present $r_i \in \mathbf{r}$ which can take only three values to indicate the real-time road conditions. If the current traffic condition vector is known, we can rank the road condition like following:

$$r_i = \begin{cases} 0 & \text{if } x_i^{nr} \leq x_i < (1+\alpha_1)x_i^{nr} & //\text{"green"} \\ 1 & \text{if } (1+\alpha_1)x_i^{nr} \leq x_i < (1+\alpha_2)x_i^{nr} & //\text{"yellow"} \\ 2 & \text{if } (1+\alpha_2)x_i^{nr} \leq x_i \leq 2x_i^{nr} & //\text{"red"} \end{cases}$$
$$(9)$$

V. SIMULATION RESULTS

Simulations are carried out to evaluate the traffic delay monitoring algorithms. We pay special attentions to the detection rate of congestion links and the tradeoffs between the measurement cost and detection accuracy.

A. "Ground Truth" Data

For evaluating the performance of Algorithm 2, we need to generate the "ground truth" data, i.e., simulate the traffic condition of the city road networks and take measurements on it. For this purpose, we built an abstracted but reasonable traffic network model in Matlab to verify the conceptional correctness of proposed algorithms. The model omits the dynamic behaviors of cars and effects of traffic lights, but grasps the essential spatial correlation of traffic delays in connected and neighboring roads. The design methodology of the model is that the congestion at road e_i will propagate slowly to its incoming links and its siblings. The releasing from congestion of this road will also reduce the congestion of incoming links and siblings. We have verified different influencing coefficients to incoming links and siblings and found that the proposed algorithm worked well in different correlation scenarios.

B. Simulation Results

We evaluate the accuracy of recovered vector $\widehat{x}$ via calculating the SNR value between $\mathbf{x}$ and $\widehat{x}$ following the formula

$$SNR_{dB} = 10 \log_{10} \frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x} - \widehat{x}\|_2^2}$$

Fig.2 was drawn for measurement number and SNR values among 9 different road numbers and 3 different congestion ratios ($\frac{k}{n}$). We can find that as the road number and congestion ratio increasing, the measurement number must increase to keep the accuracy of $\widehat{x}$. Although the SNR values indicate that the accuracy are not perfect, but almost all them are less than 15dBm, which is also a acceptable results in compressive sensing.
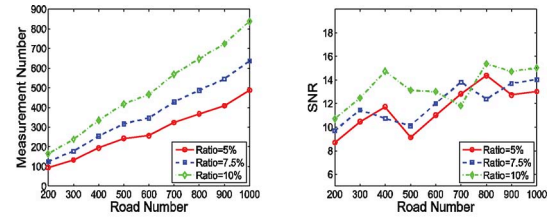
(a) Measurement Number for Different Road Number and Ratio

(b) SNR between $\mathbf{x}$ and $\widehat{x}$

Fig. 2. Measurement Number and SNR between $\mathbf{x}$ and $\widehat{x}$

Based on the recovered vector $\widehat{x}$, we can further rank the roads congestion levels following (9), because it generally true that only the congestion levels are cared by users. TABLE II shows an example when the road number n is fixed 1072. We can find that for congested roads, not only the detection rate ($\frac{Common}{Original}$) but also the accuracy rate ($\frac{Common}{Recovered}$) are greater than 90%; for semi-congested roads, the detection rate is also higher than 90%.

TABLE II
FIXED ROAD NUMBER

$n = 1072, m = 731$	Original	Recovered	Common
Green	989	990	985
Yellow	21	21	19
Red	62	61	59

Further more, we have also compared the detection and accuracy rate regarding different road number and congestion ratio. From Fig 3, we find that although the accuracy of $\widehat{x}$ is not perfect, we still can detect the congested and semi-congested roads with high probability. For example, the detection rate for congested and semi-congested roads (Fig.3(a) and 3(c)), we can easily find that almost all the red and yellow roads be checked out and the lower congestion ratio the better performance. Although the accuracy rate (Fig.3(b) and 3(d)) for congested and semi-congested roads are not very well, we can consider these results as pessimistic estimation, in other words, if some road is ranked as a green road based on our recovery vector, then basically this road can be considered as a unimpeded road.

(a) Detection Rate for Congested Roads (b) Accuracy Rate for Congested Roads

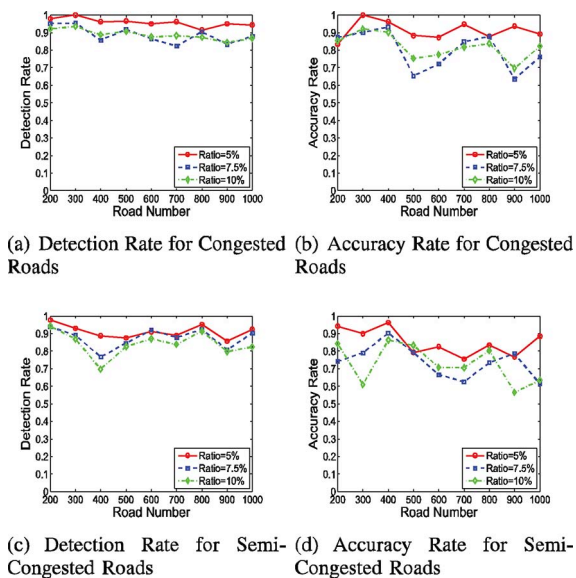(c) Detection Rate for Semi-Congested Roads (d) Accuracy Rate for Semi-Congested Roads

Fig. 3. Detection and Accuracy Rate

VI. RELATED WORK

We review related work in this section. Compressive sensing is a new paradigm in signal processing area, which offers fundamental theories and efficient recovery algorithm to recover sparse signal from reduced dimension measurements[4, 8, 10]. A promising application area of compressive sensing is *network tomography* [11–13], which investigate the problem of monitoring link properties by limited number of end-to-end measurements.

Previous studies about this problem in [1–3] showed that the end-to-end, along-path measurements in graphs introduced new *path restrictions* to the measurement matrix, so that the measurements could no longer be independent and freely random. This in some extend may contradict the incoherent requirement of observation matrix in CS. But by deliberate proofs, [1–3] proved that, despite the path constraints, in undirected graphs, $O(klog(n))$ path measurements are sufficient to identify k-sparse link vector and they proposed l_1 minimization algorithm that has theoretical guarantee in recovering the k-sparse link vector.

Some researchers have explored to apply the idea of compressive sensing onto transportation network monitoring[14, 15]. Other than the state of the art in estimating traffic utilizing static sensors such as loop detectors or traffic cameras, some recent studies have began to investigate the use of GPS devices as dynamic traffic probes for inferring traffic volume using existing mathematical models[16, 17], or they estimate traffic speed from GPS[18, 19]. The utilization of dynamic prob sensors provides a great support for the transportation monitoring via compressive sensing. However, most of these studies are based on a data mining type of approach. They either converge the transportation flow matrix by singular value decomposition or discovered the sparseness feature by

processing the traffic data statistically.

Different from these existing approaches, we at first investigate theoretically the application of CS on strongly connected digraphs, which in the first time shows the effectiveness of CS in digraphs to the best of our knowledge. We then present efficient recovery algorithms in essential model of traffic networks, which provides algorithms to recover road delays which are even not sparse.

VII. CONCLUSION AND FUTURE WORKS

In this paper, for strongly connected digraphs with low node (or edge) degrees, we prove that $O(klog(n))$ path measurements are sufficient to recover k-sparse edge vectors. It is derived by giving bounds to the stationary distribution of random walk which is different from the analysis in undirected graphs. Further more, because the urban road networks are exactly strongly connected and with low node degrees, we designed an efficient, iterative recovery algorithm, which recovers the road delays and the congestion states of the traffic networks by small number of probing cars. In future work we will study the traffic delay recovery algorithm on the real data set. Secondly, based on the current congestion state, predicting the future change of the network also falls in our research interests.

REFERENCES

[1] W. Xu and A. Tang, "Compressive sensing over graphs: How many measurements are needed?" in *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*. IEEE, 2010, pp. 615–619.

[2] M. Wang, W. Xu *et al.*, "Sparse recovery with graph constraints: Fundamental limits and measurement construction," in *INFOCOM, 2012 IEEE*. IEEE, 2012, pp. 1871–1879.

[3] M. Cheraghchi, A. Karbasi *et al.*, "Graph-constrained group testing," in *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*. IEEE, 2010, pp. 1913–1917.

[4] E. J. Candes and T. Tao, "Decoding by linear programming," *Information Theory, IEEE Transactions on*, vol. 51, no. 12, pp. 4203–4215, 2005.

[5] C. Cooper and A. Frieze, "Stationary distribution and cover time of random walks on random digraphs," *Journal of Combinatorial Theory, Series B*, vol. 102, no. 2, pp. 329–362, 2012.

[6] A. Mohaisen, H. Tran *et al.*, "On the mixing time of directed social graphs and security implications," in *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, ser. ASIACCS '12. New York, NY, USA: ACM, 2012, pp. 36–37.

[7] H. X. Nguyen and P. Thiran, "The boolean solution to the congested ip link location problem: Theory and practice," in *INFOCOM, 2007 IEEE*. IEEE, 2007, pp. 2117–2125.

[8] M. A. Davenport, M. F. Duarte *et al.*, "Introduction to compressed sensing," *Preprint*, vol. 93, 2011.

[9] K. Costello, "Random walks on directed graphs," 2005.

[10] E. J. Candès, "Compressive sampling," in *Proceedings oh the International Congress of Mathematicians: Madrid, August 22-30, 2006: invited lectures*, 2006, pp. 1433–1452.

[11] T. Bu, N. Duffield *et al.*, "Network tomography on general topologies," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 30, no. 1. ACM, 2002, pp. 21–30.

[12] Y. Chen, D. Bindel *et al.*, "Algebra-based scalable overlay network monitoring: algorithms, evaluation, and applications," *Networking, IEEE/ACM Transactions on*, vol. 15, no. 5, pp. 1084–1097, 2007.

[13] A. Gopalan and S. Ramasubramanian, "On identifying additive link metrics using linearly independent cycles and paths," *IEEE/ACM Transactions on Networking (TON)*, vol. 20, no. 3, pp. 906–916, 2012.

[14] H. Zhu, Y. Zhu *et al.*, "Seer: Metropolitan-scale traffic perception based on lossy sensory data," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 217–225.

[15] Z. Li, Y. Zhu *et al.*, "Compressive sensing approach to urban traffic sensing," in *ICDCS, 2011*. IEEE, 2011, pp. 889–898.

[16] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, June 2008.

[17] J. Aslam, S. Lim *et al.*, "City-scale traffic estimation from a roving sensor network," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*. ACM, 2012, pp. 141–154.

[18] D. B. Work, O.-P. Tossavainen *et al.*, "An ensemble kalman filtering approach to highway traffic estimation using gps enabled mobile devices," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*. IEEE, 2008, pp. 5062–5068.

[19] D. B. Work, S. Blandin *et al.*, "A traffic model for velocity data assimilation," *Applied Mathematics Research eXpress*, vol. 2010, no. 1, pp. 1–35, 2010.

VIII. APPENDIX

A. Proof of Proposition 1

In [5], there are two lemmas (Lemma 16 and 18) about the bound of stationary distribution on digraph, which is

$$(1 - o(1))\frac{deg(e)^-}{\widehat{n}} \le \pi(e) \le (1 + o(1))\frac{deg(e)^-}{\widehat{n}},$$

where $\widehat{n}$ is the edge number of a new digraph G' induced from G whose edges become the vertexes of G'. Suppose two edges $e_1 = (u_1, v_1), e_2 = (u_2, v_2)$ of G, if $v_1 = u_2$, then there exists a directed edge from e_1 to e_2 in G'. Since G is (D, c)-uniform digraph, which means $D \le deg(e)^- \le cD$ and $Dn \le \widehat{n} \le cDn$, thus we conclude that

$$\frac{1}{cn} = \frac{D}{cDn} \le \pi(e) \le \frac{cD}{Dn} = \frac{c}{n}$$

B. The Proof of Proposition 2

We know that

$$Pr[w_i = \mu | w_j = \nu, \varepsilon] = \frac{Pr[w_j = \nu | w_i = \mu, \varepsilon] \cdot Pr[w_i = \mu | \varepsilon]}{Pr[w_j = \nu | \varepsilon]}.$$

Now, according to the definition of δ-mixing time, we get that

$$|Pr[w_j = \nu | w_i = \mu, \varepsilon] - Pr[w_j = \nu | \varepsilon]| \le 2\delta,$$

because regardless of the knowledge of $w_i = \mu$, the distribution of w_j must be δ-close to the stationary distribution. Therefore,

$$|Pr[w_i = \mu | w_j = \nu, \varepsilon] - Pr[w_i = \mu | \varepsilon]| \le 2\delta/Pr[w_j = \nu | \varepsilon]$$
$$\le 2\delta/(1/cn - \delta)$$
$$\le 8\delta cN/3$$

C. The Proof of Proposition 3

Let $t' := \lfloor t/T(n) \rfloor$, for each $i \in \{0, 1, ..., t'\}$, $w_i' := w_{iT(n)}$. Denote by $W' := \{w_0', ..., w_{t'}'\}$ a subset of $t'+1$ edges visited by W. Obviously, π_e is at least the probability that $e \in W'$.

By the definition of δ-mixing time, regardless of the choice of w_0', the distribution of w_1' is δ-close to the stationary distribution π, which assigns a probability between $1/cn$ and c/n (by Proposition 1) to e. Therefore,

$$Pr[w_1' \ne e | w_0'] \le 1 - 1/cn + \delta \le 1 - \frac{1}{2cn}.$$

Similarly, $Pr[w_2' \ne e | w_0', w_1'] \le 1 - 1/2cn$, and so on. Altogether, this means that

$$Pr[w_0' \ne e, w_1' \ne e, ..., w_{t'}' \ne e] \le \left(1 - \frac{1}{2cn}\right)^{t/T(n)}$$
$$\le e^{(-t/2cnT(n))}$$
$$\le 1 - \Omega(t/cnT(n)).$$

In the last equality we used the fact that $e^{-x} \le 1 - x/2$ for $0 \le x \le 1$. Thus the complement probability is lower bounded by $\Omega(t/cnT(n))$.

D. The upper bound of ρ_3

Denote $D' = cD$, for bounding ρ_3, we firstly calculate

$$Pr[w_{i+1} \in E' | w_i = e, w_0 = e_0]$$
$$= \sum_{j=1}^{D'} \frac{j}{D'} \binom{D'}{j} \left(\frac{K}{n}\right)^j \left(\frac{n - K}{n}\right)^{D'-j}$$
$$= \frac{1}{D'} \left(\frac{n - K}{n}\right)^{D'} \sum_{j=1}^{D'} j \binom{D'}{j} \left(\frac{K}{n - K}\right)^j$$
$$\le \left(\frac{K}{n - K}\right) + \frac{1}{D'} \left(\frac{n - K}{n}\right)^{D'} \sum_{i=2}^{D'} j \binom{D'}{j} \left(\frac{K}{n - K}\right)^j$$
$$\le \left(\frac{K}{n - K}\right) + \frac{1}{D'} \left(\frac{n - K}{n}\right)^{D'} \sum_{j=2}^{D'} j \left(\frac{\sqrt{e}D'}{j}\right)^j \left(\frac{K}{n - K}\right)^j$$
$$\le \left(\frac{K}{n - K}\right) + \frac{1}{D'} \left(\frac{n - K}{n}\right)^{D'} \sum_{j=2}^{D'} \left(\frac{K}{n - K}\right) \le \frac{2K}{n - K}$$

where the second inequality is due to $\binom{D'}{j} \le \left(\frac{\sqrt{e}D'}{j}\right)^j$ for $D' \le e^2$, and the third inequality because of $j \left(\frac{\sqrt{e}D'}{j}\right)^j \left(\frac{K}{n - K}\right)^{j-1} < 1$ for $K < K_0$. Similarly,

$$Pr[w_{i+2} \in E' | w_i = e, w_0 = e_0, w_{i+1}] \le \frac{2K}{n - K},$$

regardless of w_{i+1} which means

$$Pr[w_{i+2} \in E' | w_i = e, w_0 = e_0] \le \frac{2K}{n - K},$$

and in general, for $\forall j = i + 1, ..., \gamma$,

$$Pr[w_j \in E' | w_i = e, w_0 = e_0] = Pr[w_j \in E' | w_i = e] \le \frac{2K}{n - K}.$$

Similarly $\forall j = \beta, ..., i - 1$, we have

$$Pr[w_j \in E' | w_i = e] \le \frac{2K}{n - K},$$

and by Proposition 2, conditioning on w_0 changes this probability by at most $K\delta'$. Therefore,

$$Pr[w_j \in E' | w_i = e, w_0 = e_0] \le \frac{2K}{n - K} + K\delta'.$$