Understanding (Inter-)Dependencies and Vulnerabilities in Static and Dynamic Networks

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Gyan Ranjan

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

UNDER THE SUPERVISION OF
Dr. Zhi-Li Zhang

May, 2013

Acknowledgements

There are many people that have earned my gratitude for their contribution to my stint as a graduate student at the University of Minnesota. First and foremost, and this may be bordering on cliche', is my PhD adviser, Dr. Zhi-Li Zhang, who, through his constant support and unswerving perseverance, has guided me through many a crest and trough of this academic roller-coaster. What I know of the scientific method — the rigor expected, the necessity to be self critical and the need to analyze a problem to the point that fallacies have no nooks left to slip in — is owed greatly to him. True, we have had our differences and disagreements at times, but it has always been in the interest of self betterment and the pursuit of that most tempting, yet elusive, of mistresses — the mathematical proof! Here again, Dr. Zhang's commitment to the absolute, in a mathematical sense, is both inspiring and infectious. This, above all, I shall take with me through the rest of my career.

Next, I express my sincerest regards for Dr. Daniel Boley who has been a *partenaire dans le crime* of sorts. Dr. Boley, or Dan as he is known to his students, can safely be anointed the 24/7 linear algebra *call center*. Where there is a matrix, ill conditioned or otherwise, Dan helps span the space from the origin — where one has no clue what the heck it would take to solve a problem — to a solution — where the only challenge left is figuring out how he conjured up the proof through what appears to us lesser mortals a feat of algebraic alchemy. Things get trickier when one has to decipher that most elegant handwriting of his; scanned copies of proofs can be hazardous to health. But behind all that is a great deal of learning, humility and a will to engage the other with humor and respect — I have had the good fortune of receiving a bit of each on numerous occasions — that is admirable. It is reasonable to say that my work would have been shallow at best, without his contributions. And I hope our association continues in

future.

Finally, I must acknowledge the wonderful mentors that I have had over the years. Mr. Umesh Chandra, Dr. Deepti Chafekar and Dr. Juong-Sik Lee, who made life pleasant during that glorious summer of 2010 at Nokia research center in Palo Alto. It was there that I discovered how the balance of work and pleasure must always tilt towards the latter. This approach of academic work-shirking spilled well into 2011 when I learnt the art of data mining under the supervision of Dr. Hui Zang, the most meticulous of researchers. And further into 2012, where in a brightly lit office at Bell Laboratories, only a few rooms away from that of the late Dr. Dennis Ritchie, I worked on my first business case with Dr. Mark Haner.

And to all those, mentioned above or otherwise, who have thus contributed in making this humble Hussey into something of a researcher — a million thanks!

Dedication

To my parents, my family and that green eyed damsel from Groningen ...

Abstract

This dissertation is an attempt at understanding the structural properties of static and dynamic networks from the perspective of robustness towards multiple random failures and targeted attacks.

In order to characterize the robustness of a static network, abstracted as a graph, towards random failures and targeted attacks, we transform it into a geometric object by embedding it into a Euclidean space spanned by the eigen vectors of the Moore-Penrose pseudo-inverse of the combinatorial Laplacian. This Euclidean space, endowed with a metric distance function, has several interesting properties. The nodes of the network are now transformed into discrete points in this space, each represented in terms of an n-dimensional co-ordinate (n being the number of entities), centered at the origin. We demonstrate that the position and the connectedness of an entity (node) in the network is well determined by its distance from the origin in this space. Closer a node is to the origin, more *topological central* it is, greater its tendency to lie in the larger sub-network when multiple edge failures lead to a breaking up of the network into disjoint components, thus greater its robustness in disruptive failure scenarios. We extend the same notion of centrality to the set of links (relationships) in the network as well. Last but not the least, the volume of the embedding, determined by the sum of squared lengths of the position vectors of all the nodes, yields a measure of robustness for the network as a whole. We call this the *Kirchhoff index* of the network. Once again, lower the Kirchhoff index of a network, more geometrically compact its embedding, and more difficult it is to break the network into two equal sized sub-networks; which, if possible, disrupts the greatest number of pairwise communications. The Kirchhoff index can, therefore, be used to compare the relative robustness properties of two networks with the same number of entities and relationships.

Next, we broaden our perspective by studying the sub-structures of a network within the geometric framework described above. First, on the computational front, we discuss an incremental methodology for computing the pseudo-inverse in a divide-and-conquer fashion, thus offsetting the otherwise high computational costs to within manageable

limits. Secondly, it takes our geometric framework to the realm of dynamic, time-evolving, networks.

We then apply the geometrical and topological understanding to the case of interdependent networks where the structural interplay between two or more networks determine the robustness of the overall system. Here again our geometric and topological indices provide interesting and somewhat counter-intuitive insights. We demonstrate that the manner in which interdependencies are introduced between two inter-dependent networks plays a significant role in determining the robustness of the overall system. In particular, diffusing and distributing inter-dependencies among a large number of (geographically dispersed) node pairs (sites of attachment) in the two constituent networks produces more robust systems.

Finally, we study the case of a cellular data service network (CDSN), where a different form of interdependence exists between the radio infrastructure (cellular towers) and the underlying IP (data) - network. One of the challenges in studying this class of networks is the absence of publicly available topology information, owing to security and privacy concerns. We thus motivate a means of inferring the topology through user geo-intent queries (e.g. location based services), thereby gaining some visibility in the inner workings of a CDSN. We observe that vast geo-physical territories in the radio layer map to relatively fewer and centralized IP network elements (such as NAS-gateway servers), thereby indicating potential of wide scale disruptions in the case of geo-physical attack scenarios. Last but not the least, we study the limitations of using cellular network traces to inferring human mobility patterns with an eye on urban infrastructure planning. We demonstrate that naive use of cellular traces can potentially lead to a biased view of user mobility that, in turn, may affect inferences of dynamic populations in urban areas. We demonstrate, however, that such biases can both be quantified and corrected for using an imposed sampling process, thereby obtaining better estimates.

Contents

List of Tables

List of Figures

Chapter 1

Introduction: All the World's a ...

All the world's a network,
And all the men and women actors;
They have their exits and entrances,
Their relationships and communities,
And one woman, in her network, plays many parts,
Her acts being the eight chapters of this work.

Notwithstanding the poetic absurdity, the lack of a lyrical meter or form, or even a general grace expected in works of penmanship — or indeed the fact that neither the author of this thesis nor his collaborators have any delusions of being 21^{st} century incarnations of Shakespeare — the principal claim made in the lines above stays! All the world's indeed a network. And here's why.

Simply put, a network is a collection of discrete *entities* and the *relationships* between them. It is easy to see that entities and relationships are no more than placeholders that can, depending upon the context, take the form of human beings and social ties, proteins and their interactions, atoms and chemical bonds, airports and connecting flights, routers and optical cables etc. Moreover, entities can be modeled as nodes (or vertices), and the relationships as links (or edges) thereby creating a network that can, in turn, be modeled as a graph — a combinatoric object with a rich mathematical legacy. All the world is, thus, a network.

This generality of form is in no way lost upon the scientific community. In recent years an entirely new field called *Complex Network Theory* has emerged, that cuts across several scientific disciplines including statistical physics, mathematical chemistry, neurology, epidemiology, genetics, and computer science; as indeed social sciences such as sociology and its ilk. The underlying theme in all these, despite the diversity of objectives and methodologies, is that *structure determines functionality* — a theme explored and furthered in this thesis.

To establish the truth behind this assertion, one needs to examine but a few cases. The roles played by human beings in a social context depends on their *position* and *connectedness* in the network. The physical and chemical properties of a molecule are dependent not only on its constituent atoms but also on the manner in which the atoms bond, the sites of attachment of different *functional* groups and the resulting *stereo* chemistry. Similarly, the significance of a router in the Internet is determined, in no small part, by its location (core vs. periphery). Or, the functioning of an electrical *smart-grid* that relies not only upon the power generation and distribution units but also on an underlying communication network *coupled* with it to relay control commands. All these, and numerous other analogues in literature, testify to the validity of the aforementioned claim. Needless to say, several attempts have been made, with varying degrees of success, to characterize the structural properties of networks. In this thesis, we concentrate primarily on one such aspect, that of the structural robustness of a network — as indeed of its constituents — towards multiple random failures and targeted attacks. And we do so with a little bit of algebra, some topology (graph theory), and a touch of geometry!

In order to characterize the robustness of a static network, abstracted as a graph, towards random failures and targeted attacks, we transform it into a geometric object by embedding it into a Euclidean space spanned by the eigen vectors of the Moore-Penrose pseudo-inverse of the combinatorial Laplacian. This Euclidean space, endowed with a metric distance function, has several interesting properties. The nodes of the network are now transformed into discrete points in this space, each represented in terms of an n-dimensional co-ordinate (n being the number of entities), centered at the origin. We claim that the position and the connectedness of an entity in the network is well determined by its distance from the origin in this space. To be precise, closer a node is to

the origin, more *central* it is in the network. We call this geometric notion of centrality the *topological centrality* of a node; and for a good reason. More topologically central a node is, greater its tendency to lie in the larger sub-network when multiple edge failures lead to a breaking up of the network into disjoint components. Topological centrality, thus, yields a rank-order for the entities in a network in terms of their relative immunity/vulnerability in disruptive failure scenarios. Analogously, the notion of topological centrality can be extended to the set of links (relationships) in the network as well. Here, the topological centrality of a link is determined in terms of the reciprocal of the inner product of the position vectors of its end points. Just as it is with nodes, higher the topological centrality of a link, greater its tendency to lie in the larger component when disruptive failures occur. Last but not the least, the volume of the embedding, determined by the sum of squared lengths of the position vectors of all the nodes, yields a measure of robustness for the network as a whole. We call this the *Kirchhoff index* of the network, inspired by an analogue used in mathematical chemistry that characterizes structural strengths of molecules. Once again, lower the Kirchhoff index of a network, more geometrically compact its embedding, and more difficult it is to break the network into two equal sized sub-networks; which, if possible, disrupts the greatest number of pairwise communications. The Kirchhoff index can, therefore, be used to compare the relative robustness properties of two networks with the same number of entities and relationships. In summary, the geometric transformation characterizes the robustness of a network at all granularities — from atomic entities to binary relationships all the way to the network as a whole.

Next, we broaden our perspective by studying the sub-structures of a network within the geometric framework described above. In particular, we show how the Moore-Penrose pseudo-inverse of the Laplacian for the overall network, is in fact related to that of its sub-networks (as determined by its connected bi-partitions). This observation has two interesting implications. First, from the computational point of view, it provides an incremental methodology for computing the pseudo-inverse in a divide-and-conquer fashion, thus offsetting the otherwise high computational costs to within manageable limits when the computations are done in parallel. Secondly, it takes our geometric framework to the realm of dynamic, time-evolving, networks. A dynamic network can in fact be thought of simply as a sequence of static networks across time, each of which

differs from its predecessor incrementally. Thus, analyzing the robustness of dynamic network — and its constituent nodes and links — is, in some sense, analyzing the incremental changes in the geometric properties between the different instances (snapshots). Our methods, detailed in chapter 4, provide not only for quick incremental computations, but also an insight into the effect of these incremental changes on the robustness properties of a dynamic network.

But what about the case when two (or more) networks are interdependent? For example, in modern infrastructure networks such as a *smart-grid* where there is a structural interplay between two different kinds of networks: a power generation and distribution network and a communication network. Here again our indices provide interesting and somewhat counter-intuitive insights from the point of view of structural robustness. We demonstrate that the manner in which inter-dependencies are introduced between two inter-dependent networks plays a significant role in determining the robustness of the overall system. In particular, we claim that diffusing and distributing inter-dependencies among a large number of (geographically dispersed) node pairs (sites of attachment) in the two constituent networks produces more robust systems.

But, *there are as many networks, as there are researchers!*[1] And, some other classes of networks, though amenable to a geometric treatment, require a different approach. One such class of networks is that of the cellular data service networks (CDSN). A CDSN is composed of two components — a radio communication infrastructure constituted by basestations (cellular towers that a user's mobile device directly interacts with) and an underlying IP network constituted by hierarchical proxies that receive, process and service data requests. Of great interest, therefore, is the functional dependence between the radio and the data infrastructure of a CDSN from the point of view of robustness. However, CDSNs are very secretive about their infrastructure. Thus, neither the locations of the basestations nor those of the underlying data plane, is available in the public domain. Neither are standard active probing based methods such as *traceroute* and *ping* of any avail. In chapter 6, we describe a novel approach that can

[1] Shamelessly stolen from Roger Bacon (c. 1214-1294), "There are as many bows as there are observers." Bacon, of course, was referring to rainbows. As one of the early European empiricists and a proponent of the modern scientific method, Bacon had, in a rather naughty attempt, tried to reclaim the right of creating rainbows for our own species, *Homo Sapiens*. For this, he was promptly imprisoned on charges of heresy, pending trial, for in the eyes of the Holy See of the time, the Lord had a monopoly on such acts of creativity.

be used for inferring the basestation infrastructure of a CDSN by an external agency — for example, location based services such as weather information providers and — that has access to the mobile devices of a sub-set of users.

Finally, we study in detail the patterns of human mobility within cellular networks. Here, an entirely different, yet intriguing, inter-dependence exists between user activity and its effects on the observed user mobility in the real world. A network theory interpretation in this case is to consider locations as nodes and the human populations moving between them as the links. The inferences made based on such studies have wide ranging applications in the fields of urban planning and disaster management as also in those of social media, location based services and advertising. But are the inferences representative or accurate within reasonable margins of error? Using a real-world cellular data record trace of over a million users in the San Francisco bay area, we examine the ways to answer these questions in a systematic fashion. We first assess the quality of spatio-temporal footprint of individuals as reflected in such records. We then quantify and correct for the possible biases in inferring mobility patterns that can accrue due to observational limitations.

Chapter 2

Geometry of Networks

In studying the *geometry* of networks, we first need to embed a network, represented abstractly as a graph, into an appropriate space endowed with a metric function — mathematically, a metric space. Such an embedding transforms the graph, a combinatoric object, into a geometric one. The nodes in the network are now represented as points in the embedding space while the relationships between node pairs (single or multi-hop) is reflected in the inter-point distances. Not only does such transformation make for intuitive amenability — graphs being notoriously complex to think about — it helps us exploit sophisticated tools from related literature that provide novel and valuable insights into the structural properties of the underlying network (or its graph).

In this chapter, we describe just such a metric space in terms of the eigen space of the Moore-Penrose pseudo-inverse of the combinatorial Laplacian (cf. §2.1). Next, in §2.2 we describe an interplay between the Euclidean distance metric, defined over this space, and the properties of an electrical analogue of a general network — an equivalence that is used to great advantage in the rest of this work.

2.1 Network as a Graph, the Combinatorial Laplacian and a Euclidean Embedding

Given a complex network, we represent its topology as a weighted undirected graph, $G(V, E, W)$, where $V(G)$ is the set of nodes/vertices; $E(G)$ the set of links/edges; and $W = \{w_{ij} \in \Re^+ : e_{ij} \in E(G)\}$ is a set of weights assigned to each edge of the graph

($\Re^+$: the set of non-negative real numbers). The edge weights are a measure of pairwise affinities between adjacent nodes that, depending upon the context, may represent physical quantities such as link capacity in traditional communication networks; or acquaintance, amity or reciprocity in the social counterparts. We denote by $n = |V(G)|$ the number of nodes in G, also called the *order* of the graph G. Similarly, $m = |E(G)|$ denotes the number of links. For $1 \leq i \leq n$, we define:

$$d(i) = \sum_{e_{ij} \in E(G)} w_{ij} \tag{2.1}$$

as the generalized degree of node i. Clearly, $d(i)$ is the aggregate affinity of node i with its one hop neighbors. The sum of node degrees in G, given as $Vol(G) = \sum_{i \in V(G)} d(i)$, is often referred to as the *volume* of the graph. $Vol(G)$ is, therefore, the aggregate one hop affinity in G. Without loss of generality, in the rest of this work, we deem the network represented by G to be connected. Or, in graph theoretic terms, G has exactly one connected component and, thus, at least one simple path between any pair of nodes.

The adjacency matrix of $G(V, E, W)$ is defined as $\mathbf{A} \in \Re^{n \times n}$, with elements $[\mathbf{A}]_{ij} = a_{ij} = a_{ji} = [\mathbf{A}]_{ji} = w_{ij}$, if $i \neq j$ and $e_{ij} \in E(G)$ is an edge; 0 otherwise. As stated earlier, the value w_{ij} is a measure of one hop *affinity* between nodes i and j. By definition, $\forall 1 \leq i \leq n : \quad A_{ii} = 0$. Clearly, $\mathbf{A}$ is a real and symmetric matrix.

The degree matrix of $G(V, E, W)$, is a diagonal matrix $\mathbf{D} \in \Re^{n \times n}$ such that $[\mathbf{D}]_{ii} = d_{ii} = d(i) = \sum_{j \in V(G)} a_{ij}$, is the weighted degree of node $i \in V(G)$. The combinatorial Laplacian of the graph is then given by:

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \tag{2.2}$$

Despite its simple form, the structure and eigen spectrum of $\mathbf{L}$ account for significant topological characteristics of the graph, such as minimal cuts, clustering and determining the number of spanning trees [1, 2, 3, 4]. The Laplacian thus finds use in various aspects of structural analysis [5, 6, 7, 2, 3, 8, 9, 10, 11, 12, 13, 14, 4]. It is easy to see, from the definition in (2.2) above, that the Laplacian $\mathbf{L}$ is a real, symmetric and doubly-centered matrix:

$$\sum_{i=1}^{n} [\mathbf{L}]_{ij} = \sum_{j=1}^{n} [\mathbf{L}]_{ij} = 0 \tag{2.3}$$

More importantly, $\mathbf{L}$ admits an eigen decomposition of the form:

$$\mathbf{L} = \mathbf{\Phi \Lambda \Phi'} \tag{2.4}$$

where the columns of $\mathbf{\Phi} = [\phi_1, \phi_2, ..., \phi_{n-1}, \phi_n]$ constitute the set of eigen vectors of $\mathbf{L}$. For the combinatorial Laplacian $\mathbf{L}$, this set is orthogonal [15]:

$$\phi_1 \perp \phi_2 \perp ... \perp \phi_{n-1} \perp \phi_n : \qquad \phi_i \cdot \phi_j = 0, \qquad \forall 1 \leq i \neq j \leq n \tag{2.5}$$

where $(\cdot)$ is the inner/dot product operator for vectors. Therefore, $\mathbf{\Phi}$, with appropriate normalization, constitutes the orthonormal basis of $\Re^n$. Also, $\mathbf{\Lambda}$ is a diagonal matrix with $[\Lambda]_{ii} = \lambda_i : 1 \leq i \leq n$; being the n eigen values of $\mathbf{L}$. It is well established that for an undirected graph $G(V, E, W)$, $\mathbf{L}$ is positive semi-definite i.e. all its eigen values are non-negative [15]. Further, if G is connected, as we have assumed, the smallest eigen value of 0 is unique. By convention, we shall assume a descending order for the eigen values of $\mathbf{L}$:

$$\mathbf{\Lambda} = [\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_{n-1} > \lambda_n = 0] \tag{2.6}$$

Note $\lambda_n = 0$, implies that $\mathbf{L}$ is rank deficient ($rank(\mathbf{L}) = n-1 < n$) and consequently singular. Its inverse, in the usual sense, does not exist.

However, a generalized inverse, called the Moore-Penrose pseudo-inverse of $\mathbf{L}$, denoted henceforth by $\mathbf{L}^+$, does exist and is unique [15]. Following four constitute the defining properties of $\mathbf{L}^+$:

$$a.\ \ \mathbf{L}\mathbf{L}^+\mathbf{L} = \mathbf{L} \qquad b.\ \ \mathbf{L}^+\mathbf{L}\mathbf{L}^+ = \mathbf{L}^+ c.\ \ (\mathbf{L}\mathbf{L}^+)' = \mathbf{L}\mathbf{L}^+ \qquad d.\ \ (\mathbf{L}^+\mathbf{L})' = \mathbf{L}^+\mathbf{L} \tag{2.7}$$

Like $\mathbf{L}$, $\mathbf{L}^+$ is also real, symmetric, doubly centered and positive semi-definite. Moreover, the eigen decomposition of $\mathbf{L}^+$ is given by

$$\mathbf{L}^+ = \Phi \Lambda^+ \Phi' \tag{2.8}$$

with the same set of orthogonal eigen-vectors as that of $\mathbf{L}$. The set of eigen values of $\mathbf{L}^+$, given by the diagonal matrix $\mathbf{\Lambda}^+$, is composed of $\lambda_n^+ = 0$ and the reciprocals of the positive eigen-values of $\mathbf{L}$, i.e. $[\lambda_1^{-1} \leq \lambda_2^{-1} \leq ... \leq \lambda_{n-1}^{-1}]$. It is the eigen space of $\mathbf{L}^+$, derived from the eigen space of $\mathbf{L}$, that is of interest to us.

Defining $\mathbf{X} = \mathbf{\Lambda}^{+1/2}\mathbf{\Phi'}$, we obtain:

$$\mathbf{L}^+ = \mathbf{\Phi \Lambda}^+ \Phi' = \mathbf{X'X} \tag{2.9}$$

The form in (2.9) above, together with the fact that Φ is an orthonormal basis for $\Re^n$, implies that the matrix $\mathbf{X}$ represents an embedding of the network in an $n-$ dimensional Euclidean space (cf. [8, 16] and the references therein). Next, we describe the specifics of this embedding space, with respect to the nodes and edges in the graph.

Let $\mathbf{x}_i$ denote the i^{th} column of $\mathbf{X}$. For a node $i \in V(G)$, $\mathbf{x}_i$ represents an n-dimensional co-ordinate in the embedding. In other words, $\mathbf{x}_i$ is the position vector for the node i in this n-dimensional space. Also, as $\mathbf{L}^+$ is doubly-centered (rows and columns individually sum up to 0), the centroid of set of all node position vectors, lies at the origin of the n-dimensional space. Thus, the squared distance of node i from the origin (or the squared length of the position vector) corresponds to the i^{th} diagonal entry in $\mathbf{L}^+$:

$$||\mathbf{x}_i||_2^2 = [\mathbf{L}^+]_{ii} = l_{ii}^+ \tag{2.10}$$

On the other hand, for any pair of nodes $(i, j) \in V(G) \times V(G)$, the embedding yields two geometric quantities. First, the inner product of the position vectors:

$$\mathbf{x}_i \cdot \mathbf{x}_j = [\mathbf{L}^+]_{ij} = l_{ij}^+ = l_{ji}^+ = [\mathbf{L}^+]_{ji} = \mathbf{x}_j \cdot \mathbf{x}_i \tag{2.11}$$

And secondly, the pairwise distance between nodes:

$$||\mathbf{x}_i - \mathbf{x}_j||_2^2 = l_{ii}^+ + l_{jj}^+ - l_{ji}^+ + l_{ij}^+ = ||\mathbf{x}_j - \mathbf{x}_i||_2^2 \tag{2.12}$$

Finally, we conclude by defining the volume of the overall embedding as follows:

$$\Xi(\mathbf{L}^+) = \sum_{i=1}^{n} l_{ii}^+ = Tr(\mathbf{L}^+) \tag{2.13}$$

where $Tr(\mathbf{L}^+)$ is the *trace* of the matrix $\mathbf{L}^+$. The volume of the embedding gives us a measure of *compactness* and is an aggregate quantity defined for the network as a whole.

We therefore obtain a Euclidean embedding of the network in terms of the eigen space of $\mathbf{L}^+$ with each node in the network represented as a point and a distance defined over any arbitrary node pair. This is indeed the central construct upon which this thesis relies. In particular, note that each of the geometric attributes described above (except $\mathbf{x}_i \cdot \mathbf{x}_j$), is determined by the metric distance defined for the Euclidean space. It is this function that holds the key to our analyses in the subsequent chapters; and it has an intriguing analogue in the physical world too as we explain next.

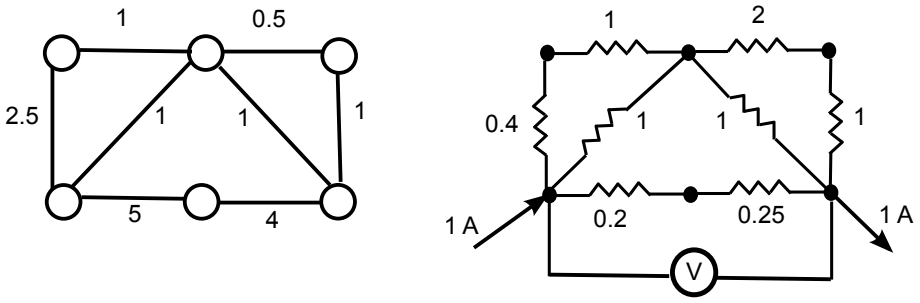


Figure 2.1: A simple graph G and its EEN.

2.2 Equivalent Electrical Network, Effective Resistance Distance and $\mathbf{L}^+$

An interesting analogy exists between simple undirected graphs and resistive electrical circuits [17, 18]. Given a simple, connected and undirected graph $G(V, E, W)$, the equivalent electrical network (EEN) of the graph can be formed by replacing each edge $e_{ij} \in E(G)$, of affinity w_{ij}, with an electrical resistance $\omega_{ij} = w_{ij}^{-1}$ *ohm* (cf. Fig. 2.1). Needless to say, each node $i \in V(G)$ has an analogous *junction* point in the EEN. A distance function can then be defined between any pair of nodes $(x, y) \in V(G) \times V(G)$ in the resulting EEN as follows:

Definition 1 *Effective Resistance (Ω_{ij}): The voltage developed between nodes i and j, when a unit current (1 amp) is injected at node i and is withdrawn at node j.*

It is well established that the square root of the effective resistance distance ($\sqrt{\Omega_{ij}}$) is a Euclidean metric [8, 18], i.e. it satisfies the following properties:

a. **Identity**: $\sqrt{\Omega_{ij}} > 0$ if $i \neq j$, 0 otherwise.

b. **Symmetry**: $\sqrt{\Omega_{ij}} = \sqrt{\Omega_{ji}}$

c. **Triangle-inequality**: $\sqrt{\Omega_{ij}} \leq \sqrt{\Omega_{ik}} + \sqrt{\Omega_{jk}}$

Most importantly, Ω_{ij} can be expressed in terms of the elements of $\mathbf{L}^+$ as follows [8]:

$$\Omega_{ij} = l_{ii}^+ + l_{jj}^+ - l_{ij}^+ - l_{ji}^+ \tag{2.14}$$

Therefore, the distance metric that our Euclidean space is endowed with is simply the effective resistance distance — an equivalence that we exploit time and again with significant rewards.

2.3 Summary

In this chapter, we introduced the preliminary notations and algebraic paraphernalia that are used throughout this thesis. We described an n-dimensional Euclidean embedding of the network in terms of the eigen space of the Moore-Penrose pseudo-inverse of the combinatorial Laplacian and established how the metric distance function characterizing this space is indeed the same as the pairwise effective resistance distance, a physical analogue in the real world. In subsequent chapters, we use these geometric notions to characterize the robustness of a network at all granularities — from individual nodes and edges to the network as a whole.

Chapter 3

Topological Centrality of Entities and Relationships in a Network

Based on the geometric embedding of the graph described in chapter 2, we now put forth three indices respectively characterizing robustness of nodes and edges in the network, as well as the network itself. However, to set the context, we begin by first reviewing some of the relevant indices of node and edge centralities popular in literature (cf. §3.1). In §3.2 , we define the *topological centralities* — $\mathcal{C}^*(i)$ for node $i \in V(G)$ and $\mathcal{C}^*(i-j)$ for edge $e_{ij} \in E(G)$. Similarly, for the network as a whole, we define a structural descriptor called the *Kirchhoff index* — $\mathcal{K}(G)$. We then establish how the topological centralities capture both the overall *position* and the overall *connectedness* of nodes and edges in the network. The position is characterized in terms of the overhead incurred in *forced random detours* (cf. §3.3), whereas the connectedness is determined in terms of the *connected bi-partitions* of the network (cf. §3.4). Analogous interpretations are also provided in the respective sections for the Kirchhoff index.

3.1 Related Work

Robustness of nodes to failures in complex networks is dependent on their overall *position* and *connectedness* in the network. Several centralities, that characterize position and/or connectedness of nodes in complex networks in different ways, have therefore been proposed in literature. Perhaps the simplest of all is degree — the number of

edges incident on a node. Degree is essentially a *local* measure i.e. a first oder/one-hop connectedness index. A second-order variant called *joint-degree*, given by the product of degrees of a pair of nodes that are connected by an edge in the network, is also in vogue. However, except in *scale free* networks that display the so called *rich club connectivity* [19, 20, 21], neither degree nor joint-degree determine the overall position or the connectedness of nodes.

A class of structural indices called *betweennesses*, namely shortest path/geodesic (GB) [22, 23], flow (FB) [24] and random-walk (RB) [25] betweenness respectively quantify the positions of nodes, with respect to source destination pairs in the network. It is easy to extend analogous concept of betweennesses to the edges in the network too. The set of betweennesses, however, reflect the role played by a node in the communication between other node-pairs in the network and are not the measures of a node's own connectedness.

Another popular centrality measure is geodesic closeness (GC)[22, 23]. It is defined as the (reciprocal of) average shortest-path distance of a node from all other nodes in the network. Clearly, geodesic closeness is a p^{th}-order measure of connectedness where $p = \{1, 2, ..., \delta\}$, δ being the geodesic diameter of the graph, and is better suited for characterizing global connectedness properties than the aforementioned indices. However, communication in networks is not always confined to shortest paths alone and GC being geodesic based, ignores other alternative paths between nodes, however competitive they might be, and thus only partially captures connectedness of nodes. Some all paths based counterparts of geodesic closeness include information centrality [26] and random-walk centrality [27], that use random-walk based approach to measure centrality. In [28], several centrality measures based on network flow, and collectively referred to as *structural centrality*, have also been discussed in great detail.

Recently, subgraph centrality (SC) — the number of subgraphs of a graph that a node participates in — has also been proposed [29]. In principle, a node with high subgraph centrality, should be better connected to other nodes in the network through redundant paths. Alas, subgraph centrality is computationally intractable and the proposed index in [29] approximates subgraph centrality by the sum of lengths of all *closed*

walks, weighed in inverse proportions by the factorial of their lengths. This inevitably results in greater correlation with node degrees as each edge contributes to closed random-walks of lengths $2, 4, 6, ...$ and thus introduces local connectivity bias. In a subsequent paper, Estrada et al introduce the concept of vibrations to measure node vulnerability in complex networks [30]. Their index called *node displacement* bears significant resemblance to information centrality and has interesting analogies to physical systems. However, once again the true topological significance of the centrality measure, in terms of connectedness is wanting.

Our aim in this work, therefore, is to provide an index for robustness of nodes in complex networks, that effectively reflects both the position and connectedness properties of nodes and edges; and consequently, by extension, of the overall network.

3.2 Topological Centrality and the Kirchhoff Index

Definition 2 *Topological centrality of node $i \in V(G)$:*

$$\mathcal{C}^*(i) = \frac{1}{l_{ii}^+} \tag{3.1}$$

Recall, that l_{ii}^+ represents the squared length of the position vector for the node i. Thus, closer a node i is to the origin in the n-dimensional space, more topologically central it is. Or in other words, lower l_{ii}^+ implies higher $\mathcal{C}^*(i)$. Similarly,

Definition 3 *Topological centrality of an edge $e_{ij} \in E(G)$:*

$$\mathcal{C}^*(i-j) = \frac{1}{l_{ij}^+} \tag{3.2}$$

Again, l_{ij}^+ represents the dot/inner product of the position vectors for the end-points of the edge e_{ij}. Lower the value of l_{ij}^+, by our definition, more topologically central the edge e_{ij} is. Just what these definitions translate to in the underlying topology is something we defer to subsequent sections. For now, a brief discussion from the point of view of the Laplacian eigen spectrum is warranted to put into context the definitions of these centralities.

Note, the general element l_{ij}^{+} in $\mathbf{L}^{+}$ can be re-written in terms of the eigen spectrum, as follows:

$$l_{ij}^{+} = \sum_{k=1}^{n-1} \frac{\phi_{ki} \cdot \phi_{kj}}{\lambda_k} \tag{3.3}$$

Putting $i = j$, we similarly obtain:

$$l_{ii}^{+} = \sum_{k=1}^{n-1} \frac{\phi_{ki}^{2}}{\lambda_k} \tag{3.4}$$

Thus, the topological centralities are a function of the entire eigen spectrum of the graph Laplacian ($\mathbf{L}$). This observation, though a straightforward artifact of algebra, is an important defining characteristic for our indices. In [31] a subset of leading eigen vectors of the combinatorial Laplacian and its normalized counterparts have been used for *localizing* a subset of closely connected nodes (or communities). Similar measures have also been defined using the spectra of the adjacency matrix and its related *ensemble* matrices (cf. [32]). Our topological centralities ($\mathcal{C}^{*}(i)$ & $\mathcal{C}^{*}(i-j)$), seen in this light, are a more generalized version of those discussed in [31], and extend previously known localization approaches (similar to that in [31]) to the granularity of individual nodes and edges.

Next, we define a structural descriptor for the overall network called *Kirchhoff index*, as:

Definition 4 *Kirchhoff index for $G(V, E, W)$:*

$$\mathcal{K}(G) = Tr(\mathbf{L}^{+}) = \sum_{i=1}^{n} l_{ii}^{+} = \sum_{i=1}^{n} 1/\mathcal{C}^{*}(i) \tag{3.5}$$

Recall, that this is simply the volume of the embedding ($\Xi(\mathbf{L}^{+})$) defined in the previous chapter. Geometrically, more compact the embedding is, lower the value of $\mathcal{K}(G)$, and we claim that more robust the network G is (details in latter sections)[1] .

Kirchhoff index has been widely used to model molecular strengths in the mathematical chemistry literature [34, 35, 36, 37, 33, 38]. It also finds mention in linear

[1] In literature, and in particular in [33] (cf. corollary 2.3), the Kirchhoff index for a network sometimes appears as $\mathcal{K}(G) = n\, Tr(\mathbf{L}^{+})$, i.e. with a scaling factor of n over what we have defined above in 3.5. However, in this work, our aim is to use $\mathcal{K}(G)$ as a comparative measure of robustness for two networks of the same order (i.e. same values of n) and volume (as described in subsequent sections). Therefore, we do away with the constant n from the definition of $\mathcal{K}(G)$ for the rest of this work.

algebra (cf. [39] and the references therein for details). However, its true topological (graph theoretic) significance has scarcely been explored and/or demonstrated. But a word on the Laplacian spectrum first. Note,

$$\mathcal{K}(G) = Tr(\mathbf{L}^{+}) = \sum_{i=1}^{n-1} \frac{1}{\lambda_i} \tag{3.6}$$

which implies that $\mathcal{K}(G)$ is a function of the entire Laplacian spectrum (albeit only the eigen values). Therefore, Kirchhoff index can be thought of as a generalized analogue of the much celebrated *algebraic connectivity* of the graph [2, 3], which is defined to be the second smallest eigen-value of the Laplacian i.e. λ_{n-1}, or, equivalently, the largest eigen value of $\mathbf{L}^{+}$.

In what follows, we demonstrate how these metrics indeed reflect robustness of nodes, edges and the overall network respectively, first through rigorous mathematical analysis resulting in closed form representations and then with empirical evaluations over simulated as well as real world network topologies.

3.3 Topological Centrality, Random Walks and Electrical Voltages

To show that topological centrality $(\mathcal{C}^{*}(i), \mathcal{C}^{*}(i-j))$ indeed captures the overall position of a node, we relate it to the lengths of random-walks on the graph. In §3.3.1, we demonstrate how $\mathcal{C}^{*}(i)$ and $\mathcal{C}^{*}(i-j)$ are related to the average overhead incurred in random *detours*. We then extend the same detour overheads to the Kirchhoff index as well. Next in §3.3.2, we provide an electrical interpretation for it in terms of voltages and the probability with which a random detour through node i returns to the source node.

3.3.1 Detours in Random Walks

A simple random walk $(i \rightarrow j)$, is a discrete stochastic process that starts at a node i, the source, visits other nodes in the graph G and stops on reaching the destination j [9]. In contrast, we define a *random detour* as:

Definition 5 *Random Detour ($i \to k \to j$): A random walk starting from a source node i, that must visit a transit node k, before it reaches the destination j and stops.*

Effectively, such a random detour is a combination of two simple random walks: $(i \to k)$ followed by $(k \to j)$. We quantify the difference between the random detour $(i \to k \to j)$ and the simple random walk $(i \to j)$ in terms of the number of steps required to complete each of the two processes given by hitting time.

Definition 6 *Hitting Time (H_{ij}): The expected number of steps in a random walk starting at node i before it reaches node j for the first time.*

Clearly, $H_{ik} + H_{kj}$ is the expected number of steps in the random detour $(i \to k \to j)$. Therefore, the overhead incurred is:

$$\Delta H^{i \to k \to j} = H_{ik} + H_{kj} - H_{ij} \tag{3.7}$$

Intuitively, more peripheral transit k is, greater the overhead in (3.7). The overall peripherality of k is captured by the following average:

$$\Delta H^{(k)} = \frac{1}{n^2 \; Vol(G)} \sum_{i=1}^{n} \sum_{j=1}^{n} \Delta H^{i \to k \to j} \tag{3.8}$$

Similarly, summing over all transits, we obtain the following analogue for a source-destination pair:

$$\Delta H^{(i-j)} = \frac{1}{n} \sum_{k=1}^{n} \Delta H^{i \to k \to j} - \frac{Tr(\mathbf{L}^+)}{n} \tag{3.9}$$

Alas, hitting time is not a Euclidean distance as $H_{ij} \neq H_{ji}$ in general. An alternative is to use commute time $C_{ij} = H_{ij} + H_{ji} = C_{ji}$, a metric, instead. We note from [18],

$$C_{ij} = Vol(G)(l^+_{ii} + l^+_{jj} - l^+_{ij} - l^+_{ji}) \tag{3.10}$$

and in the overhead form (3.7), (non-metric) hitting and (metric) commute times are in fact equivalent (see propositions $9 - 58$ in [40]):

$$\Delta H^{i \to k \to j} = (C_{ik} + C_{kj} - C_{ij})/2 = \Delta H^{j \to k \to i} \tag{3.11}$$

We now exploit this equivalence to equate the cumulative detour overhead through transit k from (3.8) to l^+_{kk} in the following theorem.

Theorem 1

$$\Delta H^{(k)} = l^{+}_{kk} \tag{3.12}$$

Therefore, a lower value of $\Delta H^{(k)}$ implies higher $\mathcal{C}^{*}(k)$ and more central the position of node k is in the network. Theorem 1 is interesting for several reasons. First and foremost, note that:

$$\sum_{j=1}^{n} C_{kj} = Vol(G)\ (n\ l^{+}_{kk} + Tr(\mathbf{L}^{+})) \tag{3.13}$$

As $Tr(\mathbf{L}^{+})$ is a constant for a given graph and an invariant with respect to the set $V(G)$, we obtain:

$$l^{+}_{kk} \propto \sum_{j=1}^{n} C_{kj} \tag{3.14}$$

Thus, lower l^{+}_{kk} or equivalently higher $\mathcal{C}^{*}(k)$, implies shorter average commute times between k and the rest of the nodes in the graph on an average. Moreover,

$$\mathcal{K}(G) = Tr(\mathbf{L}^{+}) = \sum_{k=1}^{n} l^{+}_{kk} = \frac{1}{2nVol(G)} \sum_{k=1}^{n} \sum_{j=1}^{n} C_{kj} \tag{3.15}$$

As $\mathcal{K}(G)$ reflects the average commute time between any pair of nodes in the network, it is a measure of overall connectedness in G. For two networks of the same order (n) and volume ($Vol(G)$), the one with lower $\mathcal{K}(G)$ is better connected on an average.

Theorem 2

$$\Delta H^{(i-j)} = l^{+}_{ij} \tag{3.16}$$

Once again, lower the value of l^{+}_{ij}, lower the detour overhead between the source-destination pair (i, j) when the random walk is forced through all possible transits, and more centrally located e_{ij} is.

3.3.2 Recurrence, Voltage and Electrical Networks

Interestingly, the detour overhead in (3.7) is related to *recurrence* in random walks — the expected number of times a random walk $(i \rightarrow j)$ returns to the source i [17]. We now explore how recurrence in detours related to topological centrality of nodes. But first we need to introduce some terminology.

Recall from the discussion in chapter 2, that the equivalent electrical network (EEN) [17] for $G(V, E, W)$ is formed by replacing an edge $e_{ij} \in E(G)$ with a *resistor*. The resistance of this resistor is given by $\omega_{ij} = w_{ij}^{-1}$ *amp* (see Fig. 2.1), where w_{ij} is the affinity between nodes i and j. In the EEN, let V_k^{ij} be the voltage of node k when a unit current is injected at i and a unit current is extracted from j. From [41], we have $U_k^{ij} = d(k)V_k^{ij}$; where U_k^{ij} is the expected number of times a random walk $(i \to j)$ visits node k. Substituting $k = i$ we get,

$$U_i^{ij} = d(i)V_i^{ij} \tag{3.17}$$

the expected number of times a random walk $(i \to j)$ returns to the source i. For a finite connected graph G, $U_i^{ij} > 0$. The following theorem connects recurrence to the detour overhead.

Theorem 3

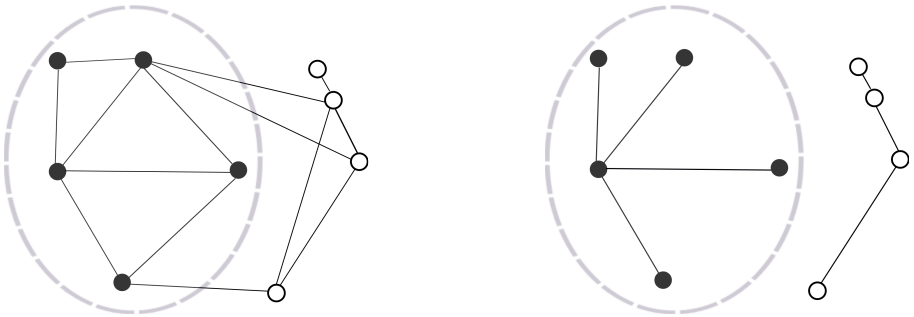
$$\begin{aligned} \Delta H^{i \to k \to j} &= \frac{Vol(G)\ (U_i^{ik} + U_i^{kj} - U_i^{ij})}{d(i)} \\ &= Vol(G)\ (V_i^{ik} + V_i^{kj} - V_i^{ij}) \end{aligned}$$

The term $(U_i^{ik} + U_i^{kj}) - U_i^{ij}$ can be interpreted as the expected extra number of times a random walk returns to the source i in the random detour $(i \to k \to j)$ as compared to the simple random walk $(i \to j)$. Each instance of the random process that returns to the source, must effectively start all over again. Therefore, more often the walk returns to the source greater the expected number of steps required to complete the process and less central the transit k is, with respect to the source-destination pair (i, j).

Therefore, $\Delta H^{(k)}$, that is the average of $\Delta H^{i \to k \to j}$ over all source destination pairs, tells us the average increase in recurrence caused by node k in random detours between any source destination pair in the network. Higher the increase in recurrence, i.e. $\Delta H^{(k)}$, lower the magnitude of $\mathcal{C}^*(k)$ and less structurally central the node k is in the network.

3.4 Topological Centrality and the Connected Bi-Partitions of a Network

Having thus far established that nodes with higher $\mathcal{C}^*(i)$ and edges with higher $\mathcal{C}^*(i-j)$, are more centrally located in the network, we now turn to establishing their average



(a) Partition $P = (S, S')$ (b) Dense spn. forest in $P = (S, S')$

Figure 3.1: Partitions and spanning forests of a graph.

connectedness characteristics too. To show how the topological centralities of nodes and edges capture their immunity/vulnerability to random failures in the network, we study their properties in all the connected bi-partitions of the graph representing the network. For simplicity of exposition, we assume the graph $G(V, E)$ to be unweighted in this section. However, we must state that all the results presented herein are extendible with minor scalings to the case of weighted undirected graphs as well.

3.4.1 Connected Bi-partitions

Definition 7 *Connected bi-partition ($P(S, S')$): A cut of the graph G which contains exactly two mutually exclusive and exhaustive connected subgraphs S and S'.*

Let, $V(S)$ and $V(S')$ be the mutually exclusive and exhaustive subsets of $V(G)$, $E(S)$ and $E(S')$, the sets of edges in the respective components S and S' of P and $E(S, S')$, the set of edges that violate P i.e. have one end in S and the other in S'. The qualifiers *mutually exclusive and exhaustive* simply imply the following properties for the set of vertices determined by $P(S, S')$:

$$a. \quad V(S) \cap V(S') = \phi \qquad\qquad b. \quad V(S) \cup V(S') = V(G) \tag{3.18}$$

Similarly, for the set of edges: $E(S) \cap E(S') = E(S) \cap E(S, S') = E(S') \cap E(S, S') = \phi$ and $E(S) \cup E(S, S') \cup E(S') = E(G)$. Also, let $\mathcal{T}(S)$ and $\mathcal{T}(S')$ be the set of spanning trees in the respective component sets S and S'. We denote by $\mathcal{P}(G)$, the set of all bi-partitions of $G(V, E)$. Clearly, a given $P(S, S')$ represents a state of the network in

which $E(S, S')$ have failed. A node $i \in V(S)$ stays connected to $|V(S)|$ nodes (including itself) and gets disconnected from $|V(S')|$ nodes. In the following relationship we show how topological centrality of node i is related to a weighted sum of $|V(S)'|$ over all the bi-partitions $P \in \mathcal{P}(G)$ of the network.

Theorem 4 *For a node $i \in V(G)$:*

$$l_{ii}^{+} \propto \sum_{P \in \mathcal{P}(G)}^{i \in V(S)} |\mathcal{T}(S)||\mathcal{T}(S')||V(S')| \tag{3.19}$$

To paraphrase, given a bi-partition $P(S, S') \in \mathcal{P}(G)$, such that $i \in V(S)$ and $j \in V(S')$, Lemma 2 yields: $\varepsilon(\mathcal{F}^{ii}_{n-2|P})/\varepsilon(\mathcal{F}^{jj}_{n-2|P}) = |V(S')|/|V(S)|$. Clearly, for a given bi-partition, nodes in the larger of the two components of P have a lower number of spanning forests rooted at them than those in the smaller component and vice versa. Hence,

$$l_{ii}^{+} - l_{jj}^{+} \quad \propto \sum_{P \in \mathcal{P}(G)}^{i \in V(S), j \in V(S')} |\mathcal{T}(S)||\mathcal{T}(S')|(|V(S')| - |V(S)|) \tag{3.20}$$

can be interpreted as a comparative measure of connectedness of nodes i and j. Note that for $P \in \mathcal{P}(G)$, the RHS of (3.20) is zero when nodes i and j belong to the same component of P or if $|V(S)| = |V(S')|$ and positive when $i \in V(S), j \in V(S')$ and $|V(S')| > |V(S)|$ or vice versa. Therefore, a node i with higher topological centrality stays connected to a greater number of nodes on an average in a disconnected network, than one with lower topological centrality and is consequently more immune to random edge failures in the network. Also, by simple extension,

Theorem 5 *For an edge $e_{ij} \in E(G)$:*

$$l_{ij}^{+} \propto \sum_{P \in \mathcal{P}(G)}^{i,j \in V(S), e_{ij} \notin E(S,S')} |\mathcal{T}(S)||\mathcal{T}(S')||V(S')| \tag{3.21}$$

Therefore, when an edge e_{ij} is not part of the set of edges that violate the cut, and $i, j \in V(S)$, this edge is only usable by the nodes in $V(S)$. In other words, $V(S') \times V(S')$ node pairs do not use e_{ij} for communications. Clearly, higher the value of l_{ij}^{+}, more the number of node pairs on an average that do not require e_{ij} in disruptive failure scenarios, and consequently lower its connectedness. Finally, we extend these results to the Kirchhoff index:

Theorem 6 *For a graph $G(V, E)$:*

$$\mathcal{K}(G) = \frac{\sum_{P \in \mathcal{P}(G)} |\mathcal{T}(S)||\mathcal{T}(S')||V(S)||V(S')|}{n|\mathcal{T}(G)|} \tag{3.22}$$

Note, the denominator $|\mathcal{T}(G)|$ is the number of spanning trees of the graph G. Higher the number of spanning trees in a graph G, more the *path diversity* in the graph and thus more difficult it is to break G. Also, the product $|V(S)| \cdot |V(S')|$ is instrumental in determining the value of $\mathcal{K}(G)$. Higher the value of this product, greater the number of node pairs that cannot communicate in disruptive failure conditions. Also, the sum $|V(S)| + |V(S')| = n$, is a constant, the product is maximized when $|V(S)| = |V(S')| \approx \frac{|V(G)|}{2}$, or in other words when the bi-partitions are equitable in size. Therefore, for two networks G_1 and G_2, $\mathcal{K}(G_1) > \mathcal{K}(G_2)$, shows that the bi-partitions of G_1 on an average are more equitably balanced than those of G_2. And, as we have argued, equitable bi-partitions imply greatest number of inter-node pair disruptions. Moreover, another way to interpret the Kirchhoff index is the following:

$$\mathcal{K}(G) = \frac{n-1}{n} \frac{\sum_{P \in \mathcal{P}(G)} |\mathcal{T}(S)||\mathcal{T}(S')||V(S)||V(S')|}{\sum_{P \in \mathcal{P}(G)} |\mathcal{T}(S)||\mathcal{T}(S')||E(S, S')|} \tag{3.23}$$

The result above follows by expressing the set $\mathcal{T}(G)$ in terms of the cut edge set $E(S, S')$ and the counts of the spanning trees defined over S and S':

$$\mathcal{T}(G) = \frac{1}{n-1} \sum_{P \in \mathcal{P}(G)} |\mathcal{T}(S)||\mathcal{T}(S')||E(S, S')| \tag{3.24}$$

Note that for a given partition $P(S, S')$, the contributing factor to the Kirchhoff index is simply the following term:

$$\mathcal{K}(G)_{|P(S,S')} = \frac{|V(S)||V(S')|}{|E(S, S')|} \tag{3.25}$$

This is largely owing to the canceling out of the term $|\mathcal{T}(S)||\mathcal{T}(S')|$ which is present in both the numerator and the denominator. Seen in this light, higher the value of $\mathcal{K}(G)$, a fewer number of edges upon removal, results in an equitable partition and thus easier it is to disrupt pairwise communications in the graph (representing the network).

Thus $\mathcal{K}(G)$ represents the average connectedness of all the nodes when a failure of a subset of edges partitions the network into two components, thereby truly reflecting overall network robustness.

3.4.2 A Case Study: When the Graph is a Tree

We now study the special case of trees to illustrate the topological (graph theoretic) interpretations presented thus far in a simpler setting. Recall, a tree $T(V, E)$ of order $n = |V(T)|$ is a connected acyclic graph with exactly $n - 1 = |E(T)|$ edges. As each of the $n-1$ edges $e_{ij} \in E(T)$, upon deletion produces a unique partition $P(S, S') \in \mathcal{P}(T)$, we conclude that there are exactly $n - 1$ connected bi-partitions of a tree. Moreover, the two sub-graphs S and S' are also trees themselves, such that $|\mathcal{T}(S)| = |\mathcal{T}(S')| = 1$ for any partition $P(S, S')$. For the nodes of the tree, we then obtain an elegant closed form for topological centrality in the following corollary.

Corollary 1

$$l_{ii}^{+} = \frac{1}{n^2} \sum_{P \in \mathcal{P}(T)}^{i \in V(S)} |V(S')|^2 \tag{3.26}$$

More importantly, in a tree, the shortest path distance $SPD(i, j)$ and the effective resistance distance Ω_{ij} between the node pair (i, j) is exactly the same i.e.

$$SPD(i, j) = \Omega_{ij} = l_{ii}^{+} + l_{jj}^{+} - l_{ij}^{+} - l_{ji}^{+} \tag{3.27}$$

The result above is simply due to the fact that a tree is an acyclic graph. It is easy to see that

$$l_{ii}^{+} = \sum_{j=1}^{n} SPD(i, j) - Tr(\mathbf{L}^{+}) \quad \Rightarrow \quad l_{ii}^{+} \propto \sum_{j=1}^{n} SPD(i, j) \tag{3.28}$$

But the node $i^* \in V(T)$ for which $\sum_{j=1}^{n} SPD(i^*, j)$ is the least, is the so called *tree center* of T. Thus the node with the highest topological centrality, is also the tree center if the graph is a tree.

But what about the edges in the tree? It is easy to see that the form for l_{ij}^{+} is the same as that in (3.27), except for an added condition that the edge $e_{ij} \notin E(S, S')$. But there is something of greater interest here. Note that $\Omega_{ij} = 1$, $\forall e_{ij} \in E(T)$. Thus,

rearranging the terms of equation (3.27), we obtain:

$$l_{ij}^{+} = \frac{l_{ii}^{+} + l_{jj}^{+} - 1}{2} = l_{ji}^{+} \tag{3.29}$$

Therefore, l_{ij}^{+} of the edge $e_{ij} \in E(T)$ is simply a linear average of the reciprocal of topological centralities of its end-points. Thus, greater the value of $\mathcal{C}^*(i-j)$, closer to the tree center the edge is. Another way to interpret this result is that the centrality of an edge is determined by the centralities of its end-points. Intuitively, an edge with both its end-points closer to the core, is more central than those that have either one or neither end-points closer to the core.

And finally, the Kirchhoff index for a tree:

Corollary 2

$$\mathcal{K}(T) = \frac{1}{n} \sum_{P \in \mathcal{P}(T)} |V(S)||V(S')| \tag{3.30}$$

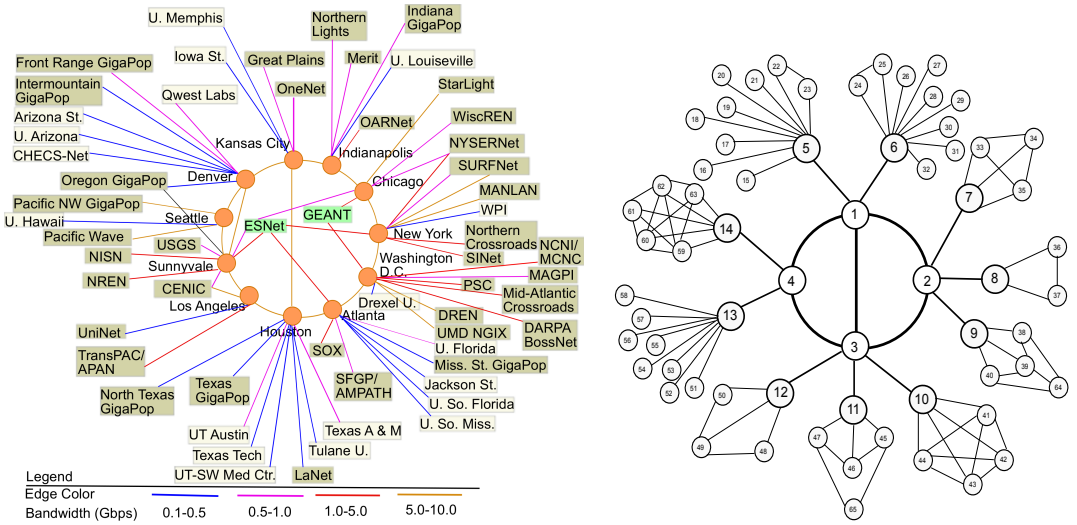
These results further knit our centrality indices into the broader body of knowledge (cf. [42]).

3.5 Empirical Evaluations

In this section, we empirically study the properties of the topological centrality index $(\mathcal{C}^*(i), \mathcal{C}^*(i-j))$ and the Kirchhoff index. We use $\mathcal{K}^*(G) = \mathcal{K}(G)^{-1}$, henceforth, to maintain *higher is better*. Also, as topological centrality of an edge/link is well approximated in terms of the node centralities of its end-points, for brevity we confine the presentation to the case of node centralities alone. We first show in §3.5.1, how a rank-order of nodes in terms of their topological centralities captures their structural roles in the network and then in §3.5.2 demonstrate how it, along with Kirchhoff index, is appropriately sensitivity to rewirings and local perturbations in the network.

3.5.1 Identifying Structural Roles of Nodes

Consider the router level topology of the Abilene network (cf. Fig. 3.2(a)) [43]. At the core of this topology, is a ring of 11 POP's, spread across mainland US, through which several networks interconnect. Clearly, the connectedness of such a network is dependent



(a) Abilene Topology (b) Simulated topology

Figure 3.2: Abilene Network and a simulated topology.

heavily on the low degree nodes on the ring. For illustration, we mimic the Abilene topology, with a simulated network (cf. Fig. 3.2(b)) which has a 4-node core $\{v_1, ..., v_4\}$ that connects 10 networks through gateway nodes $\{v_5, ..., v_{14}\}$ (cf. Fig.3.2(b)).

Fig. 3.3 shows the (max-normalized) values of geodesic closeness (GC), subgraph centrality (SC) and topological centrality $\mathcal{C}^*$ for the core $\{v_1, ..., v_4\}$, gateway $\{v_5, ..., v_{14}\}$ and other nodes $\{v_{15}, ..., v_{65}\}$ in topology (cf. Fig.3.2(b)). Notice that v_5 and v_6, two of the gateway nodes in the topology, have the highest values of degree in the network i.e. $(d(v_5) = d(v_6) = 10)$ while v_{14} has the highest subgraph centrality (SC). In contrast, $\mathcal{C}^*(i)$ ranks the four core nodes higher than all the gateway nodes with v_1 at the top. The relative peripherality of v_5, v_6 and v_{14} as compared to the core nodes requires no elaboration. As far as geodesic centrality (GC) is concerned, it ranks all the nodes in the subnetwork abstracted by v_5, namely $v_{15} - v_{23}$, as equals even though v_{22} and v_{23} have redundant connectivity to the network through each other and are, ever so slightly, better connected than the others - a property reflected in their $\mathcal{C}^*(i)$ rankings.

We see similar characterization of structural roles of nodes in two real world networks in terms of topological centrality: the western states power-grid network [44] and a social network of co-authorships [45], as shown through a color scheme based on $\mathcal{C}^*(i)$ values in Fig. 3.4. Core-nodes connecting different sub-communities of nodes in both these

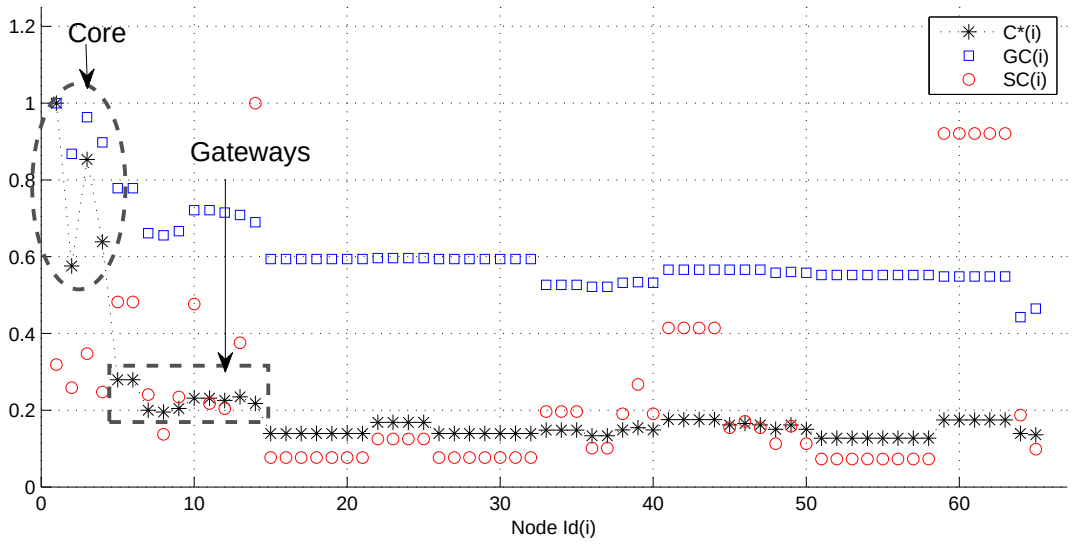


Figure 3.3: Max-normalized centralities for simulated topology.

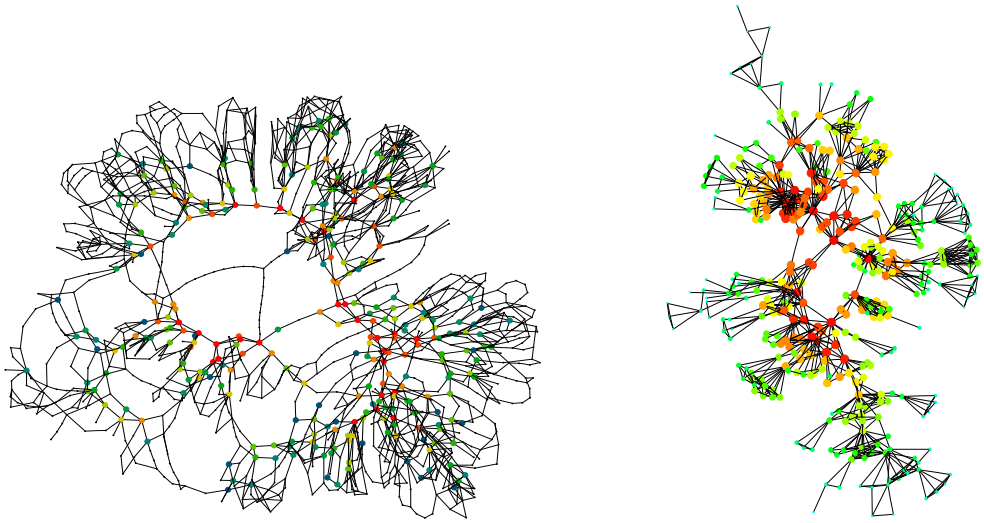
real world networks are recognized effectively by topological centrality as being more central (*Red* end of the spectrum) than several higher degree peripheral nodes.

3.5.2 Sensitivity to Local Perturbations

An important property of centrality measures is their sensitivity to perturbations in network structure. Traditionally, structural properties in real world networks have been equated to average statistical properties like power-law/scale-free degree distributions and rich club connectivity [19, 20, 21]. However, the same degree sequence $D = \{d(1) \geq d(2) \geq ... \geq d(n)\}$, can result in graphs of significantly varying topologies. Let $\mathcal{G}(D)$ be the set of all connected graphs with scaling sequence D. The generalized Randić index $R_1(G)$ [46, 47]:

$$R_1(G) = \sum_{e_{ij} \in E(G)} d(i)d(j) \tag{3.31}$$

where $G \in \mathcal{G}(D)$, is considered to be a measure of overall connectedness of G. Higher $R_1(G)$ suggests that nodes of higher degrees connect with each other with high probability thereby displaying the so-called *rich club connectivity* (RCC) in G [48]. Similarly, the average of each centrality/betweenness index (GC, SC, GB, RB averaged over the



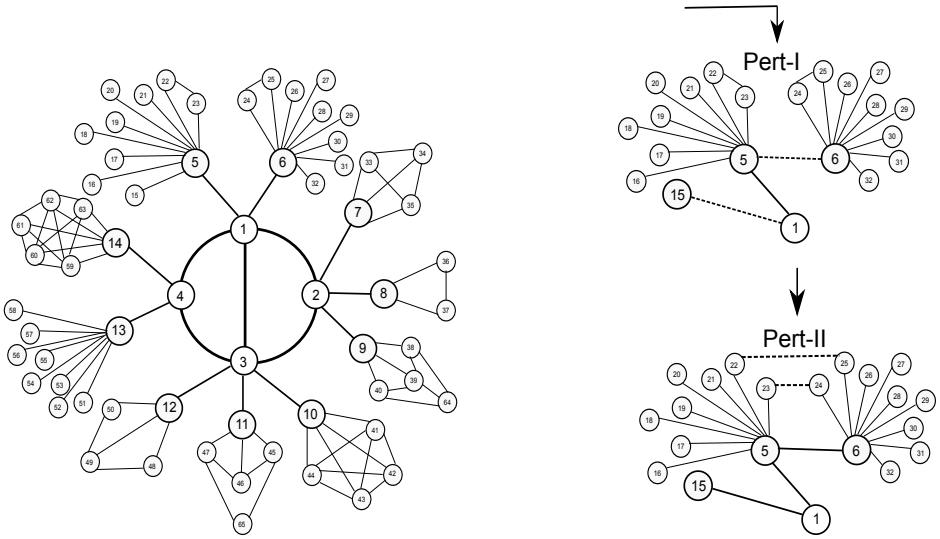
(a) The western-states power grid [44] (b) Co-authorships in net. sciences [45]

Figure 3.4: Real world networks: $Red \rightarrow Turqoise$ in order of decreasing $\mathcal{C}^*(i)$.

set of nodes), is in itself a global structural descriptor for the graph G [29]. We now examine the sensitivity of each index with respect to local perturbations in the subnetwork abstracted by the core node v_1 and its two gateway neighbors v_5 and v_6.

First, we rewire edges $e_{15,5}$ and $e_{6,1}$ to $e_{15,1}$ and $e_{6,5}$ respectively (cf. PERT-I Fig. 3.5(b)). PERT-I is a degree preserving rewiring which only alters local connectivities i.e. neither individual node degrees nor average node degree changes. Fig. 3.6(a) and (b) respectively show the altered values of centralities $(\mathcal{C}^*, GC, SC)$ and betweennesses, geodesic and random-walk i.e. (GB, RB), after PERT-I. Note, after PERT-I, v_{15} is directly connected to v_1 which makes $\mathcal{C}^*(v_{15})$ comparable to other gateway nodes while $SC(v_{15}), GB(v_{15}), RB(v_{15})$ seem to be entirely unaffected. Moreover, PERT-I also results in v_6 losing its direct link to the core, reflected in the decrease in $\mathcal{C}^*(v_6)$ and a corresponding increase in $\mathcal{C}^*(v_5)$. $\mathcal{C}^*(i)$, however, still ranks the core nodes higher than v_5 (whereas SC, GB, RB do not) because PERT-I being a local perturbation should not affect nodes outside the sub-network — v_1 continues to abstract the same sub-networks from the rest of the topology. We, therefore, observe that $\mathcal{C}^*(i)$ is appropriately sensitive to the changes in connectedness of nodes in the event of local perturbations. But what about the network on a whole?

Let G and G_1 be the topologies before and after PERT-I. G_1 is less well connected



(a) Simulated topology (b) Perturbations

Figure 3.5: Structural perturbations in the simulated topology.

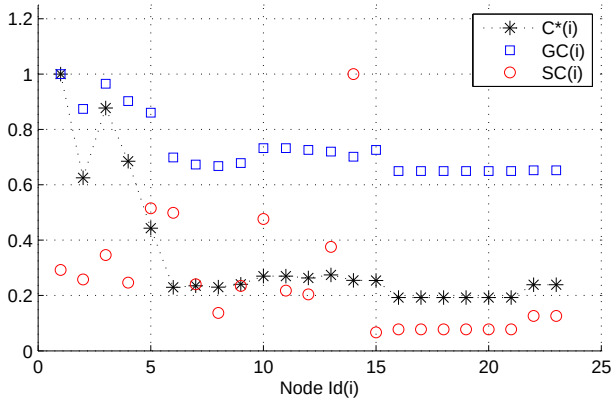
#	Structural Descriptor	PERT-I	PERT-II
1	$\mathcal{K}^*(G)$ (or $\overline{\mathcal{C}^*}$)	$\downarrow$	$\uparrow$
2	$R_1(G)$	$\uparrow$	$\leftrightarrow$
3	$\overline{GC}$	$\downarrow$	$\uparrow$
4	$\overline{SC}$, $\overline{GB}$	$\uparrow$	$\downarrow$
5	$\overline{RB}$	$\uparrow$	$\uparrow$

Table 3.1: Sensitivity to local perturbations, $\overline{X} = 1/n \sum_i^n X(i)$: Avg. node centrality for a network.

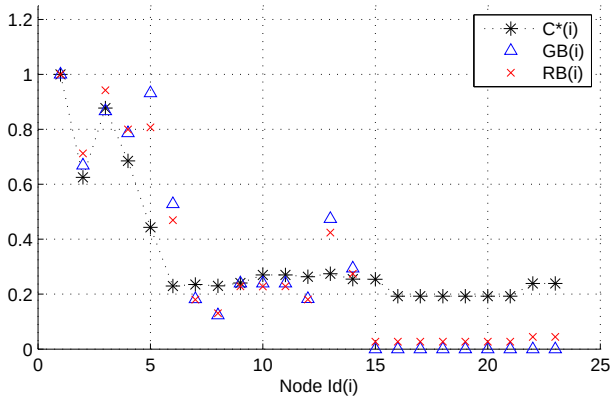
overall than G as the failure of $e_{5,1}$ in G_1 disconnects 19 nodes from the rest of the network as compared to 10 nodes in G. However,

$$\Delta R_1(G \to G_1) = \frac{R_1(G_1) - R_1(G)}{R_1(G)} = 0.029 \tag{3.32}$$

as the two highest degree nodes (v_5 and v_6) are directly connected in G_1 (cf. Table 3.5.2 for the sensitivity of other centrality based global structural descriptors). In contrast, $\Delta\mathcal{K}^*(G \to G_1) = -0.045$, which rightly reflects the depreciation in overall connectedness after PERT-I (recall $\mathcal{K}^*(G) = \mathcal{K}^{-1}(G)$. Table 3.5.2 shows the changes in the average of all centrality and betweenness indices post PERT-I.



(a) Vis-à-vis centralities



(b) Vis-à-vis betweennesses

Figure 3.6: PERT-I: Max-normalized values of centralities and betweennesses for core, gateway and some other nodes.

A subsequent degree preserving perturbation PERT-II of G_1, rewiring $e_{22,23}$ and $e_{24,25}$ to $e_{22,25}$ and $e_{23,24}$, to obtain G_2, creates two cycles in G_2 that safeguard against the failure of edge $e_{5,6}$. This significantly improves local connectivities in the sub-network. However, $\Delta R_1(G_1 \to G_2) = 0$ (and average SC decreases) while $\Delta \mathcal{K}^*(G_1 \to G_2) = 0.036$ which once again shows the efficacy of Kirchhoff index as a measure of global connectedness of networks.

#	Measure	Paths covered	Complexity
1.	Degree	-	$O(m)$
2.	GC, GB	Geodesic paths	$O(n^3)$
3.	$\mathcal{C}^*$	All paths	$O(n^3)$
4.	SC	All paths	$O(n^3)$
5.	RB	All paths	$O(m+n)n^2$
6.	FB	All paths	$O(m^2n)$

Table 3.2: Taxonomy and computational complexities of centrality measures (all nodes).

3.6 A Word on Computational Complexity

We now briefly discuss the practical aspects of computing the topological centrality measure for the set of nodes in the graph representing the complex network. Clearly, it suffices to compute the pseudo-inverse of the matrix $\mathbf{L}$, the Laplacian, to obtain the matrix $\mathbf{L}^+$. The most common method for computing the pseudo-inverse of a matrix mathematically, is to use the singular value decomposition (SVD). Indeed, mathematical software such as MATLAB, come equipped with subroutines, such as *pinv* (for pseudo-inverse), which make use of the SVD factorization. It is common knowledge that the computational complexity for the SVD method is, in terms of worst case complexities, $O(n^3)$ where n is the number of rows/columns of the matrix. Thus the base worst case complexity for computing the topological centrality measure is indeed $O(n^3)$, where n is the order of the graph. This worst case complexity is at par with the competitive centrality measures, like geodesic centrality (GC) and geodesic betweenness (GB) as well as subgraph centrality (SC). On the other hand, it is certainly better than the random-walk betweenness (RB) (cf. Table 3.2).

However, given that real world networks abstracted as graphs are sparse topological objects, the eigen spaces of the matrices $\mathbf{L}$ and $\mathbf{L}^+$ (as discussed in chapter 2), provide useful insights from a computational point of view. Exploiting the fact that $\lambda_n = 0$, the smallest eigen value of $\mathbf{L}$, is unique if the network is connected, it has been shown in [33] that $\mathbf{L}^+$ can be computed by performing a $rank(1)$ perturbation of $\mathbf{L}$ which yields an *invertible* full rank matrix. This method, though relatively faster in practice on MATLAB, still leaves the worst case complexity at $O(n^3)$. We defer a full discussion of the nuances to the next chapter where we present sub-structure analysis of networks

and provide closed form solutions which can be used in a divide-and-conquer fashion over a parallelized infrastructure [49]. For now we make do with providing some useful pointers to some interesting approximation results in the following paragraphs.

In [8], we find a way of approximating $\mathbf{L}^+$ by using fast converging Monte-Carlo algorithms. Such parallel algorithms exploit the sparsity of real world networks which in turn makes the Laplacian $\mathbf{L}$ a sparse matrix (even though $\mathbf{L}^+$ is always full). However, we observe that from the point of view of computing topological centrality alone $(\mathcal{C}^*(i),\ \mathcal{C}^*(i-j))$, all we need are the elements on the diagonal of $\mathbf{L}^+$ and the off-diagonal elements l^+_{ij} for (i,j) such that $[\mathbf{A}]_{ij} = \mathbf{A}]_{ji} > 0$. It is known that subsets of inverse for a sparse matrix can be computed to a given pattern (selective elements), using parallel and multi-frontal approaches (cf. [50] and the references therein).

Finally, another interesting approximation result has recently been proposed in [51]. When the density of edges in the network increases, the hitting time from node i to j can be well approximated as $H_{ij} \approx Vol(G)d(i)^{-1}$. By extension, the commute time becomes:

$$C_{ij} \approx Vol(G)(d(i)^{-1} + d(j)^{-1}) \tag{3.33}$$

So for dense graphs, we can approximate topological centrality for nodes and edges using the node degree distribution alone. Most importantly, this result implies that topological centrality, which is a measure of the overall position and connectedness of a node, is determined entirely by its local connectedness determined by its degree, a remarkable result indeed.

3.7 Summary

In this chapter, we presented a geometric perspective on robustness in complex networks in terms of the Moore-Penrose pseudo-inverse of the graph Laplacian. We proposed topological centrality as a measure of robustness for the nodes and edges of a network $(\mathcal{C}^*(i),\ \mathcal{C}^*(i-j))$ and Kirchhoff index $(\mathcal{K}(G))$ that respectively reflect the length of the position vector for a node and the overall volume of the graph embedding and therefore are suitable geometric measures of robustness of individual nodes and the overall network. Additionally, we provided interpretations for these indices in terms of the

overhead incurred in random detours as well as in terms of the recurrence probabilities and voltage distribution in the EEN corresponding to the network. All indices reflect the global connectedness properties of individual nodes and edges as well as the network on a whole, particularly in the event of multiple edge failures that leave the network disconnected. Through numerical analysis on simulated and real world networks, we demonstrated that topological centrality captures the structural roles played by nodes and edges in networks and, along with Kirchhoff index, is suitably sensitive to perturbations/re-wirings. In terms of computational complexity, topological centrality compares well with other geodesic and all-paths based indices in literature (cf. Table 3.2) and performs better than random-walk betweenness in the asymptotic case. In the next chapter, we investigate sub-structures of a network using a novel divide-and-conquer based approach that yields, amongst other things, a parallelizable methodology for computing $\mathbf{L}^{+}$ for networks of large orders.

Chapter 4

Sub-Structure Analysis and an Incremental Approach to Computing L^+

The Moore-Penrose pseudo-inverse and the sub-matrix inverses of the Laplacian have evoked great interest in recent times. Their applications span fields as diverse as probability and mathematical chemistry, collaborative recommendation systems and social networks, epidemiology and infrastructure planning [8, 52, 42, 18, 25, 53, 33]. Alas, despite such versatility, the pseudo-inverse and the sub-matrix inverses of the Laplacian suffer a practical handicap, as alluded to towards the end of the previous chapter. These matrices are notoriously expensive to compute. The standard matrix factorization and inversion based methods employed to compute them [15, 33], incur an $O(n^3)$ computational time, n being the order of the graph (number of vertices in the graph). This clearly impedes their utility particularly when the graphs are either dynamic, i.e. changing with time, or simply of large orders, i.e. have millions of nodes. Online social networks (OSN), typically represented as graphs, qualify on both counts. With time, the number of users as well as the relationships between them changes, thus requiring regular re-computations. As for size, a popular OSN, such as Facebook and Youtube, may easily have hundreds of millions of users. An $O(n^3)$ cost, therefore, is clearly undesirable and an approach for incremental updates is imperative, particularly given that

such changes, in most cases, may be local in nature.

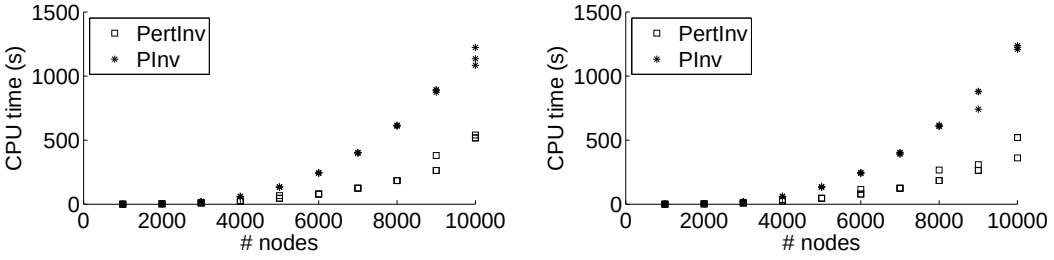
In this chapter, we provide a novel divide-and-conquer based approach for computing the Moore-Penrose pseudo-inverse of the Laplacian for an undirected graph which, in turn, determines all of its sub-matrix inverses as well. We must, however, point out that although our focus for the time being is on computational complexity, the closed form solutions obtained during this exercise have implications in the study of inter-dependent network topologies — the subject of the next chapter. Hence, the results presented here are of great importance.

4.1 Computing $\mathbf{L}^+$

A straightforward approach for computing $\mathbf{L}^+$ is through the eigen-decomposition of $\mathbf{L}$, followed by an inversion of its non-zero eigen values, and finally reassembling the matrix as discussed in chapter 2. A full eigen value decomposition is known to have a computational cost of $O(n^3)$. In practice, however, mathematical software, such as MATLAB, use singular value decomposition (SVD) to compute the pseudo-inverse of matrices (cf. *pinv* in the standard library). This general SVD based method does not exploit the special structural properties of $\mathbf{L}$, such as its sparsity for real world networks, and incurs $O(n^3)$ computational time, n being the number of nodes in the graph. An alternative has recently been proposed in [33] specifically for computing $\mathbf{L}^+$ for a simple, connected, undirected graph. A $rank(1)$ perturbation of the matrix $\mathbf{L}$ can be performed by adding a constant value (a multiple of $1/n$). This perturbation makes $\mathbf{L}$ invertible. $\mathbf{L}^+$ can then be computed from this perturbed matrix as follows:

$$\mathbf{L}^+ = \left(\mathbf{L} + \frac{1}{n}\mathbf{J}\right)^{-1} - \frac{1}{n}\mathbf{J} \tag{4.1}$$

where $\mathbf{J} \in \Re^{n\times n}$ is a matrix of all $1's$. Although the theoretical cost for this method is still $O(n^3)$, in practice it works faster for graphs of arbitrary orders and edge densities than the standard *pinv* method. But the proof of this pudding is in computing! So, to put into context the notion, we present a numerical analysis over Erdös-Rényi graphs (ER-graphs) of varying orders and edge densities. An ER-graph is a random graph determined by parameters (n, ρ), where n is the order of the graph and ρ is the uniform probability for the occurrence of any arbitrary (undirected) edge in the graph [54]. On



(a) $\rho = 0.3$ (b) $\rho = 0.5$

Figure 4.1: Computational times: Erdös-Rényi graphs of varying orders and densities. PertInv: pseudo-inverse computed through $rank(1)$ perturbation [33], PInv: pseudo-inverse computed through standard $pinv$ in MATLAB.

a dedicated machine with a quad-core AMD Opteron processor (800 $Mhz/core$) and 48 GB of primary memory, we compute the $\mathbf{L}^+$ of multiple instances of ER-graphs using the standard $pinv$ library function in MATLAB as well as the perturbed inversion based method from [33] (called $PertInv$ in the figures).

Fig. 4.1 shows a comparison of the two methods for $\rho = \{0.3,\ 0.5\}$ and $n = \{1K,\ 2K, ...,\ 10K\}$. The experiment is repeated 100 times for each parametric combination (n, ρ). The fact that the method from [33] outperforms $pinv$ is self evident, as is the fact that the computational times for both methods rise with increasing values of n. We also observe great consistency (or very little variance) across the different instances for a given (n, ρ), which is not too surprising. What is of interest, however, is that the computational times for a given value of n, do not vary significantly across $\rho = \{0.3,\ 0.5\}$, for either of the two methods. We observe the same for higher values of ρ (not shown here). This implies that the methods are insensitive to the sparsity of the graphs. Moreover, for graphs of $(n,\ \rho) = (10000,\ 0.5)$, the primary memory imprint for both methods is over 2.0 GB when run in MATLAB (a little higher, in fact, for the perturbed inversion method). Although the exact valuess may vary from machine to machine, the figures provide a good rough estimate that suffices for the problem at hand. Consequently, for dynamically changing graphs, in which small local modifications occur every now and then, such methods would incur undue heavy computational costs due to repeated re-computation of $\mathbf{L}^+$ from scratch. On the other hand, for graphs of higher orders ($n > O(10^5)$), such decomposition/inversion based methods are rendered impractical from the point of view of computational time as well

as memory requirements, if performed on a single machine.

In what follows, we show that the computation of the Moore-Penrose pseudo-inverse of the Laplacian can be done in a divide-and-conquer fashion by identifying sub-structures that constitute independent sub-problems computable in parallel. Our method allows efficient incremental updates of $\mathbf{L}^+$ for dynamically changing graphs, without having to compute $\mathbf{L}^+$ all over again. Moreover, computing $\mathbf{L}^+$ for large graphs becomes feasible, in principle, through parallelization of (smaller) independent sub-problems over multiple machines, which can then be re-combined at $O(n^2)$ cost per edge across a division (details in a subsequent section). But first we need to establish a few more preliminary results to further motivate our study.

4.2 $\mathbf{L}^+$, Sub-Matrix Inverses and Effective Resistances

In this section, we discuss the sub-matrix inverses of the Laplacian ($\mathbf{L}$) and their relationship to $\mathbf{L}^+$. We then show an interesting inversion of the relationship between the effective resistance distances and the elements of $\mathbf{L}^+$ that is instrumental to the task at hand.

4.2.1 Sub-Matrix Inverses of L

As stated previously, the combinatorial Laplacian $\mathbf{L}$ of a connected graph $G(V, E)$, is singular and thus non-invertible. However, given that its rank is $n-1$, any $n-1$ combination of columns (or rows) of $\mathbf{L}$ constitutes a linearly independent set. Hence, any $(n-1 \times n-1)$ sub-matrix of $\mathbf{L}$ is invertible. Indeed, the inverses of such $(n-1 \times n-1)$ sub-matrices are made use of in several graph analysis problems: enumerating the spanning trees and spanning forests of the graph [42], determining the random-walk betweenness of the nodes of the graph [25], to name a few. However, the cost of computing an $(n-1 \times n-1)$ sub-matrix inverse is still $O(n^3)$. To compute all such sub-matrix inverses amounts to a time complexity of $O(n^4)$. In the following, we show how they can be computed efficiently through $\mathbf{L}^+$.

Theorem 7 *Let* $\mathbf{L}(\{\overline{n}\}, \{\overline{n}\})$ *be an* $(n-1 \times n-1)$ *sub-matrix of* $\mathbf{L}$ *formed by removing*

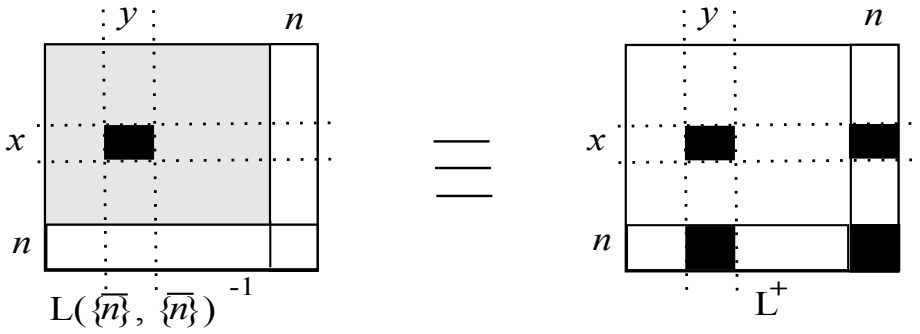


Figure 4.2: Scalar mapping: Sub-matrix inverse of $\mathbf{L}$ to $\mathbf{L}^+$.

the n^{th} row and n^{th} column of $\mathbf{L}$. *Then* $\forall(x,y) \in V(G) \times V(G)$*:*

$$[\mathbf{L}(\{\overline{n}\},\{\overline{n}\})^{-1}]_{xy} = l^+_{xy} - l^+_{xn} - l^+_{ny} + l^+_{nn} \tag{4.2}$$

The result in Theorem 7 above expresses, in scalar form, the general element (x^{th} row, y^{th} column) of the inverse of the sub-matrix $\mathbf{L}(\{\overline{n}\},\{\overline{n}\})$ in terms of the elements of $\mathbf{L}^+$, as claimed. As the choice of the n^{th} row and column is arbitrary, we can see that the result holds in general for any $(n-1 \times n-1)$ sub-matrix (permuting the rows and columns of $\mathbf{L}$ as per need). The cost of computing $\mathbf{L}(\{\overline{n}\},\{\overline{n}\})^{-1}$ for a given vertex n is $O(n^2)$. Therefore, all sub-matrix inverses can be computed in $O(n^3)$ time from $\mathbf{L}^+$, which itself can be computed in $O(n^3)$ time, even if the standard methods are used. This is clearly an order of magnitude improvement. Henceforth, we focus entirely on $\mathbf{L}^+$.

4.2.2 The Effective Resistance Distance and $\mathbf{L}^+$

Recall, that the pairwise effective resistance distance between nodes x and y (Ω_{xy}) can be expressed in terms of the elements of $\mathbf{L}^+$ as follows:

$$\Omega_{xy} = l^+_{xx} + l^+_{yy} - l^+_{xy} - l^+_{yx} \tag{4.3}$$

We now invert the elegant form in (4.3) to derive an important result in the following lemma which gives us the general term of $\mathbf{L}^+$ in terms of the distance function Ω.

Lemma 1 $\forall(x,y,z) \in V(G) \times V(G) \times V(G)$:

$$l^+_{xy} = \frac{1}{2n}\left(\sum_{z=1}^{n} \Omega_{xz} + \Omega_{zy} - \Omega_{xy}\right) - \frac{1}{2n^2}\sum_{x=1}^{n}\sum_{y=1}^{n}\Omega_{xy} \tag{4.4}$$

The RHS in Lemma 1 above is composed of two terms: a triangle inequality of effective resistances [41] and a double summand over all pairwise effective resistances in the EEN. It is easy to see that the double-summand simply reduces to a scalar multiple of the trace of $\mathbf{L}^+$ ($Tr(\mathbf{L}^+) = \sum_{z=1}^{n} l^+_{zz}$). Thus the functional half that determines the elements of $\mathbf{L}^+$, is the triangle inequality of the effective resistances, while the double summand contributes an additive constant to all the entries of $\mathbf{L}^+$. We illustrate the utility of this result, with the help of two kinds of graphs on the extremal ends of the connectedness spectrum: the star and the clique.[1]

The Star

A star of order n is a tree with exactly one vertex of degree $n-1$, referred to as the *root*, and $n-1$ pendant vertices each of degree 1, called *leaves*, (cf. Fig. 4.3). By definition, a singleton isolated vertex is also a degenerate star albeit with no leaves. It is easy to see that S_n, being a tree, is the most sparse connected graph of order n (with exactly $n-1$ edges). Also, S_n is the most compact tree of its order (lowest diameter). In the following, we show how $\mathbf{L}^+_{S_n}$ can be computed using the result of Lemma 1.

Corollary 3 *For a star graph S_n of order n, with node v_1 as root and nodes $\{v_2, v_3, ..., v_n\}$ as leaves, $\mathbf{L}^+_{S_n}$ is given by:*

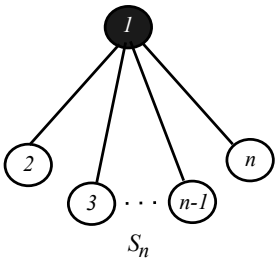
$$l^+_{11} = \frac{n-1}{n^2} \qquad \& \qquad 2 \leq x \leq n, \quad l^+_{1x} = l^+_{x1} = -\frac{1}{n^2} \tag{4.5}$$

$$2 \leq x \leq n, \quad l^+_{xx} = \frac{n^2-n-1}{n^2} \qquad \& \qquad 2 \leq x \neq y \leq n, \quad l^+_{xy} = l^+_{yx} = -\frac{n+1}{n^2} \tag{4.6}$$

The Clique

On the other end of the connectedness spectrum lies the clique. A clique K_n of order n is a complete graph with $\frac{n(n-1)}{2}$ edges. Clearly, the clique is the densest possible graph of order n, as there is a direct edge between any pair of vertices in it. It is also the most compact graph of its order (lowest diameter). Then,

[1] The graphs in these examples are assumed to be unweighted, i.e. all edges have a unit resistance/conductance.



S_n

0.16	-0.04	-0.04	-0.04	-0.04
-0.04	0.76	-0.24	-0.24	-0.24
-0.04	-0.24	0.76	-0.24	-0.24
-0.04	-0.24	-0.24	0.76	-0.24
-0.04	-0.24	-0.24	-0.24	0.76

$L^+_{S_5}$

Figure 4.3: The Star Graph: Pre-computed $\mathbf{L}^+_{S_n}$ for $n = 5$.

Corollary 4 *For a clique K_n of order n, $\mathbf{L}^+_{K_n}$ is given by:*

$$1 \leq x \leq n, \quad l^+_{xx} = \frac{n-1}{n^2} \qquad \& \qquad 1 \leq x \neq y \leq n, \quad l^+_{xy} = l^+_{yx} = -\frac{1}{n^2} \quad (4.7)$$

The results in the corollaries presented above are not just illustrative examples. They are also of interest from a computational point of view, particularly when the graph under study is an unweighted one. Both stars and cliques can occur as motif sub-graphs in any given graph. Indeed, for any non-trivial connected simple graph of order $n \geq 3$, there is at least one sub-graph that is a star. Selecting any vertex i with $d(i) \geq 2$, and conducting a one-hop breadth first search, generates a star sub-graph. Cliques, though not so universal, also occur in real world networks (e.g. citation networks). Therefore, in any divide-and-conquer methodology, both stars and cliques are likely to be found at some stage. We have already established that the cost of computing $\mathbf{L}^+_{S_n}$ and $\mathbf{L}^+_{K_n}$ is $O(1)$ (as they are determined entirely by n) and hence the solution to such a sub-problem, when found, is obtained at the lowest possible cost — a true practical gain.

To conclude, we have demonstrated that there exists a relationship between the elements of $\mathbf{L}^+$ and the pairwise effective resistances in the graph $G(V, E)$, that yields interesting closed form solutions for the pseudo-inverse for special graphs such as stars and cliques. In the subsequent sections, we demonstrate that it can be used to compute $\mathbf{L}^+$ for general graphs as well, incrementally, in a divide-and-conquer fashion.

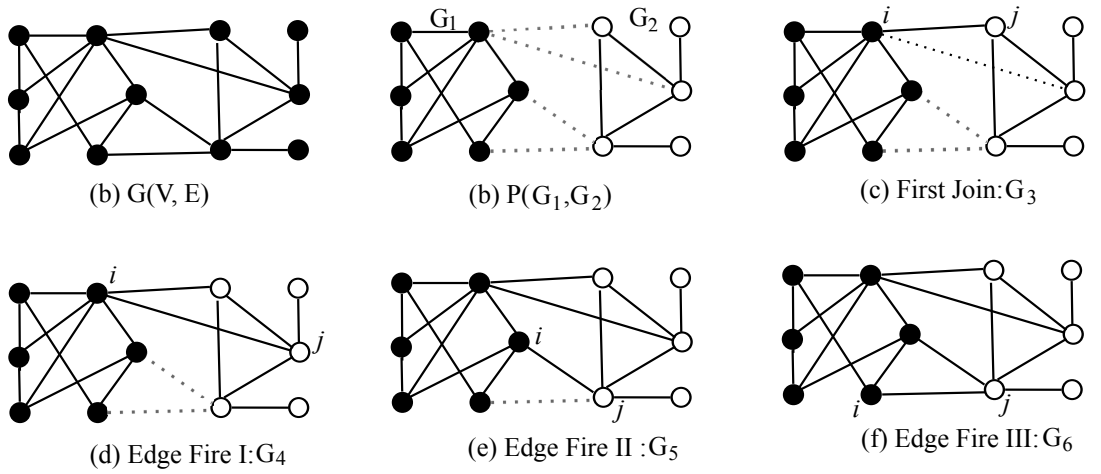


Figure 4.4: Divide-and-Conquer: Connected bi-partition of a graph and the two-stage process: first join followed by three edge firings. The dotted lines represent the edges that are not part of the intermediate sub-graph at that stage.

4.3 From Two to One: Computing $\mathbf{L}^+$ by Partitions

In this section, we present our main result — the computation of the Moore-Penrose pseudo-inverse of the Laplacian, or $\mathbf{L}^+$, by means of graph bi-partitions. In §4.3.1, we lay out a *two-stage* process — the *first join* followed by *edge firings* — that underpins our methodology. We then provide specific closed form solutions in §4.3.2.

4.3.1 Connected Bi-Partitions of a Graph and the Two-Stage Process

In order to compute the Moore-Penrose pseudo-inverse of the Laplacian of a simple, connected, undirected and unweighted graph $G(V, E)$ by parts, we must first establish that the problem can be decomposed into two, or more, sub-problems that can be solved independently. The solutions to the independent sub-problems can then be combined to obtain the overall result. But before we proceed to do so, a few notational clarifications are in order.

Fig. 4.4(a-b), shows a graph $G(V, E)$ and a connected bi-partition $P(G_1, G_2)$ of it, obtained from the graph $G(V, E)$ by removing the set of dotted edges shown. Recall, each partition $P(G_1, G_2)$ has certain defining characteristics in terms of the set of

vertices as well the set of edges in the graph. Specifically, for the set of vertices:

$$V_1(G_1) \cap V_2(G_2) = \phi \qquad \& \qquad V_1(G_1) \cup V_2(G_2) = V(G) \tag{4.8}$$

Similarly, for the sets of edges: $E_1(G_1) \cap E_2(G_2) = E_1(G_1) \cap E(G_1, G_2) = E_1(G_2) \cap E(G_1, G_2) = \phi$ and $E_1(G_1) \cup E(G_1, G_2) \cup E_2(G_2) = E(G)$. We denote by $\mathcal{P}(G)$, the set of all such connected bi-partitions of the graph $G(V, E)$.

It is easy to see that for an arbitrary connected bi-partition $P(G_1, G_2) \in \mathcal{P}(G)$ both G_1 and G_2 are themselves simple, connected, undirected and unweighted graphs. Hence, the discussion in §4.2 is applicable in its entirety to the sub-graphs G_1 and G_2 independently. Note then that $\mathbf{L}^+_{G_1}$ and $\mathbf{L}^+_{G_2}$, the Moore-Penrose pseudo-inverse of the Laplacians of the sub-graphs G_1 and G_2, must, by definition, exist. The pair $\{\mathbf{L}^+_{G_1}, \mathbf{L}^+_{G_2}\}$, constitutes the solution to two independent sub-problems represented in the set $\{G_1, G_2\}$. All that remains to be shown now is that $\{\mathbf{L}^+_{G_1}, \mathbf{L}^+_{G_2}\}$ can indeed be combined to obtain $\mathbf{L}^+_G$. It is this aspect of the methodology, that we call the *two-stage* process, as explained in detail below.

The original graph $G(V, E)$ can be thought of, in some sense, as a bringing together of the disjoint spanning sub-graphs G_1 and G_2, by means of introducing the edges of the set $E(G_1, G_2)$. Starting from G_1 and G_2, we iterate over the set of edges in $E(G_1, G_2)$ in the following fashion (cf. Fig. 4.4 for a visual reference). Let $e_{ij} \in E(G_1, G_2) : i \in V_1(G_1), j \in V_2(G_2)$, of weight w_{ij} and resistance $\omega_{ij} = w_{ij}^{-1}$ *ohm*, be an arbitrary edge chosen during the first iteration as shown in Fig. 4.4(c). We call this step the *first join* in our two-stage process, whereafter G_1 and G_2 come together to give an intermediate connected spanning sub-graph (say $G_3(V_3, E_3)$).

The first join represents a point of singularity in the reconstruction process, particularly from the perspective of the effective resistance distance. Note that before the first join, the effective resistance distance between an arbitrary pair of nodes $(x, y) \in V(G) \times V(G)$ is infinity, if $x \in V_1(G_1)$ and $y \in V_2(G_2)$, as there is no path connecting x and y. However, once the first edge e_{ij} has been introduced during the first join, the intermediate sub-graph is connected, and the discrepancy no longer exists — all pairwise effective resistances are finite.

Let $\Omega^{G_1} : V_1(G_1) \times V_1(G_1) \to \Re^+$ and $\Omega^{G_2} : V_2(G_2) \times V_2(G_2) \to \Re^+$, be the pairwise

effective resistances defined over the sub-graphs G_1 and G_2, the following holds:

$$\begin{aligned}
\Omega_{xy}^{G_3} &= \Omega_{xy}^{G_1}, && if\ x, y \in G_1 \\
&= \Omega_{xy}^{G_2}, && if\ x, y \in G_2 \\
&= \Omega_{xi}^{G_1} + \omega_{ij} + \Omega_{jy}^{G_2}, && if\ x \in G_1\ \ \&\ \ y \in G_2
\end{aligned}$$

Needless to say, this is a critical step in the process as we need finite values of effective resistances in order to exploit the result in Lemma 1. Hereafter, we can combine the solutions to the independent sub-problems, i.e. $\mathbf{L}_{G_1}^{+}$ and $\mathbf{L}_{G_2}^{+}$, to obtain $\mathbf{L}_{G_3}^{+}$. Indeed, we obtain an elegant scalar form with interesting properties (details in subsequent sections).

Following the first join, the remaining edges in $E(G_1, G_2)$, can now be introduced one at a time to obtain a sequence of intermediate connected sub-graphs ($G_4 \rightarrow G_5 \rightarrow G_6$) which finally ends in $G(V, E)$ (cf. Fig. 4.4(d-f)). We call this second stage of edge introductions, following the first join, *edge firing*. In terms of effective resistances, each edge firing simply creates parallel resistive connections, or alternative paths, in the graph. Algebraically, each edge firing is a $rank(1)$ perturbation of the Laplacian for the intermediate graph from the previous step. Thus, the Moore-Penrose pseudo-inverse of the Laplacians for the intermediate sub-graph sequence ($G_4 \rightarrow G_5 \rightarrow G_6$) can be obtained starting from $\mathbf{L}_{G_3}^{+}$ using standard perturbation methods [55] (details in subsequent sections).

To summarize, therefore, during the two-stage process we obtain a sequence of connected spanning sub-graphs of $G(V, E)$ starting from a partition $P(G_1, G_2) \in \mathcal{P}(G)$, performing the first join by arbitrarily selecting an edge $e_{ij} \in E(G_1, G_2)$, and then firing the remaining edges, one after the other, in any arbitrary order. The number of connected spanning sub-graphs of $G(V, E)$ constructed during the two-stage process is exactly $|E(G_1, G_2)|$ ($= 4$ for the example in Fig. 4.4). Note that, the order in which these sub-graphs are generated, is of no consequence whatsoever. Next, we show how to obtain $\mathbf{L}^{+}$ for all the intermediate sub-graphs in the sequence.

4.3.2 The Two-Stage Process and $\mathbf{L}^{+}$

We now present the closed form solutions for the Moore-Penrose pseudo-inverse of the Laplacians of the set of intermediate sub-graphs obtained during the two-stage process.

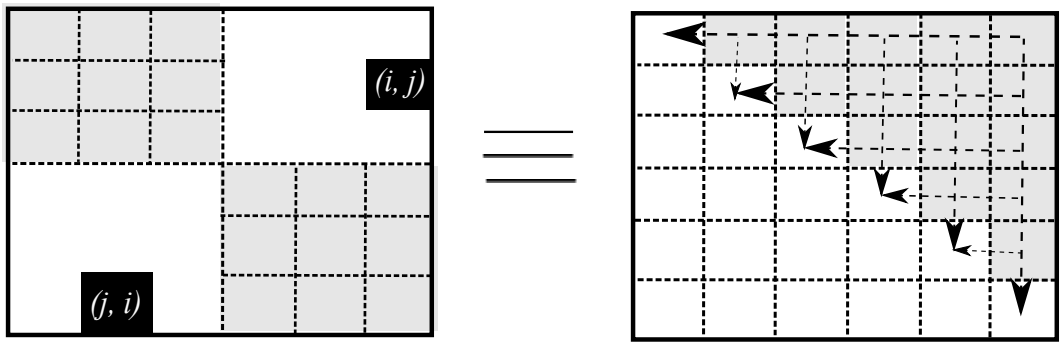


Figure 4.5: The First Join: Scalar mapping $\left(\mathbf{L}^{+}_{G_1},\ \mathbf{L}^{+}_{G_2}\right)$ to $\mathbf{L}^{+}_{G_3}$. The grey blocs represent relevant elements in $\mathbf{L}^{+}_{G_1}$, $\mathbf{L}^{+}_{G_2}$ and $\mathbf{L}^{+}_{G_3}$. Arrows span the elements of the upper triangular of $\mathbf{L}^{+}_{G_3}$ that contribute to the respective diagonal element pointed to by the arrow head: $l^{+(3)}_{kk} = -\left(\sum_{i=1}^{k-1} l^{+(3)}_{ik} + \sum_{j=k+1}^{n} l^{+(3)}_{kj}\right)$.

The First Join

Given, two simple, connected, undirected graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ let $\mathbf{L}^{+}_{G_1}$ and $\mathbf{L}^{+}_{G_2}$, be the respective Moore-Penrose pseudo-inverses of their Laplacians. Also, let $n_1 = |V_1(G_1)|$ and $n_2 = |V_2(G_2)|$ be the orders of the two graphs. We denote by $l^{+(1)}_{xy}$ and $l^{+(2)}_{xy}$ respectively the general terms of the matrices $\mathbf{L}^{+}_{G_1}$ and $\mathbf{L}^{+}_{G_2}$. Next, let the *first join* between G_1 and G_2 be performed by introducing an edge e_{ij} between the graphs G_1 and G_2 to obtain $G_3(V_3, E_3)$; where $i \in V_1(G_1)$ and $j \in V_2(G_2)$. Clearly, $V_3(G_3) = V_1(G_1) \cup V_2(G_2)$ and $E_3(G_3) = E_1(G_1) \cup \{e_{ij}\} \cup E_2(G_2)$. Thus, $|V_3(G_3)| = n_3 = n_1 + n_2$ and $E_3(G_3) = m_3 = m_1 + 1 + m_2$. By convention, the vertices in $V_3(G_3)$ are labeled in the following order: the first n_1 vertices $\{1, 2, ..., n_1\}$ are retained, *as is*, from $V_1(G_1)$ and the remaining n_2 vertices are labelled $\{n_1 + 1, n_1 + 2, ..., n_1 + n_2\}$ in order from $V_2(G_2)$. We denote by $\mathbf{L}^{+}_{G_3}$ the pseudo-inverse and $l^{+(3)}_{xy}$ its general term. Then,

Theorem 8 $\forall (x, y) \in V_3(G_3) \times V_3(G_3)$,

$$
\begin{aligned}
l^{+(3)}_{xy} &= l^{+(1)}_{xy} - \frac{n_2 n_3 \left(l^{+(1)}_{xi} + l^{+(1)}_{iy}\right) - n_2^2 \left(l^{+(1)}_{ii} + l^{+(2)}_{jj} + \omega_{ij}\right)}{n_3^2}, \qquad if\ x, y \in G_1 \\
&= l^{+(2)}_{xy} - \frac{n_1 n_3 \left(l^{+(2)}_{xj} + l^{+(2)}_{jy}\right) - n_1^2 \left(l^{+(1)}_{ii} + l^{+(2)}_{jj} + \omega_{ij}\right)}{n_3^2}, \qquad if\ x, y \in G_2
\end{aligned}
$$

$$= \frac{n_3\left(n_1 l_{xi}^{+(1)} + n_2 l_{jy}^{+(2)}\right) - n_1 n_2\left(l_{ii}^{+(1)} + l_{jj}^{+(2)} + \omega_{ij}\right)}{n_3^2}, \quad if\ x \in G_1 \ \ \& \ \ y \in G_2$$

The result in Theorem 8 is interesting for several reasons. First and foremost, it clearly shows that the general term of $\mathbf{L}_{G_3}^+$, is a linear combination of the elements of $\mathbf{L}_{G_1}^+$ and $\mathbf{L}_{G_2}^+$. This was indeed our principal claim. Secondly, $\forall (x,y) \in V_3(G_3) \times V_3(G_3)$, each individual $l_{xy}^{+(3)}$ can be computed independent of the others (barring symmetry, i.e. $l_{xy}^{+(3)} = l_{yx}^{+(3)}$, which we shall discuss shortly). They are determined entirely by the specific elements from the i^{th} and j^{th} columns of the matrices $\mathbf{L}_{G_1}^+$ and $\mathbf{L}_{G_2}^+$, depending upon the membership of x and y in the disjoint graphs. This implies that all $l_{xy}^{+(3)}$ can be computed in parallel, as long as we have the relevant elements of $\mathbf{L}_{G_1}^+$ and $\mathbf{L}_{G_2}^+$.

From a cost point of view, the first join requires $O(1)$ computations per element in $\mathbf{L}_{G_3}^+$ — constant number of $\{+,\ -,\ \times,\ /\}$ operations — if $\{\mathbf{L}_{G_1}^+,\ \mathbf{L}_{G_2}^+\}$ is given *a priori*. The common term in the numerator, i.e. $(l_{ii}^{+(1)} + l_{jj}^{+(2)} + 1)$, is an invariant for the elements of $\mathbf{L}_{G_3}^+$ and need only be computed once. This term is simply a linear multiple of the change in trace:

$$\Delta(Tr) = Tr(\mathbf{L}_{G_3}^+) - \left[Tr(\mathbf{L}_{G_1}^+) + Tr(\mathbf{L}_{G_2}^+)\right] \tag{4.9}$$

For details see the proof of Lemma 4 in Appendix. Therefore, we achieve an overall cost of $O(n_3^2)$ for the first join. Last but not the least, we need to compute and store only the upper triangular of $\mathbf{L}_{G_3}^+$. Owing to the symmetry of $\mathbf{L}_{G_3}^+$, the lower triangular is determined automatically. As for the diagonal elements, they come without any additional cost as a result of $\mathbf{L}_{G_3}^+$ being doubly-centered (cf. Fig. 4.5).

Firing an Edge

We now look at the second stage that of *firing an edge* in a connected graph. Given a simple, connected, undirected graph $G_1(V_1, E_1)$, let $e_{ij} \notin E_1(G_1)$ be *fired* to obtain $G_2(V_2, E_2)$. Clearly, $V_2(G_2) = V_1(G_1)$ and $E_2(G_2) = E_1(G_1) \cup \{e_{ij}\}$. Continuing with our convention, we denote by $\mathbf{L}_{G_1}^+$ and $\mathbf{L}_{G_2}^+$ the Moore-Penrose pseudo-inverses of the respective Laplacians. Then,

Theorem 9 $\forall (x,y) \in V_2(G_2) \times V_2(G_2)$,

$$l_{xy}^{+(2)} = l_{xy}^{+(1)} - \frac{\left(l_{xi}^{+(1)} - l_{xj}^{+(1)}\right)\left(l_{iy}^{+(1)} - l_{jy}^{+(1)}\right)}{\omega_{ij} + \Omega_{ij}^{G_1}} \tag{4.10}$$

where $\Omega_{ij}^{G_1}$ is the effective resistance distance between nodes i and j in the graph $G_1(V_1, E_1)$ — an invariant $\forall (x, y) \in V_3(G_3) \times V_3(G_3)$ that is determined entirely by the end-points of the edge e_{ij} being fired. Once again, we observe that the general term of $\mathbf{L}_{G_2}^{+}$ is a linear combination of the elements of $\mathbf{L}_{G_1}^{+}$ and requires $O(1)$ computations per element in $\mathbf{L}_{G_2}^{+}$ — constant number of $\{+,\ -,\ \times,\ /\}$ operations — if $\mathbf{L}_{G_1}^{+}$ is given *a priori*. The rest of the discussion from the preceding sub-section on first join — element-wise independence and upper triangular sufficiency — holds *as is* for this stage too. However, before concluding this section, we extend the result of Theorem 9 to the pairwise effective resistances themselves in the following corollary.

Corollary 5 $\forall (x, y) \in V_2(G_2) \times V_2(G_2)$,

$$\Omega_{xy}^{G_2} = \Omega_{xy}^{G_1} - \frac{\left[\left(\Omega_{xj}^{G_1} - \Omega_{xi}^{G_1}\right) - \left(\Omega_{jy}^{G_1} - \Omega_{iy}^{G_1}\right)\right]^2}{4(\omega_{ij} + \Omega_{ij}^{G_1})} \tag{4.11}$$

The result above is interesting in its own right. Note that computing Ω^{G_2} when the edge density of a graph increases (or the expected commute times in random walks between nodes), is pertinent to many application scenarios [56, 57, 58, 59, 8, 51, 16, 60]. Corollary 5 gives us a way of computing these distances directly without having to compute $\mathbf{L}_{G_2}^{+}$ first.

To conclude, therefore, we have established that the Moore-Penrose pseudo-inverses of the Laplacians of all the intermediate sub-graphs, generated during the two-stage process, are incrementally computable from the solutions at the preceding stage, on an element-to-element basis. We shall return to specific applications of these results to dynamic (time-evolving) graphs and large graphs in general, in a subsequent section. But first, for the sake of completeness, we present the case of structural regress.

4.4 From One to Two: A Case of Regress

We now present analogous results in the opposite direction, that of structural regress of a graph through successive deletion of edges until the graph breaks into two. These results, similar in essence to those presented in the preceding section, are particularly significant with respect to dynamically evolving graphs that change with time (e.g. social networks). Once again, we have two cases to address with respect to edge deletions

viz. (a) *Non-bridge edge*: an edge that upon deletion does not affect the connectedness of the graph (cf. §4.4.1); and (b) *Bridge-edge*: an edge that, when deleted, yields a connected bi-partition of the graph (cf. §4.4.2).

4.4.1 Deleting a Non-Bridge Edge

Given a simple, connected, undirected graph $G_1(V_1, E_1)$, let $e_{ij} \in E_1(G_1)$ be a *non-bridge* edge that is deleted to obtain $G_2(V_2, E_2)$. Clearly, $V_2(G_2) = V_1(G_1)$ and $E_2(G_2) = E_1(G_1) - \{e_{ij}\}$. Once again, we denote by $\mathbf{L}^+_{G_1}$ and $\mathbf{L}^+_{G_2}$ the Moore-Penrose pseudo-inverses of the respective Laplacians. Then,

Theorem 10 $\forall (x, y) \in V_2(G_2) \times V_2(G_2)$,

$$l^{+(2)}_{xy} = l^{+(1)}_{xy} + \frac{\left(l^{+(1)}_{xi} - l^{+(1)}_{xj}\right)\left(l^{+(1)}_{iy} - l^{+(1)}_{jy}\right)}{\omega_{ij} - \Omega^{G_1}_{ij}} \tag{4.12}$$

Note, as e_{ij} is a non-bridge edge, $\Omega^{G_1}_{ij} \neq 1$. In fact, given that $G_1(V_1, E_1)$ is connected, undirected and unweighted, we have: $0 < \Omega^{G_1}_{ij} < 1$. Also, as in the case of the two-stage process, we observe the same element-wise independence for $\mathbf{L}^+_{G_2}$ here as well. Once again, if the quantity of interest is Ω^{G_2} or pairwise expected commute times in random walks, we can simply use the following corollary.

Corollary 6 $\forall (x, y) \in V_2(G_2) \times V_2(G_2)$,

$$\Omega^{G_2}_{xy} = \Omega^{G_1}_{xy} + \frac{\left[\left(\Omega^{G_1}_{xj} - \Omega^{G_1}_{xi}\right) - \left(\Omega^{G_1}_{jy} - \Omega^{G_1}_{iy}\right)\right]^2}{4(\omega_{ij} - \Omega^{G_1}_{ij})} \tag{4.13}$$

4.4.2 Deleting a Bridge Edge

Finally, we deal with the case when a bridge edge is deleted from a graph, thus rendering it disconnected for the first time. This represents the point of singularity in the case of structural regress (analogous to the first join). Continuing with our convention, let $G_1(V_1, E_1)$ be a simple, connected, undirected graph with a bridge edge $e_{ij} \in E_1(G_1)$. Upon deleting e_{ij}, we obtain $G_2(V_2, E_2)$ and $G_3(V_3, E_3)$, two disjoint spanning subgraphs of G_1. The orders of G_1, G_2 and G_3 are respectively given by n_1, n_2 and n_3,

while $\mathbf{L}^+_{G_1}, \mathbf{L}^+_{G_2}$ and $\mathbf{L}^+_{G_3}$ are the respective pseudo-inverse matrices of their Laplacians. It is easy to see that:

$$\begin{aligned} \Omega^{G_1}_{xy} &= \Omega^{G_2}_{xy}, && if\ x, y \in V_2(G_2) \\ &= \Omega^{G_3}_{xy}, && if\ x, y \in V_3(G_3) \end{aligned}$$

and $\Omega^{G_2 \times G_3}_{xy} = \Omega^{G_3 \times G_2}_{xy} = \infty$, as G_1 and G_2 are disjoint. To obtain $\mathbf{L}^+_{G_2}$ and $\mathbf{L}^+_{G_3}$ from $\mathbf{L}^+_{G_1}$, we use the result in Lemma 1.

Theorem 11 *$\forall (x, y) \in V_2(G_2) \times V_2(G_2)$ and $\forall (u, v) \in V_3(G_3) \times V_3(G_3)$,*

$$l^{+(2)}_{xy} = l^{+(1)}_{xy} - \frac{n_2 \sum\limits_{z \in V_2(G_2)} \left(l^{+(1)}_{xz} + l^{+(1)}_{zy} \right) - \sum\limits_{x \in V_2(G_2)} \sum\limits_{y \in V_2(G_2)} l^{+(1)}_{xy}}{n_2^2} \tag{4.14}$$

$$l^{+(3)}_{uv} = l^{+(1)}_{uv} - \frac{n_3 \sum\limits_{w \in V_3(G_3)} \left(l^{+(1)}_{uw} + l^{+(1)}_{wv} \right) - \sum\limits_{u \in V_3(G_3)} \sum\limits_{v \in V_3(G_3)} l^{+(1)}_{uv}}{n_3^2} \tag{4.15}$$

Note also that $\mathbf{L}^+_{G_2} \in \Re^{n_2 \times n_2}$ and $\mathbf{L}^+_{G_2} \in \Re^{n_3 \times n_3}$. For convenience, and without loss of generality, we assume that the rows and columns of $\mathbf{L}^+_{G_1} \in \Re^{n_1 \times n_1}$ have been pre-arranged in such a way that the first $(n_2 \times n_2)$ sub-matrix (upper-left) maps to the sub-graph G_2 and similarly the lower-right $(n_3 \times n_3)$ sub-matrix to G_3.

4.5 Bringing it together: Algorithm, Complexity and Parallelization

In this section, we bring together the results obtained in preceding sections, to bear on two important scenarios: (a) dynamic (time-evolving) graphs (cf. §4.5.1), and (b) real-world networks of large orders (cf. §4.5.2). In each case, we discuss the time complexity and parallelizability of our approach in detail.

4.5.1 Dynamic Graphs: Incremental Computation for Incremental Change

Dynamic graphs are often used to represent temporally changing systems. The most intuitively accessible example of such a system is an online social network (OSN). An

OSN evolve not only in terms of order, through introduction and attrition of users with time, but also in terms of the social ties (or relationships) between the users as new associations are formed, and older ones may fade off. Mathematically, we model an OSN as a dynamic graph $G_\tau(V_\tau, E_\tau)$ where the sub-index τ denotes the time parameter. We now study a widely used model for dynamic, temporally evolving, graphs called *preferential attachment* [19, 61, 21].

The preferential attachment model is a parametric model for network growth determined by parameters (n, κ) such that n is the desired order of the network and κ is the desired average degree per node. In its simplest form, the model proceeds in discrete time steps whereby at each time instant $1 < \tau + 1 \leq n$, a new node $v_{\tau+1}$ is introduced in the network with κ edges. This incoming node $v_{\tau+1}$, gets attached to a node $v_i : 1 \leq i \leq \tau$, through exactly one of its κ edges, with the following probability:

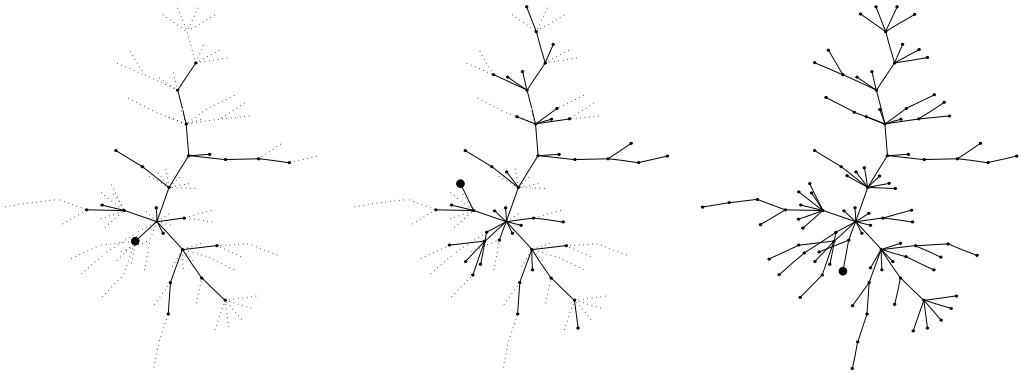
$$P_{\tau+1}(v_i) = \frac{d_\tau(i)}{\sum_{j=1}^{\tau} d_\tau(j)} \tag{4.16}$$

where $d_\tau(i)$ is the degree of node v_i at time τ. The end-points of all the edges emanating from $v_{\tau+1}$ are selected in a similar fashion. At the end of time step $\tau + 1$, we obtain $G_{\tau+1}(V_{\tau+1}, E_{\tau+1})$, and the process continues until we have a graph $G_n(V_n, E_n)$ of order n.[2]

Simplistic though it may seem, this model has been shown to account for several characteristics observed in real-world networks, including the *power law* degree distributions, the *small-world* characteristics and the logarithmic growth of network diameter with time [19, 61, 21]. We return to these in detail in the next sub-section while dealing with the more general case.

It is easy to see that in order to study the structural evolution of dynamic networks, particularly in terms of the sub-structures like spanning trees and forests [42], or centralities of nodes and edges [25, 53]; or voltage distributions in growing conducting networks [62], we require not only the final state $G_n(V_n, E_n)$, but all the intermediate states of the network. In other words, we need to compute the pseudo-inverses of the Laplacians for all the graphs in the sequence $(G_1 \to G_2 \to ... \to G_n)$. Clearly, if the

[2] In practice, for $\kappa > 1$, the process starts with a small connected network as a base substrate to facilitate probabilistic selection of neighbors for an incoming node. For $\kappa = 1$, we may start with a singleton node, and the resulting structure is a tree.



(a) $\tau = 25$ (b) $\tau = 50$ $\tau = 100$

Figure 4.6: Growing a tree by preferential attachment ($n = 100,\ \kappa = 1$). The node v_τ, being added to the tree at time step τ, is emphasized (larger circle). Dotted edges at time steps $\tau = \{25,\ 50\}$ are a visual aid representing edges that are yet to be added.

standard methods are used, the cost at time step τ is $O(\tau^3)$. The overall asymptotic cost for the entire sequence is then $O\left(\sum_{\tau=1}^{n} \tau^3 = \left[\frac{n(n+1)}{2}\right]^2\right)$.

In contrast, using our incremental approach, we can accomplish this at a much lower computational cost. Note that in the case of growing networks, we do not need an explicit divide operation at all. The two sub-problems at time step $\tau + 1$ are given *a priori*. We have, $G_\tau(V_\tau, E_\tau)$ and a singleton vertex graph $\{v_{\tau+1}\}$ as a pair of disjoint sub-graphs. The κ edges emanating from $\{v_{\tau+1}\}$ have end-points in G_τ as determined by (4.16). The conquer operation is then performed through a first join between the singleton node $\{v_{\tau+1}\}$ and the graph $G_\tau(V_\tau, E_\tau)$. We can assume that $\mathbf{L}^+_{G_\tau}$ is already known at this time step (the induction hypothesis). Also, $\mathbf{L}^+_{\{v_{\tau+1}\}} = [0]$ and $n_2 = 1$ during the first join. Substituting in Theorem 8 we obtain the desired results. The rest of the $\kappa - 1$ edges are accounted for by edge firings (cf. the discussion in §4.3). Therefore, we need only $O(\kappa \cdot \tau^2)$ computations at time step τ, and hence $O\left(\kappa \cdot \sum_{\tau=1}^{n} \tau^2 = \kappa \cdot \frac{n(n+1)(2n+1)}{6}\right)$, overall. As $\kappa << n$ in most practical cases, we have an order of magnitude lower average cost than that incurred by the standard methods. Further improvements follow from the parallelizability of our approach. Although we have not discussed it explicitly, it is evident that node and edge deletions can all be handled within this framework in the same way and at the same $O(n^2)$ cost per operation (cf. the discussion in §4.4).

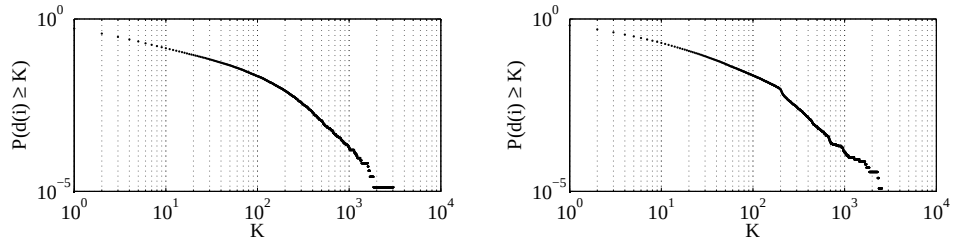
4.5.2 Large Real-World Networks: A Divide-And-Conquer Approach

In order to compute $\mathbf{L}^+$ for an arbitrary graph $G(V, E)$, in a divide-and-conquer fashion, we need to first determine independent sub-graphs of G in an efficient manner. Theoretically, an optimal divide step entails determining a *balanced* connected bi-partition $P(G_1, G_2)$ of the graph G such that $|V(G_1)| \approx |V(G_2)|$ and $|E(G_1, G_2)|$, the number of edges violating the partition, is minimized. Such balanced bi-partitioning of the graph, if feasible, can then be repeated recursively until we obtain sub-graphs of relatively small orders. The solutions to these sub-problems can then be computed and the recursion unwinds to yield the final result, using our two-stage methodology in the respective conquer steps. Alas, computing such balanced bi-partitions, along with the condition of minimality of $|E(G_1, G_2)|$, belongs to the class of *NP-Complete* problems [63], and hence a polynomial time solution does not exist. We therefore need an efficient alternative to accomplish the task at hand. Partitioning of graphs to realize certain objectives has been studied extensively in diverse domains such as VLSI CAD [64], parallel computing, artificial intelligence and image processing [14], and power systems modeling [63, 65]. Perhaps, the most celebrated results in this class of problems are the spectral method [4] and the max-flow = min-cut [66], both of which are computable in polynomial time [4, 67]. Approximation algorithms for the balanced connected bi-partitions problem, for some special cases, have also been proposed [68, 69]. Although useful in specific instances, such methods when used for the divide step may, in themselves, incur high computational costs thus undermining the gains of the conquer step. We need a simple methodology that works well on *real-world* networks.

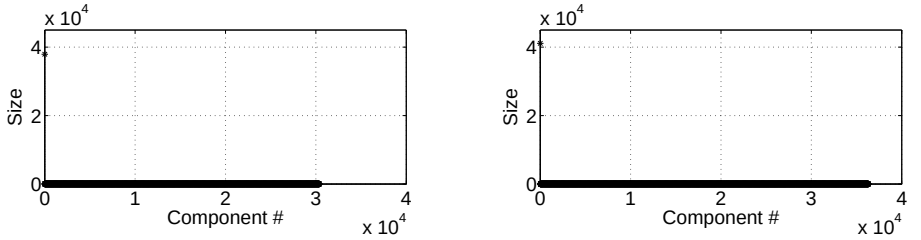
Real-world networks, and particularly online social networks, have been shown to have several interesting structural properties: edge sparsity, power-law scale-free degree distributions, existence of the so called *rich club connectivity* [19, 61, 21], small-world characteristics [44] with relatively small diameters ($O(log\ n)$). Collectively, these properties amount to a simple fact: the overall connectivity between arbitrary node pairs is dependent on higher degree nodes in the network.

$G(V,E)$	$n = \|V(G)\|$	$m = \|E(G)\|$	Leaves	Cut-off	# Comp.	$\|V(GCC)\|$	$\|E(GCC)\|$	# Cut-Edges
Epinions	75,888	405,740	35,763	4,429	30,376	37,924	61,482	102,452
SlashDot	82,168	504, 230	28,499	7,012	36,311	41,084	62,225	164,719

Table 4.1: Basic properties: Epinions and SlashDot networks.



(a) *Epinions*: Degree dist. (b) *SlashDot*: Degree dist.



(c) *Epinions*: Components at cut-off (d) *SlashDot*: Components at cut-off

Figure 4.7: Structural regress: Epinions and SlashDot Networks.

Based on these insights, we now study two real-world online social networks — the *Epinions* and *SlashDot* networks [70] — to attain our objective of a quick and easy divide step. Table 4.1 gives some of the basic statistics about the two networks.[3] It is easy to see that the networks are sparse as $m = O(n) << O(n^2)$ in both cases. Moreover, note that a significant fraction of nodes in the graphs are leaf/pendant nodes, i.e. nodes of degree 1 ($\approx 47\%$ for Epinions and $\approx 34\%$ for *SlashDot*). From Fig. 4.7 (a-b), it is also evident that the node degrees indeed follow a heavy tail distribution in both cases. Thus, there are many nodes of very small degree (e.g. leaves) and relatively fewer nodes of very high degrees in these networks. Therefore, in order to break the graph into smaller sub-graphs, we adopt an incremental regress methodology of deleting high degree nodes. Ordering the nodes in decreasing order by degree, we remove them one at a time. This process divides the set of nodes into three parts at each stage:

a. **The Rich Club**: High degree nodes that have been deleted until that stage.

b. **The Giant Connected Component** (GCC): The largest connected component at that stage.

c. **Others**: All nodes that are neither in the rich club nor the GCC.

We repeat the regress, one node at a time, until the size of the GCC is less than half the size of the original graph. We call this the cut-off point. We then retain the GCC as one of our sub-graphs (one independent sub-problem) and re-combine all the non-GCC nodes together with the rich club to obtain (possibly) multiple sub-graphs (other sub-problems). This concludes the divide step.

Table 4.1 shows the relevant statistics at the cut-off point for the two networks. Note that the cut-off point is attained at the expense of a relatively small number of high degree nodes ($\approx 5\%$ for *Epinions* and $\approx 8\%$ for *SlashDot*). Moreover, the number of nodes in the GCC is indeed roughly half of the overall order, albeit the GCC is surely sparser in terms of edge density than the overall network ($|E(GCC)|/|V(GCC)| = 1.63$ vs. $|E(G)|/|V(G)| = 5.35$ for *Epinions* and $|E(GCC)|/|V(GCC)| = 1.51$ vs. $|E(G)|/|V(G)| = 6.13$ for *Slashdot*). Fig. 4.7 (c-d) shows the sizes (in terms of nodes) of all the connected components for the respective graphs at the cut-off point. It is easy to

[3] Although the networks originally have uni-directional and bi-directional links, we symmetrize the uni-directional edges to make the graphs undirected.

see that other than the GCC, the remaining components are of negligibly small orders. Recombining the non-GCC components together (including the rich club) yields an interesting result. For the *Epinions* network, we obtain two sub-graphs of orders $37,933$ and 31 respectively while for the *Slashdot* network we obtain exactly one sub-graph of order $41,084$. This clearly demonstrates that our simple divide method, yields a roughly equal partitioning of the network — and thus comparable sub-problems — in terms of nodes. The pseudo-inverses of these sub-problems can now be computed in parallel. Albeit, as in the case of all tradeoffs, this equitable split comes at a price of roughly $\kappa = O(n)$ edges that violate the cut (cf. Table 4.1). This yields an $O(n^3)$ average cost for the *two-stage* process (cf. §4.3). However, given the element-wise parallelizability of our method, we obtain the pseudo-inverses in acceptable times of roughly 15 minutes for the *Epinions* and 18 minutes for the *SlashDot* networks.

4.6 Summary

In this chapter, we presented a divide-and-conquer based approach for computing the Moore-Penrose pseudo-inverse of the Laplacian ($\mathbf{L}^+$) for a simple, connected, undirected graph. Our method relies on an elegant interplay between the elements of $\mathbf{L}^+$ and the pairwise effective resistance distances in the graph. Exploiting this relationship, we derived closed form solutions that enable us to compute $\mathbf{L}^+$ in an incremental fashion. We also extended these results to analogous cases for structural regress. Using dynamic networks and online social networks as examples, we demonstrated the efficacy of our method for computing the pseudo-inverse relatively faster than the standard methods. The insights from our work open up several interesting questions for future research. First and foremost, similar explorations can be done for the case of directed graphs (asymmetric relationships), where analogous distance functions — such as the expected commute time in random walks — are defined, albeit the Laplacians (more than one kind in literature) are no longer symmetric [71]. Secondly, matrix-distance interplays of the kind exploited in this work, also exist for a general case of the so called *forest matrix* and its distance counterpart the *forest distance* [72, 73], both for undirected and directed graphs. he results presented here should find natural extensions to the forest matrix and the forest metric, at least for the undirected case. Finally, our closed forms can be

used in conjunction with several interesting approaches for sparse inverse computations [50], to further expedite the pseudo-inverse computation for large generalized graphs. All these motivate ample scope for future work.

Operation	Ω	$\mathbf{L}^+$
First Join	$x, y \in G_1 : \Omega_{xy}^{G_3} = \Omega_{xy}^{G_1}$ $x, y \in G_2 : \Omega_{xy}^{G_3} = \Omega_{xy}^{G_2}$ $x \in G_1, y \in G_2 : \Omega_{xy}^{G_3} = \Omega_{xi}^{G_1} + \omega_{ij} + \Omega_{jy}^{G_1}$	$l_{xy}^{+(1)} - \frac{n_2 n_3 \left(l_{xi}^{+(1)} + l_{iy}^{+(1)}\right) - n_2^2 \left(l_{ii}^{+(1)} + l_{jj}^{+(2)} + \omega_{ij}\right)}{n_3^2}$ $l_{xy}^{+(2)} - \frac{n_1 n_3 \left(l_{xj}^{+(2)} + l_{jy}^{+(2)}\right) - n_1^2 \left(l_{ii}^{+(1)} + l_{jj}^{+(2)} + \omega_{ij}\right)}{n_3^2}$ $\frac{n_3 \left(n_1 l_{xi}^{+(1)} + n_2 l_{jy}^{+(2)}\right) - n_1 n_2 \left(l_{ii}^{+(1)} + l_{jj}^{+(2)} + \omega_{ij}\right)}{n_3^2}$
Edge firing	$\Omega_{xy}^{G_1} - \frac{\left[\left(\Omega_{xj}^{G_1} - \Omega_{xi}^{G_1}\right) - \left(\Omega_{jy}^{G_1} - \Omega_{iy}^{G_1}\right)\right]^2}{4(\omega_{ij} + \Omega_{ij}^{G_1})}$	$l_{xy}^{+(1)} - \frac{\left(l_{xi}^{+(1)} - l_{xj}^{+(1)}\right)\left(l_{iy}^{+(1)} - l_{jy}^{+(1)}\right)}{\omega_{ij} + \Omega_{ij}^{G_1}}$
Non-bridge delete	$\Omega_{xy}^{G_1} + \frac{\left[\left(\Omega_{xj}^{G_1} - \Omega_{xi}^{G_1}\right) - \left(\Omega_{jy}^{G_1} - \Omega_{iy}^{G_1}\right)\right]^2}{4(\omega_{ij} - \Omega_{ij}^{G_1})}$	$l_{xy}^{+(1)} + \frac{\left(l_{xi}^{+(1)} - l_{xj}^{+(1)}\right)\left(l_{iy}^{+(1)} - l_{jy}^{+(1)}\right)}{\omega_{ij} - \Omega_{ij}^{G_1}}$
Bridge delete	$x, y \in G_k : \Omega_{xy}^{G_k} = \Omega_{xy}^{G_1}$	$l_{xy}^{+(1)} - \frac{n_k \sum_{z \in G_k} \left(l_{xz}^{+(1)} + l_{zy}^{+(1)}\right) - \sum_{x \in G_k} \sum_{y \in G_k} l_{xy}^{+(1)}}{n_k^2}$

Table 4.2: Summary of results: Atomic operations of the divide-and-conquer methodology.

Chapter 5

How to "Glue" a Robust Inter-dependent Network?

Modern infrastructure networks are becoming increasingly complex and dependent on one another. An example of such an interdependence is that of an electrical power-grid network regulated by a communication network which in turn depends on the same power-grid for its electrical supply. Due to such interdependence, failures of elements in one network, e.g., a small fraction of nodes in a communication network that is used to control and communicate elements in a smart grid, can induce failures in the other, i.e. the power grid network, which would in turn cause further failures in the communication and control network, thus producing a cascade of failures in the inter-dependent networks. In the recent past, electrical blackouts, like the one in Italy on 28 September 2003 [74], have in fact been caused by such cascaded failures.

Clearly, the extent to which *random* failures or targeted attacks can lead to a cascaded failure, of course, depends on the structural properties of the constituent networks. In their seminal work, Buldyrev et al [75] have demonstrated that inter-dependent networks can behave very differently from each of their constituents. In particular, two robust *power-law* networks when made inter-dependent via "random coupling" may become more vulnerable to random failures. Their work, and those of others, quantify the structural robustness of inter-dependent networks in terms of asymptotic statistical properties such as the existence of *giant connected components* under random failures.

As in the case of robustness of single networks, while this complex network theory characterization of network robustness provides valuable insight into the *general statistical properties of interdependences of classes of random graphs/networks*, they are not very useful in practice, as real networks are *deterministic* and *finite*. In particular, *engineered* infrastructure networks such as power-grids and communication networks, are designed to perform certain specific functions. There topological structures reflect and are constrained by the functional roles of various nodes as well as their geographical locations that are dictated by, say, user population or other resources. In other words, their structures may differ significantly from (theoretically generated) random networks, and the *interdependencies* between two networks (e.g., communication networks and power grids are not arbitrary, but often are determined by geographical and other constraints).

In this chapter, we propose a theoretical framework for assessing structural robustness of inter-dependent networks which is not dependent on specific assumptions of structure like a power-law degree distribution. In doing so, we address the following important questions related to vulnerability assessment and protection of inter-dependent networks: (a) how can we characterize the robustness of the inter-dependent network on a whole? (b) how is the overall robustness of two inter-dependent networks affected by the manner in which the two networks are *coupled* (or "glued") together? (c) how do we judicially select an appropriate *coupling* function, namely, select an appropriate collection of *coupled* node pairs to introduce "interdependence" (i.e., where the two networks are "glued" together) so that the resulting inter-dependent networks are more robust to random failures or targeted attacks? The answer to the last question provides insights as to how we can harden two inter-dependent networks.

In the following we demonstrate that these questions can be answered – at least theoretically – by studying the topological properties captured by our "geometry of networks" approach [76], namely, by using the Moore-Penrose pseudo-inverse of the graph Laplacians ($\mathbf{L}^+$) for the individual networks and that of the inter-dependent network. Based on the topological interpretations of $\mathbf{L}^+$ [77], in particular, the topological centrality metrics and Kirchhoff index, we develop a (*deterministic*) "finite-network" theory to study the *robustness of interdependence*. This theory enables us to mathematically quantify the topological centrality and roles of *coupled* nodes (as well as uncoupled

nodes) in the inter-dependent networks as well as the robustness of inter-dependent networks as a whole. More importantly, this theory allows us to explicitly study how the way node pairs in two networks are *coupled* or "glued" together (thereby introducing interdependence) affects the overall robustness of the resulting inter-dependent networks. Our study leads to some *surprising* (and somewhat counter-intuitive) results: i) simply "gluing" together of structurally most central node pairs in the two constituent networks (when considered independently) does not always result in the most structurally robust *inter-dependent network* ii) coupling a *large* number of structurally least central node pairs in the two constituent networks often leads to more robust inter-dependent networks than coupling the same number of structurally most central node pairs. Intuitively, this result suggests that by *diffusing and distributing inter-dependencies among a large number of (geographically dispersed) node pairs in the two constituent networks produce more robust inter-dependent networks.*

5.1 Modeling Interdependent Networks

In this section we briefly describe the basic notations and a simple graph model for inter-dependent networks. In particular, we introduce a *coupling function*, $\mathcal{C} = \{[u, v]$, which specifies how and where two constituent networks are *coupled* or "glued" together to introduce inter-dependencies among the two networks and form a single *inter-dependent* network.

Given two networks, N_1 and N_2, we represent them in terms of their respective graphs: $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$. Also, for the sake of simplicity, we assume that $|V_1| \approx |V_2|$, i.e., the two networks are of comparable sizes. Let $\mathcal{C}$ denote a collection of node pairs $[u, v]$, one from each constituent network, i.e. $\mathcal{C} = \{[u, v], u \in V_1, v \in V_2\}$. We refer to $\mathcal{C}$ as a *coupling function*, and each node pair $[u, v]$ in $\mathcal{C}$ a *coupled* node pair. Intuitively, the coupled node pairs are where an inter-dependency between two constituent networks are introduced. The cardinality of $\mathcal{C}$, $\kappa = |\mathcal{C}|$, represents the number of inter-dependencies, i.e., the number of *coupled* node pairs. Hence given a coupling function $\mathcal{C}$, two networks, N_1 and N_2, form as a whole a single *inter-dependent* network, denoted as $G_c(V_c, E_c)$.

As a graph, the inter-dependent network $G_c(V_c, E_c)$ can be defined as follows. Let

$[u, v], u \in V_1$ and $v \in V_2$, be a coupled node pair in $\mathcal{C}$. When glued/coupled together, u and v will result in a new node $u \otimes v$ in G_c such that each edge $e(u, x) \in E_1$ will now create an edge $e(u \otimes v, x) \in E(G_c)$, if x is an uncoupled node in N_1; it will create an edge $e(u \otimes v, x \otimes y) \in E(G_c)$ if x is also a node in another coupled pair $[x, y] \in \mathcal{C}$. Similarly, each edge $e(v, y) \in E_2$ will create an edge $e(u \otimes v, y) \in E(G_c)$, if y is an uncoupled node in N_2; it will create an edge $e(u \otimes v, x \otimes y) \in E(G_c)$ if y is also a node in another coupled pair $[x, y] \in \mathcal{C}$. In other words, where there were two vertices u and v in the individual networks, we create a macro-vertex $u \otimes v$ in the glued network with a neighbor set that is a union of the neighbors of u and v in the original networks. This representation clearly captures the inter-dependent nature of the two vertices in question, whereby the macro-vertex $u \otimes v$ fails if either u or v fail in their individual networks. Similarly, the failure of edge $e(u, x)$ in G_1, results in the failure of $e(u \otimes v, x)$ in G_c. Uncoupled nodes and their associated edges (to other uncoupled nodes) in each of the two individual networks are transported to the coupled/inter-dependent $G_c(V_c, E_c)$ as is. Thus, if the number of couplings is κ, then $|V(G_c)| = |V(G_1)| + |V(G_2)| - \kappa$ and $|E(G_c)| = |E(G_1)| + |E(G_2)|$. So the order of the glued network reduces by κ as compared to the total of its constituents, but its volume (number of edges) is the same.

Given this definition of an inter-dependent network $N_c := G_c(V_c, E_c)$ formed by two constituent networks, N_1 and N_2, via the coupling function $\mathcal{C}$, we can directly apply the results obtained in previous chapters: using the Moore-Penrose pseudo-inverse of the graph Laplacians ($\mathbf{L}_c^+$) for the inter-dependent network $G_c(V_c, E_c)$, we define the corresponding *topological centrality* metrics and the *Kirchhoff index* for the inter-dependent network to measure the structural roles of individual (coupled and uncoupled) nodes as well as the overall robustness of the inter-dependent network.

5.2 Effect of Coupling Functions on Network Robustness

Of particular importance, and one of the principal contributions of our work, is that our theory enables us to investigate the effect of the coupling function $\mathcal{C}$ on the overall robustness of the resulting inter-dependent network. In other words, it allows us to vary the manner in which we select the node pairs from the two constituent networks to be glued together, and study the "optimal" way of introducing interdependencies so

as enhance or "harden" the overall robustness of the resulting inter-dependent network. For this purpose, we adopt a topological centrality based ordering of the nodes for the selection process. First, we rank the nodes in the networks N_1 and N_2 in terms of their topological centrality values in their respective networks. Having obtained the ranks, we then define the following three ways of gluing them together: (a) *high-high* i.e. the κ-highest ranked nodes from G_1 with the κ-highest ranked nodes from G_2, (b) *low-low* i.e. the κ-lowest ranked nodes from G_1 with the κ-lowest ranked nodes from G_2 and (c) *random* i.e. κ random pairs from G_1 and G_2. The overall robustness of the coupled/inter-dependent network is then measured in terms of its Kirchhoff index i.e. $\mathcal{K}(G_c)$.

5.2.1 Atomic Coupling: The First Join

We now present closed form theoretical results on how the point of introduction of interdependent edges, influences the structural robustness of the interdependent network.

Theorem 12 *For two disjoint simple, connected, undirected networks $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, let $G_c(V_c, E_c)$ be the interdependent network resulting from coupling $i \otimes j$: $i \in V_1(G_1), j \in V_2(G_2)$. Then,*

$$\mathcal{K}(G_c) = \mathcal{K}(G_1) + \mathcal{K}(G_2) + \frac{n_1 n_2}{n_1 + n_2} \left(l_{ii}^{+(1)} + l_{jj}^{+(2)} + \varepsilon \right) \tag{5.1}$$

where $\varepsilon \to 0$.

Observe, that $\mathcal{K}(G_c)$ is a linear combination of the Kirchhoff indices of the two constituent networks; and a linear term involving $l_{ii}^{+(1)}$ and $l_{jj}^{+(2)}$, the reciprocals of topological centralities of the nodes involved in $i \otimes j$. Thus, greater the topological centrality of i and j in their respective networks, lower the value of $\mathcal{K}(G_c)$, and more robust the interdependent network is. This result suffices when there is exactly one coupling to perform. But seldom is a single coupling enough. In practice, multiple sites are selected and thus we need another result which determines, analogously, the Kirchhoff index when a subsequent coupling is introduced.

5.2.2 Subsequent Atomic Couplings

Theorem 13 *For an interdependent network $G_{c1}(V_{c1}, E_{c1})$, let $G_{c2}(V_{c2}, E_{c2})$ be another interdependent network resulting from coupling $i \otimes j$: $(i, j) \in V_{c1}(G_{c1}) \times V_{c1}(G_{c1})$. Then,*

$$\mathcal{K}(G_{c2}) = \mathcal{K}(G_{c1}) - \frac{\sum_{x \in V_{c1}(G_{c1})} \left(l_{xi}^{+(c1)} - l_{xj}^{+(c1)} \right)^2}{\varepsilon + \Omega_{ij}^{G_{c1}}} \tag{5.2}$$

where $\varepsilon \to 0$.

Needless to say, the form in Theorem 13 provides a non-trivial convex optimization factor. This is particularly important as the choice of node pairs for subsequent couplings is dependent on that of the first join. The optimal solution is thus not guaranteed by a greedy choice. In the next section, we provide some empirical insight into the general case — that of multiple simultaneous couplings.

5.3 Experiments

In the following, we first use a simple example (networks with tree topologies) and then a network with realistic network topology (the Italian power grid) to illustrate the coupling process and the effect of different coupling functions on the robustness of resulting interdependence networks.

5.3.1 When Both Networks are Trees

We consider a simple case to illustrate our theory, where two constituent networks are both trees. We study the coupling of two types of trees: stars and chains/paths of order n, which represent the most well connected (compact) and the least connected of all trees for any given n. More importantly, in the context of power-grids (and to an extent in communication networks), a star topology represents a production/distribution center i.e. the root of the star, while the pendants represent the consumers/first-hop *relay-ers*. Similarly, a chain represents a linear sub-network formed by a series of relay-ers to disseminate power or information. Recall that for a star topology:

$$l_{root}^{+} = \frac{n-1}{n^2}, \quad \textit{and} \quad l_{pendant}^{+} = \frac{n^2 - n - 1}{n^2} \tag{5.3}$$

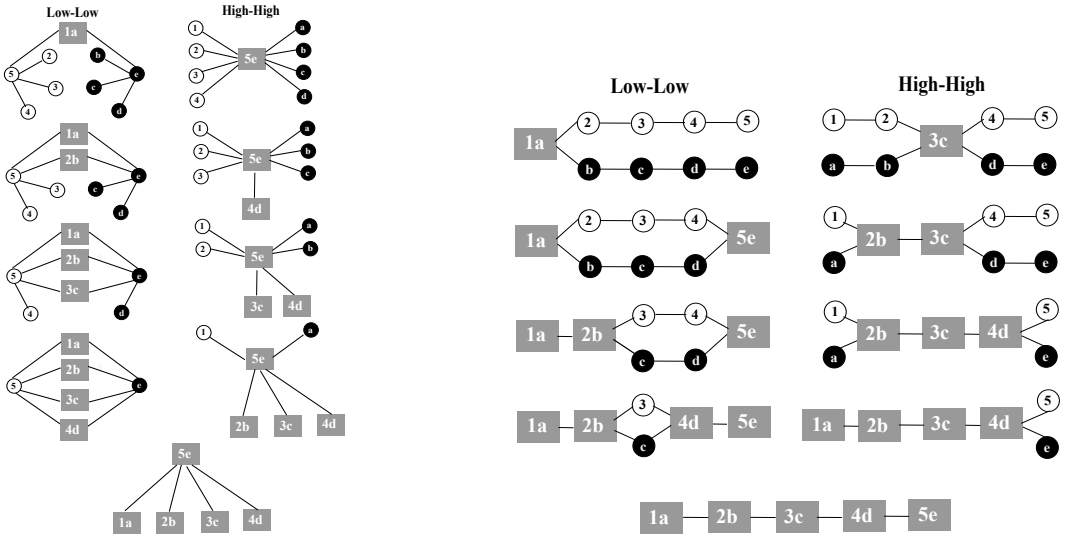


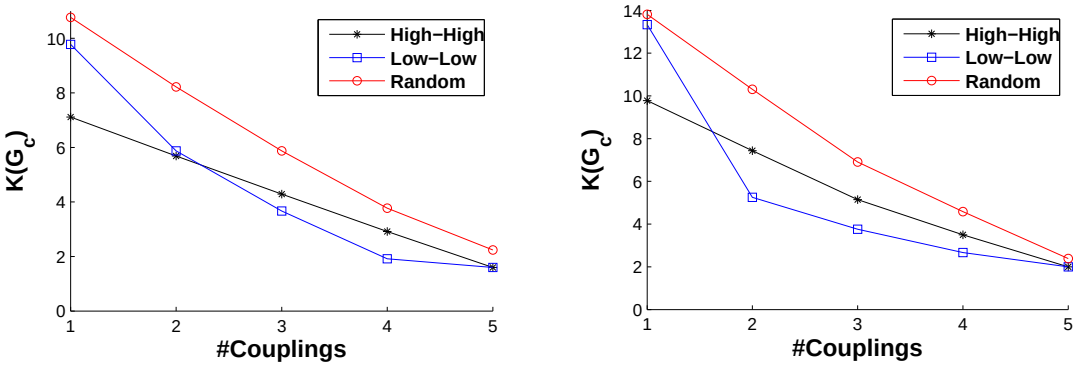
Figure 5.1: Coupling in stars and paths ($n = 5$).

It is easy to see that for $n > 2$, the root of the star has a lower l_{ii}^{+} value than the pendants and is therefore more structurally central. Similarly, for a chain of order n, the l_{ii}^{+} for the i^{th} vertex from the end is given as:

$$l_{ii}^{+} = \frac{6i^2 - 6(n+1)i + 2n^2 + 3n + 1}{6n} \tag{5.4}$$

The form in (5.4) is parabolic in l_{ii}^{+} and i. Clearly, for a given n, the minima is attained when $i = \lceil n/2 \rceil$, i.e. at the middle node/s of the chain, and the maxima is attained when $i = 1 = n$ i.e. at the pendants. Thus, the topological centrality of nodes in a chain decreases as we move from the center of the chain towards the pendants on either side.

Next, we demonstrate the $low - low$ and $high - high$ gluing strategies for a star and a chain respectively with another star and chain of the same orders in Fig. 5.1, for increasing values of $\kappa : 1 \leq \kappa \leq n$. Observe that for $\kappa \geq 2$, the $low - low$ strategy produces multiple cycles in the inter-dependent networks, thereby providing alternate connectivities between nodes and safeguarding against eventual edge failures. It is well known that greater the number of cycles in a graph, higher the count of spanning trees which signifies better redundant connectivities between node pairs. In contrast, the $high - high$ strategy produces small loops between adjacent nodes and leaves the



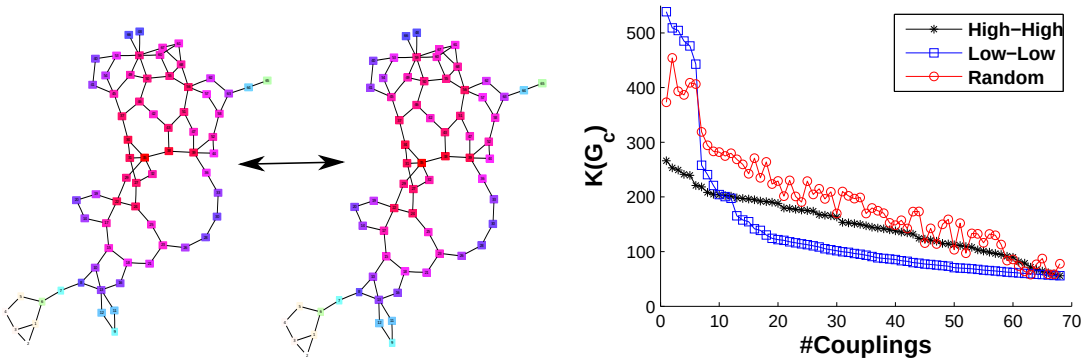
(a) Star (b) Path

Figure 5.2: $\mathcal{K}(G_c)$ for glued/coupled networks for the three different coupling functions.

overall structure rather tree like. Thus the $low - low$ strategy results in more robust glued/coupled networks for $\kappa \geq 2$ in both the star and chain topologies.This effect is well captured in the values of $\mathcal{K}(G_c)$ shown for $1 \leq \kappa \leq n$ in Fig. 5.2(a) and (b), where the $low - low$ strategy attains lowest values of $\mathcal{K}(G_c)$ for $\kappa > 2$ (note that smaller $\mathcal{K}(G_c)$ is, more robust the network is). We see that when κ is small, pairing and coupling the most structurally central nodes (in the constituent networks) produces more robust networks than other coupling strategies. However, somewhat counter-intuitively, when κ increases and becomes sufficiently large, pairing and coupling structurally least central node pairs in the two constituent networks produces more robust inter-dependent networks. This result suggests that distributing interdependencies amongst (geographically) disparate nodes may result in more robust inter-dependent networks. We explore this further with the help of a real world power-grid network in the next section.

5.3.2 The Italian Power Grid Network Example

The Italian power grid network (cf. Fig. 5.3) is a network of order $n = 68$ and $m = 93$. The nodes in fig. 5.3(a) have been colored by their structural centralities. Notice how the topological centrality reduces as we move towards the periphery of the network. We now glue the Italian power grid network with an exact copy of itself, which represents a communication network (cf. [75]) using various coupling functions $\mathcal{C}$ for increasing values of κ. Once again, (cf. Fig. 5.3(b)) we obtain the same surprising (and somewhat



(a) IPGN Topology (b) $\mathcal{K}(G_c)$ across coupling functions

Figure 5.3: The Italian power grid network coupled with itself: *Red* $\rightarrow$ *Turquoise* decreasing topological centrality.

counter-intuitive) results that only for small κ, pairing and coupling the most structurally central nodes (in the constituent networks) produce more robust (i.e., smaller $\mathcal{K}(G_c)$) networks than other coupling strategies; for $\kappa > 10$, the $low - low$ strategy produces more robust inter-dependent network than other strategies. We note that as κ increases, the $low - low$ strategy glues peripheral nodes thereby creating longer cycles in the network as compared to the other two strategies. Such longer cycles safeguard network wide connectivities against random edge failures, thus resulting in more robust inter-dependent network structures. We also observe similar results for the western states power grid network in the US.

5.4 Summary

In this chapter, we presented a theoretical framework to assess structural properties of inter-dependent networks, based on the topological properties of the Moore-Penrose pseudo-inverse of the graph Laplacians. We demonstrated that the topological centrality of nodes can be used to select nodes for gluing the two networks together and the robustness of the inter-dependent network can be measured in terms of its Kirchhoff index, given by the trace of the $\mathbf{L}^+$ of the inter-dependent network. With the help of example tree structures and the Italian power grid network, we presented comparative results for three gluing strategies based on the structural centralities of the nodes. Of the

three strategies, the $low - low$ strategy eventually wins out, producing the most robust coupled structures (inter-dependent networks). Our results suggest that by *diffusing and distributing inter-dependencies among a large number of (geographically dispersed) node pairs in the two constituent networks produce more robust inter-dependent networks.*

Chapter 6

Mapping Cellular Data Service Network Infrastructure via Geo-intent Inference

With wide adoption of smart phones and other mobile devices, *cellular data* traffic has grown tremendously in the past few years. This growth will be further precipitated by the increasing popularity of newer generations of smart-phones and mobile devices such as iPhones, Android phones and iPads. As in the case of wireline services, cellular data traffic will likely surpass the voice traffic in the not-so-distant future. Despite this tremendous growth, there have been relatively few studies on the (*operational*) *cellular data service network* (CDSN) infrastructure. Apart from various articles, papers and documentations on the architectural design and component engineering (e.g. 3G network standards), we know little, for example, about the topology and geographical distribution of IP network elements over the cellular network substrate such as basestations. The challenges in conducting *measurement-based* mapping of operational cellular data service networks (CDSNs) can be attributed, in part, to the fact that these networks are generally "closed", unlike most of the traditional Internet infrastructure. In other words, active probing (e.g. traceroute) from outside typically elicits no response from the internal network elements of a CDSN. Conducting active probing from mobile devices in general does not help much either, as IP addresses assigned to the end

users' devices are often private IP addresses [78]: the only IP addresses visible to the outside world are the IP addresses of exit routers of the CDSNs. With the rapid growth in cellular data traffic, gaining a better understanding of the CDSN infrastructure — especially the geo-spatial relationships of the IP network overlaid on top of the cellular (basestation) network substrate — is imperative. Such understanding can not only provide insights into the evolution and expansion of existing (and future) CDSNs, but also help guide the development and deployment of innovative location-aware services and applications that cater to mobile users (see more discussion on this in a latter part of this section).

In this chapter we propose and explore a novel approach to *map the CDSN infrastructure via (explicit) user geo-intent*. By *geo-intent*, we mean (explicit) geo-location information specified by users while submitting queries to certain services (e.g. weather or map services), in which they explicitly seek information regarding a specific location. Such geo-intent may be associated with the target of a user query, or the source (i.e. the user's own location). The basic intuition behind our approach is two-fold: i) mobile users often explicitly express their geo-intent when performing certain location-specific queries; and ii) their explicit geo-intent is often *local*, namely related to a location in close vicinity of their current location, e.g. a restaurant nearby or the local weather. Such queries will occur more frequently as more users adopt GPS-enabled smart phone and utilize location-based services or apps on their mobile devices. By correlating the user geo-intent expressed in location-specific queries with information regarding the CDSN infrastructure, e.g. the basestation a mobile device is currently associated with or the (first-hop) IP gateway address (such information may be obtained from mobile devices[1]), we can geo-map the CDSN infrastructure.

To investigate whether — and to what extent — our proposed approach can help geo-map the CDSN infrastructure, we employ two sources of data collected at a link inside the (wired) backbone IP network of a CDSN. The first data source comprises of the RADIUS/RADA packet data sessions which contain the basestation id's (BSIDs) and *anonymized* user id's; the second data source is collections of application sessions which contain URLs extracted from HTTP headers and (anonymized) user id's. Two

[1] For example, some smart phone mobile operating systems, e.g. Window Mobile OS, provide certain APIs via which the BSID of the basestation a mobile device is associated with, the gateway IP address as well as the IP address assigned to the mobile device can be obtained.

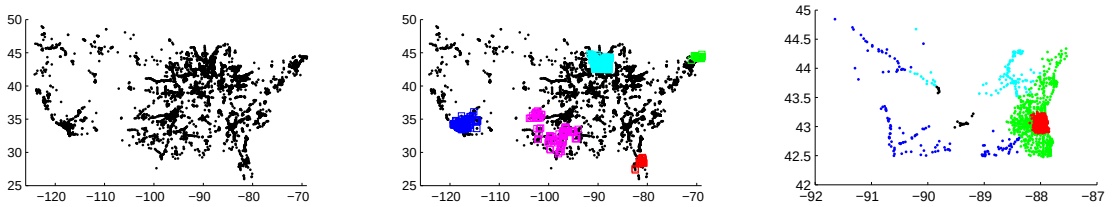
datasets (containing data from both sources), collected roughly ten months apart, are used for our study. For a subset of BSIDs, we also have the *ground-truth* GPS locations.

6.1 Related Work

Much of the existing work on localization in cellular networks has focused primarily on geo-locating mobile users or devices via signal strength based methods (e.g. triangulation) using known locations of cell towers (basestations). For a very recent study on this topic and related work, see [79] and the references therein. In contrast, we attempt to address the problem the other way around, namely, utilizing user geo-intent to map the CDSN infrastructure. The notion of user geo-intent has been proposed and studied recently in a different context, *web search*, with the goal to return search results that are more relevant to user queries. For instance, in [80], the authors analyze search queries from users, and classify them into explicit geo-intent and non-geo-intent queries.

In [81], the authors go one step further to extract (implicit) geographical information that can plausibly identify users' locations. Our work adopts a similar notion of (explicit) geo-intent and applies it to geo-map the CDSN infrastructure.

Trestian et al [82] correlate user-location (at the granularity of basestations) and application interests over time. In their analysis of user mobility patterns, they find that many users tend to move around one or a few location "hot-spots" (e.g. residence, office, or a coffee-shop). This finding indicates that a majority of users' geo-intent is likely local to their locations. The study in [78] cited earlier collects the IP addresses assigned to mobile devices as well as the IP addresses (likely those of exit routers or NAS gateways) which appear as source IP addresses in the queries sent to a web server under the authors' control, and uses them to locate mobile devices by geo-localizing the IP addresses.; only to find that the geo-mapping results using these IP addresses are very coarse-grained and often unreliable. Our study shows that the core IP network is "sparsely" distributed over the dense and geographically dispersed (see fig. 6.1(a)) cellular network substrate, thus providing a plausible explanation for their findings.



(a) US mainland (b) 5 sample SIDs (c) 5 NIDs within an SID

Figure 6.1: Illustration of geo-physical clustering of BSID's at SID/SID-NID level (Ground-truth set).

6.2 Preliminaries and Datasets

6.2.1 CDSN Infrastructure

In the traditional layered network architecture terms, a typical (3G) cellular data service network (CDSN) infrastructure consists of a (layer-1/layer-2) cellular network substrate and an IP data core network overlaid on top. The cellular network substrate comprises of a large number of basestations and radio network controllers (RNCs) geographically dispersed across the entire coverage of a cellular service provider (CSP). Each basestation is uniquely identified by its *Basestation Identifier* (BSID), which contains three parts: the System Identifier (SID), Network Identifier (NID), and Cell Identifier (CID). The BSID namespace is hierarchical and has geo-physical significance. An SID spans a large geographical region (e.g. one or more states in the US), and is composed of multiple NIDs, each representing a smaller geo-physical area. An NID, in turn, consists of many basestations, each covering a cell which is uniquely identified by a CID. Fig. 6.1(b) and (c) respectively illustrate the geo-physical clustering of five sample SIDs (represented by different shaded clusters), and five NIDs within a single SID.

SIDs are allocated to CSPs by the International Forum on ANSI-41 Standards Technology (IFAST) based on territories. A database for SIDs, publicly available on the Internet [83], provides ownership and geo-location (coarse-grained) details. A typical record in this database has five attributes: a decimal value representing the SID, the city associated with the SID (usually the name of the most populous city), the state in which the city lies, name of the CSP to whom the SID has been allocated, and the operational frequency band. Though coarse-grained, the database serves as a good cross-reference

Dataset (time)	Users	Duration	#Pkt.&App. Sess.
I (Oct 2008)	2 M	7 days	24 M & 110 M
II (Jul 2009)	1.7 M	1 day	13 M & 147 M

Table 6.1: User and traffic volume statistics.

in our analysis.

The IP network of a CDSN typically consists of IP gateways (usually referred to as *network access servers* or NAS gateways) through which data from/to mobile devices enters/leaves the IP network, (IP) home agents (for user registration and mobile IP routing), and other standard network elements such as routers, DHCP servers, DNS servers, and so forth. The IP network also includes a number of RADIUS/RADA [2] servers for authenticating users, and for logging user data access activities for billing and accounting purposes.

6.2.2 Datasets

Two datasets are used in our study, which are collected at a link *inside* the core IP network of a large North American cellular 3G service provider. The first dataset (henceforth referred to as *Dataset I*) was collected during a week-long period in October, 2008, and the second dataset (*Dataset II*) was collected over a single day in July, 2009. Table 6.1 summarizes overall statistics regarding Datasets I and II. Both datasets are *anonymized* packet traces. Each dataset consists of two sources of data: RADIUS/RADA *packet data sessions*, and *application sessions*. The RADIUS/RADA packet data sessions contain records of user activities such as the beginning and end times of a user's data session, the (anonymized) user id, the basestations (BSIDs) the user's mobile device is associated with during the data session etc. The application sessions records are the HTTP headers of users' Internet activities. We correlate the records from the two data-sources on the basis of the anonymized IP address in an HTTP application session, and match the HTTP timestamp such that it is between two consecutive RADA START and STOP messages, in the RADIUS/RADA packet data

[2] RADIUS stands for the Remote Authentication Dial In User Service protocol [84, 85], and RADA stands for the Radius Authenticated Device Access protocol. Both are used to provide centralized *Authentication, Authorization, and Accounting* (AAA) management.

Hostname	Geo-physical identifiers in URL	# of URLs
pv3.wirelessaccuweather.com	zip=54940&city=Fremont&state=Wi	510,170
mapserver.weather.com	lat=43.45&long=–88.63	273,061
maps.google.com	q=starbucks&near=Oconomowoc	9,631
addshuffle.com	zip=53946&cntry=US	6,434
geo.yahoo.com	lat=42.97&lon=–88.09	3,519

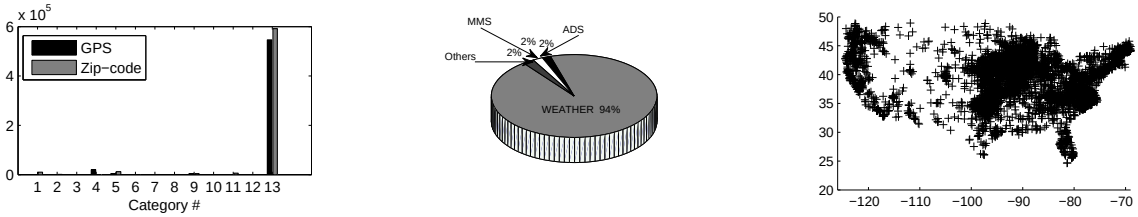
Table 6.2: Web services and sample URLs with geo-physical identifiers in Dataset I.

sessions. The URLs accessed in HTTP application sessions are extracted for identifying geo-intent queries. The BSIDs and (anonymized) user ids are extracted from the RADIUS/RADA packet data sessions. We primarily exploit the HTTP URLs, BSIDs and (anonymized) user ids, for geo-mapping the CDSN infrastructure. To verify and validate our geo-intent based mapping approach, we also utilize a collection of basestations for which we have the *ground-truth* GPS locations. Recall from fig. 6.1(a), the basestations in our ground-truth set are widely distributed across the US mainland and provide an extensive and representative set for verifying and validating the results obtained in our study.

6.3 Explicit Geo-Intent of Users

This work explores whether we can exploit "explicit geo-intent" of mobile users to learn the CDSN infrastructure, i.e. the physical locations of basestations and the IP data network elements. We define *explicit geo-intent* as location information contained in queries submitted by users to certain services (e.g. weather or map services) in which they seek information regarding a specific location. Such geo-intent in user queries may either be associated with the current (source) location of a user (e.g. *locate-me* type of features) or her target location of user (e.g. weather lookups for a region of interest).

One of the greatest challenges faced in this approach is that the geo-intent expressed in a user's query is encoded in a format meaningful for specific services and therefore varies from one service to the other. To address this issue, careful service-specific analysis is required to extract relevant explicit geo-intent from user queries. In §6.3.1, we describe our heuristics for doing this. Next, in §6.3.2, we focus on the most dominant



(a) URL count (b) Combined count (%) (c) Zip-code geo-intent

Figure 6.2: Dominance and geo-physical expanse of weather (category 13) related queries in Dataset I.

type, namely *zip-codes contained in weather-related queries*, which are primarily associated with the target locations of users' geo-intent. Lastly, in §6.3.3, we discuss the relevance of GPS-like information observed in our datasets and identify the cases when it is relevant and useful.

6.3.1 Extracting Explicit Geo-intent

We employ a set of heuristics to identify and extract geo-intent from the HTTP URLs in our datasets. Our objective is to find a set of services seen in our URL trace with a geo-intent format that can be automatically extracted, giving us a mapping between URL and the geo-intent expressed in that URL. Through a manual process of identifying a set of location-specific keywords, such as street or state names, zip-codes, and "GPS-like" coordinates [3] , we create a set of rules to perform such extraction. The output of this step are rules for extracting the embedded geo-physical identifiers in the URL string for each hostname (e.g. `www.weather.com` or `www.mapquest.com`). Table 6.2 shows some examples of services and the associated geo-identifier formats. Through such rules (heuristics), we identify over a million URLs with geo-identifiers from Dataset I and half a million from Dataset II.

We further analyze the geo-identifier information contained in the extracted URLs to understand the variety of geo-identifiers they contain. We find that zip-codes and "GPS" coordinates dominate the set of URLs with geo-identifiers (in $\approx 99\%$ of the URLs we are able to parse). Henceforth, we focus only on such URLs. Furthermore, to

[3] In this chapter we refer to any pair of latitude-longitude coordinates as "GPS" coordinates, although in fact many of these may not be directly provided by the (satellite) global position system (GPS) service. See §6.3.3 for a detailed discussion.

understand better the services associated with such URLs (with zip-code and GPS-like geo-identifiers), we classify each service (based on hostname) into 13 different categories: *1-Ads, 2-books, 3-dating, 4-maps, 5-MMS, 6-music, 7-news, 8-photo, 9-search, 10-toolbars, 11-trading, 12-video and 13-weather*. Fig. 6.2(a) shows the number of URLs in each application category separately for two types of geo-identifiers: zip-codes and GPS-like coordinates. We see that weather services constitute the most dominant category accounting for about 94% URLs with either a zip-code or a pair of GPS-like coordinates (see Fig. 6.2(b)).

In the following subsections we analyze the geo-identifiers contained in the URLs in weather category to determine whether or not these URLs indeed reveal the user geo-intent.

6.3.2 Zip-codes in Weather Queries

Weather queries are obvious candidates for finding zip-code information due to the nature of online weather services. Most phones feature a weather application allowing users to enter the zip-codes for one or more locations of interest. Quite often, these queried locations represent the user's home or place of work. Therefore, the zip-codes in weather queries provide a good, though not precise, indication of the querying user's location. We later evaluate the usefulness and accuracy of such zip-code information in our datasets for the purposes of geo-mapping the CDSN infrastructure.

In this work, we convert the zip-codes contained in geo-intent queries in terms of a GPS-like coordinate as follows. The US census bureau [86] provides GPS-like coordinates which delineate the approximate boundaries of the zip-code tabulation area (ZCTA)[4] encompassing all the zip-codes in the US. Using such boundary coordinates for a given zip-code, we compute the *centroid* (a pair of GPS-like coordinates). In the remainder of this chapter the term zip-code will be used exclusively to mean the corresponding centroid location calculated as described here. Fig. 6.2(c) shows the geographical distribution of the zip-codes (centroids) contained in all the weather queries in our datasets. We see that the set of zip-codes in the explicit geo-intent of users pervades nearly all parts of the US mainland.

[4] Some ZCTAs may span several zip-codes in less populous regions. As our results later show, for our purpose the ZCTAs provide sufficient accuracy.

Type	Geo-physical identifiers in URL	Zoom
Req.	zip=53108&city=Caledonia&state=Wi&country_code=US	-
Resp.	mzip=53108&mcity=Caledonia&mstate=Wi &mx=–87.93&my=42.82	2
Resp.	mzip=53108&mcity=Caledonia&mstate=Wi & mx=–99.76 &my=42.82	1

Table 6.3: GPS coordinates in HTTP responses from web-host.

6.3.3 GPS-like Coordinates in Weather and Other Queries vs. True Geo-Intent

Next, we investigate the URLs containing GPS-like (latitude-longitude) coordinates. As shown in Fig. 6.2, the weather category also contains an (almost) equal number of URLs with GPS-like, latitude-longitude coordinates. A majority of these GPS-like coordinates appear in the HTTP responses and not the HTTP requests. Further inspection reveals that these coordinates do not directly reflect the geo-intent of users, and show significant variance (see table 6.3). However we do observe a few services, e.g. *GPSToday* hosted by `www.geoterrestrial.com`, where the GPS coordinates contained in user queries submitted to these services do reflect *true* geo-intent[5] . Unfortunately, it represents a very small fraction of queries in our datasets. Hereafter we refer to this small set of GPS coordinates as the *GPS geo-intent* dataset.

For the remainder of the chapter, we focus on zip-code information, except where noted otherwise. We remark that our geo-mapping methodology presented later is also able to incorporate GPS coordinates and has the potential to provided greater precision as more devices and services, which use the capabilities of GPS-enabled smart-phones, are deployed.

[5] For example, careful analysis of the service provider, `www.geoterrestrial.com` and the queries submitted to this service reveals that running on GPS-enabled mobile devices, this service is associated with an application called *GeoToday* which provides topographical (e.g. altitude) and weather related information at the user' current location. Hence the GPS coordinates contained in user queries to this service reflect *explicit user geo-intent*, in this case, the source (user) location of the geo-intent. Similar analysis to a couple of other services also confirm that the GPS coordinates contained in the user queries also reflect true geo-intent.

6.4 From User Geo-Intent to Geo-Locations in the CDSN

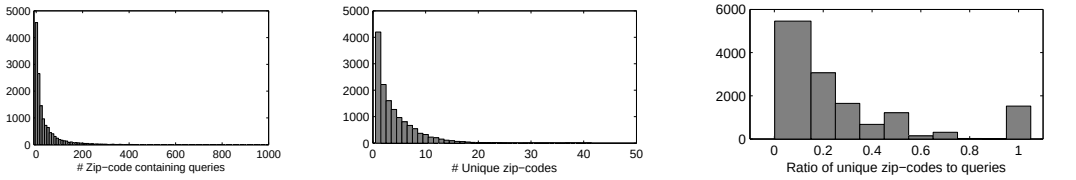
In this section we correlate the zip-codes extracted from the weather queries with the *basestation* infrastructure of the CDSN to investigate whether, and to what extent, users' geo-intent can help geo-map the CDSN infrastructure. For this purpose, we use a subset of basestations for which we have known GPS locations (the *ground-truth*). In order to make our analysis of (zip-code) geo-intent agnostic to diurnal and weekly variations (weekdays vs weekends), we use Dataset I exclusively in §6.4.1 through §6.4.3.

6.4.1 Spread of Geo-intent in the Basestation Infrastructure

To associate the geo-intent expressed in users' queries with the basestation infrastructure of the CDSN, we first need to identify and extract relevant basestation information (BSID associated with a user at the time of query). Henceforth, we say that a basestation B, *sees* a zip-code Z if at least one user queries for weather information (or any information in general) for zip-code Z while communicating with basestation B.

Table 6.4 shows some of the statistics obtained using the process of correlating (zip-code) geo-intent queries with their associated basestations. We see that although the number of users expressing their explicit geo-intent is a small fraction of the overall user-base (less than 2%), the number of basestations that see at least one zip-code query is significantly large ($\approx$ 23%). Moreover, the set formed by such basestations covers a representative fraction of SID-NID pairs, and consequently SIDs, in the network. Therefore, explicit geo-intent is pervasive not only in terms of geographic coverage (as seen in §6.3.2) but also in the CDSN infrastructure. This is particularly important because if geo-intent of users indeed captures their geo-location, we can possibly geo-map a significant fraction of the basestation infrastructure across wide geographies.

Fig. 6.3(a) and (b), respectively show the distributions for the number of user queries containing zip-codes *per basestation* and *unique* zip-codes per basestation for the URLs in Dataset I. We observe that the 50^{th} and 75^{th} percentiles for the number of queries (containing zip-codes) seen per basestation are 16 and 50 respectively. While a majority of basestations see a sizable number of geo-intent queries, the 50^{th} and 75^{th} percentiles for the number of *unique* zip-codes seen per basestation are 3 and 6



(a) # zip-code queries (b) Unique zip-codes (c) Unique zip-codes/queries

Figure 6.3: Spread of geo-intent per basestation, Y-axis: # Basestations.

# of	Users	BSID	SID-NID	SID
Overall	2 M	62, 534	506	237
Geo-Int. (Zip)	29 K	14, 224	356	219

Table 6.4: Infrastructure coverage of zip-code geo-intent in Dataset I.

respectively. Fig. 6.3(c) shows the distribution of the ratio of unique zip-codes over the total number of zip-code geo-intent queries per basestation. Once again we observe that the respective 50^{th} and 75^{th} percentiles for the ratio are 0.2 and 0.4 respectively. This result clearly indicates that when there are a number of zip-code containing weather queries seen at a basestation, many of them are associated with only a small number of zip-codes. This observation has important implications in the process of geo-mapping of the basestation infrastructure, as will be explored in the next subsection.

Further analysis shows that the spread of zip-code containing geo-intent queries across the basestation infrastructure is somewhat uneven, where basestations within urban metropolitan areas generally account for a greater fraction of geo-intent queries than those in rural areas. This can partly be explained by the difference in population densities as well as the percentages of "smart" phones and data service plans adopted by users in these areas. Due to space limitation, we do not provide detailed results (area-wise statistics) here.

6.4.2 From Geo-intent to Geo-location

With about 23% of the basestations in our dataset seeing zip-code containing weather queries, can we use the explicit geo-intent information contained therein to geo-localize the basestations in question? We note that the zip-codes contained in users' weather

queries are most likely associated with the *target* regions of users' interest; on the other hand, the basestations seeing the queries are associated with the location of users at the moment of querying. Hence the extent and accuracy of using user (explicit) geo-intent to help geo-localize the basestations will depend on how far the target location of users' interest (as specified by the zip-codes) are from the basestations where the queries are issued (the source location of users). To investigate this question, we utilize the subset of basestations for which we have the ground-truth (i.e. their GPS co-ordinates). Among the basestations with ground-truth GPS locations, we find that roughly 20% (in a similar percentage as zip-code seeing basestations to the entire basestation set) also see zip-code queries; moreover, they span 105 SID-NID pairs across 81 SIDs. In the following we will refer to the set of such basestations ($\approx 2,400$ in all), with both the ground-truth GPS locations and associated zip-code queries, as the *ground-truth-location-&-zip-code* BSID dataset.

To examine the relationship between the locations of the basestations and users' geo-intent (the zip-codes associated with the basestations), we compute the geo-physical distances between basestations and the zip-codes as follows. Given a basestation B with *known* GPS location denoted by $L_B = (lat_B, long_B)$, let $Z_B = \{Z_1, Z_2, ..., Z_k\}$ be the set of zip-codes queried by the users associated with B. Recall that we identify each zip-code Z_i with a pair of GPS-co-ordinates for its centroid in the form of $C_{Z_i} = (lat_i, long_i)$. We denote the distance between the basestation B and the zip-code Z_i by $\delta_i^B = dist(L_B, C_{Z_i})$, where the distance is computed over the surface of the earth using the (angular) latitude and longitude co-ordinates and is then mapped to the metric distance in kilometers (km)[6] . In particular, we define $\delta_{min}^B = min_{1 \leq i \leq k} \delta_i^B$ and $\delta_{max}^B = max_{1 \leq i \leq k} \delta_i^B$. Further, for basestations that are associated with multiple zip-codes ($k \geq 3$), we also compute the median of δ_i^B's, denoted by δ_{med}^B. In addition, we compute the distance between each basestation and the most frequently[7] queried zip-code associated with it, and denote this distance as δ_*^B. The distributions for δ_{min}^B and δ_{max}^B are shown in Fig. 6.4 using the *ground-truth-location-&-zip-code* BSID dataset. For 50% of the basestations in the *ground-truth-location-&-zip-code* set, the distance between basestation B and the closest queried zip-code (δ_{min}^B) is within 3.5 km

[6] We use the avg. value computed through the haversine and Vincenty formulas and assume a mean radius of 6,371 km for the earth.

[7] If there are two or more such zip-codes, we randomly pick one of them.

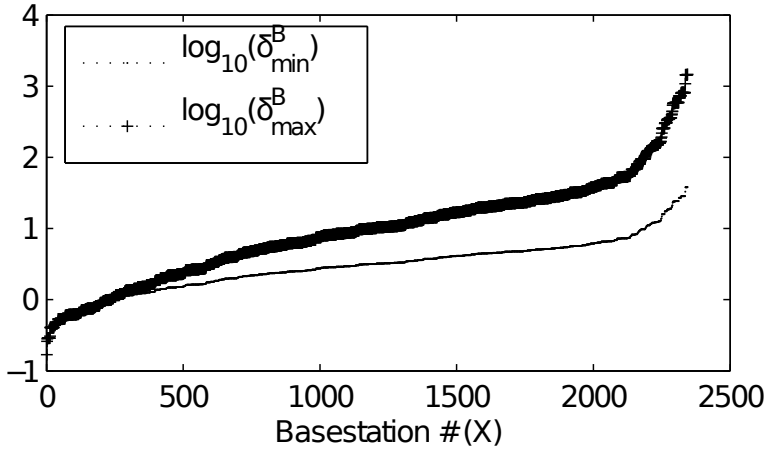
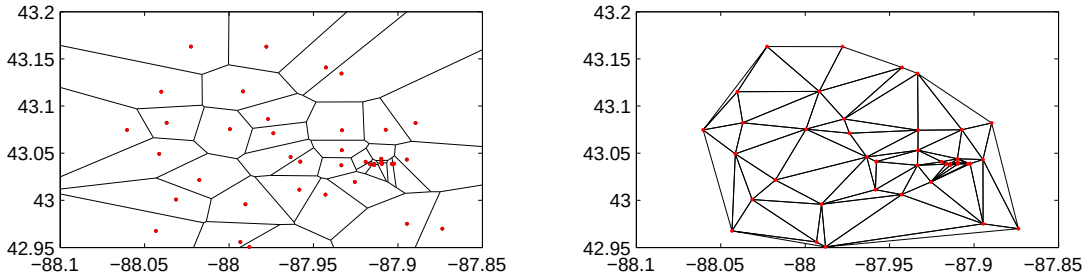


Figure 6.4: Distribution of δ^B_{min} and δ^B_{max} for basestations in *ground-truth-location-&-zip-code* set.

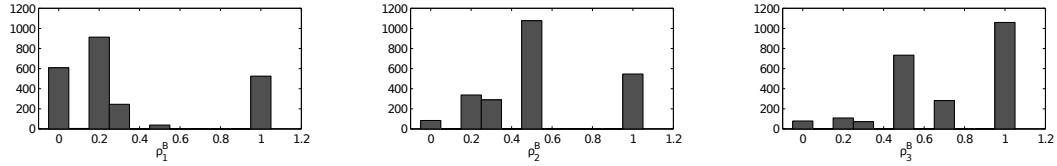
range, and for 75% of them it is less within 5.5 km range. In particular, about 25% of basestations lie within 1 km range of the nearest zip-code they see. This promises possibly high accuracy of geo-localizing a basestation on the basis of geo-intent in some cases. In contrast, δ^B_{max} (distance between basestation B and the farthest zip-code it sees), is within 12.5 km for 50% of basestations and within 20 km for 75% of them; much larger than corresponding δ^B_{min}. Similarly, δ^B_{med} is within 3.8 km range for 50% and within 6.9 km range for 75% of basestations while δ^B_* is within 3.9 km range for 50% and within 6.1 km range for 75% of the basestations. In short, we see that while the distance between the true location of a basestation and the farthest queried zip-code seen by it can be in the range of 10's km or more, that between the basestation and the closest queried zip-code is usually within 10 km (and often within 5 km or less). Moreover, when multiple zip-codes are queried by the users, more of them tend to be in the vicinity (around 7 km or less) of the basestation. The frequently queried zip-codes are often also the closest zip-code or a zip-code not much farther away.

However, using the absolute distance (in km) to correlate geo-intent (zip-codes) and geo-location (of basestations) does not paint the full picture, as zip-code areas have varying sizes, depending upon population density and other factors. For instance, large metropolitan cities tend to have more zip-code areas with smaller geo-physical sizes, while rural areas have far fewer, and larger, zip-code areas. To better understand the



(a) Voronoi partitioning (b) Delaunay (sub) graph

Figure 6.5: Milwaukee city zip-code centroids (red-dots).



(a) 1 hop (b) 2 hops (c) 3 hops

Figure 6.6: No. of basestations (Y-axis) with X (fraction) of assoc. zip-codes at most l hops away.

relationship between the location of basestation and the zip-codes their users query, we use the centroid of each zip-code and perform a *Voronoi partition* of the entire US mainland [8] . In other words, the US mainland is represented as a contiguous collection of Voronoi cells, where each zip-code centroid is exclusively contained in a single Voronoi cell. From the Voronoi diagram representation of the US mainland, we construct the corresponding *Delaunay graph*, in which the vertices are the zip-codes, and an edge is introduced between two zip-codes if and only if they are contained in neighboring Voronoi cells. As an example, Fig. 6.5 shows a portion of the Voronoi diagram (for the south-east part of Wisconsin around the Milwaukee metropolitan area) and the corresponding Delaunay (sub) graph.

We now introduce a new metric to measure the distances between basestations and zip-codes, in terms of the Voronoi diagram and the Delaunay graph introduced above, to

[8] Instead of partitioning the US mainland in terms of the ZCTA boundaries using data from the US census site *http://www.census.gov*, we use the Voronoi partition for ease of analysis and computation.

better gauge the relationship between user geo-intent and the geo-location of the associated basestation. Given a basestation B with known GPS location $L_B = (lat_B, long_B)$, we first determine the Voronoi cell in which it lies. We associate basestation B with the zip-code contained in the same Voronoi cell, say $\widehat{Z}_B$, and refer to this zip-code as the *home* zip-code of B. Now for each zip-code Z_i seen at basestation B, we compute the (hop-count) distance between B and Z_i as the shortest path distance (in terms of hop-counts) between $\widehat{Z}_B$ and Z_i in the Delaunay graph. We denote this hop-count distance by h_i^B.

In order to understand the distribution of hop-count distances (h_i^B) between a basestation B and the zip-codes $Z_i \in Z_B$, we define a multi-hop ($l = 1, 2, 3, ...$) neighborhood relationship between the nodes of the Delaunay graph shown below:

$$Neighbor_l(\widehat{Z}_B, Z_i) = 1 \text{ if } h_i^B \leq l \tag{6.1}$$

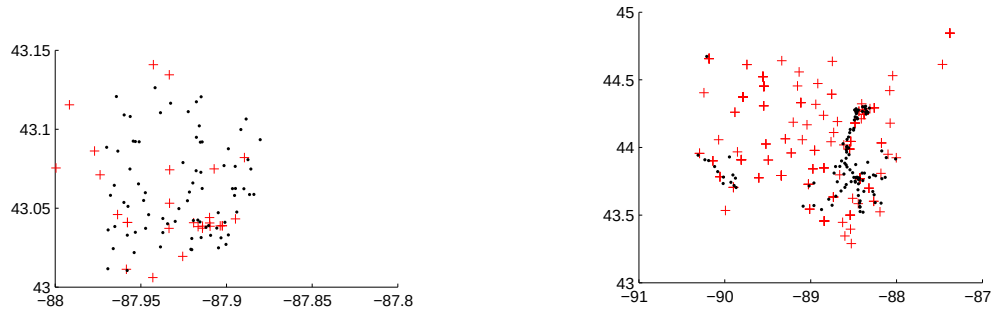
$$= 0 \text{ otherwise.} \tag{6.2}$$

Then, the following ratio:

$$\rho_l^B = \frac{\sum_{i=1}^{k} Neighbor_l(\widehat{Z}_B, Z_i)}{k} \tag{6.3}$$

where k is the number of zip-codes seen at B, provides similar insight into the hop-count distance between the home zip-code of B and the zip-codes seen at it, as the δ^B functions defined for distances over the surface of the earth. For example, ρ_1^B tells us the fraction of zip-codes seen at B that are at most 1 hop away from B in the Delaunay graph. Fig. 6.6 shows the distribution of ρ_l's for $l = 1, 2, 3$. For example, in fig. 6.6, for a given l, the value on the Y-axis (length of the bar) corresponding to bin $X = 0.5$ on the X-axis represents the number of basestations which have 50% of the associated zip-codes at most l hops away. Notice the consistent increase in the Y-values corresponding to larger X-values as we go from $l = 1$ to $l = 3$. In fact, over 90% of the basestations have more than 50% of their associated zip-codes within $l = 3$ hops. We, therefore, (in conjunction with the evidence from similar results for absolute distances) conclude that for a large majority of basestations, a significant percentage of zip-codes queried are in and around the geo-physical neighborhood of their home zip-codes.

6.4.3 Geo-intent, Geo-location and User Behavior



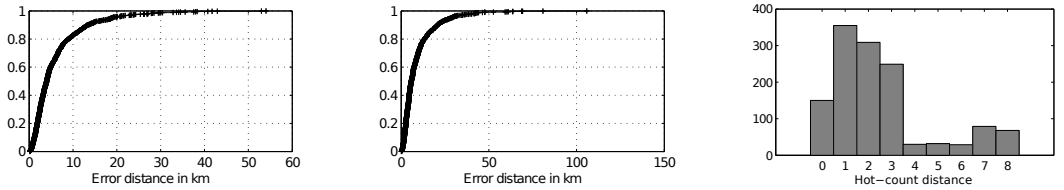
(a) Metropolitan area (b) Highway area

Figure 6.7: Most frequently queried zip-codes (red "+") seen at (sub) set of basestations (black "·").

In this section we analyze user behavior, particularly in terms of user mobility, to gain further insight into the observations obtained in the preceding sections. This analysis provides a plausible explanation as to why zip-codes in weather queries – despite being associated with the *target* of geo-intent – can help geo-map the basestation infrastructure to a large extent with a reasonable accuracy, namely, within the range of 3 km to 10 km or 1-3 neighboring zip-code areas for a large majority of basestations.

To study the user mobility behavior, we examine the number of basestations accessed by those users who express their explicit geo-intent (i.e. issuing a weather query containing a zip-code) at least once during the observation period. Using Dataset I which spans a week long period, we observe that almost 50% of the users are associated with exactly one basestation for the entire duration, while 75% of the users communicate with 4 or fewer basestations[9] . Among those who are associated with multiple (but ≤ 4) basestations, we find from the ground-truth set (when available) that such basestations are generally not far apart. This is particularly true for users within a metropolitan area. As an illustrative example, Fig. 6.7(a) shows a metropolitan area in the Midwest, where each black "·" indicates the location of a basestation within the metropolitan area, and each red "+" indicates the centroid of the most frequently queried zip-code by users associated with this (sub) set of basestations. We see that for a number of

[9] For comparison, we also perform similar analysis for those users who do *not* issue any geo-intent queries, and obtain similar results.



(a) Dist. Err. (MV) (b) Dist. Err. (DNC) (c) Hop-count Err. (DNC)

Figure 6.8: Error incurred in direct geo-mapping compared to the ground-truth set.

basestations ($\approx$ 80), the most frequently queried zip-codes are very few ($\approx$ 20) and confined to a small geo-physical region. In contrast, we find that basestations located in places with high user mobility, e.g. along major interstate highways, frequently see relatively greater numbers of zip-code queries from different users, but such zip-codes seem to be far more geographically dispersed (see Fig. 6.7(b) for an example).

Besides user mobility patterns, we also examine user query patterns. We find that among the users who express their explicit geo-intent at least once during the observation period, 90% of them query just one zip-code, while 96% query two or fewer zip-codes. In summary, our analysis shows that a majority of users tend to have limited mobility (when they access mobile data services) with respect to the basestation infrastructure (especially in metropolitan/urban areas), and their data access patterns are fairly stable with respect to the data access points (basestations). As a consequence of such user behavior and mobility patterns, namely, a majority of users tend to move around one or a few spots within a relatively limited radius and typically query for a default zip-code (close to their places of residence or work), their explicit geo-intent (in this case, the target location of their interest) can indeed help geo-localize the basestations they are associated with, albeit the extent and accuracy of the geo mapping hinges on the type of the geo-intent information available.

6.5 Geo-mapping the Basestation Infrastructure

Based on the analysis and observations made in the previous section, we now present some heuristics to geo-map the basestation infrastructure. In §6.5.1 we first describe two simple heuristics for geo-localize the basestations which see at least one user zip-code weather query. We then extend the heuristics to geo-map those basestations that

do *not* see any explicit geo-intent queries but share the user base with those that do, by exploiting user movement over short time intervals. The evaluation results are presented in § 6.5.2. As a proof of concept, we will also present some results obtained from similar analysis applied to the GPS co-ordinates related to a particular service (`www.geoterrestrial.com`) from Dataset II in §6.5.3.

6.5.1 Geo-Mapping Heuristics

Direct Geo-mapping via Geo-Intent

For those basestations which see at least one zip-code containing weather query, we directly geo-localize them using the user explicit geo-intent by means of the following two simple heuristics.

Given a basestation B, let $Z_B = \{Z_1, Z_2, ..., Z_k\}$ be the set of valid zip-codes queried by its users $U_B = \{U_1, U_2, ..., U_l\}$. The first heuristic, the *Majority Voting* (MV) scheme, selects the most probable location (or locations) from all possible zip-code locations (Z_i)'s as follows: Each user $U_i \in U_B$ has one simple vote. Recall that a given user U_i may query the same zip-code Z_j multiple times. In order to prevent such frequent voters from skewing the vote count, we permit a user to vote only once. Also, a given user U_i may possibly query multiple zip-codes from the set Z_B. In such cases, we split the simple vote of U_i either equally or proportionally among all the zip-codes s/he queries. For example, if U_i queries zip-code Z_1 thrice and Z_2 twice, in equal vote-splitting, both zip-codes receive 0.5 votes from U_i while in proportional vote-splitting, Z_1 receives 0.6 vote and Z_2 receives 0.4 vote from U_i. The winner of the election, i.e. the zip-code receiving most votes, is chosen as the most probable geo-location for the basestation B. When there are multiple winners (ties), all of them are chosen as probable locations (with equal probability).

The second heuristic, the *Dense Neighborhood Clustering* (DNC) scheme, uses both the frequencies of zip-codes queried as well as the neighborhood relationship among the zip-codes. Given the Delaunay graph of the US mainland, the zip-codes in Z_B induce a subgraph, denoted by $G_Z(B)$, with vertices Z_i, $1 \leq i \leq k$, and there is an edge between Z_i and Z_j if and only if the zip-code areas they represent border each other. Furthermore, we assign each node Z_i a weight w_i equal to the votes received by it during

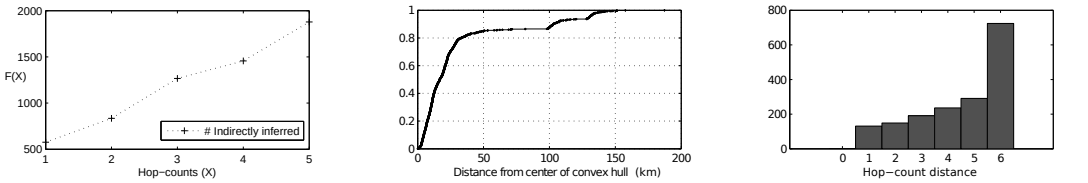
the Majority Voting scheme above. In general, the subgraph $G_Z(B)$ consists of multiple connected components, $C_1, \ldots, C_m$, where $1 \leq m \leq k$ ($k = |Z_B|$), each a probable candidate for the location of B. For each C_p, we define $w(C_p) = \sum_{Z_i \in C_p} w(Z_i)$. We select the component C_p with the largest $w(C_p)$ as the probable location (a connected zip-code neighborhood) for the basestation B. We note that in the special case where $m = k$, i.e. $G_Z(B)$ consists of k disjoint vertices, this scheme reduces to Majority Voting. A further refinement of this heuristic also filters out cases where the total weight $w(C_p)$ of the winner component is too small (below a threshold) and $G_Z(B)$ consists of mostly disconnected vertices that are spread over a large geographical area. In such cases, the heuristic simply labels the location of B as "undecided" instead.

Indirect Geo-mapping based on User Mobility

The direct geo-mapping via geo-intent helps geo-localize around 20% of the basestations in our datasets. To map other basestations, those not mapped during direct geo-mapping due to lack of geo-intent queries, we exploit user movement. To do so, we introduce the *basestation-user-mobility graph*, $\mathcal{G}_M$, where the vertices are the basestations (BSIDs) and an edge $e = (B_i, B_j)$ is introduced between two vertices B_i and B_j if at least one user[10] accesses both of them (regardless of order) within a short interval of time ΔT (say 5 minutes). Given $\mathcal{G}_M$ thus defined, let $\mathcal{B}_{mapped}$ denote the set of basestations geo-located via the two direct geo-mapping heuristics described above, and $\mathcal{B}_{unmapped}$ be the set of *unmapped* basestations. For each basestation $B \in \mathcal{B}_{unmapped}$, if it is connected to some basestation $\overline{B} \in \mathcal{B}_{mapped}$ via some paths, we define $h(B, \overline{B})$ as the shortest path distance (hop-count) from B to $\overline{B}$. Then, let $h(B, \mathcal{B}_{mapped}) = \min_{\overline{B} \in \mathcal{B}_{mapped}} h(B, \overline{B})$. Note that $h(B, \mathcal{B}_{mapped}) = \infty$ if B is not connected to any $\overline{B} \in \mathcal{B}_{mapped}$.

In our datasets, we have about 22% basestations in $\mathcal{B}_{unmapped}$ that are connected to at least one basestation in $\mathcal{B}_{mapped}$ at a 1-hop distance (i.e. $h(B, \mathcal{B}_{mapped}) = 1$). Hence, we geo-localize them first by exploiting their connectivities to the basestations in $\mathcal{B}_{mapped}$. The challenge here is to map the connectivity in $\mathcal{G}_M$ to geo-locations or zip-code neighborhoods in the Delaunay graph of US zip-codes. To control the mapping

[10] More generally, for each edge $e = (B_i, B_j)$, we count the number of users associated with both B_1 and B_2 within a short time interval ΔT, and filter out edges that have a very small common user count to prevent spurious connections due to noisy data. If a lot of users access B_i and B_j within a short interval they are likely close to each other



(a) # unmapped BSIDs (b) Dist. Err. (DNC) (c) Hop-count Err. (DNC)

Figure 6.9: Coverage and error incurred in indirect geo-mapping compared to the ground-truth set.

accuracy of a basestation $B \in \mathcal{B}_{unmapped}$, we introduce two parameters, the *hop-count threshold* d, and the *(mapped) neighborhood size* s. For any $B \in \mathcal{B}_{unmapped}$ such that $h(B, \mathcal{B}_{mapped}) \leq d$ and it is connected to at least s basestations in $\mathcal{B}_{mapped}$ that are at most d hops away from B, we geo-localize B by constructing a connected zip-code neighborhood in the Delaunay graph of zip-codes. Let $N_d(B)$ be the set of home zip-codes of the (directly mapped) basestations $\overline{B}$'s in $\mathcal{B}_{mapped}$ that are at most d-hops away from B (note that $|N_d(B)| \geq s$). Using the centroids of $\widehat{Z}_{\overline{B}}$'s, we construct a convex hull H_B, covering all $\widehat{Z}_{\overline{B}}$'s, as the most probable geo-location for B. Alternatively, we construct a connected zip-code neighborhood (subgraph), also denoted by H_B, which is formed by the zip-codes whose centroids fall within the convex hull H_B. We refer to H_B as the inferred zip-code neighborhood for basestation B.

6.5.2 Evaluation and Validation

To evaluate the efficacy of our heuristics for geo-mapping the basestation infrastructure, we use the collection of basestations with *ground-truth* GPS locations. In particular, we use the basestations in the *ground-truth-&-zip-code* set for Dataset II to evaluate the two direct geo-mapping heuristics. Then, we use the other basestations in the *ground truth* set, which do not see zip-code queries by their users, in both Dataset I and Dataset II to evaluate the indirect mapping heuristics.

Using the *ground-truth-&-zip-code* basestation dataset from Dataset II, Fig. 6.8(a) shows the distribution of geo-mapping errors, namely, the distance between the inferred location and the ground-truth location, using the *Majority Voting* heuristic. In case of multiple inferred locations (zip-codes) available for a basestation, we compute the error for each inferred location. We observe that the 50^{th} and 75^{th} percentiles are around

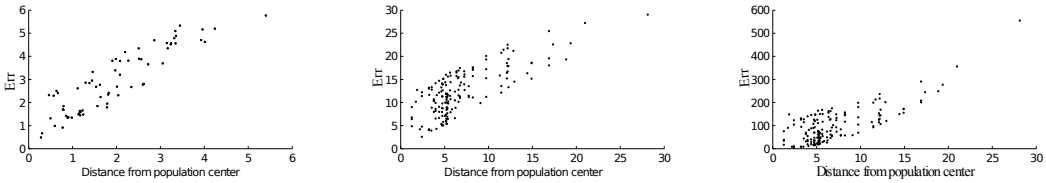
3.9 and 6.1 km respectively (Dataset II), quite similar to what we observed for δ^B_{min} in Dataset I (see § 6.4.2).

For the *Dense Neighborhood Clustering* heuristic, we measure the errors in terms of both absolute distance and hop-count. For a basestation B, let $C(B)$ be the inferred zip-code neighborhood. We compute the centroid of $C(B)$ and use the distance (ground-truth GPS location) between B and the centroid as the absolute distance error. In terms of hop-count distance error, we use the home zip-code $\widehat{Z}_B$ of the basestation B, and compute the (shortest distance) hop-count from $\widehat{Z}_B$ to $C(B)$, namely, $h(\widehat{Z}_B, C_B)$ as defined in the indirect geo-mapping heuristics. Figs. 6.8(b) and (c) respectively show the distributions of absolute and hop-count errors. We see that the errors in absolute distances are comparable to those obtained for absolute distances in Majority Voting for most basestations. This is not surprising as most of the clusters in our dataset are of small sizes (made of 4 or less zip-codes) and span relatively small geo-graphical areas, especially in urban locations. We also see that the 50^{th} and 75^{th} percentiles for the hop-count distance are 2 and 3 respectively.

Next, we evaluate the heuristic for the indirect geo-mapping of basestations in $\mathcal{B}_{unmapped}$. To do so, we fix the (mapped) neighborhood size s to 3[11] , and vary the hop-count threshold d from 1 to 5. We measure the geo-mapping errors in terms of the absolute distance (i.e. the distance from the ground-truth GPS location of B to the centroid of the inferred convex hull H_B) and hop-counts (i.e. $h(\widehat{Z}_B, H_B)$). Fig. 6.9(a) shows the percentage of additional basestations that can be indirectly geo-mapped as a function of d. Fig. 6.9(b) and (c) show the distributions of absolute distance errors and hop-count errors, respectively. We see that the errors incurred go up in terms of distances, even though hop-counts go up only by a few hops. The reason for this is that the edges in user mobility graph cover long distances in highway areas. Add to it, the long distances between the vertices of the convex hull H_B in such areas due to far apart zip-codes. Even for small values of d and s, the error incurred in such areas is high. In contrast, in urban areas, users cannot travel very long distances in short intervals of time. This means shorter edges in the mobility graph. Also, zip-codes in such areas are geo-physically close to each other. A combination of the two results in relatively

[11] In our dataset, increasing s does not significantly improve the accuracy of the geo-mapping while considerably reduces the coverage

lower error in urban areas for the indirect method both in terms of hop-counts (usually $1-3$) as well as absolute error ($5-10$ km or less). This helps us realize our objective of geo-localizing basestations in $\mathcal{B}_{unmapped}$ at city level granularities.



(a) Metropolitan area (b) Urban area (c) Highway

Figure 6.10: Relationship between population and basestation density and error in geo-mapping.

To explore the effects of population density in a region, which determines both the size of zip-codes and basestation densities, on the accuracy of geo-mapping, we conduct a case study. We select three non-overlapping regions - a metropolitan area (34 zip-codes, 235 basestations), an urban area with a relatively lower population density than a metropolitan area (20 zip-codes, 115 basestations), and a stretch of an interstate highway connecting several urban centers along south-eastern Wisconsin (6 towns, 130 basestations). In each case, we identify high population density centers (7 most populous zip-codes in the metro, 5 in the urban area and the centroids of the 6 towns in the case of the highway). Fig. 6.10 shows the (absolute distance) errors incurred in geo-mapping basestations via explicit geo-intent in all the three cases as a function of distance between the basestations and the nearest population center. We see that the error varies almost linearly with the distance from the population centers in the case of metropolitan and the urban areas (average errors smaller for the metropolitan area than the urban area). This seems to show that users usually query for information in and around themselves for some preferred target locations (e.g. downtown, residential areas) represented by the population centers. On the other hand, the errors incurred in the case of the highway are substantially high even for relatively short distances from the nearest city in the vicinity. This is possibly because people usually query information related to the regions they are coming from or, more likely, going towards, while on the highway.

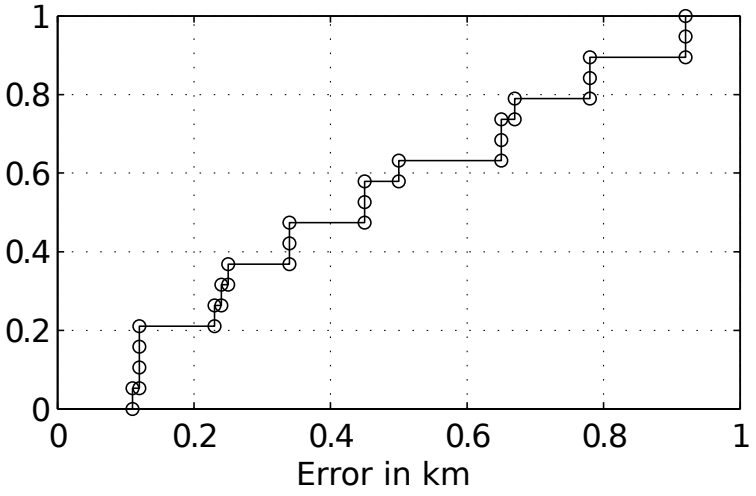
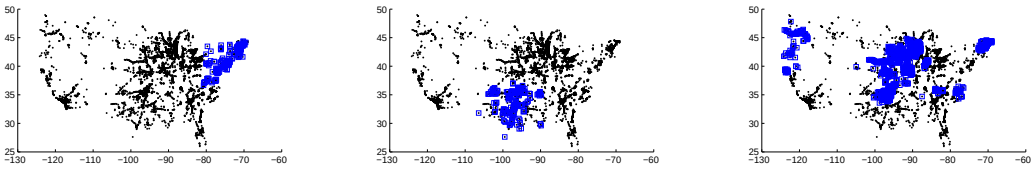


Figure 6.11: CDF of the mean errors (km) in geo-mapping using the small GPS geo-intent data.

6.5.3 Geo-mapping using GPS Geo-intent

Lastly, we use the small *GPS geo-intent* dataset discussed in section §6.3.3 to illustrate the efficacy of our approach when GPS-based geo-intent (in particular, when the GPS coordinates are associated with the source (user) locations of the geo-intent. We extend the direct geo-mapping heuristics from §6.5.1 to the case of GPS coordinates, and apply tessellation and density estimation to geo-localize basestations by computing a (small) neighborhood area (rectangular cell) as their most probable locations. Due to the space limitation, the details are omitted. Fig. 6.11 shows the mean distance (error) between the ground-truth and the inferred (centroid) locations of the two dozen basestations in the small GPS geo-intent dataset (and for which we have the ground-truth locations). We see that the overall accuracy is within 0.5 - 1 km. Hence we believe that with the increasing popularity of newer generations of GPS-enabled smart phones and location-aware services, geo-mapping based on user geo-intent will yield more accurate results than what can be obtained using zip-codes alone.



(a) (b) (c)

Figure 6.12: Geo-physical coverage of NAS gateways.

6.6 IP Infrastructure in the CDSN

In this section we examine the IP network infrastructure in the CDSN with the help of inferred (and ground-truth) locations for the set of geo-mapped basestations. Our goal is two-fold: i) to infer and understand the relationship (e.g. the geo-spatial distribution) of the IP network elements with the basestation substrate, and ii) to investigate whether we can geo-map the network elements in the CDSN to some degree. Here, we use the packet data sessions to extract the relevant relationships. Recall, the RADIUS/RADA packets contain basestation ids. In addition, they also contain four types of IP addresses of interest to us[12] . They are: *framed* IP addresses (assigned to end users' devices); the RADIUS/RADA server IP addresses (assigned to servers responsible for authenticating a user's session), IP addresses of *NAS gateways* (gateway servers to the IP data distribution backbone network in the CSDN) and IP addresses of home-agents (servers that maintain certain user information e.g. user registration, credentials, and current locations for mobile IP routing).

Not surprisingly, a predominant majority of the IP addresses in the datasets are framed IP addresses. They mostly come from the private address realm of the IP space; this is consistent with the findings in [78] where the authors collect and analyze the IP addresses seen at the end users' devices. The framed IP addresses appear to be assigned randomly from the private address ranges agnostic to the geo-location of the basestations. The number of the other three IP addresses are far smaller: both NAS gateways and home-agent IP addresses number within 100, and RADIUS/RADA servers below 10 – in stark contrast to the number of basestations (in tens of thousands). As the NAS gateways and home-agents are more likely to correlate with user/basestation

[12] Note that since the data is collected through *passive* measurement, we do not have IP addresses of IP routers, DNS servers, etc., and we are unable to conduct active measurements in the CDSN.

locations, in the following we explore the geo-spatial distribution of these IP addresses and their relation to the basestation infrastructure.

For each NAS gateway/home-agent IP address, we extract all the BSIDs which appear in the same RADIUS/RADA data packets containing the said IP address – these basestations are where the user data sessions originate. Hence each IP address (NAS gateway server or home-agent) is associated with a collection of basestations. We study the geo-spatial distribution of these basestations to investigate whether there is any significant geo-spatial correlation between the NAS gateways/home agents and locations of the basestations. We further analyze the relationships between NAS gateways by clustering them based on the number of basestations they share in common, i.e. the size of intersection between the basestation collections associated with the two NAS gateway IP addresses.

As representative examples, Fig. 6.12 depicts the geo-spatial distribution of the basestations associated with three different NAS gateways. We observe that these NAS gateways cover rather large geographical areas (spanning multiple states, and in terms of the basestation infrastructure, multiple SIDs). These areas are typically contiguous (as in Figs. 6.12 (a) and (b)), but sometimes can be disparate too (as in Fig. 6.12(c)). Furthermore, two or more NAS gateways may share a large overlapping set of basestations; it appears that these gateways serve the same large geographical region for load-balancing. We also performed similar analysis for home-agent IP addresses (where we also take into account the user activities to account for user mobility and roaming). We find that each home-agent IP address also covers a large geographical region, and multiple home-agents may cover the same or similar regions for load-balancing. Due to space limitation, we do not provide these results here.

In summary, we find that in contrast to the basestation infrastructure, the numbers of NAS gateways and home-agents are far smaller. While these gateways/home-agents are geo-spatially distributed, each covers a large geographical region spanning multiple states and corresponds to a large collection of the basestations in the CDSN substrate. Our findings point to several challenges in attempting to geo-map the CDSN infrastructure *from the outside* (cf. [78]), and in deploying *location-aware* content distribution services *outside the CDSN* to serve users inside the CDSN.

Last but not the least, we remark that comparing the two datasets collected about

10 months apart, we observe that the amount of cellular data activity and traffic has grown tremendously: for instance, the numbers of data sessions and application sessions increased over 3 and 10 times, respectively (see table 6.1). Moreover, the number of cellular data users have also increased considerably. With the increasing popularity of new generations of smart phones, the growth in cellular data traffic will further spur expansion of the IP networks within cellular service provider networks, and we may see more finer-grain geographic coverage to better cater to the growing user demand within a CSDN infrastructure.

6.7 Summary

In this chapter we put forth a novel approach for mapping the CDSN infrastructure via *(explicit) user geo-intent*, which circumvents the challenges plaguing conventional approaches (e.g. active probing). The intuition behind the proposed approach is to exploit specific geo-locations (i.e. geo-intent) contained in user queries to location-based services, and correlate them with basestation id's and gateway IP addresses to geo-map the CDSN infrastructure. To investigate whether — and to what extent — our approach can help geo-map the CDSN infrastructure, we employed the data (RADIUS/RADA packet data sessions and HTTP application sessions) collected at the core IP network inside a CDSN. We developed heuristics for identifying user geo-intent to geo-map the CDSN infrastructure — in particular, the basestations and IP NAS gateways — and evaluated their efficacy using a subset of basestations with known *ground-truth* GPS locations. Using zip-codes contained in user weather queries, we demonstrated that a large portion of basestations can be geo-mapped within a $3.9 - 6.1$ km range in general and within $1.5 - 2$ km range in densely populated urban areas. Furthermore, the geo-mapping accuracy is far better (often within 1 km) in large metro-areas with dense population and smaller zip-code areas. Using the inferred and ground-truth GSP locations of the basestations, we also examined the geo-spatial distribution of IP network elements such as NAS gateways and IP home-agents, and their relationship to the cellular network substrate.

With growing popularity of newer generations of GPS-enabled smart phones and increasing prevalence of location-specific and location-aware services and apps, we expect

our geo-intent based mapping approach to yield more precise results, as was illustrated using a small set of user geo-intent queries with GPS coordinates in our datasets.

Given the unprecedented growth in cellular data traffic, mapping the CDSN infrastructure is a critical step in understanding how to best expand and evolve the CDSN infrastructure to better meet growing user demands, and to guide the development and deployment of innovative location-aware services and applications that cater to mobile users and devices. Our study is only an initial step in this direction.

Chapter 7

On Activity and Mobility in Cellular Networks

Recent years have seen a surge in the number of studies related to human mobility patterns [87, 88, 89, 90]. For *large-scale* human mobility studies, one of the primary data sources is the call detail records (CDRs) collected by cellular services providers for billing and troubleshooting purposes. Consequently, several such CDR databases, appropriately anonymized for privacy, have been used by researchers to explore and quantify the basic laws governing human mobility at different scales and in various contexts [87, 88, 90].

In this chapter, we take a step back to inquire the limitations of using CDR data for human mobility analysis. Our intuition in doing so is based on two basic reasons. First, most of the datasets analyzed in the literature are usually voice-call [87, 88, 90], or additionally, short messaging service (SMS) datasets [91, 92]. Each time a user makes or receives a voice-call or an SMS message, the user's location is recorded in terms of the position of the cell-tower (base-station) that the user is communicating with at that time. Thus, the sample of observed locations for a user, in such datasets, is largely dependent upon user initiated activity and requires user participation. The number of times a user is observed in the CDR dataset is determined completely by the frequency of his/her voice-call and/or SMS activity. This leads to very sparse representation for most users as voice-calls have been reported to be bursty in nature[87]. Secondly, and

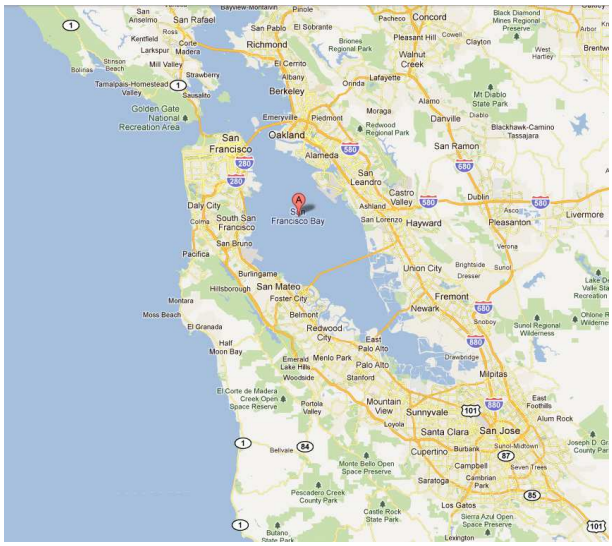


Figure 7.1: Territorial expanse of the dataset (SF-bay area).

this is an extension of the previous argument, most studies in literature resort to user sampling whereby high-frequency voice-callers or SMS-users are often selected to study human mobility patterns. Recently in [93] the authors observe a linear correlation between the number of voice-calls made by a user and the number of times the users change locations in Paris. A natural question, therefore, that arises is whether and to what extent does the selection of high-voice-call activity users skew the overall statistics for a population?

With rapid growth in mobile data and increasing adoption of smart phones, user *data activities* provide another rich source of information to study human mobility, in particular, to answer the aforementioned question. Unlike voice-calls and SMS activities, (user) data activities do not always require user initiation, nor user participation. For example, a plethora of applications running on 3G enabled cellular devices invoke themselves periodically or sporadically. These include push-mail notifications, periodic software updates and weather services, to name a few. The *data access records* which record such data activities by cellular providers, therefore, provide an unprecedented opportunity to investigate the limitations, if any, of the voice-call and SMS activities with respect to studies related to human mobility. However, compared to the number of studies using voice-calls and SMS CDRs, the number of studies exploiting data-access

records are far and few in between, notable exceptions being [94, 82].

In contrast, we utilize the user data-access records as well as the conventional CDRS (containing both voice-call and SMS activities) and take a *joint activity-mobility* perspective to study human mobility. In addition to answering the question posed earlier, we are also interested in studying whether there are distinct human *mobility and activity* patterns associated with different types of cellular activities: data, voice-call, SMS activities. But first, as always, we must provide a context by reviewing some pertinent works from literature.

7.1 Related Work

The analysis of human mobility patterns from empirical data has been an active area of research. Much of the early work reported in the literature had focussed on tracking devices in wireless LANs, in particular in WiFi university and corporate campuses, both of which provide a reasonable amount of user data [95]. The analysis of mobility data from wireless LANs delivered many significant results, too long to list exhaustively here, but which include the spectral analysis of mobility patterns [96], the evaluation of movement prediction schemes [97], the derivation of trace-based models [98], and the heavy-tail nature of movement and pause times (for example lognormal in [98]).

Other work has focused on measurement data collected on short-range networks, in particular on Bluetooth networks, with insightful results derived on the heavy tailed nature of inter-contact times (for example [99]). The recent availability of cheap GPS receivers led others to fit a few dozen willing participants with such receivers to obtain high-quality GPS mobility traces. They revealed walking patterns consistent with Lévy flights and heavy tailed inter-contact times, in agreement with earlier work (e.g. [100]).

All the references listed above analyze relatively small amounts of mobility data, typically from a few dozen to a few thousand users, monitored over periods ranging from a few weeks to a couple of years. In contrast, the call records collected by wireless operators provide orders of magnitude larger amounts of data. As the privacy and anonymization issues are being incrementally sorted, availability of voice-call data has become greater in the past few years. Much of the work on mobile voice-call records has focused on the structural analysis of the mobile call graph, for data mining purposes (see

[101]), and, to a lesser extent, in the study of mobility at aggregate population levels using statistical parameters like radius of gyration [88] and different kinds of entropies [90]. In such studies, the sample set of users chosen is usually the frequent voice-callers as only they provide enough sample points for any meaningful study.

The latest spree of papers that use voice-call CDRs to study human mobility patterns include [91, 88, 92, 90]. In most of these studies, the CDRs are the primary source of data used to infer locations of user populations with emphasis on either significant locations and/or population-wide statistics.

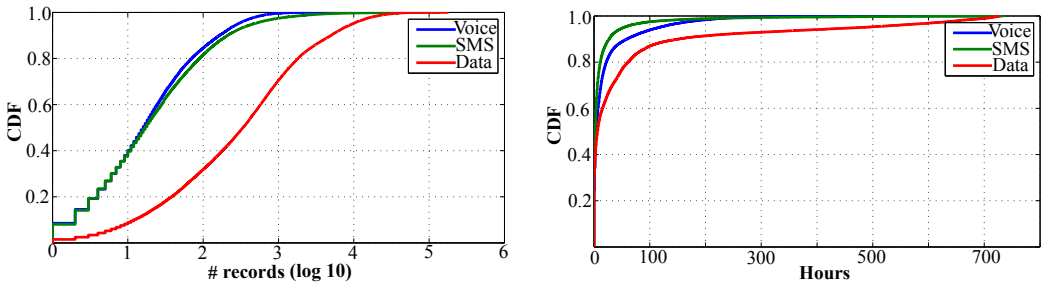
There are some notable exceptions however [102, 94, 82] in which data from location based services (mobile data records) have been used for mobility related studies. In particular, the authors in [82] study the spatio-temporal aspects of application usage patterns for cellular data users in a metropolitan city while the authors in [94] use the explicit geo-intent expressed by the users of a cellular data network to infer the location of the cellular infrastructure itself. Such studies are, however, few and far in between.

7.2 Preliminaries

In this section we introduce some of the preliminaries of our work. In §7.2.1, we provide details of our dataset, followed by a discussion of the activity rates of data-users versus non-data users in §7.2.2.

7.2.1 Dataset

Our primary dataset consists of *anonymized* cellular voice-call, SMS and data-session (2G and 3G) records collected from an operational CDMA 1xRTT-EVDO cellular network. Such records, also referred to as Per-Call Measurement Data (PCMD), are usually collected by cellular services providers for billing and trouble-shooting purposes. PCMD contains records of voice-calls, SMS and data activity of each cellular user. CDRs, or voice CDRs, mainly used for billing, are formed based on the PCMD records for voice sessions. Each PCMD record is a per-user-per-session record and consists of over 100 fields with information related to both the mobile device and the cellular network. In our data set, we use a selected set of fields from PCMD and among which, the user identifier field is anonymized beforehand. In addition, we deal with these fields: the



(a) # records (b) # active-hours

Figure 7.2: User activity: Overall volume and active-hours for data vs. non-data users.

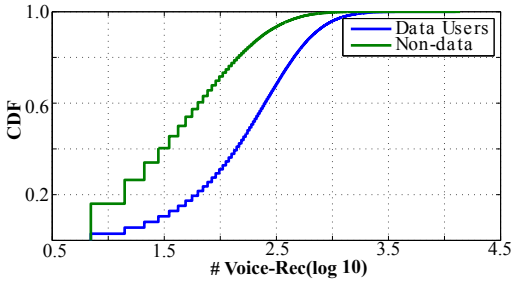


Figure 7.3: User activity: Voice-calls data vs. non-data users.

beginning and end timestamps for each call, the base-stations (cell-tower) associated with the beginning and end of each call, and the call-type that recognizes whether the session in progress is a voice-call, an SMS message or a data-session (2G and 3G). Note, that the location of the base-stations is known *a priori* and these are used as proxies for users location. Spatially, our dataset covers a 7,000 sq. mile wide territory in the San Francisco bay area, for over a million mobile users studied over a month long period (July, 2011).

7.2.2 Activity Volumes and Data Users

A user's activity rate in the cellular network often determines whether or not he is selected for a study. Studies which use only voice-call CDRs sometimes set the threshold as high as 0.5 calls per hour on an average [90], to ensure temporal completeness [1] .

[1] We also discretize the activity of users into 15 minute long time-slots thereby preventing over count bias at a location due to bursty activity.

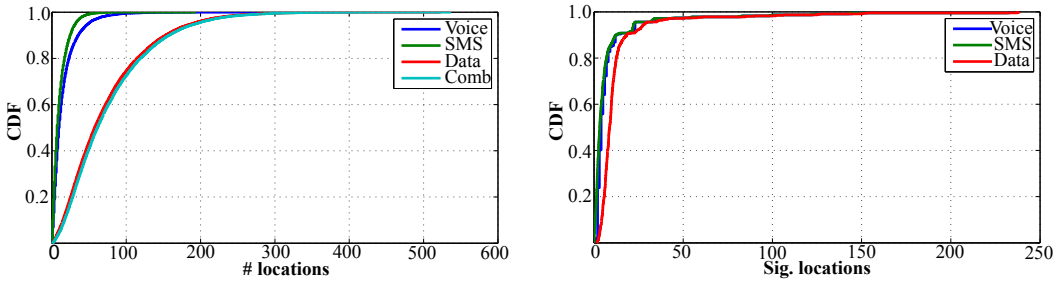
Figs. 7.2(a) and (b), respectively show the cumulative distribution frequency (CDF) of the number of records and number of hours of activity per user for each of the three activity types. Note that the voice-call activity contributes the least in terms of volume as well as the number of active hours, while the data-access activity contributes the most. Such high volumes and temporal spread for the data activity can be attributed in part to automated applications such as push-mail notifications and software updates, that usually occur in the background without the user's active participation. In contrast, voice-calls and SMS activity are largely user initiated, either by the user herself, or by the party at the other end of the communication. Data activity, therefore, potentially helps make the overall record of a user more complete in time, imperative for our study.

Thus, we divide the users into two types: users who have data-activity records (henceforth called data-users), and those who do not (non-data-users). We now show that selecting data-users for this study will in itself introduce no selection bias. The average voice-call and SMS activity volumes for data-users is in fact higher than that of non-data-users (Fig. 7.3) which has two important implications: first, the adoption of data plans by users does not seem to deter their voice-call and SMS activity volumes. Second, by using data-users as representatives of the overall population, we do not discriminate against the high frequency voice-callers or SMS users at all. They are as well represented in the set of data-users as they are in the set of non-data-users.

In the remainder of this study, unless otherwise mentioned, we focus on the data-user set which contains over 500 K users in it.

7.3 Is There a Possible Bias In Voice-Call Based Studies?

In this section we explore the question as to whether there is indeed a possibility of bias if voice-calls are used to study human mobility — individual or of populations. In §7.3.1, we look at the location profile of a user's spatial footprint — observed and significant locations — with particular emphasis on *home* and *work* locations. Next we explore the spatio-temporal aspects of mobility — entropy and radius of gyration — in §7.3.2 and §7.3.3.



(a) Distinct locations per user (b) Significant loc.

Figure 7.4: Spatio-temporal footprints for individual users.

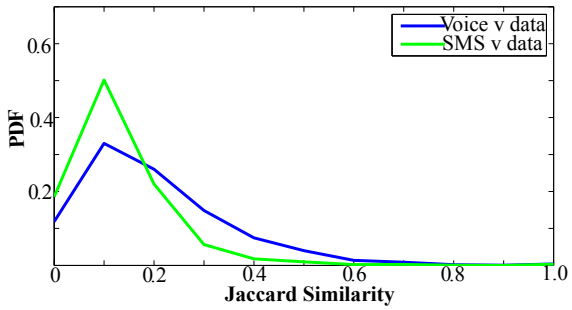


Figure 7.5: Overlap in significant locations.

7.3.1 Locations in a Cellular Network

We first analyze the number of distinct locations (N) visited by a user during the observation period which provides some insight into the diversity of a users spatial footprint. Fig. 7.4(a) shows the CDF for the number of locations visited by each data-user ($500K$ in number) as revealed by their voice-call, SMS and data activities respectively. We also plot the combined count for comparison. Note that the number of distinct locations revealed by the data-activity is clearly higher than those revealed by the voice-call and SMS activities. Interestingly, the SMS activity, despite being higher than the voice-call activity in terms of volume, fares no better than the voice-call activity in accounting for the diversity of a user's spatial footprint. Thus, for an individual user the voice-call and the SMS activities only partially account for, or equivalently underestimate, the set of locations where a user can possibly be found at random.

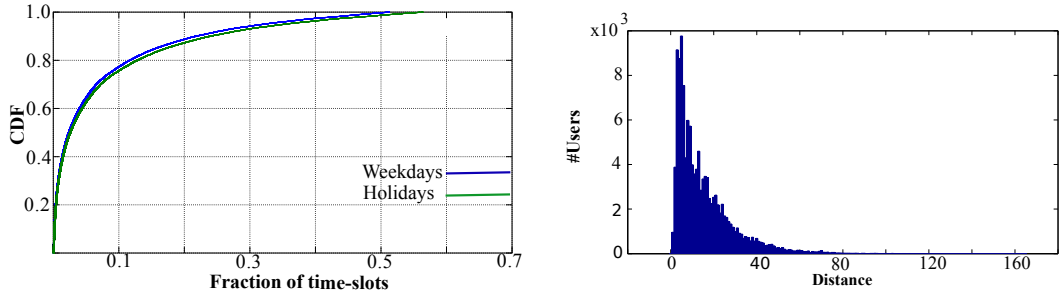
Significant Locations

However, not all locations are equal. Most users display a great degree of loyalty to certain locations (such as home, school and work) as compared to other infrequently visited ones, for example say a cinema theater. The set of *significant locations* [92] for a user is defined as the subset of all locations visited by a user that account for over 90% of his/her observed activity in the cellular network. In other words, a user is more likely to be found in one of these significant locations at a random point in time than the remaining 10% peripheral or not-so-significant locations. Indeed we find that the number of such significant locations as revealed by the voice-call and SMS activities, is 10 or fewer for over 80% individuals in our dataset (cf. Fig. 7.4(b)). In contrast, the significant location sets are relatively larger for the same population as revealed by the data-activity with the 80^{th} percentile at 18 locations per user [2] . Let V be the set of significant locations revealed by the voice-call activity (and similarly S: SMS and D: data respectively). We now compute the Jaccard-similiarity between these sets as follows:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{7.1}$$

Fig. 7.5 shows the probability distribution function for the Jaccard similarities of the voice-call and SMS activities with respect to the data activity for individual users. Note that the peaks of the Jaccard similarity are attained at as low as $X = 0.1$ accounting for 30% users when comparing the voice-calls and data activities. In other words, for 30% data-users, the overlap between the set of significant locations as revealed by their voice-call and data activities is as low as 10%. Similarly, the value for $X = 0.1$ is 50% when we compare the significant location sets for the SMS and data activities. This observation clearly suggests that for a significant portion of the user-base, the set of significant locations may differ significantly (no pun intended). To what extent this difference matters is what we explore next.

[2] Note that this difference may be also be due to the geographic expanse of the San Francisco bay area which is, in some sense, an extended metropolitan area.



(a) Time spent away from *home* and *work* (b) Distance (km) from home to work

Figure 7.6: Home and Work: Share of time and distance between locations.

Home and work

Of all the significant locations of a user, *home* and *work* locations are intuitively the most significant. We first select all users whose overall activity (combination of voice-calls, SMS and data-records) is spread across 250 or more hours out of the 744 hours in the observation period and who have at least three significant locations. Our dataset contains about 300 K users who fulfill this criteria, who will be used henceforth throughout this study for empirical analysis.

Next for each of these users, we consider the 20 working days from the month long period (excluding weekends and July 4), and divide the day into working hours (9:00 am to 6:00 pm) and non-working hours (the rest). We now compute the work and home locations of each user by using Hartigan's leader selection algorithm [103, 92] over the working-hours and non-working hours respectively. The *work* and *home* locations revealed by all three processes, voice-calls, SMS and data-sessions, quite remarkably, do not vary across the three processes for over 95% (nearly all) users. Also, we observe that number of time-slots in which the user is *not* at either his home or work locations on weekdays is less than 10% for over 80% users while on holidays it is a close match (cf. Fig. 7.6(a)).

The difference in the set of significant locations as described in the previous subsection must then be accounted for by transient locations for example the locations between home-work commute. Next we look at the length of this commute. For over 57% users out of the 300 K users, the home and work locations are either the same or fall within the same zip-code. Fig. 7.6(b) shows the histogram of the home-work distances of users

whose home and work locations are not within the same zip-code. We observe that the peak of the distribution lies between $4-8$ km, while 50^{th} and 75^{th} percentiles lie at 10 km and 21 km respectively.

Next we explore the implications of these observations over the observed spatio-temporal footprint of users.

7.3.2 Spatio-temporal Footprint

We now describe two metrics from literature [88, 90] to analyze the spatio-temporal characteristics of individual users as well as populations.

The Shannon entropy

Like the random entropy S^R, another entropy measure that is commonly used in literature is the Shannon entropy (also referred to as the temporally uncorrelated entropy S^U in [90]). Precisely,

$$S^U = -\sum_{i=1}^{N} P_i \; log_2 P_i \tag{7.2}$$

where P_i is the probability that an activity was observed at location i from the set of N locations that a user visits. S^U is, therefore, a measure of the spread of a user's activity over his/her spatial footprint (locations).

The radius of gyration

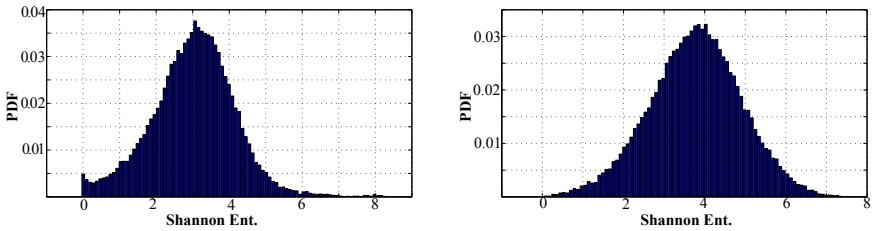
To quantify the range of a user's trajectory[88, 90] the so called the radius of gyration (R_G) is often user. Let $\overrightarrow{R_i}$ denote the position of the user at time i (say time-slot i if the observation period is discretized). Then the radius of gyration of the user is given by:

$$R_G = \sqrt{\frac{1}{L}\sum_{i=1}^{L}(\overrightarrow{R_i} - \overrightarrow{R_{cm}})^2} \tag{7.3}$$

where $\overrightarrow{R_{cm}}$ is the center of mass for all the temporally recorded locations for the user (L in total).

Percentile	25^{th}	50^{th}	75^{th}
# Voice-calls	246	437	695

Table 7.1: Quartile-wise break-down of number of voice calls made by high-activity users.



(a) Lowest frequency users (b) Highest frequency users

Figure 7.7: Shannon entropy (S^U) comparison across voice-caller classes based on frequency.

7.3.3 Looking for Possible Biases

We now look at the following two questions: (i) does using voice-calls CDRs to study individual mobility patterns potentially introduce a bias in the observed properties? And (ii) does selecting high-frequency voice-callers to study the mobility characteristics of the population potentially introduce a bias?

For an individual user

We now compute the S^U and R_G values for each individual using first only the voice-call records and then the overall record. Figs. 7.8 and 7.9 (a) respectively show the CDF distributions of absolute errors incurred in the computation of S^U and R_G respectively for individual users. We observe that for over 50% users S^U incurs an absolute error of around 0.25 and above. In contrast, the R_G values are estimated to within a 1 km error range by the voice-call process for over 80% users. Therefore, the only possible bias voice-call process seems to incur is in terms of the entropy, which we shall look at greater detail in a subsequent section.

User-classes by voice-call frequency

Next we explore the question of whether (and to what extent) high-frequency voice-callers are representatives of the population on a whole? To do so, we partition the set of 300 K users cited above into four quartiles each of 75 K users, by the number of voice-calls made by them (see table 7.3.3).Thus we have the low-frequency voice-callers with fewer than 246 voice-calls in a month (the first quartile) to compare against those in the other three quartiles. Figs. 7.7(a) and (b), respectively show the probability distribution of S^U for first and the fourth quartile users. Note that S^U is computed using the overall activity record for the user and not just the voice-call activity. We observe that the peak of the probability distribution shifts from around 3.00 for the first quartile users to around 4.00 for the fourth quartile (in fact this increase is consistent across the quartiles). Thus, as far as the population is concerned, the uncorrelated entropy measure might be overestimated if we select high-frequency voice-callers as representatives of the population.

Finally, we look at the distribution of the radii of gyration for the users of the four classes by voice-call frequency. Fig. 7.9(c), shows the log-log distribution of R_G of the first and the fourth quartile users. We observe that the distributions nearly overlap suggesting a lack of variance across user classes by voice-call frequency (the same is true for the second and third quartile users). R_G distribution of a population is often characterized in terms of a truncated power-law [88]. We observe that the exponents for the power-law fit across the four classes are in the range $\beta = [1.76 - 1.79]$, consistent with that in [88].

To summarize therefore, using high-frequency voice-callers to study aggregate mobility of user populations can incur possible biases for the uncorrelated entropy but the radius of gyration distributions seem to be immune to such selection. But Shannon entropy is only a number, and thus although it provides a hint into possible differences, we need better means to characterize these differences. In what follows, therefore, we motivate mobility studies in the form of a sampling problem to understand as to why and under what conditions the entropy of a user differs across the activities and how, if possible, to rectify for it.

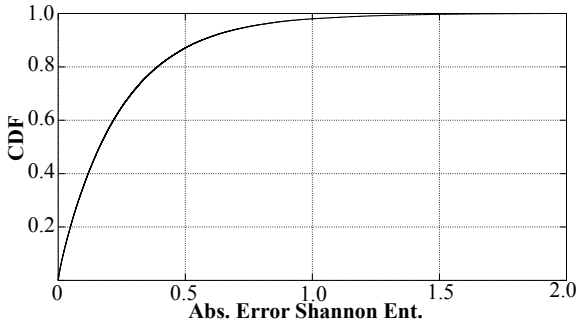
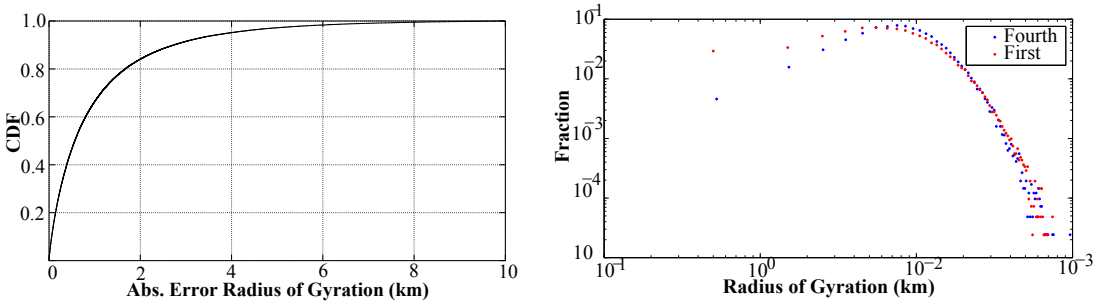


Figure 7.8: Comparing relative errors: Shannon Entropy.



(a) Radius of gyration (R_G) (c) Radius of gyration across classes

Figure 7.9: Comparing relative errors: Radius of gyration.

7.4 A Sampling Problem

In this section we look at the nuances of the spatio-temporal footprint of individual users and the underlying biases in terms of a sampling problem. In §7.4.1 we provide a case-study to show that preferential locations for different activity types may indeed lead in an over-counting bias. Then, in §7.4.2, we formally state the sampling problem as well as motivate an imposed sampling process to compare the voice-call process against.

7.4.1 An Illustrative Example

We now present a case study of a frequent voice-caller to put into perspective the sampling problem. Our example user, has 510 voice-call records spread over 218 hours in the observation period. This is higher than the 90^{th} percentile of the number of voice-calls per user. The user also has over 780 SMS records and $7,300$ data-records amounting to nearly $8,500$ activity-records in total (voice-calls, SMS and data) spread

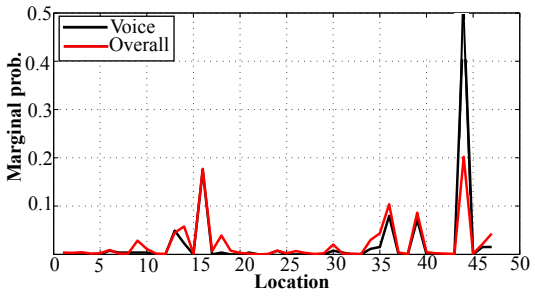
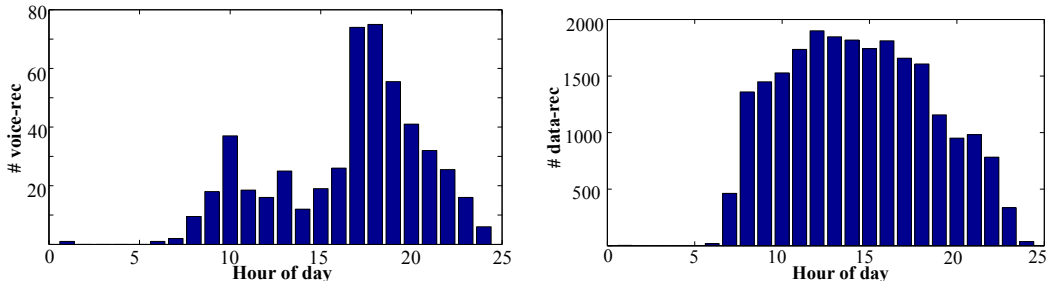


Figure 7.10: Example user: Prob. of observation per location.

over 466 hours. Moreover, the spatial-footprint of the user encompasses 47 distinct locations in the San Francisco bay area, which is close to the mean number of locations visited by the users in our dataset. Out of these, the number of locations accounted for by the voice-call activity alone is 24, once again higher than the mean number of locations accounted for by voice-calls for the user-base. In short, this user is likely to be sampled for a mobility study of individuals, with high probability, based on either selection criteria: activity as well as mobility, irrespective of the kind of activity under consideration.

Next we compute the probability for this user to make a voice-call over the set of overall locations visited by the user (cf. Fig. 7.10). Two observations stand out: the voice-call activity misses some significant locations, and location 44 alone accounts for almost 50% of the marginal distribution for the voice-call activity. On further inspection, we discovered that the user is mostly present at location 44 during the evening hours (cf. Figs. 7.11(a) and (b)) i.e. his home locations. Thus, we observe that this is a preferential location for the voice-call activity for this user. In contrast, the data activity (and consequently the overall activity) is more evenly spread over the hour-of-day.

Such preferential behavior for voice-activity clearly may lead to biased estimates of both Shannon entropy (an absolute error of 0.34 in this case) as well as the radius of gyration (0.25 km). Whereas the absolute error in R_G is intuitive to understand, we need a better insight into the Shannon entropy error and if possible make an attempt to correct for it. This we do in the next subsection.



(a) Hour of day (Voice-calls) (b) Hour of day (Data-records)

Figure 7.11: Example user: Temporal Activity.

7.4.2 Mobility as a Sampling Problem

A user's activity-profile (say the voice-call activity), is in fact a segmentation of the observation period whereby events (such as voice-calls) occur at certain times, interspersed with periods of inactivity of varying lengths. Our view of a user in the cellular network is entirely dependent on this event-pause sequence. We observe a user and his location if an only if there is an event and not during the pauses. The observed mobility can therefore be posed as a sampling problem in the following way. Given an observation period of T hours, define a discretized partitioning of T in terms of time-windows of length M minutes each. The number of maximum observations is given by $W = (T * 60)/M$. By abuse of notation, we will use W to represent the set of time-slots as well. A sampling process $\mathcal{S}(W)$ is then defined over the set W of time-slots, that samples a user's location at discrete time intervals determined by a rate function ρ. The output of the sampling function $\mathcal{S}(W)$ is a subset $\overline{W} \subset W$, of the overall set of time-slots, where the number of sampled windows is determined by the rate function ρ. From the point of view of mobility, if L be the set of all locations visited by a user, this hypothetical sampling function $\mathcal{S}(W)$ only records a subset of locations $\overline{L} \subset L$, which the user visits during the time-windows of the set $\overline{W}$.

The *goodness* of the sampling process $\mathcal{S}(W)$ can then be determined in the following way. Let $P_{\mathcal{S}} \in \Re^N$ be the marginal probability distribution of the sampling process $\mathcal{S}(W)$, over the set of all locations that constitute the spatial footprint of the user. The entry $p_i \in P_{\mathcal{S}}$, is probability of sampling location i. Similarly, let P_O be the marginal probability of the overall activity distributed over the set of locations. Then the marginal

distributions can now be compared in terms of the Jensen-Shannon divergence [104] between the two distributions as follows:

$$JSD(P_{\mathcal{S}}||P_O) = \frac{1}{2}(D(P_{\mathcal{S}}||P_M) + D(P_O||P_M)) \tag{7.4}$$

where $P_M = \frac{P_{\mathcal{S}}+P_O}{2}$ and $D(P_{\mathcal{S}}, P_M)$ is the Kullback-Leibler divergence between $P_{\mathcal{S}}$ and P_M[3] . In information theory, Jensen-Shannon divergence is often used as a measure of mutual information between two probability distributions (lower the Jensen-Shannon divergence, more similar the two probability distributions are). Additionally, we can also compare one or more of the popular metrics in literature (discussed earlier) such as the set of locations visited by a user, the Shannon entropy and/or the radius of gyration. We therefore have several ways of quantifying the bias incurred by a sampling process as compared to the overall observed activity-mobility profile (which in itself is a sampling over the true mobility of a user).

In view of the above, it is easy to see that the voice-call activity (or for that matter SMS, data-activity and the overall activity) clearly fits the description of a sampling process. And we have a measure, namely the Jensen-Shannon divergence of the voice-call process against the overall activity process, to quantify the bias. However, the Jensen-Shannon divergence is only a relative measure of difference. In order to make sense of the difference, we need at least one other process to compare against, and we choose an artificially imposed one. Can a sampling process defined with the same average rate as that of the voice-call activity perhaps perform better? This is the question that we now deal with in detail.

We now formalize the problem of assessing the *suitability/goodness* of the voice-call activity as a sampling process. For a given user, let $\mathcal{S}^V(W)$ be the sampling process representing the user's voice-call activity. If the number of voice-calls made by the user during the observation period be V, then the average rate-function is simply $\rho = V/|\Delta T|$ i.e. the number of calls made by the user between the first and final hour during which there is a voice-call record associated with him/her. We define the imposed sampling process $\mathcal{S}^I(W)$, as an instance of the set of all sampling processes with the same average rate ρ as exhibited by the voice-call activity of the cellular user. For convenience we

[3] We choose the Jensen-Shannon divergence for these comparisons purely because it is bounded in the interval [0,1] [105] and also measures mutual information between the two marginal probability distributions.

choose a Poisson process with the same intensity function as the average voice-call rate as our imposed sampling function. The reason for this choice is simple: a Poisson process is the most evenly spread out random process with a given rate function. The average call rates of users are easy to estimate and this is the only parameter required to define the Poisson process, thus making the choice quite obvious.

A second imposed sampling process that we study is one with varying fractions of the overall activity-rate for a user. Our aim, in doing so, is to determine the sampling rate at which an imposed activity sampling process provides a *good* sample of the user's observed mobility behavior.

7.5 Experiments

In this section, we describe the experimental results for the various sampling processes described previously. In §7.5.1 we compare the voice-call process against the imposed Poisson sampling process with the same intensity followed by a study of varying rate of sampling with fractions of overall activity rates in §7.5.2.

7.5.1 Voice-call Sampling vs. an Imposed Poisson Process

We now compare the voice-call based sampling for individual users vis-a-vis a Poisson process with the same intensity as the average number of voice-calls per active hour for the user. However, before doing so, we first need to pick a relevant sample of users from the dataset with enough number of voice-calls to make any sensible comparisons. As observed previously, for data-users the 25^{th} percentile for the number of voice-calls is 68, the 50^{th} percentile is at 153 while the 75^{th} percentile lies at 315. We now classify the data-users for this comparative study into low-activity (68 to 153 calls), medium-activity (153 to 315 calls) and high-activity (315 calls and above), based on the quartile margins. Note that our high activity group is quite similar to the one picked in [90] where the selection criteria is 0.5 calls per hour which is roughly 372 calls in our case.

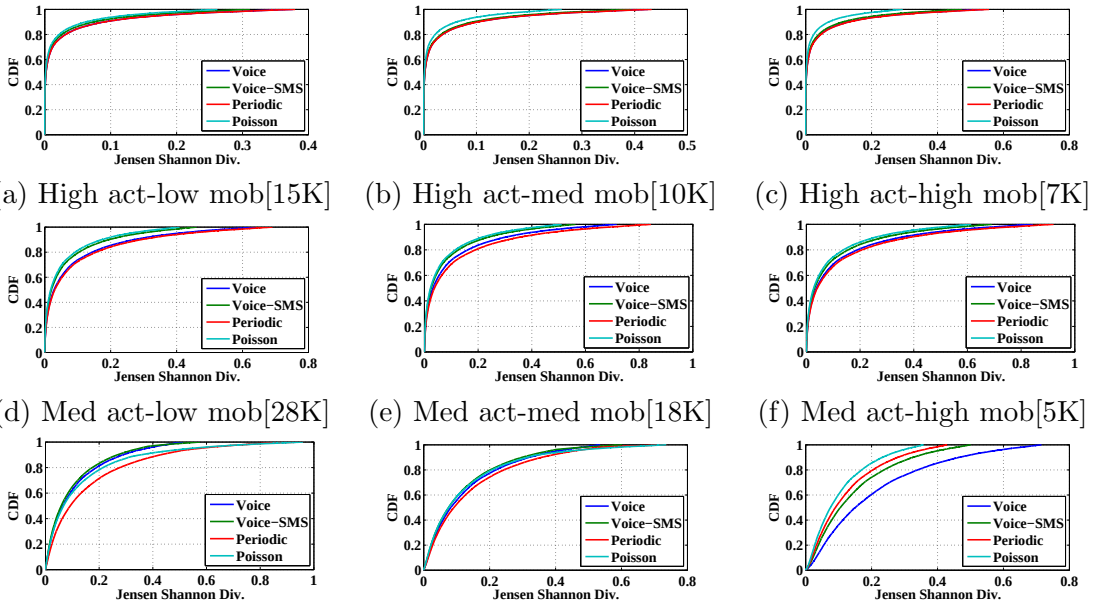
Similarly, we also divide each of the three activity classes described above into three mobility-classes based on the number of locations that constitute the set of *significant locations* for each user (as accounted for by his/her overall activity). Recall that despite a remarkable difference in the number of locations observed by the data process, the

set of significant locations is 15 or fewer for over 75% of the data users. Once again, we define as low, medium and high mobility classes for users whose significant location sets contain $3-8$, $8-15$ and 15 and above locations. Note that although considering significant locations reduces the impact of extremely low probability locations, there is always a caveat that not all significant locations are *equally* or *competitively* significant. We defer the details to a latter paragraph.

Given user i whose number of voice-calls in the entire duration is at least above the 25^{th} percentile, denoted by V_i. We also note the overall activity span of user i i.e. the difference in between the times at which user i makes the first and the last voice-calls, (say ΔT_i in terms of the number of 15 minute intervals separating the first and last voice-calls). This yields the rate $\rho = V_i/\Delta T_i$ for the Poisson (and periodic) sampling processes that we will impose to sample the locations at which user i is active (voice-SMS-data). Note that the aim is to sample (approx.) the same number of these discrete time windows as the number of voice-calls made by user i. We also require that the active interval ΔT_i represent at least a two-week long (14 days) period in order to avoid random visitors in our population. Also, for the Poisson and periodic sampling process, we generate 10 different random starting points (determined by overall activity and not just the voice-activity) for sampling and then take the average of the 10 instances to avoid *temporary* void periods.

Marginal distributions

We now present the results of this comparative analysis using the Jensen-Shannon divergence for the marginal probability distributions (cf. Fig. 7.12). Observe that all the four processes are competitive when the voice-call activity is high, for a large fraction of the users. The Poisson and the voice-cum-SMS sampling processes perform ever so slightly better than voice and periodic processes in the high activity category. This is not surprising as a significant population of data-users with high voice-call rates also have higher data activity rates. Therefore, the overall activity (voice-SMS-data) only improves on the temporal spread of the voice-activity in the average case, that is subsequently reflected in higher *hit-rates* for the imposed Poisson and periodic processes (cf. Fig. 7.12). Moreover, we observe that the divergences increase for each of the four processes, on an average, when we move from low to high mobility classes. However,



(a) High act-low mob[15K] (b) High act-med mob[10K] (c) High act-high mob[7K]

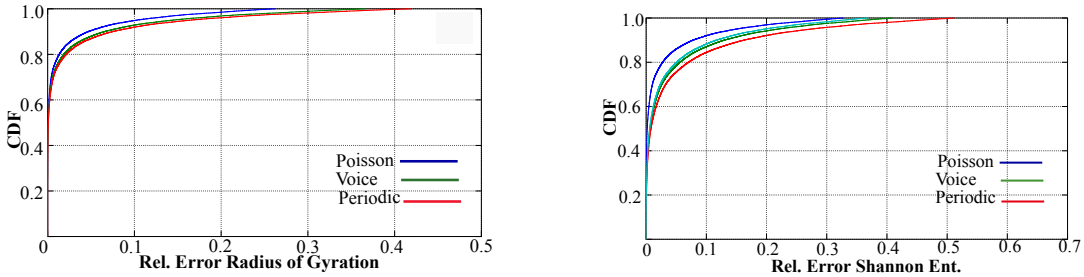
(d) Med act-low mob[28K] (e) Med act-med mob[18K] (f) Med act-high mob[5K]

(g) Low act-low mob[31K] (h) Low act-med mob[6K] (i) Low act-high mob[2.5K]

Figure 7.12: CDF: Inter-class comparisons mobility and activity; numbers in brackets indicate users per class.

the Poisson process continues to perform better, again ever so slightly, than the other three. Particularly, for the high-activity-high-mobility class, we observe that the difference between the Poisson and the other processes is more pronounced. This clearly indicates that for users whose observed spatial diversity is higher, and spread over a number of locations, the voice and SMS processes tend to have selective bias towards certain specific locations (as shown for the example user in the previous section). This is important to note as in most previous studies the high-activity class is the only one that is studied.

For the medium activity group, we observe that the Poisson and voice-cum-SMS processes combined tend to perform better than the voice process with increasing mobility. Predictably as the number of sample points decrease and the location diversity increases, the performance of the imposed sampling processes decreases due to lower *hit-rates*. Yet, overall we observe that the Poisson and voice-cum-SMS processes perform better. This may be a result of the fact that the SMS process augments the voice process at locations where the users tend to make fewer voice-calls.



(a) Radius of gyration (R_G) (b) Shannon entropy (S^U)

Figure 7.13: CDF: Relative errors in radius of gyration and uncorrelated entropies of data users.

We observe similar trends in the low activity group barring the low-activity-low-mobility users for whom the obvious handicap is the extreme sparsity of data.

Therefore, despite several competing factors, we observe that the Poisson and the voice-cum-SMS processes perform better on an average than the voice process, particularly as the number of significant locations increases.

Other mobility parameters

We now look at the relative error incurred in computing the radius of gyration and Shannon entropies of users by the various sampling process in Figs. 7.13(a) and (b). Notice once again, that the relative errors incurred by the Poisson process are comparatively lower than that incurred by the others (even if ever so slightly).

7.5.2 Imposed Sampling Processes with Varying Intensities

We now explore the imposed Poisson sampling process from another perspective. This time we take into consideration the overall activity rate for individual users (instead of their voice-call activity) to determine the intensity function for the imposed Poisson sampling process. We study the performance of this imposed Poisson sampling process for decreasing rates of the intensity function ρ. We decrease ρ in successive integral steps of the average activity rate for the user through a decay coefficient $\kappa = \{2, 4, 8, 16, ...\}$. Our aim in doing so is to determine the least rate of sampling (or equivalently the highest value of κ) at which the imposed Poisson process incurs a Jensen-Shannon divergence below a certain threshold ϵ (< 0.1 say).

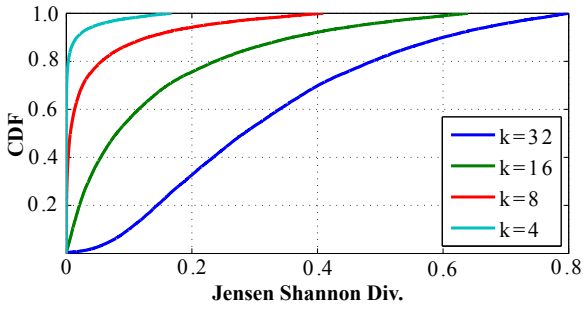


Figure 7.14: CDF: Jensen-Shannon divergence between marginal distributions for Imposed Poisson processes with varying intensities(10 K Data-users).

However, before we elaborate on the results, we need to select our sample user set carefully. The 75^{th} percentile for the number of records (voice-SMS-data combined) per hour lies at 21 records per hour, or equivalently one record every three minutes on an average. As, we need users for whom we can explore a wide range of κ values, it is reasonable to select only those users who have high activity rates to begin with (or else we might end up with the same issue as demonstrated in the previous subsection). Therefore, for the purposes of this experiment, we first concentrate on the high-activity-high-mobility group of users, imposing the same restriction that the user's first and final activity must span a period of two weeks at the very least.

Fig. 7.14 shows the Jensen-Shannon divergence of the imposed Poisson sampling processes vs. the observed marginal distribution. We observe that as the rate of the sampling process decreases, the Jensen-Shannon divergence increases with regularity, which in itself is not surprising. However, notice that the divergence becomes greater than 0.1 for 90% of users only at $\kappa = 32$ i.e. when the intensity function for the imposed Poisson sampling is 1/32 of the average activity rate for these users. Therefore, we conclude, from the evidence at hand, that an imposed Poisson sampling process with an intensity function much lower than the overall activity rate for users, performs well as a sampling process for most users.

7.6 Summary

In this chapter, we discussed the possible caveats of using voice-call detail records (CDRs) for studying individual human mobility patterns. While CDRs provide an unprecedented source for user locations at large population scales, there are some obvious limitations on them, largely due to the underlying nature of the voice-call process, which being human initiated depends on the calling frequencies of an individual. This may lead to a skewed view of the spatio-temporal distribution of an individual over the set of all locations visited. Using the dataset of over a million cellular users from the San Francisco bay area, covering several thousands of square miles, for a month long period, we demonstrated that the voice-call activity does well in inferring significant locations like *home* and *work*, even though it may fail to capture the nuances. When compared with a Poisson sampling process with the same intensity, the voice-call process compares reasonably well for high call-activity users, but the Poisson process certainly improves on the performance, particularly as the activity rates vary. Thus when designing location-sensing applications on a mobile device to sample users' locations a similar imposed process might come handy. From the point of view of populations, we observe that while the radius of gyration does not show variation across different classes of users by activity, the Shannon entropy values may in fact be over-estimated. Therefore, the use of voice-calls for human mobility patterns should be taken with advised caution depending upon the nature and objectives of the study.

Chapter 8

Conclusion and Discussion

In this thesis, we discussed a geometry of networks approach to studying the structural properties of networks at the granularity of individual nodes and edges, as well as that of the network as a whole. We demonstrated how the geometric properties of the eigen space associated with the Moore-Penrose pseudo-inverse of the combinatorial Laplacian translates to significant topological characteristics at different scales. We also demonstrated how a network (or the graph associated with it), can be viewed in terms of its sub-network structures. This observation led to useful computational developments in the form of a divide-and-conquer methodology for computing the submatrix and pseudo-inverses of the Laplacian for dynamic time-evolving graphs as well as networks of high orders. We then used these observations to study the case of interdependent (infrastructure) networks where the overall robustness of the system was shown to depend significantly on the choice of the coupling function. We deduced, through analytical and empirical analysis, that a seemingly optimal strategy is to diffuse the inter-dependencies across geographically disparate sites. Needless to say, our methods are generally applicable to all set-ups where a system can be modeled as a simple, undirected graph. Also, when seen in totality, the several approaches used in our analyses are derived from diverse numerical and computational disciplines: from random walk/ Markov chain literature to linear algebra and graph theory. Therein lies the universality of our approach.

Finally, we studied the case of cellular data networks — a class of networks with its own nuances of structural interdependence. First and foremost, a daunting challenge

in this case, as discussed previously, is the opaqueness of the underlying infrastructure from any observer outside the network. We demonstrated how this handicap can be circumvented by exploiting the explicit geo-intent of cellular users as expressed in their application data, particularly in weather queries. A critical insight of the study is that large geo-physical territories, spanned by hundreds of basestations, are often served by a few NAS server pairs. This leads us to conclude that potential catastrophic failures in the data plane can lead to significant disruptions in the overall communication capability. Last but not the least, we studied the problem of inferred human mobility and the possible biases incurred due to observational limitations in the call-detail-record datasets. Through real-life trace driven analysis, we showed how we can not only quantify the bias, but also correct for it through an imposed sampling process. These observations are important as the dynamics of human populations are vital for urban planning and disaster management in urban and metropolitan locations. Our work is therefore a relevant contribution to the body of scientific literature in related areas.

References

[1] N. Biggs. *Algebraic Graph Theory.* Cambridge University Press, 1993.

[2] M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Math. J.*, 23:298–305, 1973.

[3] M. Fiedler. A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory. *Czechoslovak Math. J.*, 25(100):619–633, 1975.

[4] U. V. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, December 2007. Max Planck Institute for Biological Cybernetics. Technical Report No. TR-149.

[5] Noga Alon. Eigenvalues and expanders. *Combinatorica*, 6(2):83–96, 1986.

[6] F. R. Chung. *Spectral Graph Theory.* Am. Math. Soc., 1997.

[7] W. E. Donath and A. J. Hoffman. Lower bounds for the partitioning of graphs. *IBM J. of Res. and Dev.*, 17:410–425, 1973.

[8] F. Fouss, A. Pirotte, J. M. Renders, and M. Saerens. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 19, 2007.

[9] F. Gobel and A. Jagers. Random walks on graphs. *Stochastic Processes and Their Applications*, 2:311–336, 1974.

[10] B. Mohar. The Laplace spectrum of graphs. In *Graph Theory, Combinatorics, and Applications*, volume 2, pages 871–898. Wiley, 1991.

[11] B. Mohar. Laplace eigenvalues of graphs - a survey. *Discrete Math.*, 109:171–183, 1992.

[12] B. Mohar. Some applications of Laplace eigenvalues of graphs. In *Graph Symmetry: Algebraic Methods and Applications*, volume NATO ASI Ser C 497, pages 225–275. Kluwer, 1997.

[13] A. Nilli. On the second eigenvalue of a graph. *Discrete Mathematics*, 91(2):207–210, 1991.

[14] J. Shi and J. Malik. Normalised cuts and image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8):888–905, 1997.

[15] A. Ben-Israel and T. Greville. *Generalized Inverses: Theory and Applications, 2^{nd} edition*. Springer-Verlag, 2003.

[16] M. Saerens, F. Fouss, L. Yen, and P. Dupont. The principal components analysis of a graph, and its relationships to spectral clustering. In *Proc. 15th European Conf. Machine Learning (ECML '04)*, pages 371–383, 2004.

[17] P. G. Doyle and J. L. Snell. *Random Walks and Electric Networks*. The Math. Assoc. of America, 1984.

[18] D. J. Klein and M. Randić. Resistance distance. *J. Math. Chemistry*, 12:81–95, 1993.

[19] R. Albert, H. Jeong, and A. L. Barabási. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[20] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *Proc. of the ACM SIGCOMM*, pages 251–262, 1999.

[21] I. J. Farkas, I. Derényi, A. L. Barabási, and T. Vicsek. Spectra of real world graphs: Beyond the semicircle law. *Physical Review E*, 64(2), 2001.

[22] L. C. Freeman. A set of measures of centrality based upon betweenness. *Sociometry*, 40:35–41, 1977.

[23] L. C. Freeman. Centrality in social networks: Conceptual clarification. *Social Networks*, 1:215–239, 1979.

[24] L. C. Freeman, S. P. Borgatti, and D. R. White. Centrality in valued graphs: A measure of betweenness based on network flow. *Social Networks*, 13:141–154, 1991.

[25] M. E. J. Newman. A measure of betweenness centrality based on random walks. *Social Networks*, 27(1):39–54, 2005.

[26] K. A. Stephenson and M. Zelen. Rethinking centrality: Methods and examples. *Social Networks*, 11:1–37, 1989.

[27] J. D. Noh and H. Rieger. Random walks on complex networks. *Phys. Rev. Lett.*, 92, 2004.

[28] S. P. Borgatti. Centrality and network flow. *Social Networks*, 27(1):55–71, 2005.

[29] E. Estrada and J. A. Rodríguez-Velázquez. Subgraph centrality in complex networks. *Phys. Review E*, 71, 2005.

[30] E. Estrada and N. Hatano. A vibrational approach to node centrality and vulnerability. *Physica A*, 389, 2010.

[31] M. Mitrović and B. Tadić. Spectral and dynamical properties in classes of sparse networks with mesoscopic inhomogeneities. *Physical Review E*, 80, 2009.

[32] S. N. Dorogovtsev, A. V. Goltsev, J. F. F. Mendes, and A. N. Samukhin. Spectra of complex networks. *Physical Review E*, 68(4), 2003.

[33] W. Xiao and I. Gutman. Resistance distance and laplacian spectrum. *Theoretical Chemistry Accounts*, 110:284–289, 2003.

[34] J. L. Palacios. Closed-form formulas for kirchhoff index. *Intl. Journal of Quantum Chemistry*, 81:135–140, 2001.

[35] J. L. Palacios. On the kirchhoff index of regular graphs. *Intl. Journal of Quantum Chemistry*, 110:1307–1309, 2001.

[36] J. L. Palacios. Resistance distance in graphs and random walks. *Intl. Journal of Quantum Chemistry*, 81:29–33, 2001.

[37] J. L. Palacios and J. M. Renom. Bounds for the kirchhoff index of regular graphs via the spectra of their random walks. *Intl. Journal of Quantum Chemistry*, 110:1637–1641, 2001.

[38] B. Zhou and N. Trinajstić. A note on kirchhoff index. *Chemical Physics Letters*, 455(1-3):120–123.

[39] E. Bendito, A. Carmona, A. M. Encinas, J. M. Gesto, and M. Mitjana. Kirchhoff indexes of a network. *Linear Algebra and its Applications*, 432(9):2278–2292, 2010.

[40] J. G. Kemeny, J. L. Snell, and A. W. Knapp. *Denumerable Markov Chains*. Van Nostrand, New York, 1966.

[41] P. Tetali. Random walks and effective resistance of networks. *Journal of Theoretical Probability*, pages 101–109, 1991.

[42] S. J. Kirkland, M. Neumann, and B. L. Shader. Distances in weighted trees and group inverse of laplacian matrices. *SIAM Journal of Matrix Anal. Appl.*, 18:827841, 1997.

[43] www.stanford.edu/services/internet2/abilene.html.

[44] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

[45] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Preprint Physics/0605087*, 2006.

[46] B. Bollobás and P. Erdős. Graphs of extremal weights. *Ars Combin.*, 50:225–233, 1998.

[47] M. Randić. On characterization of molecular branching. *J. Amer. Chem. Society*, 97:6609–6615, 1975.

[48] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the internet's router-level topology. In *Proc. of the ACM SIGCOMM*, 2004.

[49] G. Ranjan, Z.−L. Zhang, and D. Boley. Incremental computation of pseudo-inverse of laplacian: Theory and applications. *arXiv:1304.2300*, 2013.

[50] Y. E. Campbell and T. A. Davis. Computing the sparse inverse subset: An inverse multifrontal approach. *Tech. report TR-95-021, Univ. of Florida, Gainesville*, 1995.

[51] U. V. Luxburg, A. Radl, and M. Hein. Getting lost in space: Large sample analysis of the commute distance. *NIPS*, 2010.

[52] D. Isaacson and R. Madsen. *Markov Chains Theory and Applications*. John Wiley and Sons, 1976.

[53] G. Ranjan and Z.−L. Zhang. Geometry of complex networks and topological centrality. *arXiv:1107.0989*, 2011.

[54] B. Bollobás. *Random Graphs*. Cambridge University Press, 2001.

[55] C. D. Meyer. Generalized inversion of modified matrices. *SIAM Journal on Applied Mathematics*, 24(3):315–323, 1973.

[56] M. Brand. A random walks perspective on maximizing satisfaction and profit. In *Proc. 2005 SIAM Int'l Conf. Data Mining*, 2005.

[57] A. K. Chandra, P. Raghavan, W. L. Ruzzo, R. Smolensky, and P. Tiwari. The electrical resistance of a graph captures its commute and cover times. In *Proc. of Annual ACM Symposium on Theory of Computing*, pages 574–586, 1989.

[58] F. Fouss, A. Pirotte, J. M. Renders, and M. Saerens. A novel way of computing similarities between nodes of a graph, with application to collaborative recommendation. In *Proc. 2005 IEEE/WIC/ACM Int'l Joint Conf. Web Intelligence*, pages 550–556, 2005.

[59] F. Fouss, L. Yen, A. Pirotte, and M. Saerens. An experimental investigation of graph kernels on a collaborative recommendation task. In *Proc. of the 6th International Conference on Data Mining*, 2006.

[60] B. Sarwar, G. Karypis, J. Konstan, , and J. Riedl. Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In *Proc. Fifth Int'l Conf. Computer and Information Technology*, 2002.

[61] A. L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[62] B. Tadić and V. Priezzhev. Voltage distribution in growing conduction networks. *European Physical Journal B*, 30:143–146, 2002.

[63] A. Sen, P. Ghosh, B. Yang, and V. Vittal. A new min-cut problem with application to electric power network partitioning. *European Transactions on Electrical Power*, 2008.

[64] C. J. Alpert and A. B. Kahng. Recent directions in netlist partitioning: A survey. *INTEGRATION(the VLSI journal)*.

[65] S. Tiptipakorn. A spectral bisection partitioning method for electric power network applications. *PhD dissertation, Dept. of Elec. and Comp. Engineering, Univ. of Wisconsin*, 2001.

[66] L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.

[67] J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM*, 19(2):248–264, 1972.

[68] J. Chlebikova. Approximating the maximally balanced connected partitions problem in graphs. *Information Processing Letters*, 60:225–230, 1996.

[69] E. Amir, R. Krauthgamer, and S. Rao. Constant factor approximation of vertex-cuts in planar graphs. In *Proceedings of the 35th annual ACM symposium on Theory of computing*, STOC '03, pages 90–99, New York, NY, USA, 2003. ACM.

[70] http://snap.stanford.edu/data/.

[71] D. Boley, G. Ranjan, and Z.−L. Zhang. Commute times for a directed graph using an asymmetric laplacian. *Linear Algebra and its Applications*, 435(2):224–242, 2011.

[72] R. Agaev and P. Chebotarev. The matrix of maximum out forests of a digraph and its applications. *Automation and Remote Control*, 61(9):1424–1450, 2000.

[73] P. Chebotarev and E. Shamis. The matrix-forest theorem and measuring relations in small social groups. *Automation and Remote Control*, 58(9):1505–1514, 1997.

[74] V. Rosato and et al. Modelling interdependent infrastructures using interacting dynamical models. *International Journal on Critical Infrastructure*, 4:63–79, 2008.

[75] S. V. Buldyrev and et al. Catastrophic cascade of failures in interdependent networks. *Nature*, 464:1025–1028, 2010.

[76] Gyan Ranjan and Zhi-Li Zhang. A geometric approach to robustness in complex networks. In *Proc. of SIMPLEX'11 (co-located with IEEE ICDCS'11)*, June 2011.

[77] G. Ranjan and Z.-L. Zhang. Geometry of complex networks and structural centrality. *arXiv:1107.0989*, 2011.

[78] M. Balakrishnan, I. Mohomed, and V. Ramasubramanian. Where's that phone? Geolocating IP addresses on 3g networks. *Proc. of ACM Internet Measurement Conference*, 2009.

[79] H. Zang, F. Baccelli, and J. C. Bolot. Bayesian inference for localization in cellular networks. In *Proc. of IEEE INFOCOM 2010*, March 2010.

[80] Q. Gan, J. Attenberg, A. Markowetz, and T. Suel. Analysis of geographic queries in a search engine log. *Proc. of LocWeb*, 2008.

[81] X. Yi, H. Raghavan, and C. Leggetter. Discovering user's specific geo intention in web search. *Proc. of World Wide Web*, 2009.

[82] I. Trestian, S. Ranjan, A. Kuzmanovic, and A. Nucci. Measuring serendipity: Connecting people, locations and interest in a mobile 3G network. *Proc. of ACM Internet Measurement Conference*, 2009.

[83] http://www.roamingzone.com.

[84] C. Rigney, S. Willens, A. Rubens, and W. Simpson. Remote authentication dial in user service (RADIUS). *Internet RFC 2865*, 2000.

[85] C. Rigney. RADIUS accounting. *Internet RFC 2866*, 2000.

[86] http://www.census.gov.

[87] A.-L. Barabasi. The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207–2011, 2005.

[88] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 435:779–782, 2008.

[89] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong. On the levy-walk nature of human mobility: do humans walk like monkeys? In *Proc. IEEE Infocom'08*, 2008.

[90] C. Song, Z. Qu, N. Blumm, and A.-L. Barabasi. Limits of predictability in human mobility. *Science*, 327:1018–1021, 2010.

[91] R. Becker, R. Cáceres, K. Hanson, J. M. Loh, S. Urbanek, A. Varshavsky, and C. Volinsky. Classifying routes using cellular handoff patterns. *Proc. of Netmob 2011*, 2011.

[92] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Varshavsky. Identifying important places in people's lives from cellular network data. *9th International Conference on Pervasive Computing Pervasive*, 2011.

[93] T. Couronné, Z. Smoreda, and A.-M. Olteanu. Chatty mobiles: Individual mobility and communication patterns. *Proc. of Netmob 2011*, 2011.

[94] G. Ranjan, Z.-L. Zhang, S. Ranjan, R. Keralapura, and J. Robinson. Un-zipping cellular infrastructure locations via user geo-intent. *Proc. of Infocom*, 2011.

[95] http://crawdad.cs.dartmouth.edu/.

[96] M. Kim and D. Kotz. Periodic properties of user mobility and access-point popularity. *Journal of Personal and Ubiquitous Computing*, 11(6), August 2007.

[97] L. Song, D. Kotz, R. Jain, and X. He. Evaluating next-cell predictors with extensive WiFi mobility data. *IEEE Transactions on Mobile Computing*, 5(12), December 2006.

[98] M. Kim, D. Kotz, and S. Kim. Extracting a mobility model from real user traces. *Proc. IEEE Infocom'06*, April 2006.

[99] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of human mobility on the design of opportunistic forwarding algorithms. In *Proc. IEEE Infocom'06*, Barcelona, Spain, Apr. 2006.

[100] D. Borckmann, L. Hufnagel, and T. Geisel. The scaling laws of human travel. *Nature*, 439:462–465, 2006.

[101] A. Nanavati, S. Gurumurthy, G. Das, D. Chakraborty, K. Dasgupta, S. Mukherjea, and A. Joshi. On the structural properties of massive telecom call graphs: findings and implications. In *Proc. of 15th ACM Conference on Information and Knowledge Management*, pages 435–444, 2006.

[102] A. Noulas, S. Scellato, R. Lambiotte, M. Pontil, and C. Mascolo. A tale of many cities: universal patterns in human urban mobility. *PLoS One*, 7, 2012.

[103] J. A. Hartigan. Clustering algorithms. *John Wiley & Sons, New York*, 1975.

[104] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience, 1991.

[105] J. Lin. Divergence measures based on the Shannon entropy. *IEEE transactions on information theory*, 37(1):145–151.

[106] P. Chebotarev and E. Shamis. On proximity measures for graph vertices. *Automation and Remote Control*, 59(10):1443–1459, 1998.

Appendix A

Proofs for Chapter 3

A.1 Proof of Theorem 1

Using $\Delta H^{i \to k \to j} = (C_{ik} + C_{kj} - C_{ij})/2$:

$$\Delta H^{(k)} = \frac{1}{2n^2\ Vol(G)} \sum_{i=1}^{n} \sum_{j=1}^{n} C_{ik} + C_{kj} - C_{ij}$$

Observing $C_{xy} = Vol(G)\ (l^+_{xx} + l^+_{yy} - 2l^+_{xy})$ [18] and that $\mathbf{L}^+$ is doubly centered (all rows and columns sum to 0) [8], we obtain the proof.
□

A.2 Proof of Theorem 2

Using $\Delta H^{i \to k \to j} = (C_{ik} + C_{kj} - C_{ij})/2$:

$$\Delta H^{(k)} = \frac{1}{n} \sum_{k=1}^{n} C_{ik} + C_{kj} - C_{ij} - \frac{Tr(\mathbf{L}^+)}{n}$$

Observing $C_{xy} = Vol(G)\ (l^+_{xx} + l^+_{yy} - 2l^+_{xy})$ [18] and that $\mathbf{L}^+$ is doubly centered (all rows and columns sum to 0) [8], we obtain the proof.
□

A.3 Proof of Theorem 3

From [41] we have, $\Delta H^{i\to k\to j} = d(i)^{-1}\ Vol(G)\ U_i^{jk}$. The rest of this proof follows by proving $U_i^{jk} = U_i^{ik} + U_i^{kj} - U_i^{ij}$.

From the *superposition principle* of electrical current, we have $V_x^{xz} = V_y^{xz} + V_y^{zx}$. Therefore,

$$\begin{aligned} V_i^{ik} + V_i^{kj} - V_i^{ij} &= V_j^{ik} + V_j^{ki} + V_i^{kj} - V_k^{ij} + V_k^{ji} \\ &= V_j^{ik} + (V_j^{ki} + V_i^{kj} - V_k^{ij} - V_k^{ji}) \end{aligned}$$

From the *reciprocity principle*, $V_z^{xy} = V_x^{zy}$. Therefore, $V_i^{ik} + V_i^{kj} - V_i^{ij} = V_i^{jk}$. Multiplying by $d(i)$ on both sides we obtain the proof.

□

A.4 Proof of Theorem 4

The following result, due to Chebotarev et al. [73, 106], forms the basis of our proof. Let $\mathcal{F}_x$ be the set of spanning rooted forests of $G(V, E)$ with x edges. Precisely, $F_x \in \mathcal{F}_x$, is a spanning acyclic subgraph of G with the same node set as G and is composed of exactly $n - x$ trees with one node marked as a *root* in each of these x trees. Let $\mathcal{F}_x^{ii}$, be the subset of $\mathcal{F}_x$ in which node i is the root of the tree in which it belongs. Then,

$$l_{ii}^{+} = \frac{\varepsilon(\mathcal{F}_{n-2}^{ii}) - \frac{1}{n}\varepsilon(\mathcal{F}_{n-2})}{\varepsilon(\mathcal{F}_{n-1})} \tag{A.1}$$

Here, $\varepsilon(\cdot)$ simply represents the cardinality of the input set (see [73, 106] for details). It is easy to see that $\varepsilon(\mathcal{F}_{n-1}) = n|T(G)|$, as a spanning forest with $n-1$ edges is a spanning tree, and each spanning tree has exactly n possible choices of roots. Also, $\varepsilon(\mathcal{F}_{n-2})$ and $\varepsilon(\mathcal{F}_{n-1})$ are invariants over the set of vertices $V(G)$ for a given graph. Hence, $l_{ii}^{+} \propto \varepsilon(\mathcal{F}_{n-2}^{ii})$. The rest of the proof follows from the results of the following lemmas:

Lemma 2 *Let $\mathcal{F}_{n-2|P}^{ii}$ be the set of spanning forests with $n-2$ edges (or exactly two trees) rooted at node i in a given bi-partition $P = (S, S')$ and $\mathcal{T}(S)$, $\mathcal{T}(S')$ be the set of spanning trees in S and S' respectively. If $i \in V(S)$ then,*

$$\varepsilon(\mathcal{F}_{n-2|P}^{ii}) = |\mathcal{T}(S)|\ |\mathcal{T}(S')|\ |V(S')|$$

Proof : Let $T_1 \in \mathcal{T}(S)$ and $T_2 \in \mathcal{T}(S')$. Clearly, $|E(T_1)| = |V(S)| - 1$ and $|E(T_2)| = |V(S')| - 1$. As $|V(S)| + |V(S')| = |V(G)|$ and $|E(T_1) \cap E(T_2)| = 0$, $|E(T_1) \cup E(T_2)| = |V(S)| - 1 + |V(S')| - 1 = n - 2$. Each such pair (T_1, T_2) is a spanning forest of $n - 2$ edges. Given i is the root of T_1 in S, we can choose $|V(S')|$ roots for T_2 in S'. There being $|\mathcal{T}(S)|\ |\mathcal{T}(S')|$ such pairs: $\varepsilon(\mathcal{F}^{ii}_{n-2|P}) = |\mathcal{T}(S)|\ |\mathcal{T}(S')|\ |V(S')|$.

□

By symmetry, for $j \in V(S')$:

$$\varepsilon(\mathcal{F}^{jj}_{n-2|P}) = |\mathcal{T}(S)|\ |\mathcal{T}(S')|\ |V(S)|$$

Lemma 3 *Given $\mathcal{P}(G)$, the set of all bi-partitions of G:*

$$\varepsilon(\mathcal{F}^{ii}_{n-2}) = \sum_{P \in \mathcal{P}(G)}^{i \in V(S)} |\mathcal{T}(S)||\mathcal{T}(S')||V(S')|$$

Proof: By definition, $\forall F \in \mathcal{F}_{n-2}$, F belongs to exactly one of the partitions of G. Hence, $\mathcal{F}^{ii}_{n-2} = \coprod_{P \in \mathcal{P}} \varepsilon(\mathcal{F}^{ii}_{n-2|P})$. As the RHS is a disjoint union, counting members on both sides we obtain the proof.

□

Evidently, combining the results of the two lemmas above, we obtain the proof for Theorem 4.

□

A.5 Proof of Theorem 5

The proof follows exactly in the same logical order as that of the proof of Theorem 5.

□

A.6 Proof of Theorem 6

Once again, the proof follows by summing up the proof of Theorem 4 $\forall i \in V(G)$.

□

A.7 Proof of Corollary 1

The proof follows simply by making the following observations about trees: $\varepsilon(\mathcal{F}_{n-1}) = n \cdot 1 = n$. Also,

$$\varepsilon(\mathcal{F}^{ii}_{n-2}) = \sum_{P(S,S')\in\mathcal{P}(G)}^{i\in V(S)} |V(S')| \tag{A.2}$$

and

$$\varepsilon(\mathcal{F}_{n-2}) = \sum_{P(S,S')\in\mathcal{P}(G)} |V(S)||V(S')| = \sum_{P(S,S')\in\mathcal{P}(G)} (n - |V(S')|)|V(S')| \tag{A.3}$$

Thus substituting these values in $(A.1)$, we obtain the proof.

□

Appendix B

Proofs for Chapter 4

B.1 Proof of Theorem 7

Given, $\mathbf{L} \in \Re^{n \times n}$, we note that $\mathbf{L} \cdot \mathbf{1} = \mathbf{0}$ and $\mathbf{1}' \cdot \mathbf{L} = \mathbf{0}'$, where $\mathbf{1} \in \Re^n$ and $\mathbf{0} \in \Re^n$ are vectors of length n containing all $1's$ and $0's$ respectively. From [71], we have:

$$\mathbf{L}(\{\overline{n}\}, \{\overline{n}\})^{-1} = [\mathbf{I}_{n-1}, -\mathbf{v}]\ \mathbf{L}^{+} \begin{bmatrix} \mathbf{I}_{n-1} \\ -\mathbf{u}' \end{bmatrix} \tag{B.1}$$

where $\mathbf{I}_{n-1}$ is the identity matrix of dimension $(n-1 \times n-1)$ and $\mathbf{u} = \mathbf{v} = \mathbf{1} \in \Re^{n-1}$ are vectors of all $1's$ of length $n-1$. Expanding we obtain the following scalar form:

$$[\mathbf{L}(\{\overline{n}\}, \{\overline{n}\})^{-1}]_{xy} = l^{+}_{xy} - l^{+}_{xn} - l^{+}_{ny} + l^{+}_{nn} \tag{B.2}$$

□

B.2 Proof of Lemma 1

Given, $\mathbf{L}^{+}$ is symmetric and doubly centered and $\Omega_{xy} = l^{+}_{xx} + l^{+}_{yy} - l^{+}_{xy} - l^{+}_{yx}$, we have:

$$\begin{aligned}
\sum_{z=1}^{n} \Omega_{xz} + \Omega_{zy} - \Omega_{xy} &= \sum_{z=1}^{n} [(l^{+}_{xx} + l^{+}_{zz} - 2l^{+}_{xz}) + (l^{+}_{zz} + l^{+}_{yy} - 2l^{+}_{zy}) - (l^{+}_{xx} + l^{+}_{yy} - 2l^{+}_{xy})] \\
&= 2 \sum_{z=1}^{n} [l^{+}_{zz} + l^{+}_{xy}] \\
&= 2n\ l^{+}_{xy} + 2\ Tr(\mathbf{L}^{+})
\end{aligned}$$

Rearranging terms, we get:

$$l^+_{xy} = \frac{1}{2n}\left(\sum_{z=1}^{n}\Omega_{xz} + \Omega_{zy} - \Omega_{xy}\right) - \frac{1}{n}\,Tr(\mathbf{L}^+) \tag{B.3}$$

Substituting $Tr(\mathbf{L}^+) = \frac{1}{2n}\sum_{x=1}^{n}\sum_{y=1}^{n}\Omega_{xy}$, in the expression above, we obtain the proof.

□

B.3 Proof of Corollary 3

Given, a star S_p with node 1 as root and nodes $\{2, 3, ..., p\}$ as leaves, we have:

$$\forall x : 2 \le x \le p, \quad \Omega^{S_p}_{1x} = 1 \qquad \& \qquad \forall x \ne y : 2 \le x, y \le p, \quad \Omega^{S_p}_{xy} = 2 \tag{B.4}$$

Therefore,

$$\sum_{x=1}^{p}\sum_{y=1}^{p}\Omega^{S_p}_{xy} = 2(p-1)^2$$

Also,

$$\sum_{z=1}^{p}\Omega^{S_p}_{1z} + \Omega^{S_p}_{zp} - \Omega^{S_p}_{1p} = 2(p-2) \quad \& \quad \forall x \ne y : 2 \le x, y \le p, \quad \sum_{z=1}^{p}\Omega^{S_p}_{xz} + \Omega^{S_p}_{zy} - \Omega^{S_p}_{xy} = 2(p-3)$$

Substituting for these values in Lemma 1, and noting that $\mathbf{L}^+_{S_p}$ is doubly-centered, we obtain the proof.

□

B.4 Proof of Corollary 4

Given a clique K_p of order p, $\forall x \ne y : 1 \le x, y \le p, \Omega_{xy} = \frac{2}{p}$ [1]. Therefore,

$$\sum_{x=1}^{p}\sum_{y=1}^{p}\Omega^{K_p}_{xy} = 2(p-1)$$

Also,

$$\forall x : 1 \le x \le p, \quad \sum_{z=1}^{p}\Omega^{K_p}_{xz} + \Omega^{K_p}_{zx} - \Omega^{K_p}_{xx} = \frac{4(p-1)}{p}$$

Substituting in Lemma 1, we obtain: $l^+_{xx} = \frac{p-1}{p^2}$ and, from the fact that $\mathbf{L}^+_{K_p}$ is doubly-centered, $\forall x \neq y : 1 \leq x, y \leq p, \;\; l^+_{xy} = -\frac{l^+_{xx}}{p-1} = -\frac{1}{p^2}$.
□

B.5 Proof of Theorem 8

We present the proofs for the following two cases: (a) $x, y \in V_1(G_1)$ and, (b) $x \in V_1(G_1)$ and $y \in V_2(G_2)$. It is obvious that the other two cases, viz. (c) $x, y \in V_2(G_2)$ and (d) $x \in V_2(G_2)$ and $y \in V_1(G_1)$, follow from symmetry. But first we must express $Tr(\mathbf{L}^+_{G_3})$ as a function of $(Tr(\mathbf{L}^+_{G_1}), Tr(\mathbf{L}^+_{G_2}))$, which is useful to us in both cases.

Lemma 4 *For two disjoint simple, connected, undirected graphs $G_1(V_1, E_1)$ & $G_2(V_2, E_2)$, let $G_3(V_3, E_3)$ be the graph resulting from the first join between G_1 and G_2 by means of introducing an edge $e_{ij} : i \in V_1(G_1), j \in V_2(G_2)$. Then,*

$$Tr(\mathbf{L}^+_{G_3}) = Tr(\mathbf{L}^+_{G_1}) + Tr(\mathbf{L}^+_{G_2}) + \frac{n_1 n_2}{n_1 + n_2}\left(l^{+(1)}_{ii} + l^{+(2)}_{jj} + \omega_{ij}\right) \tag{B.5}$$

Proof of Lemma 4

For an arbitrary node $x \in V_1(G_1)$:

$$\begin{aligned} \Omega^{G_3}_{xy} &= \Omega^{G_1}_{xy}, && if\ y \in V_1(G_1) \\ &= \Omega^{G_1}_{xi} + \omega_{ij} + \Omega^{G_2}_{jy}, && if\ y \in V_2(G_2) \end{aligned}$$

Therefore,

$$\begin{aligned} \sum_{y \in V_3(G_3)} \Omega^{G_3}_{xy} &= \sum_{y \in V_1(G_1)} \Omega^{G_1}_{xy} + \sum_{y \in V_2(G_2)} \left(\Omega^{G_1}_{xi} + \omega_{ij} + \Omega^{G_2}_{jy}\right) \\ &= n_1\ l^{+(1)}_{xx} + +n_2\left(l^{+(1)}_{xx} + l^{+(1)}_{ii} - 2\ l^{+(1)}_{xi}\right) + n_2\ \omega_{ij} + n_2\ l^{+(2)}_{jj} + \delta \end{aligned}$$

where $\delta = Tr(\mathbf{L}^+_{G_1}) + Tr(\mathbf{L}^+_{G_2})$. Summing up over all nodes $x \in V_1(G_1)$:

$$\sum_{x \in V_1(G_1)} \sum_{y \in V_3(G_3)} \Omega^{G_3}_{xy} = (2\ n_1 + n_2)\ Tr(\mathbf{L}^+_{G_1}) + n_1 n_2\ (l^{+(1)}_{ii} + l^{+(2)}_{jj} + \omega_{ij}) + n_1\ Tr(\mathbf{L}^+_{G_2})$$

By symmetry,

$$\sum_{x \in V_2(G_2)} \sum_{y \in V_3(G_3)} \Omega^{G_3}_{xy} = (2\ n_2 + n_1)\ Tr(\mathbf{L}^+_{G_2}) + n_1 n_2\ (l^{+(1)}_{ii} + l^{+(2)}_{jj} + \omega_{ij}) + n_2\ Tr(\mathbf{L}^+_{G_1})$$

Therefore,

$$\begin{aligned}\sum_{x\in V_3(G_3)}\sum_{y\in V_3(G_3)}\Omega_{xy}^{G_3} &= \sum_{x\in V_1(G_1)}\sum_{y\in V_3(G_3)}\Omega_{xy}^{G_3}+\sum_{x\in V_2(G_2)}\sum_{y\in V_3(G_3)}\Omega_{xy}^{G_3}\\ &= 2\ (n_1+n_2)\delta+2\ n_1n_2\left(l_{ii}^{+(1)}+l_{jj}^{+(2)}+\omega_{ij}\right)\end{aligned}$$

where $\delta = Tr(\mathbf{L}_{G_1}^{+}) + Tr(\mathbf{L}_{G_2}^{+})$. Substituting $Tr(\mathbf{L}_{G_3}^{+}) = \dfrac{1}{2n_3}\displaystyle\sum_{x\in V_3(G_3)}\sum_{y\in V_3(G_3)}\Omega_{xy}^{G_3}$, we obtain the result.

□

Rest of the Proof of Theorem 8

Case a: $x, y \in V_1(G_1)$

From Lemma 1, we have:

$$l_{xy}^{+(3)} = \frac{1}{2n_3}\left(\sum_{z\in V_3(G_3)}\Omega_{xz}^{G_3}+\Omega_{zy}^{G_3}-\Omega_{xy}^{G_3}\right)-\frac{1}{n_3}Tr(\mathbf{L}_{G_3}^{+}) \tag{B.6}$$

For the triangle inequality in the RHS above:

$$\begin{aligned}\sum_{z\in V_3(G_3)}\Omega_{xz}^{G_3} &= \sum_{z\in V_1(G_1)}\Omega_{xz}^{G_1}+\sum_{z\in V_2(G_2)}\left(\Omega_{xi}^{G_1}+\omega_{ij}+\Omega_{jz}^{G_2}\right)\\ &= (n_1+n_2)\ l_{xx}^{+(1)}+n_2\ l_{ii}^{+(1)}-2n_2\ l_{xi}^{+(1)}+n_2\ \omega_{ij}+n_2\ l_{jj}^{+(2)}+\delta\end{aligned}$$

where $\delta = Tr(\mathbf{L}_{G_1}^{+}) + Tr(\mathbf{L}_{G_2}^{+})$. By symmetry,

$$\sum_{z\in V_3(G_3)}\Omega_{zy}^{G_3} = (n_1+n_2)\ l_{yy}^{+(1)}+n_2\ l_{ii}^{+(1)}-2n_2\ l_{yi}^{+(1)}+n_2\ \omega_{ij}+n_2\ l_{jj}^{+(2)}+\delta$$

Finally, for the last of the three terms:

$$\sum_{z\in V_3(G_3)}\Omega_{xy}^{G_3} = (n_1+n_2)\left(l_{xx}^{+(1)}+l_{yy}^{+(1)}-2\ l_{xy}^{+(1)}\right)$$

Summing the three individual terms along with the value of $Tr(\mathbf{L}_{G_3}^{+})$ from $(B.5)$ and substituting the result in $(B.6)$, we obtain the proof.

Case b: $x \in V_1(G_1)$ **and** $y \in V_2(G_2)$

Once again,

$$l_{xy}^{+(3)} = \frac{1}{2n_3}\left(\sum_{z\in V_3(G_3)} \Omega_{xz}^{G_3} + \Omega_{zy}^{G_3} - \Omega_{xy}^{G_3}\right) - \frac{1}{n_3} Tr(\mathbf{L}_{G_3}^{+}) \tag{B.7}$$

For the triangle inequality in the RHS above:

$$\begin{aligned}
\sum_{z\in V_3(G_3)} \Omega_{xz}^{G_3} &= \sum_{z\in V_1(G_1)} \Omega_{xz}^{G_1} + \sum_{z\in V_2(G_2)} \left(\Omega_{xi}^{G_1} + \omega_{ij} + \Omega_{jz}^{G_2}\right) \\
&= (n_1 + n_2)\ l_{xx}^{+(1)} + n_2\ l_{ii}^{+(1)} - 2n_2\ l_{xi}^{+(1)} + n_2\ \omega_{ij} + n_2\ l_{jj}^{+(2)} + \delta
\end{aligned}$$

where $\delta = Tr(\mathbf{L}_{G_1}^{+}) + Tr(\mathbf{L}_{G_2}^{+})$. Similarly,

$$\begin{aligned}
\sum_{z\in V_3(G_3)} \Omega_{zy}^{G_3} &= \sum_{z\in V_1(G_1)} \left(\Omega_{yj}^{G_2} + \omega_{ij} + \Omega_{iz}^{G_1}\right) + \sum_{z\in V_2(G_2)} \Omega_{zy}^{G_1} \\
&= (n_1 + n_2)\ l_{yy}^{+(2)} + n_1\ l_{jj}^{+(2)} - 2n_1\ l_{jy}^{+(2)} + n_1\ \omega_{ij} + n_1\ l_{ii}^{+(1)} + \delta
\end{aligned}$$

And finally,

$$\begin{aligned}
\sum_{z\in V_3(G_3)} \Omega_{xy}^{G_3} &= \sum_{z\in V_3(G_3)} \Omega_{xi}^{G_3} + \omega_{ij} + \Omega_{jy}^{G_3} \\
&= (n_1 + n_2)\left(l_{xx}^{+(1)} + l_{ii}^{+(1)} - 2\ l_{xi}^{+(1)} + \omega_{ij} + l_{yy}^{+(2)} + l_{jj}^{+(2)} - 2\ l_{jy}^{+(2)}\right)
\end{aligned}$$

Summing the three individual terms along with the value of $Tr(\mathbf{L}_{G_3}^{+})$ from $(B.5)$ and substituting the result in $(B.7)$, we obtain the proof.
□

B.6 Proof of Theorem 9

We prove Theorem 9 in two steps. First, in the following lemma, we provide a general result for a perturbation of a positive semi-definite matrix, which is then used to prove the overall theorem.

Lemma 5 *Given $V \in \Re^{n\times n}$ is a symmetric, positive semi-definite matrix and $X \in \Re^{n\times q}$ a perturbation matrix, such that $(I+\alpha X^T V^+ X)$ has an inverse where $\alpha = \{1, -1\}$, and $VV^+X = X$, the following holds:*

$$(V + \alpha XX^T)^+ = V^+ - \alpha V^+ X(I + \alpha X^T V^+ X)^{-1} X^T V^+ \tag{B.8}$$

Proof of Lemma 5:

Observe that the positive semi-definiteness of V guarantees that $I + X^T V^+ X$ has an inverse.

Let $W = V^+ - \alpha V^+ X(I + \alpha X^T V^+ X)^{-1} X^T V^+$. Therefore,

$$\begin{aligned}
(V + \alpha XX^T)W &= VV^+ - \alpha VV^+ X(I + \alpha X^T V^+ X)^{-1} X^T V^+ \\
&\quad + \alpha XX^T V^+ - XX^T V^+ X(I + \alpha X^T V^+ X)^{-1} X^T V^+ \\
&= VV^+ - \alpha X(I + \alpha X^T V^+ X)^{-1} X^T V^+ \\
&\quad + \alpha X \left[I - \alpha X^T V^+ X(I + \alpha X^T V^+ X)^{-1}\right] X^T V^+ \\
&= VV^+ - \alpha X(I + \alpha X^T V^+ X)^{-1} X^T V^+ \\
&\quad + \alpha X \left[((I + \alpha X^T V^+ X) - \alpha X^T V^+ X)(I + \alpha X^T V^+ X)^{-1}\right] X^T V^+ \\
&= VV^+ - \alpha X(I + \alpha X^T V^+ X)^{-1} X^T V^+ \\
&\quad + \alpha X \left[(I + \alpha X^T V^+ X)^{-1}\right] X^T V^+ \\
&= VV^+ - \alpha X(I + \alpha X^T V^+ X)^{-1} X^T V^+ \\
&\quad + \alpha X \left[(I + \alpha X^T V^+ X)^{-1}\right] X^T V^+ \\
&= VV^+
\end{aligned}$$

From this identity, and the symmetry of V, W & (XX^T), it follows easily that W satisfies the four conditions required for a Moore-Penrose pseudo-inverse.

□

Rest of the Proof of Theorem 9

Note that the firing of the edge e_{ij} in $G_1(V_1, E_1)$ to obtain $G_2(V_2, E_2)$, results in the following scalar relationships between the Laplacians of the two graphs:

$$a.\ \ [\mathbf{L}_{G_2}]_{ij} = [\mathbf{L}_{G_2}]_{ji} = -\frac{1}{\omega_{ij}}, \quad b.\ \ [\mathbf{L}_{G_2}]_{ii} = [\mathbf{L}_{G_1}]_{ii} + \frac{1}{\omega_{ij}}, \quad c.\ \ [\mathbf{L}_{G_2}]_{jj} = [\mathbf{L}_{G_1}]_{jj} + \frac{1}{\omega_{ij}} \tag{B.9}$$

For ease of exposition, we permute the rows and columns in $\mathbf{L}_{G_1}$ and $\mathbf{L}_{G_2}$ in such a way that $i = 1$ and $j = 2$. The above perturbations can then be rewritten as:

$$\mathbf{L}_{G_2} = \mathbf{L}_{G_1} + \frac{1}{\omega_{12}} \begin{bmatrix} 1 & -1 & 0 & ... & 0 \\ -1 & 1 & 0 & ... & 0 \\ 0 & 0 & 0 & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & ... & 0 \end{bmatrix} \tag{B.10}$$

$\mathbf{L}_{G_2}$ is therefore a sum of $\mathbf{L}_{G_1}$, a real, symmetric, positive semi-definite matrix, and a $rank(1)$ perturbation matrix, referred to henceforth as Y. It is easy to see that for a simple, connected, undirected graph $G_1(V_1, E_1)$, $\mathbf{L}^+_{G_1}$ satisfies all the preconditions in Lemma 5. Substituting $V = \mathbf{L}_{G_1}$, $\alpha = 1$ and $X = \sqrt{Y} = X^T$, in Lemma 5, we get:

$$\mathbf{L}^+_{G_2} = (\mathbf{L}_{G_1} + XX^T)^+ = \mathbf{L}^+_{G_1} - \mathbf{L}^+_{G_1} X \ (I + X\mathbf{L}^+_{G_1} X)^{-1} \ X\mathbf{L}^+_{G_1} \tag{B.11}$$

All that remains now is to obtain the scalar form for the term:

$$\mathbf{L}^+_{G_1} X \ (I + X\mathbf{L}^+_{G_1} X)^{-1} \ X\mathbf{L}^+_{G_1}$$

Note:

$$\mathbf{L}^+_{G_1} X = \frac{1}{\sqrt{2\ \omega_{12}}} \left[\begin{array}{cc|ccc} l^{+(1)}_{11} - l^{+(1)}_{12} & -(l^{+(1)}_{11} - l^{+(1)}_{12}) & 0 & ... & 0 \\ l^{+(1)}_{21} - l^{+(1)}_{22} & -(l^{+(1)}_{21} - l^{+(1)}_{22}) & 0 & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ l^{+(1)}_{n-1,1} - l^{+(1)}_{n-1,2} & -(l^{+(1)}_{n-1,1} - l^{+(1)}_{n-1,2}) & 0 & ... & 0 \\ l^{+(1)}_{n1} - l^{+(1)}_{n2} & -(l^{+(1)}_{n1} - l^{+(1)}_{n2}) & 0 & ... & 0 \end{array}\right] \tag{B.12}$$

Similarly,

$$X\mathbf{L}^+_{G_1} = \frac{1}{\sqrt{2\ \omega_{12}}} \left[\begin{array}{cccc} l^{+(1)}_{11} - l^{+(1)}_{21} & l^{+(1)}_{12} - l^{+(1)}_{22} & ... & l^{+(1)}_{1n} - l^{+(1)}_{2n} \\ -(l^{+(1)}_{11} - l^{+(1)}_{21}) & -(l^{+(1)}_{12} - l^{+(1)}_{22}) & ... & -(l^{+(1)}_{1n} - l^{+(1)}_{2n}) \\ \hline 0 & 0 & ... & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & ... & 0 \end{array}\right] \tag{B.13}$$

Or,

$$X\mathbf{L}_{G_1}^{+}X = \left[\begin{array}{cc|ccc} \frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & -\frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & 0 & ... & 0 \\ -\frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & \frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & 0 & ... & 0 \\ \hline 0 & 0 & 0 & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & ... & 0 \end{array}\right] \tag{B.14}$$

where $\Omega_{12}^{G_1} = l_{11}^{+(1)} + l_{22}^{+(1)} - l_{12}^{+(1)} - l_{21}^{+(1)}$, which yields:

$$(I + X\mathbf{L}_{G_1}^{+}X)^{-1} = \left[\begin{array}{cc|ccc} 1+\frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & -\frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & 0 & ... & 0 \\ -\frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & 1+\frac{\Omega_{12}^{G_1}}{2\ \omega_{12}} & 0 & ... & 0 \\ \hline 0 & 0 & & & \\ \vdots & \vdots & & I_{n-2,n-2} & \\ 0 & 0 & & & \end{array}\right]^{-1} \tag{B.15}$$

$$= \left[\begin{array}{cc|ccc} \frac{2\omega_{12}+\Omega_{12}^{G_1}}{2(\omega_{12}+\Omega_{12}^{G_1})} & \frac{\Omega_{12}^{G_1}}{2(\omega_{12}+\Omega_{12}^{G_1})} & 0 & ... & 0 \\ \frac{\Omega_{12}^{G_1}}{2(\omega_{12}+\Omega_{12}^{G_1})} & \frac{2\omega_{12}+\Omega_{12}^{G_1}}{2(\omega_{12}+\Omega_{12}^{G_1})} & 0 & ... & 0 \\ \hline 0 & 0 & & & \\ \vdots & \vdots & & I_{n-2,n-2} & \\ 0 & 0 & & & \end{array}\right] \tag{B.16}$$

Multiplying on the left and right sides of the RHS above with $\mathbf{L}_{G_1}^{+}X$ and $X\mathbf{L}_{G_1}^{+}$ respectively, the following scalar form is obtained:

$$l_{xy}^{+(2)} = l_{xy}^{+(1)} - \frac{(l_{x1}^{+(1)} - l_{x2}^{+(1)})(l_{1y}^{+(1)} - l_{2y}^{+(1)})}{\omega_{12} + \Omega_{12}^{G_1}} \tag{B.17}$$

Substituting $i = 1$ and $j = 2$ back into the equation above, we obtain the proof.

□

B.7 Proof of Corollary 5

Noting $\Omega_{xy}^{G_2} = l_{xx}^{+(2)} + l_{yy}^{+(2)} - l_{xy}^{+(2)} - l_{yx}^{+(2)}$ and substituting into the result of Theorem 9, we obtain the proof.

□

B.8 Proof of Theorem 10

Deleting a non-bridge edge $e_{ij} \in E_1(G_1)$ from $G_1(V_1, E_1)$ to obtain $G_2(V_2, E_2)$, results in the following scalar relationships between the Laplacians of the two graphs:

$$a.\quad [\mathbf{L}_{G_2}]_{ij} = [\mathbf{L}_{G_2}]_{ji} = \frac{1}{\omega_{ij}}, \qquad b.\quad [\mathbf{L}_{G_2}]_{ii} = [\mathbf{L}_{G_1}]_{ii} - \frac{1}{\omega_{ij}}, \qquad c.\quad [\mathbf{L}_{G_2}]_{jj} = [\mathbf{L}_{G_1}]_{jj} - \frac{1}{\omega_{ij}} \tag{B.18}$$

Once again, for convenience, we rearrange the rows and columns of $\mathbf{L}_{G_1}$ and $\mathbf{L}_{G_2}$ in such a way that $i = 1$ and $j = 2$. Thus,

$$\mathbf{L}_{G_2} = \mathbf{L}_{G_1} - \frac{1}{\omega_{12}} \begin{bmatrix} 1 & -1 & 0 & ... & 0 \\ -1 & 1 & 0 & ... & 0 \\ 0 & 0 & 0 & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & ... & 0 \end{bmatrix} \tag{B.19}$$

The rest of the proof follows as in the proof of Theorem 9, with the following modification. Substituting $V = \mathbf{L}_{G_1}$, $\alpha = -1$ and $X = \sqrt{Y} = X^T$, in Lemma 5, we get:

$$\mathbf{L}^+_{G_2} = (\mathbf{L}_{G_1} - XX^T)^+ = \mathbf{L}^+_{G_1} + \mathbf{L}^+_{G_1} X \ (I + X\mathbf{L}^+_{G_1} X)^{-1} \ X\mathbf{L}^+_{G_1} \tag{B.20}$$

which yields the following scalar form:

$$l^{+(2)}_{xy} = l^{+(1)}_{xy} + \frac{(l^{+(1)}_{x1} - l^{+(1)}_{x2})(l^{+(1)}_{1y} - l^{+(1)}_{2y})}{\omega_{12} - \Omega^{G_1}_{12}} \tag{B.21}$$

Substituting $i = 1$ and $j = 2$ back into the equation above, we obtain the proof.
□

B.9 Proof of Corollary 6

Noting $\Omega^{G_2}_{xy} = l^{+(2)}_{xx} + l^{+(2)}_{yy} - l^{+(2)}_{xy} - l^{+(2)}_{yx}$ and substituting into the result of Theorem 10, we obtain the proof.
□

B.9.1 Proof of Theorem 11

We present the proof for the case: $x, y \in V_2(G_2)$ as the other case follows by symmetry. Once again, we need a lemma to determine $Tr(\mathbf{L}^+_{G_2})$ in terms of the elements of $\mathbf{L}^+_{G_1}$.

Lemma 6 *Let $G_1(V_1, E_1)$ be a simple, connected, unweighted graph with a bridge edge $e_{ij} : i \in E_1(G_1)$ which upon deletion produces two disjoint simple graphs $G_2(V_2, E_2)$ and $G_3(V_3, E_3)$. Then,*

$$Tr(\mathbf{L}^+_{G_2}) = \sum_{x \in V_2(G_2)} l^{+(1)}_{xx} - \frac{1}{n_2} \sum_{x \in V_2(G_2)} \sum_{y \in V_2(G_2)} l^{+(1)}_{xy} \tag{B.22}$$

Proof of Lemma 6

The proof follows simply by observing $\Omega^{G_2}_{xy} = \Omega^{G_3}_{xy}$, $\forall (x, y) \in V_2(G_2) \times V_2(G_2)$ and substituting values in terms of the elements of $\mathbf{L}^+_{G_1}$.
□

Rest of the Proof of Theorem 11

Follows similarly from the triangle inequality in Lemma 1, by confining to node pairs $(x, y) \in V_2(G_2) \times V_2(G_2)$, and then substituting the result of Lemma 6 and other relevant effective resistance values in terms of $\mathbf{L}^+_{G_1}$.
□

Appendix C

Proofs for Chapter 5

C.1 Proof of Theorem 12

Recall, by Lemma 4, for two disjoint simple, connected, undirected graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, let $G_3(V_3, E_3)$ be the graph resulting from the first join between G_1 and G_2 by means of introducing an edge $e_{ij} : i \in V_1(G_1), j \in V_2(G_2)$. Then,

$$Tr(\mathbf{L}^+_{G_3}) = Tr(\mathbf{L}^+_{G_1}) + Tr(\mathbf{L}^+_{G_2}) + \frac{n_1 n_2}{n_1 + n_2}\left(l^{+(1)}_{ii} + l^{+(2)}_{jj} + \omega_{ij}\right) \tag{C.1}$$

The process of merging two nodes is then the equivalent of introducing an infinitesimally small resistance $\varepsilon = \omega_{ij}$ between $i \in V_1(G_1)$ and $j \in V_2(G_2)$. It is easy to see that $G_3 = G_c$ here is then the coupled network post first join.

□

C.2 Proof of Theorem 13

Recall, by Theorem 9

$$l^{+(2)}_{xy} = l^{+(1)}_{xy} - \frac{(l^{+(1)}_{xi} - l^{+(1)}_{xj})(l^{+(1)}_{iy} - l^{+(1)}_{jy})}{\omega_{ij} + \Omega^{G_1}_{ij}} \tag{C.2}$$

Substituting $y = x$, summing up $\forall x \in V_{c1}(G_{c1})$ and substituting $\omega^{G_{c1}}_{ij} = \varepsilon$, we obtain the proof. Once again, we note that coupling i and j, is the same as introducing an infinitesimally small resistance between the two nodes.

□