ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

EXTENDING OPENSTREETMAP USAGE FOR ADVANCED ROUTING SERVICES

Ph.D. THESIS

Mohammed ZIA

Department of Geomatics Engineering

Geomatics Engineering Programme

APRIL 2018

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

EXTENDING OPENSTREETMAP USAGE FOR ADVANCED ROUTING SERVICES

Ph.D. THESIS

Mohammed ZIA
(501132610)

Department of Geomatics Engineering

Geomatics Engineering Programme

Thesis Advisor: Prof. Dr. Dursun Zafer ŞEKER
Co-advisor: Prof. Dr. Ziyadin ÇAKIR

APRIL 2018

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

OPENSTREETMAP KULLANIMINI İLERİ YÖNLENDİRME HİZMETLERİ İÇİN GELİŞTİRMEK

DOKTORA TEZİ

Mohammed ZIA
(501132610)

Geomatik Mühendisliği Anabilim Dalı

Geomatik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Dursun Zafer ŞEKER
Eş Danışman: Prof. Dr. Ziyadin ÇAKIR

NİSAN 2018

Mohammed ZIA, a Ph.D. student of ITU Graduate School of Science Engineering and Technology 501132610 successfully defended the thesis entitled "EXTENDING OPENSTREETMAP USAGE FOR ADVANCED ROUTING SERVICES", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. Dursun Zafer ŞEKER** Istanbul Technical University	.....
Co-advisor :	**Prof. Dr. Ziyadin ÇAKIR** Istanbul Technical University	.....
Jury Members :	**Prof. Dr. Azime TEZER** Istanbul Technical University	.....
	Prof. Dr. Cem GAZİOĞLU Istanbul University	.....
	Assoc. Prof. Dr. Aslı DOĞRU Bogazici University	.....
	Assist. Prof. Dr. Ahmet Özgür DOĞRU Istanbul Technical University	.....
	Assist. Prof. Dr. Hüseyin Can ÜNEN Maltepe University	.....

Date of Submission : **12 March 2018**
Date of Defense : **02 April 2018**

To everyone,

FOREWORD

I would like to convey my sincere thanks to both of my supervisors, Dr. Dursun Zafer ŞEKER and Dr. Ziyadin ÇAKIR, for their immense support, patience, and trust on me. Without their able guidance, this PhD work would not have been possible to compile. Also, my sincere respect to all jury members, Prof. Dr. Azime TEZER, Prof. Dr. Bülent BAYRAM and Prof. Dr. Cem GAZİOĞLU, for their time and guidance in this pursuit. Thanks ton to all!

The research presented in this thesis is primarily funded by The Scientific and Technological Research Council of Turkey under 2215 - Graduate Scholarship Programme for International Students (TUBITAK, *URL*: `https://www.tubitak.gov.tr/en`). I am also thankful to the Istanbul Technical University - GIS Research and Innovation Center for additional support.

APRIL 2018

Mohammed ZIA
(Geospatial Engineer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

E-GTSP	**:** Equality Generalised Travelling Salesman Problem
FOSS4G	**:** Free and Open Source Software for Geospatial
GIS	**:** Geographic Information Systems
GTSP	**:** Generalised Travelling Salesman Problem
GTSPLIB	**:** Generalised Travelling Salesman Problem Library
OSM	**:** Open Street Map
TSP	**:** Travelling Salesman Problem
VGI	**:** Volunteered Geographic Information
VRS	**:** Vehicle RoutingService

LIST OF TABLES

Page

LIST OF FIGURES

Page

EXTENDING OPENSTREETMAP USAGE FOR ADVANCED ROUTING SERVICES

SUMMARY

The last two decades have evidently witnessed a sudden boom in Information and Communication Technologies (ICT), which include any communication device or application like cellular phones, computer hardware, satellite systems and so on. This has resulted into a massive flooding of geo-tagged information, efficiently being handled by specialized Information Systems, aka Geographic Information System (GIS). This massive or big data has contemporaneously led to the opportunities of various geo-services, targeted for specific use-cases. However, because of primarily being collected, managed, stored and distributed by Governmental and National Mapping Agencies, there has always been a data access check for general users and other low/mid class service-providers, except for those who can afford data's lofty pricing. The advent of Web2.0 technology, in this sense, has proven to be a game changer, by allowing any end-user to generate, upload and disseminate his/her own geo-data. Systems designed for these kind of data management are termed as Volunteered Geographic Information (VGI) (introduced by Michael F. Goodchild). One such famous, if not the famous, VGI project is OpenStreetMap (OSM), founded in 2004 by Steve Coast. Since genesis, this open-geo-data project has gone too far with worldwide 9 billion mapped locations (nodes), 0.4 billion traced lines (ways), and 4.5 million sketched polygons (relations), generated by 3.1 million registered users, approximately (November 2016 stats). It has clearly generated huge opportunities for researchers to test their hypotheses on real-data, developers to structure meaningful geo-services, analysts to study factual trends, and so on.

Primarily, OSM services can be classified into two categories, namely, Thematic Mapping Service (TMS) and Vehicle Routing Service (VRS). One popular and currently active VRS is Open Source Routing Machine (OSRM), which runs on top of the OSM data to provide shortest routes between destinations, along with many other small-scaled services targeted for specific users. Generally, these providers fail to assimilate the state-of-the-art scientific findings into their services and lag more advanced routing options. This substantial gap between developers and researchers has conceptualized this presented thesis. It has been understood that a more structured study of OSM's suitability for VRS will bring forth better routing-services in future. An attempt has been done to understand the current trend in OSM evolution and where the project is heading towards. Few existing lags in its road data are identified, which are supposed to tamper its usage applicability. Furthermore, one advanced routing query is attempted to solve from an spatial perspective, backed by scientific evidences in order to structure more featured services. The following four chapters are documented in this fashion, with corresponding gist provided in the following paragraphs.

The 2^{nd} chapter of this thesis, entitled ***Analysing the growth and governing factors of Turkey OpenStreetMap dataset***, has tried to understand how good OSM data-set is for any plausible routing-service's formulation. Because of having a no single criteria to measure it, it has always been a topic of revision. Nevertheless, it has been tried to picture this hypothesis by scrutinizing its spatial evolution with time for the political region of Turkey, as backed by other researchers too (Pengxiang Zhao for Beijing city, China) as one solid proxy. Although, the initial attempt was to provide a global commentary, Turkey region is studied as one missing link in on-line literature. Likewise conducted researches are quite discrete so far, leaving a room for global study for more generalized commentary. Yet, conducted analysis has sufficed data's usability at regional level. Furthermore, an attempt has been done to relate this evolution as a function of human-based parameters of the region, namely, literacy level, population density, tourism activity, internet usage, and human development index, as recommended by Neis Pascal as possible governing factors. The time-series statistical analysis of a region helps to answer many questions which govern shaping that OSM's ecosystem. An attempt to answer many relevant questions could be find in the final sections of the next chapter.

The key parameter to improve a routing-service is its underlying road length's precision. In chapter 3, entitled ***Improving OpenStreetMap derived road length on a global scale using Curve Fitting approach***, an attempt has been done to improve the existing methodology to estimate curved-road length by adopting piecewise cubic parametric polynomial curve fitting approach, which is proven to be a good way to enhance best fit to series of data-points. Existing OSM data handling tools estimate euclidean length between nodes, constituting a curved-road section, and thereby, avoid road curvature factor altogether. Things get severe at places like round-abouts. Practically, it is unattainable to trace down these kind of sections with tolerable precision as it requires extensive mapping efforts by mappers, and will remain unmanageable for OSM servers too. Unfortunately, no conducive work to tackle this problem is done by other researchers in any peer-reviewed journal, may be because of its simplicity to grasp and fix, nevertheless, it has been perceived to be one valid gap which is necessary to be filled with scientific proofing. Key findings and proposed methodology's limitations are discussed comprehensively in the final sections of 3^{rd} chapter. Additionally, a web-GUI is developed as one handy data visualization platform.

Once the OSM's usage possibility understood and road's data quality improved, a detailed study has been done to answer a more advanced routing query called the Equality-Generalized Traveling Salesman Problem (E-GTSP), which is an extension of the world famous Traveling Salesman Problem (TSP), by testing over real-city's OSM data. It is one NP-hard combinatorial optimization problem, with plethora of literature already available on-line with sub-optimal to optimal solutions. A more recommended dynamic programming approach to solve the E-GTSP is by transforming it to corresponding TSP, as there already exists a range of TSP solvers. However, all existing E-GTSP to TSP transformation algorithms are mathematical by origin, leaving applicability difficult for untrained users for routing models. A new approach to achieve this transformation for near-optimal solution involves considering the spatial spread of vertices within a given city's road-network's graphs, as explained thoroughly in 4^{th} chapter, entitled ***A new spatial approach for Efficient Transformation of E-GTSP to TSP***. 5 different search algorithms are developed for possible tests on

real-data of 15 cities worldwide, with 5 instances each, where each instance represents the total number of groups withing that routing model.

The 5^{th} chapter, entitled ***An attempt to reduce an E-GTSP instance size for GLKH solution***, is all about improving the model presented in 4^{th} chapter. A new custom cost of vertex, *Cost Product*, is coined to reduce the dimension of instance before solving it a given GLKH solution. The shrinked matrices generated using this cost are compared with tested matrices that were obtained from GTSPLIB, for cost error, time, and space. It is observed that for time and space measurements, shrinked matrices are better of the order of 2^{nd} degree polynomial than original ones. It is observed that percentage cost error is a function of average number of vertex per cluster and is bounded within specific range for different scenarios. The Cost Product is observed to be one custom cost that could be used to reduce the size of a given E-GTSP instance before solving it using GLKH. Key findings and general commentary are provided in the final section.

Findings of the subsequent four chapters have helped us to partially answer the following questions: *How good OSM data-set is for an advanced E-GTSP-solver routing-service, and how underlying route length's precision can be improved for better results?*. This study is expected to open future research possibilities for scientists and researchers in the field of VGI, OSM, VRS, and open geo-data, and assist developers to adopt good practices for improved services, as stated in each chapter's conclusion section. Nevertheless, a general conclusion is provided in the last chapter. Future work might involve the identification of better socio-economic proxies for OSM node density evolution, along with the identification of street network of different kind. The developed R-Search methodology could be improved by the use of machine learning and heuristic concepts. Statistical analysis to identify overshot nodes would be useful to check out road sections not suitable for curve fitting. A general possibility of future work is provided in the conclusion chapter.

OPENSTREETMAP KULLANIMINI İLERİ YÖNLENDİRME HİZMETLERİ İÇİN GELİŞTİRMEK

ÖZET

Son yirmi yıldır, cep telefonu, bilgisayar donanımı, uydu sistemleri gibi her türlü iletişim cihazını veya uygulamaları içeren Bilgi ve İletişim Teknolojilerinde (BIT) hızlı bir artış yaşanmıştır. Bu durum, Coğrafi Bilgi Sistemi (CBS) gibi özel bilgi sistemleri tarafından coğrafi etiketli verilerin büyük bir yoğunlukla etkin bir şekilde ele alınmasına neden olmuştur. Bu devasa ya da büyük veri eşzamanlı olarak, belirli kullanım durumları için hedeflenen çeşitli coğrafi hizmetlerin ortaya çıkmasını sağlamıştır. Ancak, bu tür verilerin genel olarak devlet ve ulusal harita kurumları tarafından öncelikli olarak toplanması, yönetilmesi, depolanması ve dağıtılması gerçekleştirildiği için bu verileri satın alma gücüne sahip olamayan genel kullanıcılar ve diğer düşük ya da orta sınıf hizmet sağlayıcılar için her zaman bir erişim kontrolü olmuştur. Bu anlamda, Web2.0 teknolojisinin ortaya çıkması, herhangi bir son kullanıcının kendi coğrafi verilerini oluşturmasına, yüklemesine ve yaymasına izin vererek mevcut sistemi değiştirebileceğini kanıtlamıştır. Bu tür veri yönetimi için tasarlanan sistemler Gönüllü Coğrafi Bilgi (GCB) (Michael F. Goodchild tarafından tanıtılmıştır) olarak adlandırılmaktadır. Çok bilinen bir GCB projesi Steve Coast tarafından 2004 yılında kurulan OpenStreetMap (OSM) dir. Bu açık konum verisi projesi, Kasım 2016 istatistiklerine göre dünya genelinde 9 milyar lokasyon (nokta), 0,4 milyar çizgi (yol) ve 3.1 milyon kayıtlı kullanıcı tarafından oluşturulan 4.5 milyon poligon (ilişkiler) verisi ile çok fazla gelişme göstermiştir. Böylece, araştırmacıların gerçek veriler hakkındaki hipotezlerini test etmek, geliştiricilerin anlamlı coğrafi hizmetlerini yapılandırabilmeleri ve analistlerin fiili eğilimleri incelemek gibi bir çok alanda oldukça büyük fırsatlar geliştirilmiştir.

OSM hizmetleri öncelikle, Tematik Harita Servisi (THS) ve Araç Yönlendirme Servisi (AYS) olmak üzere iki sınıfa ayrılabilir. Günümüzde aktif ve oldukça yaygın AYS, belirli kullanıcılar için hedeflenen bazı küçük ölçekli servislerin yanı sıra, hedefler arasında en kısa rotaları sağlamak için OSM verileriyle çalışan Açık Kaynak Kodlu Yönlendirme Aracıdır (AKYA). Genel olarak, bu sağlayıcılar son teknoloji ürünü bilimsel bulguları kendi hizmetlerine ve daha gelişmiş yönlendirme seçeneklerine sağlamakta başarısız olmaktadırlar. Geliştiricilerle araştırmacılar arasındaki bu önemli farklılık, sunulan bu tez çalışmasında kavramsallaştırılmıştır. OSM'nin AYS'ye uygunluğunun daha yapılandırılmış bir çalışma ile ele alınmasının gelecekte daha iyi yönlendirme hizmetleri sağlayacağı anlaşılmıştır. OSM evrimindeki mevcut eğilimi anlamak ve OSM projesinin nereye doğru gittiğini anlamak için bir girişimde bulunulmuştur. Kullanım olanaklarını etkilemesi beklenen yol verilerindeki az sayıda eksiklik tanımlanmıştır. Dahası, daha gelişmiş hizmetlerin yapılandırılması için bilimsel kanıtlarla desteklenen, bir mekânsal perspektiften, bir gelişmiş yönlendirme sorgusu çözülmeye çalışılmıştır. Tezde yer alan sonraki dört bölüm, bu bakış açısıyla hazırlanmıştır.

Türkiye OpenStreetMap veri setinin büyüme ve yönetim faktörleri başlıklı bu tezin 2. bölümünde herhangi bir makul yönlendirme hizmetinin formülasyonu için ne kadar iyi bir OSM veri setinin olduğu anlaşılmaya çalışılmıştır. Ölçmek için tek bir ölçüt olmamasından dolayı, her zaman bir revizyon konusu olmuştur. Yine de, bu hipotezi, diğer araştırmacılar tarafından da (Pekin şehri, Çin için Pengxiang Zhao) tek bir örnekle desteklendiği gibi, Türkiye'nin siyasi bölgesi için zamanla mekansal evrimini inceleyerek açıklamaya çalışılmıştır. İlk girişim küresel bir yorum sağlamak olsa da, Türkiye bölgesi çevrimiçi literatürde eksik bir bağlantı olarak ele alınmıştır. Benzer şekilde, araştırmalar şu ana kadar oldukça ayrıktır ve daha genelleştirilmiş yorumlar için küresel çalışma için bir alan bırakmaktadır. Ancak, yapılan analizler, verilerin bölgesel düzeyde kullanılabilirliğini göstermiştir. Dahası, bu gelişmeyi, Neis Pascal'ın olası bir yönetim olarak önerdiği gibi, bölgenin insan temelli parametrelerinin, yani okur-yazarlık seviyesinin, nüfus yoğunluğunun, turizm faaliyetinin, internet kullanımının ve insani gelişme endeksinin bir fonksiyonu olarak ilişkilendirmek için bir girişimde bulunulmuştur. Bir bölgenin zaman serisi istatistik analizi, OSM'nin ekosistemini şekillendiren ve yöneten birçok soruyu cevaplamaya yardımcı olmaktadır. Birçok ilgili sorunun cevaplanması, bir sonraki bölümün son bölümlerinde yer almaktadır.

Bir yönlendirme servisini iyileştirmek için anahtar parametre, temel yol uzunluğunun doğruluğudur. Eğri uydurma (Curve Fitting) yaklaşımı kullanılarak OpenStreetMap ten türetilmiş yol uzunluğunun küresel ölçekte geliştirilmesi başlıklı 3. bölümde, mevcut olduğu kanıtlanmış, parçalı kübik parametrik polinom eğri uydurma yaklaşımını benimseyerek, kavisli yol uzunluğunu tahmin etmek için mevcut metodolojiyi geliştirmek için bir girişimde bulunulmuştur. Mevcut OSM veri işleme araçları, noktalar arasındaki Öklid uzunluğunu tahmin ederek eğri yol kesiti oluşturur ve böylece yol eğriliği faktörünü tamamen ortadan kaldırır. Bu durum keskinliklerin ortadan kalması durumunda daha da artmaktadır. Pratik olarak, bu tür bölümleri haritacılar tarafından kapsamlı haritalama çabaları gerektirdiğinden tolere edilebilir hassasiyetle takip edemez ve OSM sunucuları için de yönetilemez duruma gelir. Ne yazık ki, bu sorunun çözümüne yönelik bir çalışma, belki de kavramanın ve düzeltmenin basitliğinden dolayı herhangi bir hakemli dergide başka araştırmacılar tarafından yapılmamıştır. Ancak bilimsel kanıtlama yoluyla doldurulması gereken bir boşluk olduğu düşünülmektedir. Temel bulgular ve önerilen metodolojinin sınırlamaları 3. bölümün son kısmında kapsamlı olarak ele alınmıştır. Ayrıca, bir web tabanlı grafik kullanıcı arayüzü bir kullanışlı görselleştirme platformu olarak geliştirilmiştir.

OSM'nin kullanım olasılığı anlaşıldıktan ve yolun veri kalitesi iyileştirildikten sonra, dünyaca ünlü Gezgin Satıcısı Probleminin bir uzantısı olan Eşit-Genelleştirilmiş Gezgin Satıcısı Problemi (E-GGSP) adı verilen daha gelişmiş bir yönlendirme sorgusuna cevap vermek için gerçek uygulama alanında OSM verilerini test ederek ayrıntılı bir çalışma yapılmıştır. Bu optimal çözümler için literatürde yer alan bir optimizasyon problemidir. E-GGSP'ni çözmek için daha fazla önerilen bir dinamik programlama yaklaşımı, bunun mevcut birçok GSP dönüştürücüsü kullanılarak karşı geldiği GSP ne dönüştürmektir. Ancak, mevcut tüm E-GGSP'nin GSP dönüşüm algoritmalarına göre kökeni matematikseldir ve yönlendirme modelleri için eğitimsiz kullanıcılar için uygulanabilirliğini zorlaştırmaktadır. Bu dönüşümün optimale yakın çözümü için gerçekleştirecek yeni bir yaklaşım, E-GGSP'nin GSP'ye verimli dönüşümü için yeni bir mekânsal yaklaşım başlıklı 4. bölümde ayrıntılı olarak

açıklandığı gibi, belirli bir şehrin karayolu ağı grafikleri içindeki köşe noktalarının mekansal yayılımını göz önünde bulundurmayı kapsar. Dünya çapında 15 şehrin gerçek verileri üzerinde olası testler için 5 farklı arama algoritması geliştirilmiştir. Her bir örnekte, bu yönlendirme modeline sahip grupların toplam sayısı gösterilmektedir.

Başlığı GLKH çözümü için bir E-GGSP örnek büyüklüğünü azaltma girişimi olan 5. bölüm, 4. bölümde sunulan modeli geliştirmekle ilgilidir. Kırık noktalarının yeni bir maliyeti, , belirli bir GLKH çözümünü çözmeden önce örnek boyutlarını azaltmak için üretilir. Bu maliyet kullanılarak üretilen daraltılmış matrisler, maliyet hatası, zaman ve alan için GTSPLIB'den elde edilen test matrisleri ile karşılaştırılmıştır. Zaman ve uzay ölçümleri için, büzülmüş matrislerin orijinal derecelere göre 2. derece polinoma göre daha iyi olduğu gözlenmiştir. Yüzde maliyet hatasının, kümelenme başına ortalama kırık sayısının bir fonksiyonu olduğu ve farklı senaryolar için belirli bir aralık içinde sınırlandığı gözlemlenmiştir. Maliyet ürününün, belirli bir E-GGSP örneğinin boyutunu GLKH kullanarak çözmeden önce azaltmak için kullanılabilecek bir özel durum olduğu görülmektedir. Son bölümüde temel bulgular ve genel yorumlar yer almaktadır.

Birbirini izleyen dört bölümün bulguları, aşağıdaki soruların kısmen yanıtlanmasına yardımcı olmuştur. Gelişmiş bir E-GGSP çözücü yönlendirme hizmeti için ne kadar iyi OSM veri kümesi kullanılmalıdır ve daha iyi sonuçlar için yol uzunluğunun hassasiyeti nasıl geliştirilebilir? Bu çalışmada, GCB, OSM, AYS ve açık coğrafi veriler alanındaki bilim adamları ve araştırmacılar için yeni araştırma olanaklarını açması ve her bölümün sonuç kısmında belirtildiği gibi, geliştiricilerin daha iyi hizmet sunabilmeleri için uygulamaları benimsemelerine yardımcı olması beklenmektedir. Bununla birlikte, son bölümde genel sonuçlara yer verilmiştir. Gelecekteki çalışmalar, OSM nokta yoğunluğu evrimi için daha iyi sosyo-ekonomik verilerin tanımlanmasını ve farklı türdeki yol ağlarının tanımlanmasını içerebilir. Geliştirilmiş R-Arama yöntemi, makine öğrenimi ve sezgisel kavramlar kullanılarak geliştirilebilir. İstatistiksel analiz yardımıyla gereksiz noktaları tanımlamak ve eğri uydurma için uygun olmayan yol bölümlerini kontrol etmek mümkün olacaktır. Sonuç bölümünde genel olarak gelecekte gerçekleştirilebilecek çalışma olasılığı sunulmuştur.

1. INTRODUCTION

With global urban population expected to grow from 4.9 to 7.4 billion between the years 2014 and 2050, there is a worldwide need for smart urban transportation solutions. Almost 90% of this increase is expected to take place in developing nations, primarily from Asia and Africa [155]. It is believed that by 2050, there would be around 2-3 billion vehicles running in the world [156]. Smart urban solution refers to the use of Information and Communication Technologies (ICT) to optimize city functions and drive economic growth, primarily by the use of emerging automation, machine learning and advanced routing algorithms. [73] has defined smart urban mobility as the use of technology to generate and share data, information and knowledge that influences decisions to enhance vehicles, infrastructure and services. Smart mobility is a system that requires optimised geo-data and optimised routing algorithms to run on top of it. With over 50 different public and private routing projects using the free availability of OpenStreetMap (OSM) dataset, it is believed to be one continuously evolving dataset to built any routing service on top of. Few aspects, however, are necessary to consider before understanding its suitability for routing services. These are how the data is changing and behaving with time and how accurate the dataset is for routing use-case. Routing algorithm, however, remains independent of these aspects as any dataset could be used with it. Nevertheless, it should be studied along with underlying dataset to find any semantics dependency. Improving routing algorithm involves optimising mathematical ways to find a near optimal route within a given time and space. The whole thesis is structured in this manner. The first two chapters, primarily, talk about the dataset and the last two chapters talk about routing algorithm's optimisation.

This age of online-data dissemination with no physical constraints and bottlenecks is allowing narrowly targeted goods and services as economically attractive as mainstream businesses ([128], [2]). For example, Wikipedia holds a huge pile of online free-content and competes with proprietary Encyclopedia Britannica. This approach

has even allowed amateur cartographer or citizen to upload tagged geo-data of any region to online servers [104]. OpenStreetMap (OSM) is one classic example of this kind which started in 2004, with objective to establish a free and editable map of the world. This VGI has recently gained huge popularity by providing big volume data with limited or no restrictions [163]. By the end of 2015, there were around 5 billion GPS points, 3 billion nodes and 4 million relations in OSM dataset [127]. It is important to understand what spatial evolutionary pattern does Turkey-OSM dataset have followed since 2007 to predict the future of it as has already been done by researchers for other cities ([167], [41], [19], [18], [162], [60]). The time-series statistical analysis of a region helps to answer the following questions, which govern shaping that OSM's ecosystem: What is the mapping behaviour at more refined zoom level? • What are the few local restrictions to be handled separately? • Which areas require more mapping efforts? • What is the psychology of a mapper behind volunteered activity? • How much data is prone to vandalism or already affected by it? • How the project should be treated locally in future? Factors used for this analysis are Literacy Level (graduated students), Population Density, Tourism Activity, Internet Usage [84], and Human Development Index (HDI). 2^{nd} chapter primarily talks about the evolution of dataset and reports crucial finding. No acknowledged study regarding Turkey-OSM dataset spatial evolution, socio-economic factors impact on such evolution, contributors involvement and general commentary on data health is present in online literature. Conducting this study is therefore essential to bring forth VGI responses at higher resolution (spatial and temporal) in developing country ([163], [66], [162]) like Turkey with fairly rich OSM dataset (17 million points, 1.3 million edges AKA lines, and 0.4 million polygons [124]). This analysis is also important for any VRS development that leverages its open geo-dataset for underlying graphs. 2^{nd} chapter, thus, primarily talks about how to obtain an improved geo-attribute from OSM dataset for VRS and other urban planning tasks, like road network visualisation, urban construction etc.

The broad range of applications of OSM has encouraged researchers and developers to collaboratively develop tools like osm2pgsql, osmium, osmosis, osm2pgrouting etc. to handle, manipulate and tweak its Extensible Markup Language (XML) data. Many services online leverage its raster and vector dataset for different use cases [131].

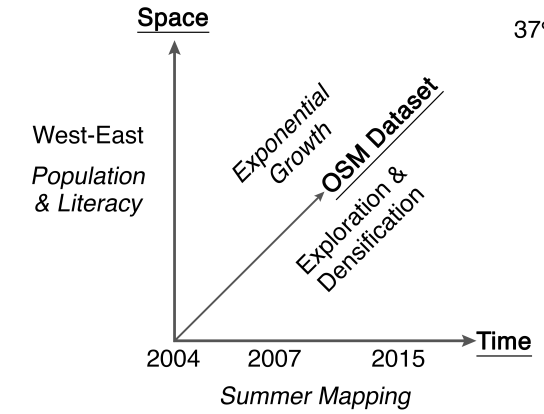
These services could be broadly divided into Thematic Mapping and VRS [54]. In cartography, map generalisation is used to reduce feature details [96]. It is being done by mappers while uploading features in VGI projects like OSM. There is no one reason for mappers generalising features, ranging from time availability etc. to poor background imagery ([98], [78]). It is observed that a reverse-generalisation of linear features is fruitful for better geo-attribute estimation. Chapter 3 talks about a curve fitting approach to obtain improved geo-attributes, that otherwise is missing in raw XML data. For curved road sections this kind of curve fitting is highly crucial. Instead of using Euclidean formula, a cubic parametric polynomial curve fitting algorithm, based on Pythagoras and Mean-Value theorem, is derived. The main objective of this chapter is to explain a data processing limitation in OSM ecosystem and a quick fix to it. The derived equation is applicable for any VGI with XML data format. Quality assessment done by other researchers primarily work on raw data set by comparing it with other governmental/proprietary data set ([43], [66]), by using indirect approaches like Linus Law ([42], by using contributors count as quality proxy ([84], [84], [78]), by developing intrinsic quality assessment parameter/tools ([82], [36], [31]) and by reviewing its change set dump file [5]. This way, they do not consider the derived attributes after download post-processing. No online literature in peer-reviewed journal is available to be used as benchmark for this study. Once improved, this value is crucial for services where road length value is primarily used like vehicle navigation and routing like the world famous TSP.

Travelling Salesman Problem (TSP) is a well-known and thoroughly studied combinatorial optimisation problem that asks to find a minimum-cost *Hamiltonian* cycle in G [46] connecting each node in a graph exactly once. It has numerous applications in areas like vehicle routing, networking, sequencing, scheduling, communication etc. [67] and therefore has attracted researchers for decades. A detailed classification of different type of TSPs and their solutions is done by [91]. Generalized-TSP (GTSP) is a direct extension of TSP [51] where the set of nodes is further divided into a number of groups and the task is to find a minimim-cost cycle passing each group exactly once. It has huge relevance in location based problems like routing, logistics, urban planning, telecommunication etc. Many advanced heuristics to solve its complexity include Ant Colony algorithm [161], Memetic algorithm ([10] and

[38]), Variable Neighbourhood Search algorithm [53], Random Key Genetic algorithm [100], Reinforcing Ant Colony system, Efficient Composite heuristic [93] etc. Chapter 4 talks about a spatial approach to reduce an E-GTSP instance to TSP before solving it using some TSP solver like the state-of-the-art Lin-Kernighan-Helsgaun [50]. Since the attempt is to see how good or bad the OSM dataset is for vehicle routing services, a spatial approach is followed for this transformation problem. Five possible search algorithms are proposed and tested on OSM street-network dataset to decide how to reduce the instance size. No key literature for this type of spatial transformation of E-GTSP is available online.

Finally, we have tried to improve the proposed R-Search model by defining a new cost value, called as Cost Product. It is formulated to reduce the overall cost-matrix size for fast and low-spaced computation. This new matrix systematically reduces the cluster size by keeping the probability of finding the best and optimal vertex in each cluster high. Results in terms of time, space, and cost error are compared with results drawn from the state-of-the-art GLKH solution [150]. Chapter 5 talks about it's nature of being bounded and how coefficient of variation limits the probability of finding optimal node within X%. Cost Product, thus, is applicable for any kind of instances, and not just instances representing routing problem. The proposed hypothesis of relationship between cost error, probability, coefficient of variation and X% is justified by the observed curve, which turned out to be inversely proportional.

All chapters attempt to answer the question of how efficiently the OSM dataset of a developing country, like Turkey, could be used for advanced vehicle routing problems. It has also been attempted to see if derived geo-attributes from OSM XML format could be improved with any tweak in existing algorithms. Results are promising and it has been observed that although OSM project has a long way to go, there are ways to understand the quality of derived attributes and, indeed, it could be used for advanced routing problems like E-GTSP, especially for rich graphs. A general conclusion and recommendation is provided by the end of each chapter and the last one. Developed codes and concepts are freely available online at my Github repository and is open under the MIT license.



37% Mappers -> 75% Data Upload

Nutshell

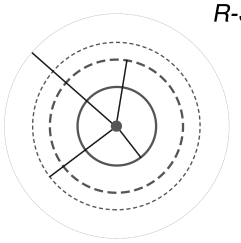
1. Strong proxies possible to predict OSM future
2. Need to compare different VGIs for quality
3. OSM -> potential source of vector data for VRS
4. Follows similar urban expansion mechanism

Derived *Piece-wise Cubic Parametric Polynomial Curve Fitting* Approach

Nutshell

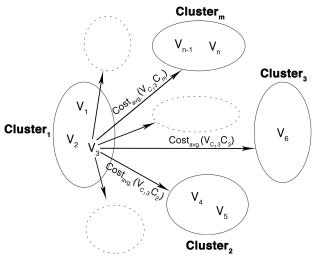
1. Overall Gain -> 0.7%
2. Simpson's Rule -> 10 # Segments
3. Curvature Overestimated -> Limitation
4. Better than Spline -> Simple and Fast

R-Search



Nutshell

1. Urban features uniformaly distributed
2. Convoluted city street-network
3. ↑ E-GTSP Dim. -> Saturated Optimal route length
4. Machine Learned R-Search possible



Cost-Product

Nutshell

1. Polynomial gain -> Time & Computing (RAM)
2. Bounded Cost Error = Function(CV and X%)
3. Suitable for Large Instances
4. Test different GTSP instances
5. Structure, Nature, & Origin of GTSPLib.

Figure 1.1 : Graphical abstract of the whole thesis.

2. ANALYSING THE GROWTH AND GOVERNING FACTORS OF TURKEY OPENSTREETMAP DATASET

2.1 Abstract

Plethora of studies have already been conducted in recent years about OpenStreetMap (OSM) project covering many aspects for developed countries and major world cities with limited attention on the developing ones. This paper presents an analysis of the spatial evolution of Turkey-OSM dataset from 2007 to 2015 year, and shows how it is related to different socio-economic factors of the region. Major key findings from this analysis are (a) an east-west spatial biasedness in OSM features density is observed in the country, (b) there is a high direct correlation between Population Density and Literacy Level with features density in the region, (c) there is an increase in socio-economic factors correlation with features density and OSM registered users mapping involvement with time, (d) Exploration and Densification processes are found responsible for the evolution pattern of street network dataset, (e) there is a high participation inequality among contributors among editing activities, (f) only around 5 *Crazy mappers* are found responsible for country's 50% OSM geo-data upload, however there varied data sources (e.g. other mapping projects, governmental data, in-person data acquisition etcetera) support the usefulness for specific case scenarios. Present study, thus, has opened up research paradigm for data quality assessment of Turkey-OSM dataset and modelling relationships between different Volunteered Geographic Information (VGI) mapping projects.

2.2 Introduction

Advent of Web2.0 [130] technology has allowed the Twenty-first century mankind to exploit more intelligently the Long Tail theory according to which our economy and culture is profoundly getting drifted away from a focus on a relatively small number of "hits" (mainstream solutions and domains) at the crest of the demand curve and towards a glut of niches in the tail. The call for one-size-fits-all bucket to encapsulate products

and users is waning away with reduction in the production- and distribution-cost of online content. In this age of no physical constraints and bottlenecks of online-data dissemination narrowly-targeted goods and services are getting as economically attractive and lucrative as mainstream fare ([128], [2]). For example, we now have Skype and Viber in Telecommunication sector, which was predominantly governed by Telecom industries for decades. Wikipedia is holding a huge stock of online free-content, competing proprietary Encyclopedia Britannica. Government and National Intelligence Agencies were regulating confidential information in the past but recent free publication of classified information by WikiLeaks and OpenLeaks has left a foot-print on human history [74]. Similarly, Volunteered Geographic Information (VGI) ([34], [69], [43]) or Crowdsourcing Geographic Data ([47], [23]) has evolved due to easy geo-data generation and circulation by human-beings acting as a sensor [33] as an additive to geographic information which is conventionally being compiled and retained by National Mapping Agencies and other private cartographic companies [26]. This contemporary approach has allowed even a naive cartographer/citizen with limited or no mapping experience to collect, map, and upload geo-data with extendable tagging options [104] of any region to online-servers. A classic live VGI example is OpenStreetMap (OSM) project [45] with genesis in 2004 with the objective to establish a free and editable global street map, in addition to many other similar examples like Wikimapia, Wikiloc, Foursquare, Google Map Maker etcetera but with distinct objectives. This classic example has recently gained immense popularity because of its big volume data (mapped by contrasting geo-data producers, AKA *NeoGeographers* ([35], [44]) because of no defined editing restrictions), heterogeneity, abundance, and free data access and thus has attracted extensive interest from researchers of ranging domains [163].

Present article is an attempt to answer what spatial evolutionary pattern does Turkey-OSM dataset have followed since 2007 until 2015 considering the necessity which has already been proven in plethora in the past ([167], [41], [19], [18], [162], [60]) for live/future VGI project's formulation and growth rate speculation, and has tried to understand selected socio-economic factors' impact on it. Selected factors', i.e. Literacy Level (graduated students), Population Density, Tourism Activity, Internet Usage [84], and Human Development Index (HDI), effect on VGI projects has not

been discerned so far in previously acclaimed studies with few mere speculations [84]. Furthermore, a commentary on the correlation between the number of active contributors and size of OSM dataset and the health of Turkey-OSM dataset is provided. To the best of authors' knowledge this kind of high resolution (at provincial level) OSM statistical analysis for a whole country backing existing theories along with current case specific vital observations on an eight year time-frame is first of its kind and authors believe that this will bring forth compelling pattern, trend, proxy parameters, and future research paradigm with usefulness to restructure existing and future similar projects.

The rest of the article's body is documented into the following sections • Review of latest notable OSM research and different facets of tackled analysis; • Study set-up: Data sources/processing/adjustment and adopted hypothesis; • Results and Discussions; and • Conclusions and Future Work.

2.3 Review of Research on OpenStreetMap

By the end of 2015 OSM dataset has had an enormous amount of geo-tagged world data in the form of approximately 5 billion GPS points, 3 billion nodes, 3 billion ways, and 4 million relations contributed by around 2.5 million registered users worldwide [127]. One of the many possible reasons for this popularity hype is Google's partial pulling out its Map APIs from public domain in 2012 [126], thus encouraging Apple iPhoto, FourSquare, Craiglist, Flickr [166], and many more to switch to the OSM. This massive dataset has brought forth possibilities of investigating a broad spectrum of domains such as data accuracy, data exhaustiveness, possible usage, time-series data evolution, motivated psychology and elements governing this psychology of contributors, and relationship with other VGI projects [83].

Regional/Global OSM data accuracy has already been studied by researchers in great detail from many perspectives in recent years, for example by comparing with governmental/proprietary dataset ([43], [66]), by using indirect approaches like Linus Law [42] or using contributors count as a quality proxy ([84], [86], [78]), by developing intrinsic quality assessment parameter/tools ([82], [36], [31]), and by reviewing its changeset dump file [5]. Linus law (formally *"Given enough eyeballs, all bugs are shallow"* [92]) which explains the direct relationship between the number of

developers assigned to a project and its bugs detection rate was also found applicable for OSM data reliability and credibility assurance by [42] where below 6 m positional accuracy is reported for regions with more than 15 contributors per km^2 area. OSM participation inequality was stated by few researchers ([163], [78], [84] by reporting top 3% OSM members as *Senior Mappers* in 12 urban cities, and [86] by reporting 5 active per 100 members with majority being located in Europe as a proxy for data quality check-up [$data\_quality \propto 1/participation\_inequality$]). On the contrary, a thought-provoking conclusion was drawn by [79] by negating the idea of counting the number of contributors as a quality (metadata) proxy ([42], [84], [86], [78]). Researchers like [82] (comprehensive rule-based prototypical tool for vandalism), [36] (VGI quality assessment approaches), and [31] (key French-OSM spatial data quality assessment parameters, extending the work of [43] for London) have tried to exploit the OSM intrinsic parameters for plausible data scrutiny and boost, thus adding additional quality assurance mechanism.

Sagacious delve on deciphering OSM dataset completion has always been impeded by strict licensing policy, bounded usage, reserved availability, and lofty pricing of governmental/proprietary geo-data sources which act as a reference dataset ([27], [43], [66], [167]). Nonetheless, recent years have witnessed some notable contributions on this in scientific literature (for example, Germany-OSM street network data by [85] using proprietary dataset, USA-OSM bicycle trail and lane data by [52] using data from local planning agencies, and USA-OSM street network data by [166] using TIGER/Line data [125]). Few of the many OSM use-case scenarios include measuring urban sprawl with its street data as a population proxy [55], developing Location Based and Emergency Medical Services ([80], [1], [4]), generating interactive 3D City Models using Shuttle Radar Topography Mission height data [89], extracting Image-based road network [15] and multilane roads data [70], calculating shortest routes within urban cities [165], and validating/reforming existing Land Use/Land Cover (LULC) data like Global Land Cover Maps [29]. Recent adoption of OSM data during Haiti earthquake relief operations has further stretched out its plausible usage in natural calamities as well [17]. [33] has argued personal satisfaction and community serving as two key motivating fuels behind crowd sourcing activities and VGI gain.

[11] and [12] have further segregated these and other motivating aspects into intrinsic and extrinsic categories.

2.3.1 OSM effort in Turkey

No acknowledged study regarding Turkey-OSM dataset spatial evolution, socio-economic factors impact on such evolution, contributors involvement, and general commentary on data health is present in online literature. Conducting current study is therefore essential to bring forth VGI responses at higher resolution (spatial and temporal) in a developing country ([163], [66], [162]) i.e. Turkey with fairly rich OSM dataset (17 million points, 1.3 million edges AKA lines (in the following text, edge terminology is used for any line/polyline feature), and 0.4 million polygons [124]), and authors do believe that this will subsequently add insights to help popularize/expand the same. Five aspects of dataset analysis are:

1. *Time-series spatial evolution*: One of the first aspect which has been studied thoroughly in this article is how the dataset has spatially evolved in a ten year of time-span. This will advance the work of [18], [163], [19], and [162] and discuss if the mapping activity is regionally biased or not.

2. *Effect of region's socio-economic factors on its spatial evolution*: In order to identify the impact of socio-economic factors on VGI activities five major factors (namely, Population Density [85], Literacy Level, *Tourism Activity, Internet Usage [84]*, and HDI) as also discussed by other researchers are compared with OSM features density on a time-scale.

3. *Processes governing the evolution of country's road network*: What evolutionary trend [41] does the Turkey-OSM street network dataset has followed in the given span of time is observed and reported? This aspect is studied considering the famous *Exploration and Densification* elementary processes concept for road networks evolution [103] which has already been propped in other similar studies ([19] and [18]).

4. *OSM-contributors mapping behaviour*: The forth aspect of conducted analysis is to determine the confidence level by which the density of the number of distinct contributors with at least one contribution can be used as a proxy for region's OSM

features denstiy. This proxy has been thoroughly studied by other researchers but at different zoom-level and region ([85], [83], [42], [74], [79], [78], [167], [163], [166]).

5. *Quality of the dataset*: It is fairly important to report the quality of a given OSM dataset in order to identify the usefulness for specific use-cases. Past researchers have used the participation inequality in editing processes as one strong proxy for VGI data accuracy/health ([163], [78], [84], [86]) and therefore in this study also authors have used this proxy along with mappers varied geo-data sources to provide a general commentary on Turkey-OSM dataset usefulness.

2.4 Study Set-up: Data Sources and Processing

2.4.1 Source and Format of Data

High interoperability of OSM dataset by having various data sources (Full Planet dump file [124], Geofabrik downloads [121], Overpass API [122]) and formats (ESRI-Shapefiles $*$.shp, Extensible Markup Language (XML) $*$.osm, Protocolbuffer Binary Format $*$.pbf) is considered as one another reason for its popularity hype. This was further facilitated by upgrading its *Editing API* (latest v0.6) in due course of time [120] depending upon the technological advancements and user requirements and switching its license from *Creative Commons Attribution-ShareAlike 2.0* to *Open Database License (OdbL)* in 2012 [119], thus allowing users and others to freely share, modify, and use the database [118]. Full Planet dump file does not consist of edits before 2007 since object history feature was introduced in *Editing API* v0.5 [120] as also reported in Section 2.5.1 and has lost 1% of data during conflict of users interest during 2012 licensing event [119]. Nonetheless, dump files are proven to be the best source of OSM dataset to study time-series evolution ([78], [79], [81], [85], [52], [166], [81], [5]) as other sources only reflect the latest snapshots for a specialized region, and authors do not believe 2012 data loss to bias it.

Full Planet dump file (size approximately 67 GB and 1.5 TB when compressed and uncompressed, respectively) dated *September 02, 2015* (last stable history release at the time of data processing) was downloaded from [124] which contained OSM complete database including editing history from as far back as 2007 until

September 2015. The file is in a human-readable XML format containing three primitive data elements/features: node (point), way (polyline and polygon), and relation (logical combination of nodes, ways and/or other relations), annotated with tags in a key-value structure of free format text fields [123]. Provincial statistical data of socio-economic factors for last ten years was downloaded freely from TUIK (Turkish Statistical Institute) online portal [108] which is a national level government organization for data (demographic/geographic/scientific/economic etcetera) collection, storage, processing, and distribution for policy formulation, educational, and scientific purposes. The variable of related domain downloaded from TUIK statistics portal are as follows: *Number of students for vocational training school and undergraduate programs of higher education institutions: Graduates / Total* from *Education >Higher Education* domain as an indicator of *Literacy Level* (Figure 2.1a), *Annual growth rate and population density of provinces* from *Population and migration >General Population Censuses* as *Population Density* (Figure 2.1b), *Number of arriving foreigners by province of border gate and mode of transport : Air way* from *Tourism* as *Tourism Activity* (Figure 2.1c), *The proportion of individuals regularly using the Internet* from *Transportation and Communication* as *Internet Usage* (Figure 2.1e), and *Per capita gross value added (GVA) : Per capita GVA ($)* from *National Accounts* as a proxy of *HDI* (equation 5.2) (Figure 2.1d). The ill-famed Syrian Refugee Crisis has caused heavy foreigners influx via Road way at the south-eastern (South-East Anatolia, Figure 2.1e) part of country in recent years contaminating data of *Tourism Activity* and therefore only Air way as the mode of transport was selected to get a better statistical picture (Figure 2.1c). Data corresponding to the length of provincial roads in order to negate the idea of roads scarcity as a reason for spatial biasedness of *Edges* feature density (Section 2.5.1) was obtained from General Directorate of Highways [107] website (Figure 2.1f).

2.4.2 Data Processing Steps

OSM XML file could be made processable by a variety of command line tools such as osmosis (Java application for reading/writing databases [117]), osmium (multipurpose tool for data interoperability and time-series analysis [116]), osm2pgsql (tool to convert XML data to PostGIS-enabled PostgreSQL databases [115]), osm2postgresql

(to simplify rendering with QGIS and other GIS/web servers [114]), osm2pgrouting (to import data file into pgRouting databases [113]) etcetera; however because of being designed to work on recent data version of a given region for specialized tasks these are not suitable for the carried out processing. Instead, osmium based open-source *osm-history-splitter* tool [112] which is laid out to help split the Full Planet dump files for any world region using its bounding-boxes, .poly files, or .osm polygon files was used to crop *September 02, 2015* dump file using bounding-box covering the political region of Turkey by softcut-algorithm. Subsequently, country's ESRI-shapefile for provincial boundaries was used to further crop down the data into 81 different provinces excluding Cyprus which is a land of conflict with finally loading up each cropped province into different schemas of PostGIS enabled PostgreSQL database. In order to expedite the process of data management and querying each provincial dump file data was classified into three databases delineating point, edge, and polygon (covering all primitive geometry elements for analysing individual dataset evolution) with each one having 81 schemas. Finally, each schema was divided into 18 time-tagged tables depending upon the features' date of creation ($valid\_from$ column), thus, making $(18 \times 81 \times 3)$ 4374 tables in totality. The time intervals used for feature categorization are as follows: *between April 01, 2004 to April 01, 2007*, *between April 01, 2007 to September 01, 2007*, *between September 01, 2007 to April 01, 2008* and so on, till *between April 01, 2015 to September 01, 2015*, comprising 18 such intervals (April and September months were selected to divide a year into two halves in order to represent summer (Tourism and out-door activity) and winter (arm-chair mapping activity) season).

It is possible to attach any kind/type of tag to an OSM feature using its online [111] or stand-alone JOSM [110] editor making it prone to noise intrusion. It was therefore necessary to be specific in features selection using pre-defined tags [104] for current analysis. It was observed that only 2% of the whole point data under investigation have some sort of tags associated with it and authors decided to break it down into *Points(all)* (all Point features present in the dump file) and *Points(tagged)* (all Point features with not null tags) subcategories. For edges and polygons, features with *Highway* (describing all roadways and footpaths including motorways, residential roads, primary roads etcetera, except cycleways and railways) and *Building, Landuse,*

Natural (describing major LULC feature and man-made structure) key were selected, respectively, making four final feature categories, i.e. *Points(all)*, *Points(tagged)*, *Edges*, and *Polygons*. Only relevant keys, with no values legitimacy estimation because of the unavailability of any reference dataset, are favoured for *Edges* and *Polygons* knowingly the fact that it may influence the true picture marginally. Author has developed three Python scripts in order to automate data querying and results storage processing which took around two months to finish on an Ubuntu-14.04 Trusty Server with 40 GB total RAM, downloadable from his Github account [105] along with a comprehensible README file. However the author has advocated the scripts development on language more closer to the hardware (mainly `C++`) for accelerated big data processing.

2.4.3 Data Adjustment and Normalization

It is not straight-forward to compare two elementary geometry features, i.e. point, edge, and polygon, when the idea behind is to determine contribution effort (Figure 2.2). Production of a point, a line, and a polygon does not involve similar toil from contributors side making them incomparable to each other in this perspective. Similar argumentation is valid for comparing lines ([163], [18], [103], [43]) and polygons of different length and area, respectively. As a matter of fact, lines and polygons are digitally stored and defined as a collection of nodes and therefore the number of *nodes* (vertices) constituting *Points(all)*, *Points(tagged)*, *Edges*, and *Polygons* were counted as a proxy for respective feature's count in order to make them comparable and entertain contribution activity more reasonably [163] (Figure 2.2). Additionally, to nullify the geographic effect of varied provincial area on *nodes* count for each province it was divided by the respective area. For example, two regions, namely A and B, with area of 1 unit^2 and 100 unit^2 and *nodes* count (for let's say *Edges*) of 10 and 500, respectively, can not be correlated regarding feature density by mere count of *nodes* because of varied spatial sizing of the two regions. Division of *nodes* count with corresponding area will give us a better sum (count per unit^2) for correlation calculated as 10 and 5 for this example, thus exhibiting region A as doubly crowded with *Edges* features as region B. On the other hand, demographic effects are not considered as a potential threat for ongoing OSM dataset spatial analysis, although researchers like

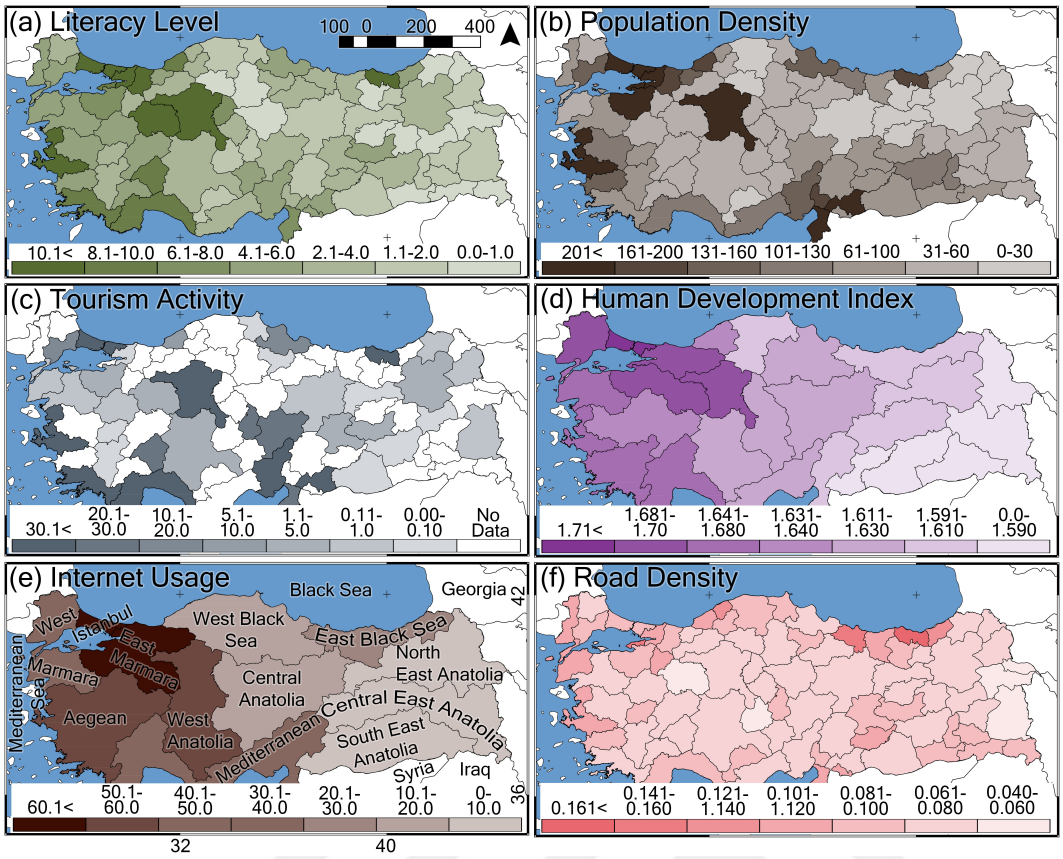


Figure 2.1 : Color legends are (a)Total number of students in the year 2013, (b) Population density in the year 2014, (c) Total number of arriving foreigners in the year 2014, (d) Human Development Index as a function of Gross Value Added per capita ($), (e) Total number of person in the year 2013 using internet, (f) Road (state+provincial) length in km, per km^2 area.

[103] and [18] have tried to remove it in similar studies regarding street network evolution.

JOSM [110] application even lets one to import/edit features offline and bulk upload the same to online databases through OSM API despite the fact that the community discourages this approach for database gain, if not performed by *Senior Mappers*, for world regions with underdeveloped datasets as it may severely propagate errors and affect other contributors' efforts of manual data acquisition and editing. Past researchers have accounted the necessity and process of bulk imports removal as an outlier element in order to avoid unexpected data spikes and observations ([78], [19], [1], [166], [18], [5]); however, the true definition of bulk import documented as *"Bulk import means more than a few hundred nodes or for a larger area like a whole country"* [106] is itself pretty vague and researchers have used their own explanations at various

past scenarios ([1] have used tens of thousands of edits by a single user in a single day as a model for bulk import). Therefore, authors have selected 25000 *nodes* contribution by a single user in a week as a model of bulk import event and have subsequently removed it from all calculations. Less than 10 *Crazy Mappers* are identified as bulk importers during this study throughout the country and removed accordingly.

In graph theory the degree k_i of a node i is the number of nodes adjacent to it, i.e. $\left[k_i = \sum_{j=1}^{N} a_{ij}\right]$, in terms of the adjacency matrix [13]. In real street networks degree of a street junction is the number of road segments having it as their starting or terminating node. To a great extent distribution of different degree values in a street network is a manifestation of its topological structure and how densely or sparsely the street segments are connected. Degree distribution $P(k)$, defined as $P(k) = N(k)/N$ where $N(k)$ is the number of nodes with degree k and N is the total number of nodes, in a primal graph or real road network is therefore necessary to understand its evolved stage, i.e. how much the region has been explored and densified with roads at any given point in time. Decrease in the degree distribution value of low degree junction represents early stage Exploration, whereas increase in the same for high degree junction indicates late stage Densification [163]. These are the two elementary mechanisms governing the evolution of road networks [103]. These two mechanisms are used in Section 2.5.3 to support observations.

Finally, in order to compare provincial HDI value as one of the socio-economic factors with OSM feature density per capita GVA ($) value as an HDI proxy was used since no direct measurements are available on TUIK portal. HDI is defined as the geometric mean of Life Expectancy Index (LEI), Education Index (EI), and Income Index (II) [129]; mathematically:

$$
\begin{aligned}
&HDI \\
&= (LEI \times EI \times II)^{1/3} \\
&= \left(\frac{LE - 20}{85 - 20} \times \frac{\frac{MYS}{15} + \frac{EYS}{18}}{2} \times \frac{\ln(GNI_{percapita}) - \ln(100)}{\ln(75000) - \ln(100)}\right)^{1/3}
\end{aligned} \tag{2.1}
$$

where, LE is Life expectancy at birth (years), MYS is Mean years of schooling for ages 25 and above (years), EYS is Expected years of schooling (years), and GNI is Gross National Income per capita ($). Since:

$$GNI_{percapita} = GVA_{percapita} + T + IF - S - ID$$

where, T is Taxes on products ($), S is Subsidies on products, IF is factor Incomes earned by foreign residents, and ID is Income earned in the domestic economy by non-residents [68]; substituting GNI value in equation 5.1 will give:

$$\begin{aligned} &HDI \\ &= \left(\frac{LE-20}{85-20} \times \frac{\frac{MYS}{15}+\frac{EYS}{18}}{2} \times \frac{\ln(GVA_{percapita}+T+IF-S-ID)-\ln(100)}{\ln(75000)-\ln(100)}\right)^{1/3} \\ &\approx \left(\ln\left(\frac{GVA_{percapita}}{100}\right)\right)^{1/3} \end{aligned} \tag{2.2}$$

considering other variables as constant.

Although, authors are aware of the fact that the above approximated relationship at equation 5.2 between HDI and GVA is a biased depiction and will affect final observations but because of being one of the first of its kind (one previous work was by [84] using GNP per capita) this comparison will lead future clues about the influence of income/living standard per capita as the motivational sources for VGI involvement.

2.5 Results and Discussions

This section is divided into five aspects of current analysis of Turkey-OSM dataset and key observations are discussed accordingly. The sections follow the trend presented by previous researchers on understand how a dataset evolves with space and time. Each section follows figures to explain observed findings.

2.5.1 Time-series spatial evolution

Figure 2.3 shows the time-series evolution of *nodes* density for 81 provinces of Turkey constituting *Points(all)*, *Points(tagged)*, *Edges*, and *Polygons* since 2007 until 2015. It

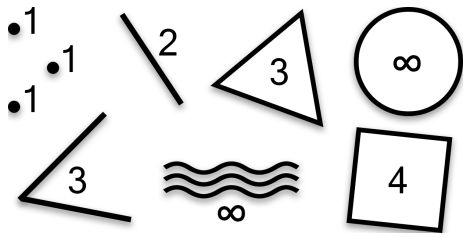


Figure 2.2 : Different geometrical features with corresponding *nodes* count.

can be seen that data before *April, 2007* is absent in the dump file as all the graphs are tapering towards the origin as one moves back in time, especially in Figure 2.3b. This observation was expected as *Editing API* v0.5 has introduced the object history feature into the project in 2007 [120] meaning no history data before it. Between 2007 and 2012 the curves are following a gradual increment (Figure 2.3a,c,d) with almost horizontal trend in case of *Points(tagged)* Figure 2.3b (points with attributes and no bulk imports), thus depicting limited contributions by dormant contributors because of limited editing flexibility by old OSM license ([119]) which was then followed by an exponential growth in 2012 ([163] has also reported similar growth rate for both the number of *nodes* and *Edges* for Beijing, China) because of the inception of *Odbl* license and increased OSM usage in ranging mapping projects [166]. Although the sudden boost in data contribution activity is exponential it is not equally powered for all provinces as it is a function of the number of active contributors in the region [163]. A closer look at graphs (especially for provinces with high *nodes* density) illustrates that the exponential curve itself is a partial exponential-step curve (exponential curve growing step-wise). This is because time-span between *September, 2007* and *April, 2007* every year has witnessed fewer *nodes* edit through different mapping events as compared to between *April, 2007* and *September, 2007* because of low level Tourism and out-door activity in winter. However, this observation is exclusively visual.

Another observation can be drawn regarding *nodes* density value across the country. Figure 2.3 also shows *nodes* density map at 2009, 2012, and 2015 time-slices. At each snap-shot it can be visualized that the eastern and south-eastern part of the country is less densified as compared to the western and south-western part ([162] has also reported the spatial distribution of China's OSM road network) with some outliers. Socio-economic factors can be attributed for this spatial biasedness of *nodes* density as similar biasedness is present in Figure 2.1. Density maps of Literacy Level per

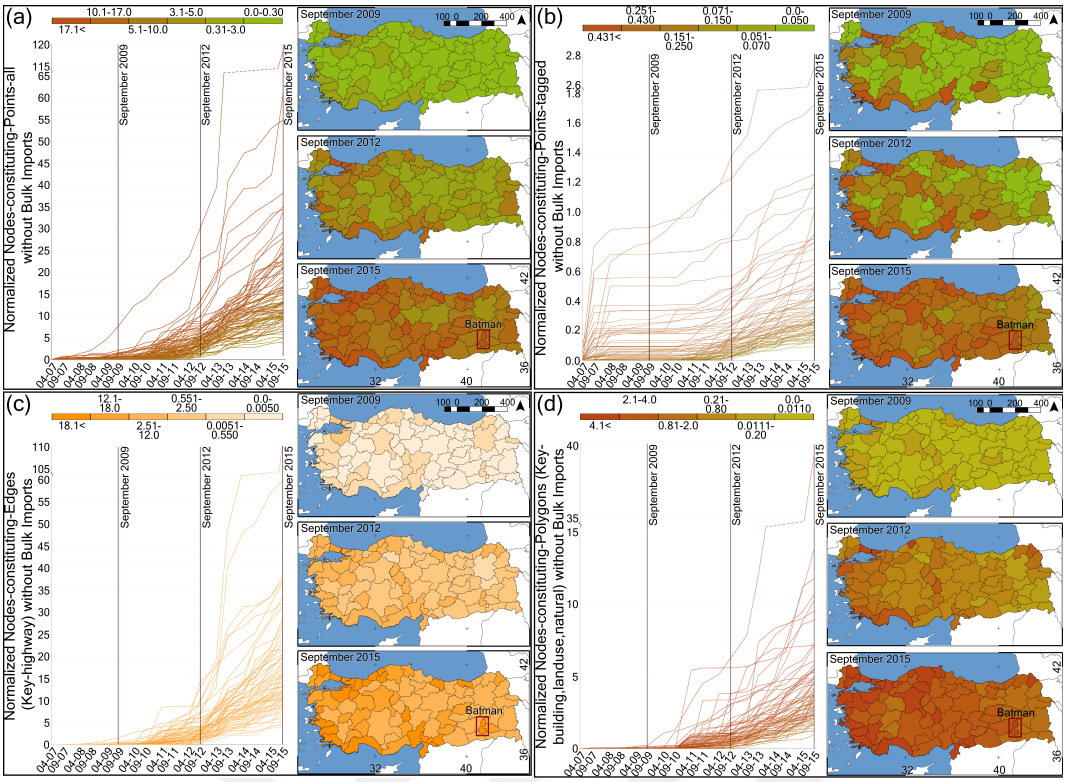


Figure 2.3 : Features' *nodes* density evolution with time.

km^2 area of year 2013, Population Density of year 2014, Tourism Activity per km^2 area of year 2013, Internet Usage per km^2 area of year 2014, and HDI of year 2011 were drawn showing similar density pattern which was observed in Figure 2.3. In order to negate the notion that this spatial biasedness in OSM *Edges nodes* density is because of the absence of the features itself on the ground provincial road length density map was drawn from *General Directorate of Highways* [107] website (Figure 2.1f). Cross-validation is not possible for *Points(all)*, *Points(tagged)*, and *Polygons* since these features generally do not exist de facto onto the ground.

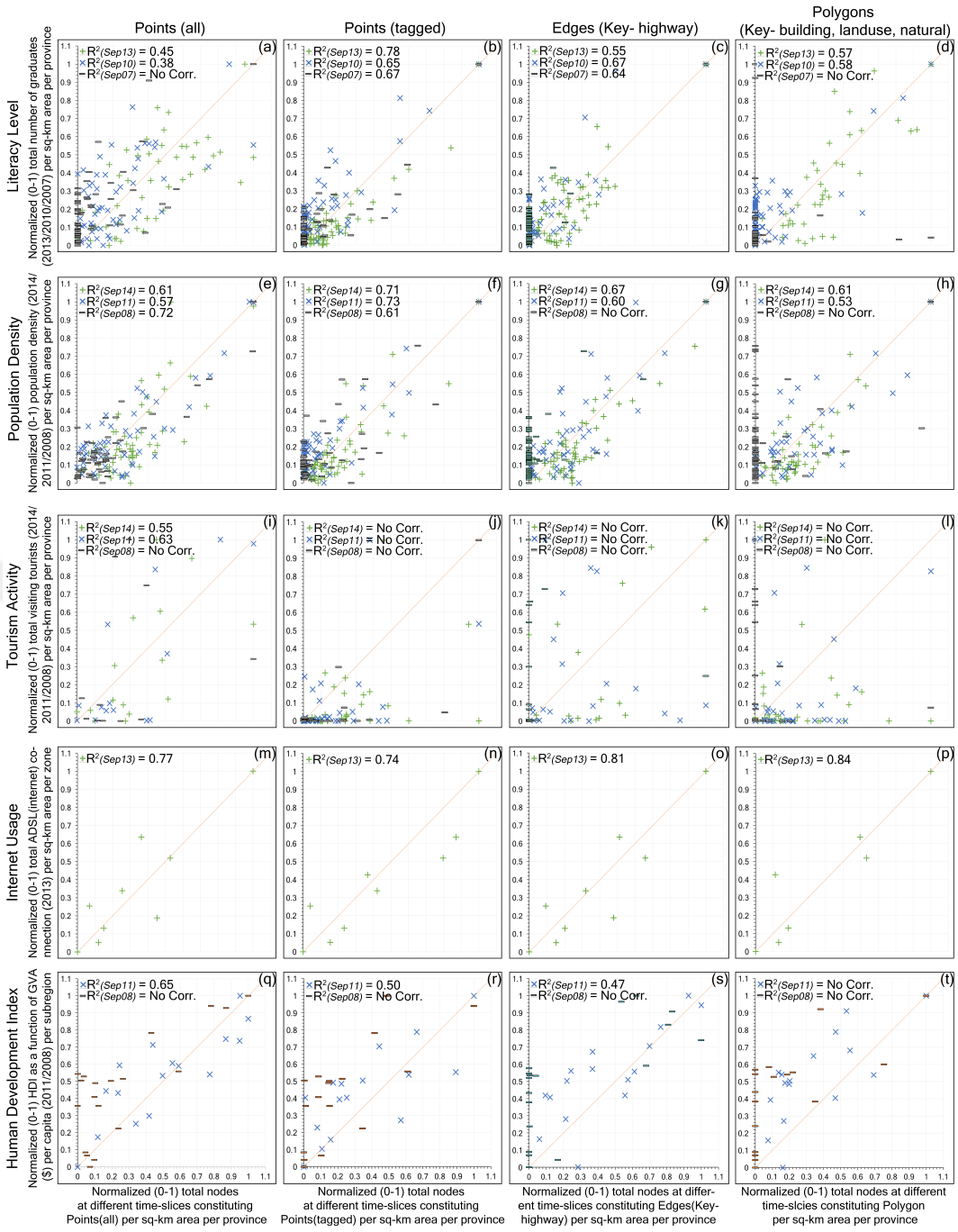


Figure 2.4 : Graphs between the socio-economic factors and OSM features density under study.

Except buildings *Polygons*, landuse and natural *Polygons* are mobile and sporadic features and no reference dataset or satellite imageries could be used for cross-validation. Figure 2.1f illustrates a uniform road length density throughout the country depicting that the road itself is not scarce rather OSM *Edges* features are non uniformaly mapped by contributors ([19], [163], and [18] have also studied OSM road network evolution). Some provinces in the eastern and south-eastern part show *nodes*

density spikes, thus acting as an outlier because of *Senior mappers* being active there. One such region is Batman province, red box in Figure 2.3, which shows a high *nodes* frequency, especially in 2015, because of a mapper (a university student, Table 5.1) who is responsible for around 4 and 3% Turkey-OSM contributions for *Points(all)* and *Edges*, respectively. Since this mapper is currently a university student no spikes are present for this province for earlier years.

2.5.2 Effect of region's socio-economic factors on its spatial evolution

A normalized (0-1) scatter plots between *nodes* density and all five socio-economic factors under consideration at different time-slices are plotted in Figure 2.4 to better relate which was merely limited to visual interpretation in the previous Section 2.5.1. The Figure shows a matrix of graphs of *Points(all)*, *Points(tagged)*, *Edges*, *Polygons* vs Literacy Level, Population Density, Tourism Activity, Internet Usage, HDI, with all possible pairs. Mathematically, the more the two variables (x- and y-axis) in a graph are directly related to each other the closer the normalized (0-1) scattered points are to a 45 degree line passing through the origin (orange line in Figure 2.4) with exactly on the line for highest correlation with coefficient of determination (R^2) being equal to 1. Although, statistically, R^2 (which always fall between 0 and 1) values between 0.0-0.5, 0.5-0.6, 0.6-0.7, 0.7-0.8, and 0.8-1.0 depicts No, Weak, Moderate, Good, and High correlation between physical parameters, socio-economic factors are based on human activities and therefore even Weak correlation with others are good enough to deduce definite conclusion [41]. Authors have manually removed few provinces (between 4-6 depending upon the feature which were homogeneously distributed throughout the region) which were exhibiting spikes and contrary trend between the two axis. It can be seen that R^2 value is quite high (more than 0.6) for all features against Population Density (Figure 2.4e,f,g,h) showing a Moderate correlation between them. For *Points(tagged)* features this value is even stronger (more than 0.7 for recent years (Figure 2.4f)). It should be noted that *Points(tagged)* features can be treated as a sample of an ideal dataset, out of *Points(all)*, *Points(tagged)*, *Edges*, and *Polygons*, because of having attributes associated with it. Similar observations are drawn for graphs against Literacy Level (Figure 2.4a,b,c,d). It can be said that Population Density and Literacy Level are better proxy for OSM features density and contribution activity in a region

(Figure 2.4a-h). Although [84] has guessed some dependency of Tourism Activity (and Internet Usage) on project's evolution, No correlation was observed between *nodes* density and the number of tourists visiting that province per km^2 area (Figure 2.4i,j,k,l). In spite of having high R^2 value for Internet Usage no concrete statement can be drawn (Figure 2.4m,n,o,p) because of limited data availability (only 2013 data for the number of people regularly using internet at zonal level in Turkey is available at TUIK portal [108]), similar explanation is possible for No correlation in case of Tourism Activity as only data from 37 provinces out of 81 is available (Figure 2.1c). Weak correlation, on the other hand, is reported for GVA per capita as a proxy for HDI (Figure 2.4q,r,s,t) (although [84] has reported a Moderate R^2 value of 0.664 between the number of members in OSM and the GNP per capita for major world cities). Since HDI factor is a function of life expectancy, mean year of schooling, and gross national income (the three parameters strongly represents the health and economic status of a region) a better analysis would be to directly compare HDI value (instead of some proxy parameter) with *nodes* density. It can said that in Turkey-OSM dataset Population Density has showed a high correlation with the number of features present/mapping activity in the region, followed by Literacy Level, and finally HDI, with no strong remarks about Internet Usage and Tourism Activity.

R^2 value for Population Density against *Points(tagged)*, *Edges*, and *Polygons* increases as one moves from older to recent time-slices (2008-2011-2014), i.e. from 0.61 to 0.73 to 0.71, from No corr. to 0.60 to 0.67, and from No corr. to 0.53 to 0.61, respectively (Figure 2.4e,f,g,h), similarly for Literacy Level it grows against *Points(all)*, *Points(tagged)*, and *Polygons* for three time-slices (2007-2010-2013), i.e. from No corr. to 0.38 to 0.45, from 0.67 to 0.65 to 0.78, and from No corr. to 0.58 to 0.57, respectively. For Population Density vs *Points(all)* the R^2 value for the year 2007 is quite high (0.72 (Figure 2.4e)) because of all the scattered points lying close to the origin. On the other hand, for Literacy Level vs *Edges* R^2 value is high for the year 2010 as compared to the year 2007 and 2013 (Figure 2.4c) and this can not be explained and can be considered as an outlier. Generalizing for all VGI projects it can be reported that as project grows in time the socio-economic factors get better correlated with database features density. The scatter-points mainly lies on the y-axis in the early years of OSM project's inception and as one moves further forth in time

these points get more and more concentrated over the 45 degree line which is an indicator of correlation improvement. The Turkey-OSM dataset size is synchronizing with socio-economic factors with time and the same notion can be generalized for any kind of VGI projects governed by crowd sourcing activities.

2.5.3 Processes governing the evolution of country's road network

The elementary processes governing the evolution of any real life road network was explained in depth by [103] by devising two scientific terminologies, i.e. Exploration and Densification processes, according to which any road network's evolution first experiences an exploration phase where unexplored regions and regions with scare road connectivity are explored followed by a densification phase where further secondary and tertiary level roads get popped up around the initially developed primary network. These phases of network evolution are also visible in VGI projects where mappers edit road networks on a geographic information system [163]. The y-axis of Figure 2.5 is the slope value of graphs between degree distribution ($P(1)$, $P(2)$, $P(3)$, $P(4)$, $P(5)$, and $P(6)$) (Section 2.4.3) and time-series from 2007 to 2015, for all 81 provinces in Turkey; whereas the x-axis is the provinces, grouped together into respective zones and plotted from west to east of the country (Figure 2.1e). Authors have only plotted graphs for junctions upto 6 degree, since 7 or higher degree junctions are practically not possible in real life scenarios, although [163] have entertained maximum 5 degree junctions. The y value which are all negative (with mean value of -0.00016, Figure 2.5 1 degree histogram) except for one vertex in Central East Anatolia which could be considered as an outlier for vertices of blue line (Figure 2.5) which belongs to 1 degree junctions exhibits a decrease in degree distribution for all the provinces, an indicator of initial phase Exploration process (Section 2.4.3). 3 degree junctions (grey line in Figure 2.5), on the other hand, exhibits an increase in degree distribution for all the provinces by having positive y values for all its vertices (with mean value of 0.0001, Figure 2.5 3 degree histogram) a cursor of later phase Densification process. For 2 and 4 degree junctions (orange and yellow line, respectively, Figure 2.5), the y value is not definite (i.e. positive and negative and lying very close to the x-axis (having mean values of 2.09E-5 and 2.89E-5, respectively, Figure 2.5 2 and 4 degree histograms) as compared to 1 and 3 degree junctions for

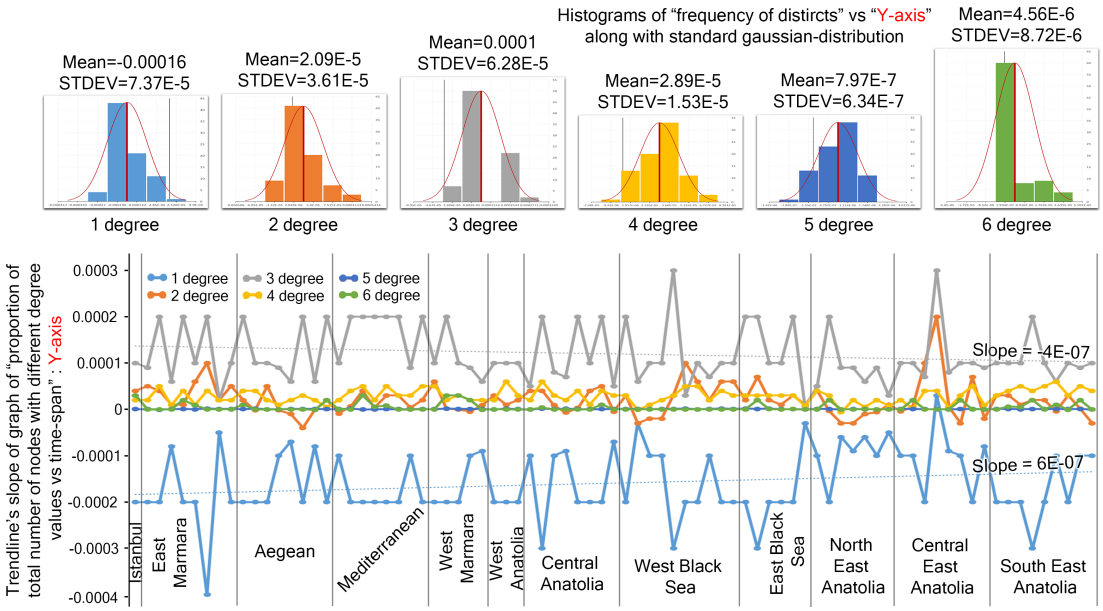


Figure 2.5 : Graph supporting the Exploration and Densification processes for street network evolution of Turkey-OSM dataset.

different provinces). This is because the $P(k)$ vs time-series graph for them is almost parallel to the x-axis which shows a constant density of such kind of junctions in the dataset throughout the time. Whereas for junctions with degree 5 and 6 (dark-blue and green line, respectively, Figure 2.5) the slope value is almost zero (having mean value of 7.97E-7 and 4.56E-6, respectively, Figure 2.5 5 and 6 degree histograms) because they are absent or highly scarce in real road networks in Turkish provinces. The high frequency of graphs with zero slope (Figure 2.5 6 degree histogram) is because of the scarcity of 6 degree junctions in Turkish road networks since no junctions for a given degree will result graphs to lie on the x-axis with zero slope. It can be concluded that the Turkey-OSM does have followed the elementary Exploration and Densification processes for street network evolution in an eight year time span which has already been reported for Beijing, China [163] and Ireland [18]; and the 3 degree road junctions are the most abundant one in country's urban network setup which represents an organic street layout [30] which has also been evidenced by [146] for developing countries.

Furthermore, a linear trendline was drawn over blue and grey plots and respective slope values were calculated which came out to be positive and negative, respectively (Figure 2.5). Since the provinces on the x-axis are plotted from west to east of Turkey (Figure 2.1e), the tapering of trendline towards the x-axis (the trendline slope is positive for 1

degree and negative for 3 degree) as one moves further right direction shows the high slope values (both positive and negative) of degree distribution ($P(1)$, and $P(3)$) vs time-series graph for western part of the country as compared to the eastern part. This attributes to the high mapping activity by contributors resulting into large *nodes* density in western part as compared to the eastern part which has already been documented in Section 2.5.1 (Figure 2.3). One might doubt about the acceptance of small absolute values of trendline slope and y-axis for any concrete conclusion but the idea behind this current discussion is not based upon the magnitude of the slope but sign. It is important to note that human behaviours are generally cumbersome to map, unlike physical parameters.

2.5.4 OSM-contributors mapping behaviour

Figure 2.6 is the graph between the total number of distinct contributors with atleast one contribution of a particular feature in Turkey-OSM dataset per $1000 \times km^2$ area and the total number of nodes constituting that feature per km^2 area for three different time-slices normalized on a 0-1 scale. Selection of only those contributors who did atleast one contribution will filter out inactive, fake, and those registrations which happened because of some trail attempts, which are noises in the data. Orange line is again a 45 degree line passing through the origin (Figure 2.6). High R^2 value is calculated for each graph corresponding to the *Points(all)* (0.67), *Points(tagged)* (0.69), *Edges* (0.83), and *Polygons* (0.78) features for the year 2015 (Figure 2.6a,b,c,d) which characterizes a Moderate, Moderate, High, and Good correlation, respectively. It can be reported that the active contributors density is a strong proxy for respective features density in a region by being directly proportional, thus backing the findings of [167], [84], and [163] who have also reported a direct relationship between total points and registered users density. As a matter of fact, the logarithmic or exponential relationship will cause the scatter points to deviate from the 45 degree line. Another point to be noted is that contributors responsible for bulk imports (Section 2.4.3) are not considered in counting the y values (Figure 2.6).

The R^2 value for all four graphs against 45 degree line increases as one moves from old to recent time-slice. For *Points(all)*, *Points(tagged)*, *Edges*, and *Polygons* the value increases from 0.61 to 0.74 to 0.67, from No corr. to 0.66 to 0.69, from 0.59

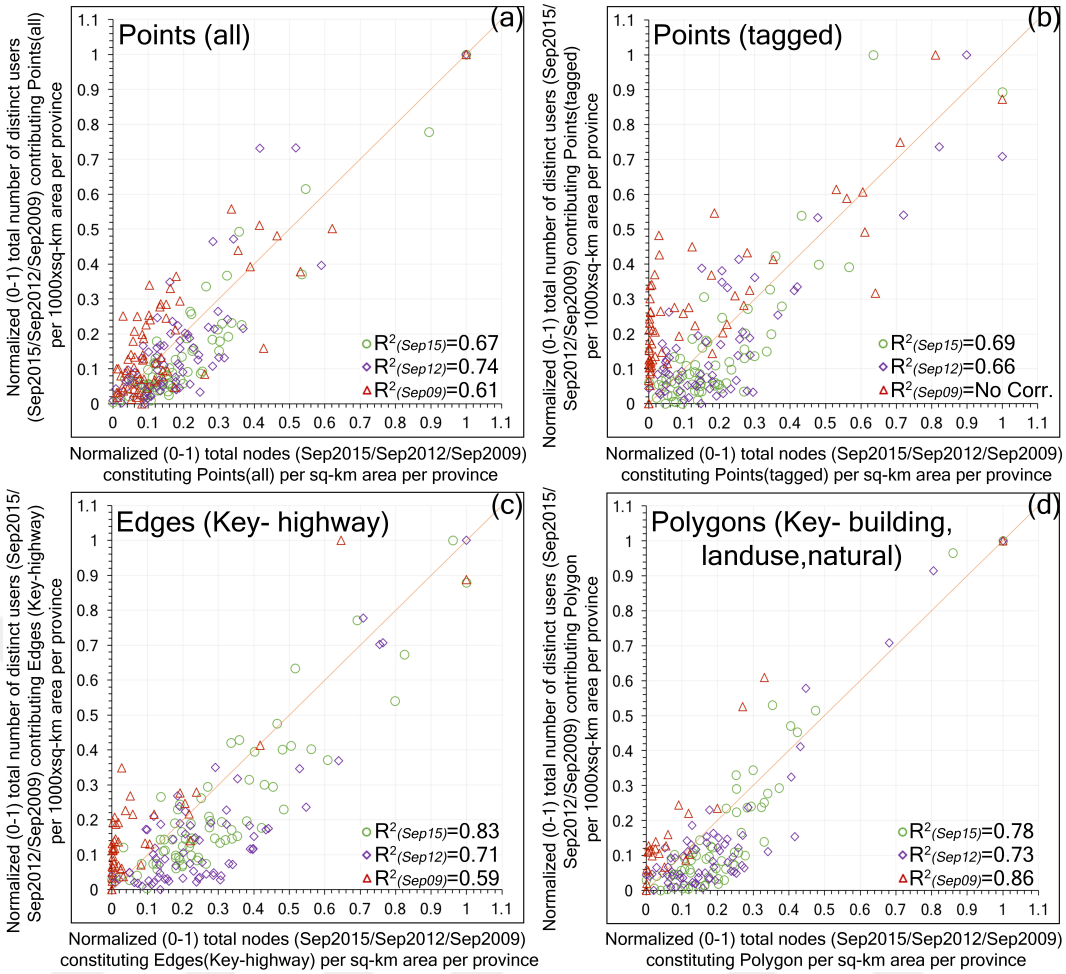


Figure 2.6 : Graph showing a direct fairly good correlation between the number of active contributors edited a particular feature and the density of *nodes* frequency constituting that feature.

to 0.71 to 0.83, and from 0.86 to 0.73 to 0.78 for the time-slice 2009, 2012, and 2015, respectively. It must be noted that although the R^2 value for graphs against *Polygons* for the year 2009 is 0.86, it is in fact a bad correlation and this high value is because of the clustering of all scattered points close to the origin. This happened because very limited contributors were there who did *Polygons* features mapping back in 2009 in Turkey-OSM dataset. The increase in R^2 value in time indicates later stage involvement in contribution activity by dormant contributors. For 2009, all four features, especially *Points(tagged)* and *Edges*, show the scattered points clustering over the y-axis (red triangle, Figure 2.6b,c) which is because of limited mapping activities (no mapping activity is not possible as authors have intentionally selected distinct contributors with atleast one feature contribution) by users in the early days of OSM project. As one moves forth in time, the scattered points get drifted from y-axis towards $y = x$ line which shows a later staged mapping involvement by inactive

contributors, thus creating a direct relationship between the frequency of mappers and mapped events. It can be stated that although users do perform some initial staged editing as soon as they register to any VGI project these edits are only limited to a few number of features as a result of trial/testing activity and they generally take some time to get started editing and contributing significantly which is evident by the increase in R^2 value from 2009 to 2012 to 2015. Because of the ease of online registration and editing in VGI projects, thanks to Web2.0 [130], people do register and perform some initial edits but typically do not show any serious personal commitment or community service motivation. Another possible reason for later staged contribution activity in Turkey-OSM dataset is the introduction of *Odbl* license in 2012 [119] which has permitted more flexibility in data edits and uploads. This pattern of R^2 increase with time has also been observed during the correlation of socio-economic factors with *nodes* density (Section 2.5.2) (Figure 2.4).

2.5.5 Quality of the dataset

Participation inequality is accounted as a proxy for VGI project's accuracy according to Linus Law [42]. Several researchers ([79], [86], [163], [84]) have reported that heavy mapping by few selected users, also called as *Crazy mappers*, generally through bulk imports impedes the suitability of data practice for specialized GIS tasks. *Crazy mappers* are different from *Senior mappers* by the fact that they generally do not have any proficiency in mapping activities yet contribute heavy chunk into mapping VGI projects. Figure 2.7 shows the percentage contribution by all bulk importers and remaining mappers without any bulk import event. The Figure illustrates that almost three fourth ($\approx$75%) of the total *nodes* density for *Points(all)*, *Edges*, and *Polygons* is uploaded by bulk importers by the end of 2015. For *Points(all)* and *Edges* there are around 37 bulk importers (Figure 2.7a,c) whereas for *Polygons* only 12 (Figure 2.7d) which shows that point and edge features are more easily identifiable on the satellite imageries and are readily available from other geo-data sources for on-line uploads. On the contrary, *Points(tagged)* pie chart (Figure 2.7b) shows no bulk import event since attributed point features dataset is difficult to generate or obtain from other portals. Thus, the credibility of tagged point features in Turkey-OSM dataset is high as compared to other features, i.e. *Points(all)*, *Edges*, and *Polygon*,

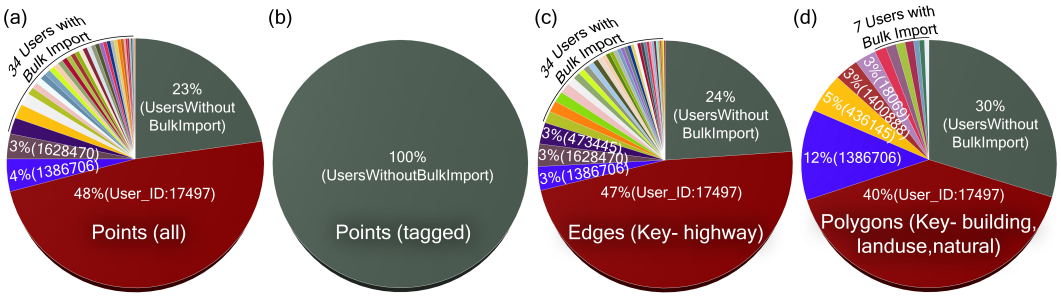


Figure 2.7 : Pie charts showing the participation inequality and bulk importers for all four OSM features.

because of the inverse relationship between data quality and participation inequality ($data\_quality \propto 1/number\_of\_bulk\_importers$) (Section 2.3).

Additionally, from Figure 2.7a,c,d only one user (User ID: 17497) is accountable for almost 50% of the whole dataset generation for *Points(all)*, *Edges*, and *Polygons*, along with few other mappers contributing between 3 to 12% of the database. It is of utmost importance to understand these *Crazy mappers'* geo-data sources in order to control the origin and propagation of random/systematic errors, if any. Table 5.1 contains general information about few of these mappers, i.e. their nationality (to understand if they had local knowledge about the mapped region), technical background (to understand if they have performed the crowd sourcing activities soundly), and geo-data sources (to check if their parent data sources are reliable), which was obtained during in person communications by the author. Although two out of five mappers (Table 5.1) are from Germany their contributions can be trusted because of being resident of Turkey which brings forth local knowledge about the surrounding regions (although they are responsible not only for their locality but also for the whole country's data evolution). Two of the remaining mappers are from Turkey and data is not available for the fifth one. Technically these mappers are sound by being System Programmer, University student, and GIS specialist which is positive for the online bulk upload events as blunders are less prone to happen. Parent geo-data source information is crucial to provide some commentary about Turkey-OSM dataset accuracy (Table 5.1). It can be seen that all of these mappers are using proprietary satellite images, i.e. Google, Bing, and Mapbox basemaps which are open for general public use, from private companies which are one of the reliable geo-data sources. Apart from that they are mostly using data from other kind of similar VGI mapping projects (wikiloc, geofabrik tools for data correction) for bulk imports. It can commented that VGI projects share data

among themselves and once an error get inserted it is hard to fix completely as it propagates, sometimes automatically. Therefore, users must be highly careful while performing any crowd sourcing activity as easy digital data dissemination will not limit their contribution to a particular project. User *Nesim Is* from Batman University (and *Penom*, Table 5.1), on the other hand, have used more authentic geo-data obtained from local municipalities (Konya and Denizli Municipality) for bulk imports, thus, causing heavy contributions in Batman province in recent years (red boxs in Figure 2.3 for 2015 time-slice). Authors believe these kind of bulk importers to be responsible for disrupting the general picture of mapping evolution in a country ([78], [166], [5]) by creating outlier regions (e.g. Batman province) which can be seen in Figure 2.3 at the eastern and south-eastern part of the country which otherwise have showed low socio-economic growth (Figure 2.1). In spite of having high participation inequality in Turkey-OSM contribution dataset can be considered suitable for fairly detailed GIS purposes, especially for *Points(tagged)* features since it's free from bulk import, because of having varied geo-data sources, in-person data collection events, and data acquisition processes backed by local community help. However, there is a high urge for data's accuracy assessment using some proprietary dataset or data acquired from ground truth campaigns.

2.6 Conclusions and Future Work

This article presents an analysis of the spatial evolution of Turkey-OSM dataset and its correlation with different socio-economic factors of the region in an eight year time span (2007-2015). The five facets of this analysis are: (a) how spatially the dataset has evolved in the given course of time?; (b) how socio-economic factors are correlated with this evolution?; (c) What road network evolutionary pattern does the street network dataset has followed?; (d) How the active contributors are related to this evolution?; and (e) How reliable the dataset is for any given purpose, a general commentary on its quality?

It has been observed that the dump files do not have history data before 2007 because of the absence of object history feature in *Editing API* v0.4 or earlier. In Figure 2.3 the curves are horizontal between 2007 to 2012 which shows a period of immobility in contribution activity, however, there is an exponential rise after year 2012 because

Table 2.1 : General information about Turkey-OSM *Crazy mappers*

OSM User ID	OSM Username	Nationality	Technical Background	Geo-data Sources
17497	Roman	Germany	IBM System Programmer	• Bing-Maps and Mapbox satellite imageries to trace over. • .jpg images for boundary data. • .gpx files from Wikiloc.
1386706	Nesim Is	Turkey	Batman University Student	• Kentrehberi Konya Belediyesi. • Denizli Buyuksehir Belediyesi. • Google Street View. • In-person data collection. • Local community help. • Various literature sources. • Different websites.
18069	Claudius Henrichs	Germany		• Geofabrik Tools - OSM Inspector. • In-person data collection. • Mapped remotely using satellite images. • Publicly available data.
1400888	Summerson		GIS specialist	• Mostly polygons for buildings. • Bing satellite maps. • Online maps by municipalities. • Perform regular bulk imports.
436145	Penom	Turkey		• Regional Municipality for detailed mapping.

of the change in OSM license from *Creative Commons Attribution-ShareAlike 2.0* to *Odbl*. After 2012, the curves are following a partial exponential-step function because of less contribution activities in winter seasons. The spatial analysis has revealed that there is an spatial biasedness from west to east of the country towards the evolution of dataset at any given point in time (Figure 2.3), with some exceptional provinces. Provinces along the Mediterranean sea (western and south-western provinces) have experienced more *nodes* density at three selected time-slices (2009, 2012, and 2015) as compared to the eastern and south-eastern part of the country which were always under-developed in the past. This pattern in *nodes* density is believed to be the consequence of socio-economic factors, i.e. Literacy Level, Population Density, Tourism Activity, Internet Usage, and HDI, in the region, where similar pattern in factors density is observed (Figure 2.1).

The R^2 value for recent years for the graph between different socio-economic factors and OSM features under analysis has revealed that the Population Density and Literacy Level in a region are highly correlated with the success of OSM project in Turkey (Figure 2.4). Pictorially, all factors with their degree of impact (by different font sizes) on OSM database can be summed up like:

Population Density >Literacy Level >HDI >Internet Use $\approx$ *Tourism Activity*

However, it should be noted that further analysis in different regions around the world is required to generalise this trend in the degree of impact. For the moment, this has only been observed in Turkey-OSM. Furthermore, it can commented out that with time the impact of socio-economic factors on any VGI project grows stronger.

Regarding street network evolution, it has been found that the country's street network does have followed the primary Exploration and Densification processes for its expansion in time (Figure 2.5). The country has organic street layout with abundance of 3 degree road junctions. It has also been observed that the western and south-western part of the country has followed a sudden change in degree distribution values for $P(1)$ and $P(3)$ in time as compared to the eastern and south-eastern part depicting more mapping activities which has already been commented in Figure 2.3.

The active contributors density is a strong proxy of *nodes* density in a region (Figure 2.6). It has been observed that early staged inactive contributors do start contributing as the project evolves or policies change.

Finally, a profound participation inequality is observed in the given OSM dataset with only 37 *Crazy mappers* responsible for around 75% of whole dataset upload through bulk imports (Figure 2.7). A personal, one on one, communication with these mappers has revealed that they themselves are dependent on other similar VGI mapping projects for geo-data sources (Table 5.1). This opens a concern about how and upto what extent different VGI projects are intermingled with each other. A part from that, their varied other data sources, like governmental data portals, proprietary satellite images, data collected in-person and with local community help, are positive elements to reduce the impact of participation inequality upto some extent, although this effect of different elements on the health of OSM is a matter of statistical analysis which is beyond the scope of current study.

Future work may include considering more socio-economic factors with direct or indirect association with OSM growth. What motivation does these factors bring forth to the users will help better structure future VGI projects and formulate corresponding licenses. Study to relate different mapping projects will aid VGI project administrators to check possible sources of error and provide data accuracy indicator.

3. IMPROVING OPENSTREETMAP DERIVED ROAD LENGTH ON A GLOBAL SCALE USING CURVE FITTING APPROACH

3.1 Abstract

In OpenStreetMap (OSM) ecosystem, derived length of road is an imperative attribute necessary for successful analysis of street network and development of geo-services. Conventionally, this geometrical attribute is calculated after data download using FOSS4G tools that operate on Euclidean formulation based upon Pythagoras Theorem. Therefore they ignore the road curvature factor altogether. In this study, a piece-wise cubic parametric polynomial curve fitting approach is presented which incorporates curviness into account for improved data visualization and better road length estimation. The approach operates on OSM (a famous VGI project) highway feature's *node* set on an iterative basis by considering four *nodes* at a time. When tested on highway features of four urban cities, the developed methodology has bestowed better results than Euclidean method. An overall 0.70% improvement in length estimation is observed over tested cities. Computational cost is substantially reduced by selecting 10 number of segment between each pair of *nodes* in order to solve the developed definite integral using Simpson's rule. Further commentary on mapping precision of curved-sections in OSM and VGI, in general, is also provided with three defined classes, i.e. *Precise Mapped*, *Curvature-Underestimated*, and *-Overestimated Mapped*; and discussion is done on how fit the developed methodology is for each one of these classes. This study opens few potential research areas where OSM derived attribute could be improved, attribute that otherwise is absent in OSM raw XML file.

3.2 Introduction

In past few years, the near-ubiquitous presence of Global Positioning System (GPS) enabled devices, remote internet availability, advent of Web2.0 technology [130] and Information and Communication Technology (ICT) advancement has facilitated the sudden increase in free online geographic content. When generated voluntarily

by individuals or teams primarily on a local basis this geo-content is termed as Volunteered Geographic Information (VGI) ([34], [69], [43]) or Crowdsourcing Geographic Data ([47], [23]). Although a lot of VGI projects with different aims, scope and restriction/licensing policy thrive these days, like Wikimapia, Wikiloc, Foursquare, Google Map Maker etc., one classic example is OpenStreetMap (OSM) [45] with objective to establish a free and editable map of the world. This over a decade old project with genesis in 2004 allows anybody from anywhere to map almost any physical or virtual (like ship trajectory) geo-features online in abstracted form using flexible specifications [104] and varied web/desktop platforms ([111], [110]). Its sudden hype in popularity was led by its big volume data, heterogeneity, abundance and open access (Open Database License [119]).

By end of June 2017, the project had almost 5.8 billion uploaded GPS points, 4.0 billion nodes, 0.4 billion ways, 5.1 million relations and 4.0 million registered users [127]; although only a small fraction of these users actually contribute to the project regularly ([84], [163], [78], [84]). It has recently attracted extensive interest from researchers and service providers from ranging domains ([163]). Its broad range of application has encouraged developers to collaboratively develop tools, like osm2pgsql [115], osm2postgresql [114], osmium [116], osmosis [117], osm2pgrouting [113] etc., mainly on social coding websites like GitHub to handle, manipulate and tweak its Extensible Markup Language (XML) data file and to derive inherited geo-attributes useful for different use-cases. Plethora of OSM based web/mobile-GIS services could be find online [131] leveraging its raster and vector data set. They can be broadly divided into Thematic Mapping and Vehicle Routing Services (VRS) [54]. For VRS the precision of derived road length is of great importance.

3.2.1 OSM Way Tagged Length

Figure 3.1 shows the structure of an OSM XML file format with *node*, *way* and *relation* tags [132]. During feature generation in OSM ecosystem user generates collection of *nodes* representing each particular feature in WGS84 reference system by collecting either by GPS devices or manual tracing over satellite imageries. For *way* and *relation* features, OSM refers to the set of *nodes* constituting them and avoids tag redundancy and improves data readability. By default, the raw XML data does not

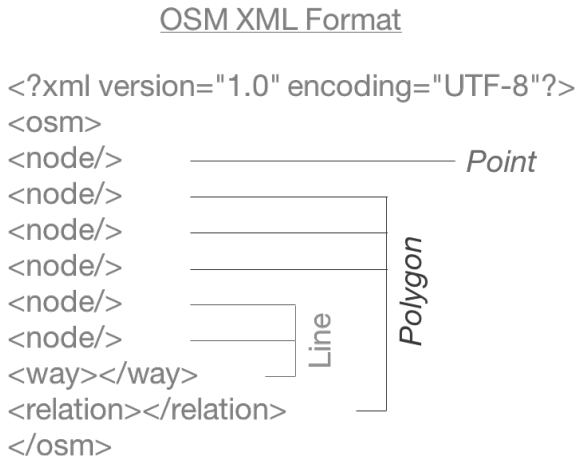


Figure 3.1 : A shortened sample OSM XML file format showing the dependency of *way* and *relation* features on to the corresponding *nodes* set.

contain geometrical attributes of *ways* and *relations*, mainly length and area, because of the way being structured syntactically. In order to derive these attributes, user needs to perform some post-download steps to the file. There are almost 320 different kind of features that could be marked by *way* tag inside OSM specifications. Among them, 28 features represent roads and footpaths and are marked by highway keys [104]. Although highway key represents only a small fraction of all possible *way* features, they represent one of the biggest chunk of *way* tags. We are interested in highway features (mainly road) primarily because of the extent to which they are mapped in OSM and used in different geo-services.

In order to estimate the length of a road for different applications, like to feed to the underlying graph for weighting purposes in VRS [1] etc., user commonly calculates the Euclidean Length (EL) using Pythagoras Theorem that gives EL aka straight-line length [147]. Since OSM XML format describes *way* as collection of *nodes*, EL approach eventually sums up all individual *node*-pair EL values falling adjacent to each other for overall length. Figure 3.2 shows how existing tools treat the shape and length of curved road sections in OSM. This way the curvature of a road which is critical for curved roads is circumvented altogether, hampering length estimation and feature visualization. Practically, it is unattainable to map these curved sections with great precision as it demands huge mapping effort by mappers and computational resource by servers. Therefore EL of roads is unsuitable for specialized purposes and advanced VRS, although it is also a factor of city street pattern [77]. Since in practice

Figure 3.2 : A plotted sample road section over satellite imagery.

the curvature of road is always generalized in OSM, EL is always less than or equal to Actual ground Length (AL).

$$EL \leq AL \tag{3.1}$$

In eq. 5.1 the equality holds true for cases where section is precisely linear with only two *nodes* representing the start and end point of the section. Otherwise, the length error, i.e. $AL - EL$, solely dependents upon the extent of generalization or, in other words, the mapped precision of the mapper. It should be noted that representation of linear features as collection of point is a global phenomena and is not limited to OSM or VGI and although this study talks only about road features in OSM the developed algorithm could be applied to any other linear feature in order to reduce the affect of generalization.

3.2.2 Problem Encountered

Map generalization is a technique used in cartography to reduce the detail of feature. Whatever map we see at whatever scale is generalized upto certain extent [96]. Possible reasons for map or feature generalization are better readability, low data size, fast rendering etc. [159]. One famous feature generalization technique is Ramer-Douglas-Peucker algorithm where a subset of points that defines the original curve is identified to represent a simplified version of it [24]. Researchers like [90] and [75] have tried to compare the errors produced by different line simplification

algorithms and developed a hybrid approach for better results by segmenting and simplifying linear features based on quantitative characteristics of the line. In one decent study, [102] have developed a methodology to check the requirement and limitation of automated map generalization in various commercial software. In case of VGI geo-data generation, mappers generalize these features too but within their mind and submit a simplified version of it manually, generally speaking, to the server. There is no quantitative way to say the extent of generalization in any VGI data set, like famous OpenStreetMap, as there are many different reasons for simplification done by mappers, ranging from lack of mapping experience, background imagery's resolution, time availability etc. ([98], [78]). In normal generalization, we reduce a detailed data set to something handy for both the user and the computer, but in VGIs a simplified version of features is all what we get from servers and sometimes that is not what we want for spatial use-cases. It has been observed that for VGI data set a reverse-generalization or curve fitting of linear feature is fruitful. Researchers have already used curve fitting approaches for linear feature detailing on different venues ([56], [7], [14] and [25]) but no such study exists around VGI or OSM project. We believe that this curve fitting application in linear feature is beneficial for end-users/citizen scientists who better want to leverage the power of public geo-data.

In this study, a methodology is developed to reduce the affect of generalization, done by mappers within their head or by GPS device because of limited mapping rate, on road sections by using a cubic polynomial curve fitting approach by using underlying *nodes* set taken from OSM file (discussed in detail in Section 4.4). This becomes crucial for sections that are curved on the ground. Instead of calculating the EL pair-wise, a mathematical formula is derived based upon cubic polynomial equation, Pythagoras and Mean-Value Theorem in order to calculate the Curved Length (CL) pair-wise (Section 3.4.1). The implementation is done in `C++` environment (Section 3.4.2) and tested over around 400 road sections from four major world cities (Section 3.5). The results are discussed graphically in Section 5.5 to present the usefulness of this easy to understand and implement approach. The essence of this study is to explain a data processing limitation in OSM related tools and a quick fix to it. The curve fitting approach itself is not novel as past researchers have already used similar mathematical ways to improve best fitting curves on series of data points, possibly

subjected to constraints ([158], [135], [136]). However, the implementation is hoped to bring forth improved tools for better services, thus increasing the usefulness of OSM data set altogether. The workflow is scalable for whole Planet-OSM and other similar data set where feature is stored in XML format. Furthermore, an identified limitation is discussed in Section 3.6.2. We have also developed a web-GUI for data visualization (Figure 3.6). This short demonstration opens future research possibilities for improved OSM road length estimation as discussed in Section 3.7.

The rest of the sections of the chapter are documented as follows: • Related Work (Section 3.3); • Methodology (Section 4.4); • Study Set-Up (Section 3.5); • Results and Discussions (Section 5.5); • Conclusions and Future Work (Section 3.7).

3.3 Related Work

In recent years, the quality checkup of OSM data set has been done by plethora of researchers by comparing it with other governmental/proprietary data set ([43], [66]), by using indirect approaches like Linus Law ([42], by using contributors count as quality proxy ([84], [84], [78]), by developing intrinsic quality assessment parameter/tools ([82], [36], [31]) and by reviewing its change set dump file [5]. However, these kind of quality assessments basically work on raw data set by comparing accuracy of existing tags and do not assess derived values calculated after download post-processing. We could not find any online literature in peer-reviewed journal in this regard to use as benchmark. Although tackled problem and presented solution might appear trivial to advanced users in terms of complexity and novelty, we believe its usefulness for better services and data usage.

3.4 Methodology

A typical OSM curved road section consists of at least four or more number of underlying *nodes*. For two and three number of *nodes* it could be primarily treated as straight-line, if not an artifact. In order to piece-wise apply a curve fitting approach to a section user has to select four consecutive *nodes* at a time. Figure 3.3 shows how the fitting is done on step-basis on sample curved section. The initial four *nodes* are selected out of seven representing RJ-RJ section (RJ = Road Junction, Figure 3.3b). A piece-wise cubic parametric polynomial equation is derived to best fit these

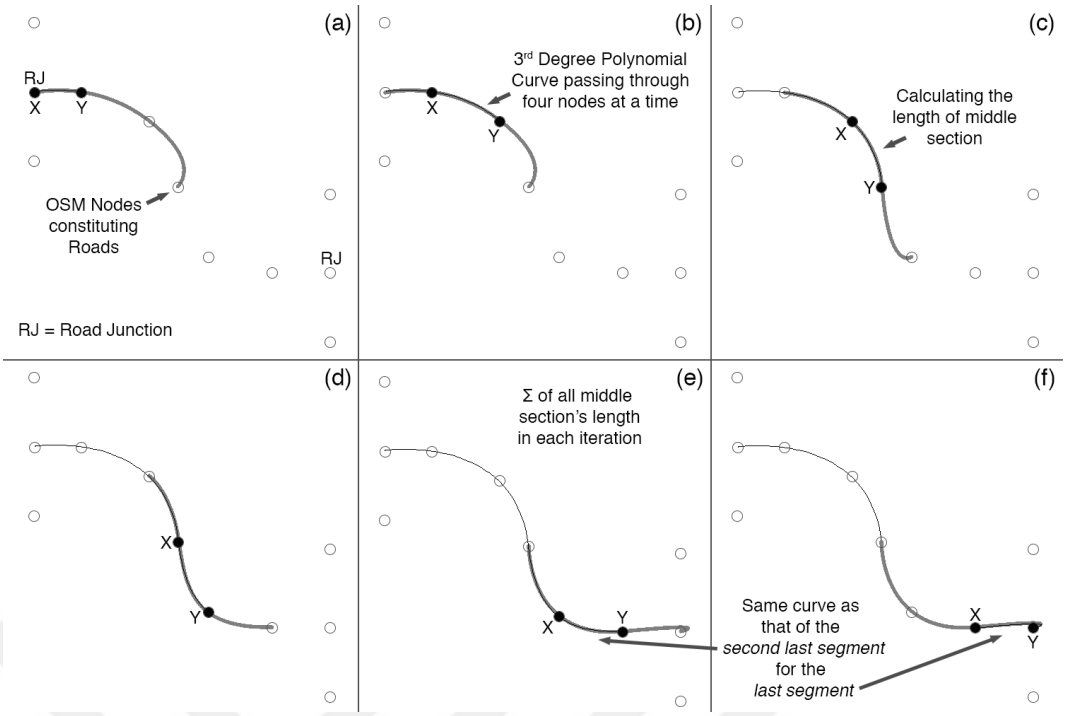


Figure 3.3 : A step-wise CL (Curved Length) estimation of sample road section using presented curve fitting approach.

selected four *nodes*. Finally, the mid XY section is selected for visualization and length estimation using derived integral as explained in Section 3.4.1. This step is performed programmatically (Section 3.4.2) on an iterative basis for all possible adjacent four *nodes*-pair for the road section under consideration. Each iteration derives the CL of XY for final summation (Figure 3.3). For the first and last XY section (Figure 3.3a,f), the cubic curve from immediate neighbouring four *nodes*-pair is used. Only mid-sections (XY) are used to obliterate unrealistic spikes/bends in road features that can be seen at *node* Y (Figure 3.3f). These spikes are inevitable to prevent because of them being at the junction. This way one obtains the overall CL value after $n-3$ iterations where n is the number of *nodes* in that road.

3.4.1 Derived Mathematical Equation

For a valid mathematical representation of road section, a parametric piece-wise cubic polynomial schema is adopted. The coefficients of this cubic polynomial are calculated for each consecutive quadruple points and mid section of each curve is represented with obtained formula. There are exceptions for the first and last quadruple points for which the same coefficients are used from the immediate neighbouring curve. The method ensures the value of consecutive curve segments on either side of the control point

to be same at control point but does not guarantee the value of first derivative to be equal. However, the results obtained by this method are analyzed visually and found acceptable. The method is chosen over a more widely used cubic spline interpolation for its simple implementation and cheap resource demand in order not to rule out the possibility of deployment on light hardware machines such as mobile devices or embedded systems that contain pretty limited computational resources.

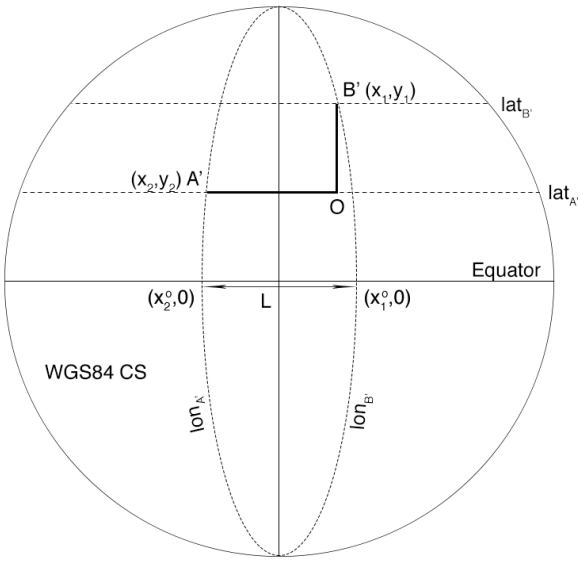


Figure 3.4 : A modelled Earth sphere in WGS84 CS with two given points (A' and B').

The following section derives a mathematical formula to calculate the length of a given cubic polynomial parametric equation, applicable in WGS84 Coordinate System (CS). Let's assume A, B, C, D to be four points with known latitude and longitude through which we want to pass a 3rd degree curve and further assume the parametric equation of this curve to be:

$$\begin{aligned} long(some-point) = x = f(t) = a_1t^3 + b_1t^2 + c_1t + d_1 \\ lat(some-point) = y = g(t) = e_1t^3 + f_1t^2 + g_1t + h_1 \end{aligned} \tag{3.2}$$

where, t is the parameter. It means that the point A, where $x = long(A)$ and $y = lat(A)$, is on a 3rd degree curve if and only if there is a value of t such that the above two equations generate that point. Since the coordinates of A, B, C, D are known from OSM, eq. 5.2 can be easily solved for constants a_1,...,h_1. This gives a parametric equation of

3rd order representing curved shape defined by points A, B, C, D. This equation is also used by us for visualization of section *BC* (Section 3.4.2). Now, given a continuous parametric function $x = f(t)$ and $y = g(t)$ of curve A, B, C, D, eq. 3.3 will determine its length based upon Pythagoras Theorem (PT) and Mean-Value Theorem within interval $[\alpha, \beta]$ by assuming the derivative to be continuous within the range [133].

$$CurveLength(units) = \int_{\alpha}^{\beta} \sqrt{\left\{\frac{df(t)}{dt}\right\}^2 + \left\{\frac{dg(t)}{dt}\right\}^2}\, dt \tag{3.3}$$

where, $df(t)$, $dg(t)$ and dt are respective differentials. Eq. 3.3 is valid only for a 2-D system where plane is in regular grid pattern, i.e. length$(x_n - x_{n-1}) =$ length$(y_n - y_{n-1})$, which is not the case with WGS84 CS. In latitude-longitude plane, length$(lat_n - lat_{n-1}) \neq$ length$(lon_n - lon_{n-1})$ and, therefore, eq. 3.3 is not directly applicable to OSM *node* set. Figure 3.4 represents two random locations in WGS84 CS. In order to use the PT for length estimation we need $B'O$ and OA' length. Because of the latitudes being mutually parallel $B'O = y_1 - y_2 = lat_{B'} - lat_{A'}$ equality holds true. As for OA' length we have to consider the sinusoidal shape of longitudes into account as they are not mutually parallel. Hence,

$$\begin{aligned} OA' &= x_1 - x_2 \\ &= \left\{\frac{L}{2}\right\}\cos y_1 - \left\{-\frac{L}{2}\right\}\cos y_2 \\ &= (x_1^0 - x_2^0) \times \left\{\frac{\cos y_1 + \cos y_2}{2}\right\} \\ &= (lon_{B'} - lon_{A'}) \times \left\{\frac{\cos lat_{B'} + \cos lat_{A'}}{2}\right\} \end{aligned} \tag{3.4}$$

Please refer to Figure 3.4 for notations of eq. 3.3. The $\left\{\frac{\cos lat_{B'} + \cos lat_{A'}}{2}\right\}$ component in eq. 5.4 is the Equirectangular Approximation (EA) factor which considers the non-parallelism of longitudes into account applicable for small distances ([134]). Hence, by incorporating EA factor into eq. 3.3, the new curve length equation applicable to WGS84 CS for section *BC* becomes:

$$CurveLength(radians) = \int_{B}^{C} \sqrt{\left\{\frac{df(t)}{dt} \times \frac{\cos\{g(t_B)\} + \cos\{g(t_C)\}}{2}\right\}^2 + \left\{\frac{dg(t)}{dt}\right\}^2}\, dt \tag{3.5}$$

where, t_B and t_C are the parameters for B and C points, respectively. Unfortunately, the above definite integral is not integrable in its current form and we need to use Simpson's rule [137] to solve it into disintegrated form which involves division of *BC* into n-number of segments. Mathematically, the larger the number of segment is the better the approximated integral value would be. The optimal number of segmentation for presented workflow is turned out to be 10, as discussed in Section 3.6.1 (Figure 3.7). Once CL for each segment is calculated, it is multiplied by the radius of Earth at that latitude to get the value in metric scale. Eq. 5.5 gives the radius of earth at any given latitude in meters.

$$\begin{aligned} \text{Radius of Earth}(lat_x) &= \left\{ \left[\frac{\cos(lat_x)}{a}\right]^2 + \left[\frac{\sin(lat_x)}{c}\right]^2 \right\}^{-0.5} \\ where, \tan(lat_x) &= \left\{\frac{c}{a}\right\}^2 \times \tan(lat_{mean}) \end{aligned} \tag{3.6}$$

where, lat_{mean} is the mean latitude of a given segment, i.e. $\frac{lat_{A'} + lat_{B'}}{2}$ for Figure 3.4, and a and c are the equatorial (6378137 meter) and polar (6356752.3 meter) radii of WGS84 Earth Ellipsoid. Iteratively, all segments are measured for all XY sections (Figure 3.3) to estimate the overall CL of given road section.

3.4.2 Code Implementation

The software part of this study should be examined in two halves. Firstly, the OSM road data is processed with native application to benefit from system resources as efficiently as possible and secondly, the results are visualized using a web application to benefit from platform mobility and development ease [139]. The native application is coded in an object oriented fashion using `C++` language by highly utilizing the curve fitting library previously developed by us. Section's length, cumulative road length (Figure 3.3) and frequency histogram (Figure 3.5) are calculated using mentioned code by exploiting data set acquired from OSM server. The application for visualization is coded using web technologies, mainly JavaScript for algorithm

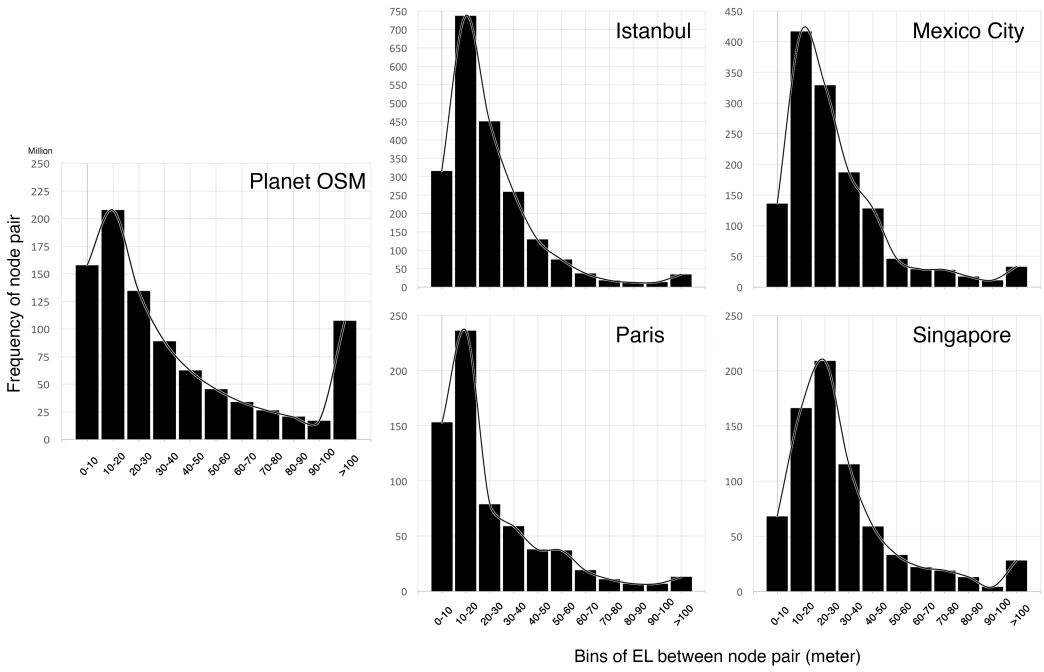


Figure 3.5 : Graph between frequency and *node*-pair EL (Euclidean Length) for selected cities and Planet-OSM.

implementation, HTML for GUI structuring and CSS for GUI visual aestheticism. The application inputs OSM XML file containing road data and displays improved curves on top of Bing Satellite Imagery powered by Leaflet v1.0.1 library [138]. The web-GUI is currently in development phase as more features are needed to enrich the user-experience.

3.5 Study Set-Up

In order to test the proposed methodology on real OSM road data set, around 100 sections are randomly selected from each four cities with large street network data set, i.e. Mexico City (Mexico), Paris (France), Istanbul (Turkey), and Republic of Singapore [84]. These selected cities represent urban setup at different latitudes. The attempt of this study is not to show case how much gain in road length a city graph might observe once the presented CL methodology is applied onto it but to exhibit how much gain a single road section might attain using this technique. The overall gain in length is primarily a function of street pattern too with grid pattern gaining least because of few curved sections and cul-de-sacs gaining most [77]. For selected road section's visualization JOSM [110] editor is used. Downloaded data is directly imported into the editor along with satellite imagery. Randomly selected sections,

Figure 3.6 : A visual comparison of proposed curve fitting approach (yellow line) with existing linear approach (red line).

suitable for analysis, are observed and corresponding IDs are noted down for later referencing. Since existing tools measure EL and presented methodology estimates CL, we need a reference value, ideally $\approx$ AL (eq. 5.1), for length comparison; although visually speaking CL is less erroneous than EL (Figure 3.6). So, in order to calculate AL we have manually traced over all randomly selected sections with high precision by mapping more than 50 *nodes* per 50 meter ground length, thus considering curvature factor extensively into account. For subsequent discussion, this length is going to be referred to as Ground Length (GL) and used as reference value.

3.6 Results and Discussions

In this section, we have discussed preliminary results obtained by applying proposed methodology to estimate length of given roads. Sections are divided into parts where we have tried to explain, first, the optimal number of segmentation for Simpson's rule, then to classify curved sections into three road categories. Each section follows figures to explain respective findings.

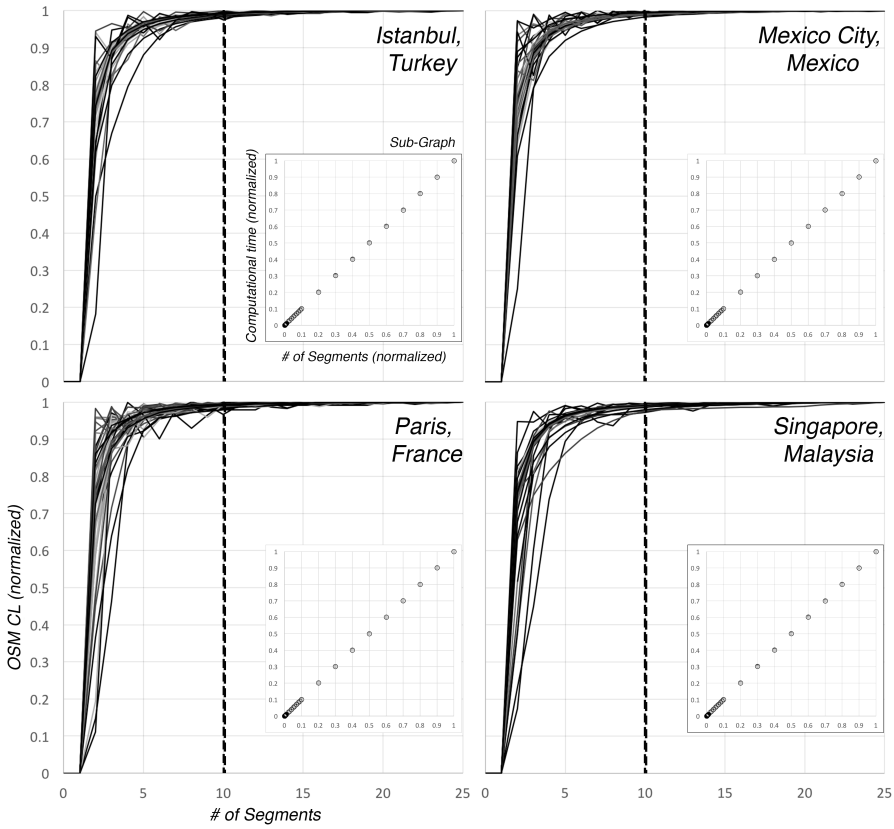


Figure 3.7 : A graph between the total number of segments and CL (Curved Length) for selected *way*s showing plateau at around 10 (dashed-line).

3.6.1 Optimal Number of Segmentation

As explained in Section 3.4.1, in order to solve eq. 3.5 using Simpson's rule one had to divide BC section into n-number of segments. Although precision of definite integral is directly proportional to the number of segments in Simpson's rule, practically speaking, taking large number of segments is computationally not possible. The sub-graph in Figure 3.7 shows a direct relationship between total number of segments and total amount of time taken by hardware to solve it. This Figure also presents the relationship between number of segments and CL for all sections taken from selected cities. It is observed that beyond 10 number of segments, CL attains a saturation point. For higher number of segments, it is computationally and precision-wise not beneficial. Although this testing is done on selected cities, similarity of frequency histogram (Figure 3.5) of different cities and Planet-OSM suggests that this observation could be extended to other world cities as well. Figure 3.5 shows the frequency histogram of *node*-pair EL for all four cities, including Planet-OSM. It can be seen that for selected

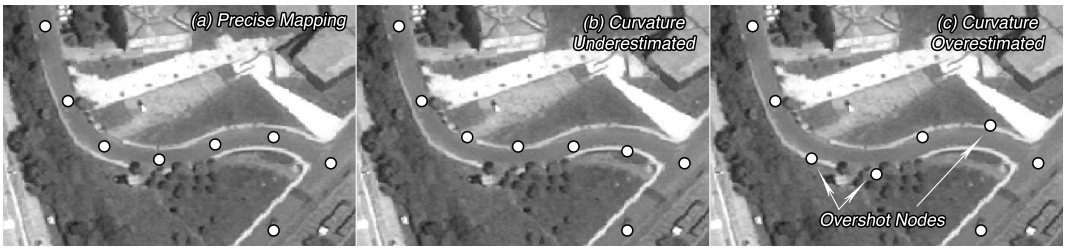


Figure 3.8 : Satellite imagery of a road section showing three different categories of mapping precision, as discussed in Section 3.6.2.

urban setups as well as for Planet-OSM, the *node*-pair EL with maximum frequency is between 10 to 30 meter. Since 10 number of segments is apt for given cities (Figure 3.7), it can be generalized for whole OSM data set as well since XY sections (Figure 3.3) are all mostly of similar EL. For all subsequent analysis for curve fitting the mid section, i.e. XY, is divided into 10 number of segments.

3.6.2 Precise, Underestimated and Overestimated Mapped Curved Road Sections

Another topic worth consideration is how strictly different road sections are mapped in given VGI project. As explained in Section 4.4, the primary hypothesis behind derivation for CL calculation is that all mapped *nodes* of given section are mapped onto the underlying satellite imagery as shown in Figure 3.2. By this the derived cubic curve will also fall onto the actual road section. But, as the scope and definition of VGI project says [34], anybody can plot these sections and, therefore, mapping errors do exist ([5], [66], [36], [43]). A close inspection of selected roads suggests that there are three different categories of mapped sections, classified by us as *Precise Mapped Section (PMS)*, *Underestimated-Curvature Mapped Section (UMS)*, and *Overestimated-Curvature Mapped Section (OMS)* (Figure 3.8a,b,c). PMS is a category where respective *nodes* are precisely mapped onto the road section in sat-imagery (Figure 3.8a). UMS, on the other hand, are cases where mappers have underestimated the road curvature because of limited mapping experience, poor road visibility or laziness and added *nodes* in more linear fashion (Figure 3.8b). Finally, the third category, i.e. OMS (Figure 3.8c), are cases where road curvatures are overestimated by mapper and, therefore, as can be seen in Figure 3.8c, there are two bends where *nodes* get overshot beyond curvature. From the derivation point of view (Section 3.4.1) presented approach is applicable for PMS and UMS but not for OMS as it brings in more error than already is there by Euclidean formulation. This is the only

known limitation of presented methodology. It is necessary to check how precisely a mapper has mapped roads onto the underlying imagery. An automated way to identify these OMS roads is beyond the scope of this study, although this definitely opens new research topics for us. For current research, it is necessary to manually filter out these roads from selected ones before applying any curve fitting.

3.6.3 Removing OMS

In order to identify OMS, graphs are plotted between Error and GL for all selected sections from four cities (Figure 3.9). GL is the EL precisely mapped by us (Section 3.5) and Error is the difference between GL and EL for each section. Keeping the following relationship in mind:

$$EL \leq CL \leq GL \approx AL \tag{3.7}$$

where, EL, CL, GL and AL are Euclidean, Curved, Ground (plotted length) and Actual Length, respectively. Error value, i.e. $GL - EL$, must remain positive for PMS and UMS. However, for OMS it should be negative. Therefore, sections marked by triangular marker on the positive side of y-axis represent PMS (lying close to x-axis) and UMS (Figure 3.9) and sections marked by ovals represent OMS. For subsequent analysis OMS sections are not used and needed to be accurately mapped first. Automated identification of these roads is necessary using image analysis and/or proprietary data comparison ([166], [43]) and for current study this is beyond the scope.

3.6.4 Comparison between Euclidean and Curve Fitting Methodology

We have calculated the EL and CL of selected road sections that belong to either PMS or UMS (Section 3.6.3) by using existing OSM tool and developed algorithm, respectively. Calculated values are compared with GL that was estimated after precise mapping. Figure 3.10 is a set of graphs showing absolute Errors introduced by EL and CL formulation for selected roads. Axes originating from the center represents Error in meter, i.e. $GL - EL$ and $GL - CL$, from existing and developed methodology. Vertices represent each analyzed road for each city. Dashed curve exhibits Euclidean

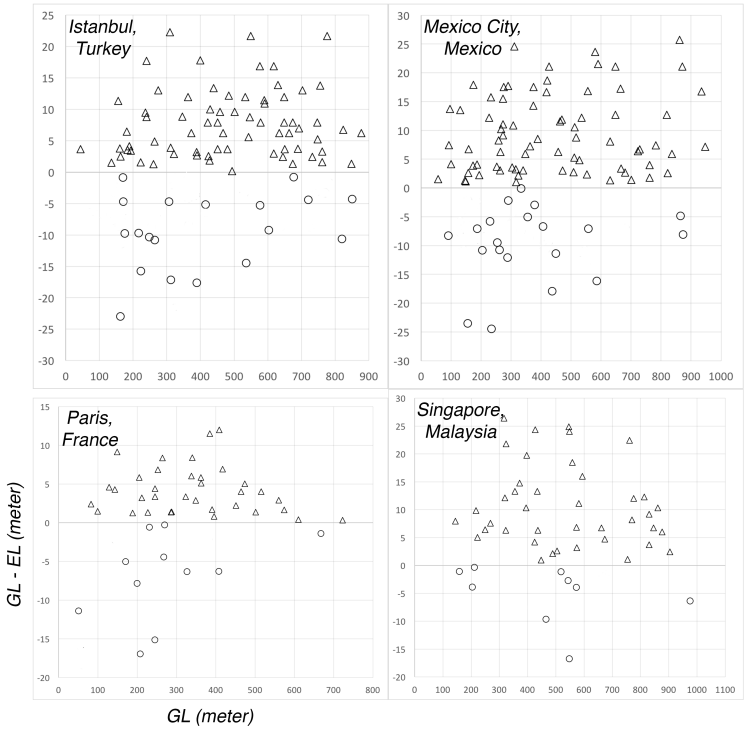


Figure 3.9 : A graph between GL (Ground Length) and Euclidean Error (i.e. GL - EL) for selected *ways* for all four cities.

method whereas solid curve represents curve-fitting. It is clear from all four graphs that CL is more close to GL than EL. For all cities the error by curve fitting approach is smaller than by Euclidean with few sections being equal. For PMS and UMS, the presented methodology brings forth more precision in terms of visualization and attribution. It must be noted that at least one road gives negative error (Figure 3.10) because of it being too close to the x-axis (Figure 3.9). These sections are nothing but a transition case between PMS and OMS and, therefore, curve fitting could be applied. However, for sections lying far enough to the positive side above x-axis (Figure 3.9) it is preferable to use curve fitting for better results.

Another way to observe similar behavior is by plotting a stock bar graph between Errors and GL for selected roads (Figure 3.11a). X-axis shows the GL of analyzed curved section in meter and y-axis shows the absolute Error in EL and CL. In stock bar graph, the lower limit is error in CL and the upper limit error in EL. It is clear that for all cases the curve fitting approach bestows improved results which can be observed by noticing the absence of any solid bar (Figure 3.11a). It is also clear from the Root Mean Square (RMS) Error graph (Figure 3.11b) where for all cities the overall RMS Error is less for CL. It is evident by observing the slope of the trendline of the two methods that

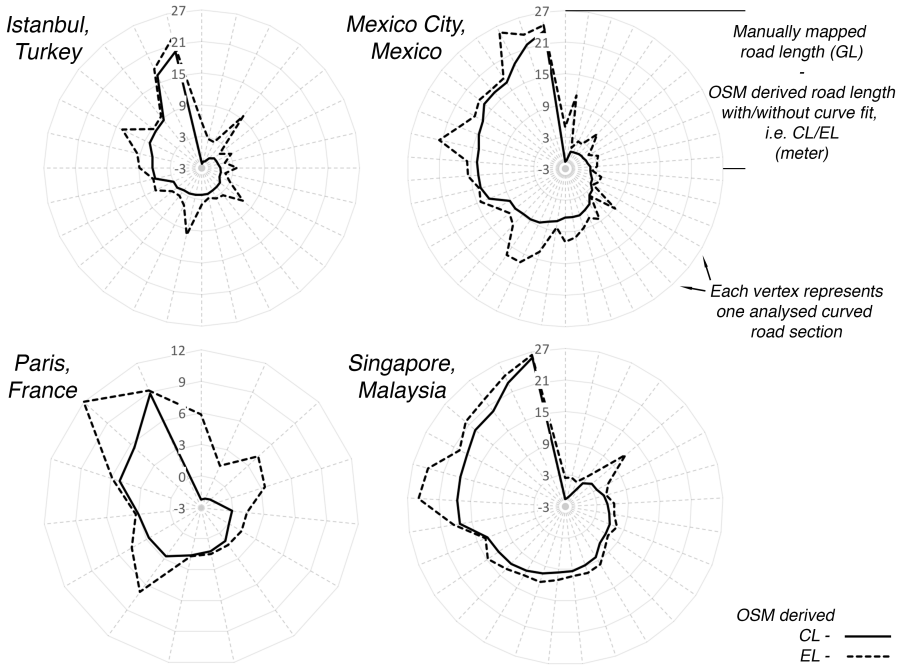


Figure 3.10 : Graphs comparing the absolute Error of CL (Curved Length) and EL (Euclidean Length) for tested roads for given cities (Section 3.6.3).

with increase in AL error of EL gets bigger than error of CL (Figure 3.11c). It is noted that, except Paris (France), for all cities the slope value corresponds to curve fitting approach is smaller than the other one. This demonstrates how length estimation gets more inaccurate for longer roads while using existing tools. Exceptional behavior of Paris could be marked by its least number of analyzed sections, i.e. only 35 (Figure 3.9), and further analysis is required.

Finally, Figure 3.12 is a plot showing the percentage gain in length value after migrating from Euclidean to curve-fitting approach. This comparison is necessary to quantify the value brought in by proposed algorithm. For all four tested cities, the average % length improvement randomly fluctuates between [0.50-0.90]. The numbers might look trivial for a while but when compared on a city/country level scale with millions of road sections the impact caused by them would be staggering. Overall % length improvement for all tested roads sums up to 0.70%. Those couple of data points for Paris, Istanbul and Mexico that are above the general data point population (Figure 3.12, solid points) are cases where road section is highly curved and its mapped precision in OSM data is poor. These two factors have collectively made the percentage length improvement for them more than 1.5%. A detailed analysis of Planet-OSM is

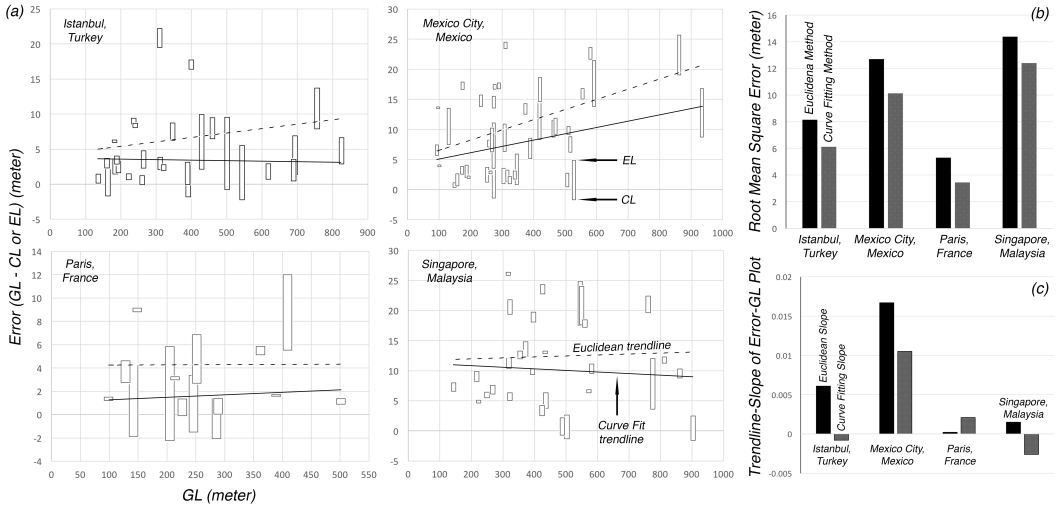


Figure 3.11 : **(a)** Graphs showing direct relationship between absolute EL (Euclidean Length) Error and GL (Ground Length). **(b)** Overall RMS Error from Euclidean and Curve Fitting formulation for selected cities. **(c)** Higher trendline slope of EL Error for all cities, except Paris (discussed in Section 3.6.4).

believed to give us overall picture of current state of OSM and how better its vector data could be used for services where road length value is used.

We have discussed the suitability of proposed methodology over existing ones by comparing various graphs. Proposed formulation is recommended for existing tools for better road length estimation for improved OSM street data usage. However, how precisely a section is mapped by a mapper (Section 3.6.2) is still a matter of concern for applicability of this workflow for Planet-OSM and future research might improve presented tool (Section 3.4.2) or provide a better one to control OMS road sections.

3.7 Conclusions and Future Work

OSM and other VGI data opens copious doors to deploy geo-services by leveraging their freely downloadable vector data. However, data quality has always been a concern in these services. Plethora of online literature is available discussing its usefulness for spatial use-cases, although little or no study related to quality estimation of derived attributes is available. One such attribute is length value of road feature that one derives after data download. Existing open source tools limit curvature factor of these roads and measure lengths using Euclidean approach (Section 3.2.1). This brings error that is necessary to be ameliorated for any usage. The attempt of this study is to derive and propose a piece-wise cubic parametric polynomial curve fitting approach

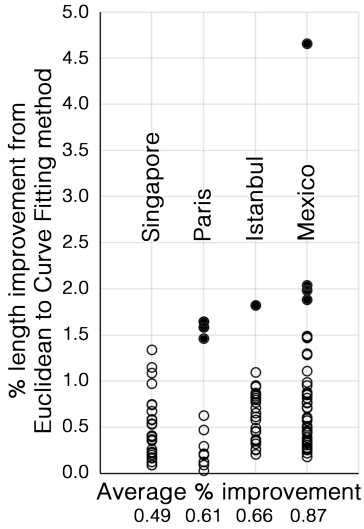


Figure 3.12 : Graph showing the percentage gain in length value of all tested roads by shifting to the Curve-Fitting. Overall percentage gain is found to be 0.70%.

to consider road curvature during download post-processing in order to extract more accurate geometrical attribute. This easy to understand and apply approach brings forth better way to visualize and calculate road length and, as discussed in Section 3.6.4, is identified to be better over existing approach that is solely based upon PT. An overall 0.70% gain in length value is observed by using proposed curve-fitting methodology over existing one. It is, therefore, suggested to implement in existing tools. However, it should be noted that because of various level of precision of mapping presented approach should not be applied to OMS (discussed in Section 3.6.2). This is the only observed constraint of this algorithm. Another observation is that in Planet-OSM the EL between each adjacent *node*-pair generally lies between 10 to 30 meter and, therefore, 10 number of segments for each *node*-pair is most suitable for any CL calculation (Section 3.4.1, eq. 3.5).

Current research opens few potential research doors to improve OSM derived road length. We have also developed a public web-GUI to visualize these redrawn curves on top of underlying Bing imagery and currently the development is under-progress (Section 3.4.2). This GUI will assist users to better analyze how good presented methodology is for any geo-data where features are stored in OSM XML format. Future work might include developing an automated way to identify the discussed OMS sections from raw OSM file or developing a way to statistically identify overshot *nodes* (Figure 3.8) to fix them on the fly. In this study, we have manually mapped

roads to obtain GL for referencing purposes but cross validation with other proprietary or governmental data is equally conducive for better comparison. Another work might include testing presented workflow on other major world cities in order to help develop a more generalized commentary and to better understand limitations.

4. A NEW SPATIAL APPROACH FOR EFFICIENT TRANSFORMATION OF EQUALITY - GENERALISED TSP TO TSP

4.1 Abstract

The Equality - Generalized Travelling Salesman Problem (E-GTSP), which is an extension of the Travelling Salesman Problem (TSP), is stated as follows: Given a groups of points within a city, like banks, supermarkets etc., find a *Hamiltonian* cycle visiting each group exactly once. It is an NP-hard problem which can model many real-life combinatorial optimization scenarios like planning, logistics, etc. more efficiently than TSP. An E-GTSP instance can be successfully transformed into its equivalent TSP instance before solving with a given TSP solver. This paper presents 5 novel spatially driven search-algorithms for possible transformation which consider the spatial spread of points in a given urban set-up. Algorithms are tested over 15 different cities, classified with their street-network's fractal-dimension, with 5 instances of different group-counts each. The obtained results point out that the R-Search algorithm, which selects station (i.e. selected point) from each group based upon its radial separation with respect to the start-end point, is the best search criterion for any given city or group-count instance with an average length error of 8.8%. This will help geo-developers to answer complex routing-queries and researchers to solve graph problems from a spatial perspective.

4.2 Introduction

The Travelling Salesman Problem (TSP) is by far one of the well-known and extensively studied combinatorial optimization problem, used as a benchmark for new developments for decades. It demands to find the shortest (in terms of length, time, or any custom cost) route that visits each vertex in a given set exactly once before returning to the starting vertex. Formally, it could be stated as follows: Given a directed/undirected graph $G = (V, E)$ with set of vertices V and set of weighted edges E, find the shortest path between start vertex s and end vertex e (e could be same as

s for closed path) that visits each vertex for a given set $V' \subseteq V - \{s, e\}$ exactly once. It is equivalent to finding the minimum-cost *Hamiltonian* cycle in G [46], which is an NP-hard problem. It can be formulated as an integer linear program. It has huge applications in ranging areas including vehicle routing, communication networking, sequencing and scheduling, to name a few [67], and therefore has always remained a great source of attraction from varied disciplines, especially in last three decades. [91] has documented a detailed classification of different types of TSP and their possible solutions.

A variety of heuristic and tabu search algorithms to tackle it are developed by researchers in the past [148] like • Nearest Neighbour, also known as Greedy algorithm, which was the first of its kind [40], • Clarke-Wright heuristic [16], • Minimum Spanning Tree heuristic (eg. Kruskal's algorithm) [95], and • Christofides heuristic [37]. All four approximate algorithms above are constructive, i.e. a tour is improved iteratively. Other algorithms include • K-OPT [20], • Tabu search [32], • Simulated Annealing [59], • Held-Karp lower bound [157], • Lin-Kernighan algorithm [48], • Lin-Kernighan-Helsgaun [49], and • Cutting Plane and Branch-Bound techniques. Readers are furthermore encouraged to visit [141] for a quick and suffice in-depth introduction. GIS users may also read [21].

The Generalized-TSP (GTSP) or Set-TSP or Travelling Politician Problem is a useful exemplary for selection and sequence related problems. It is one practical direct extension of TSP introduced by [51], where V is further segmented into n number of groups and task is to find a minimum-cost route passing through at least one vertex from each group before reaching to the destination. For almost all inherently hierarchical real-world problems, it offers a more precise model than TSP. The GTSP could be mathematically defined as follows: Let $G = (V, E)$ be a graph where $V = \{v_1, v_2, ..., v_n\}$ is the set of vertex, $E = \{(v_i, v_j) \mid i \neq j; v_i, v_j \in V\}$ is the edge set, and $W = \{w_{ij}\}$ is the non-negative cost or weightage defined on E. If E is undirected then directions are irrelevant and $(v_i, v_j) = (v_j, v_i)$. Now, in the GTSP V is partitioned into x mutually exclusive and exhaustive groups such that $V^g = \{V_1, V_2, ..., V_x\}$ and $V = V_1 \cup V_2 \cup ... \cup V_x$ with $V_\alpha \cap V_\beta = \varnothing$ for all $\alpha, \beta = 1, 2, ..., x$ and $\alpha \neq \beta$. It asks to determine the shortest *Hamiltonian* circuit passing through each group *at least once* (introduced independently by [51], [97], and [101]) or *exactly once* (introduced by

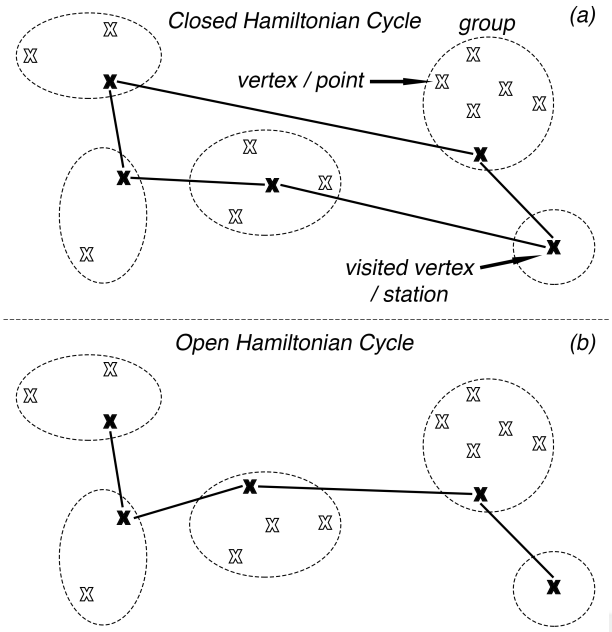


Figure 4.1 : Representation of one possible (a) closed and (b) open *Hamiltonian* cycle in a given symmetric E-GTSP instance.

[87], and [62]). If the matrix W is symmetrical, i.e. $w_{ij} = w_{ji}$ for all $i, j = 1, 2, ..., n$ and $i \neq j$, the problem is prefixed with *symmetric*, otherwise *asymmetric*. This results to many vertices from each group left being visited. The *exactly once* variant of the GTSP is also known as Equality-GTSP or E-GTSP [50], where the shortest route contains exactly one vertex, i.e. station, from each V^g group. The E-GTSP is an NP-hard problem [58], as it reduces down to the famous TSP (also NP-hard) whenever $(|V_\alpha| = 1 \; \forall \; \alpha = 1, 2, ..., x)$ condition met, i.e. individual groups become singleton. Figure 4.1a represents one possible closed *Hamiltonian* cycle, also called as a *g-tour*, which visits exactly one vertex from each group before returning to the starting vertex, and therefore this vertex could be anyone from any group. In Figure 4.1b the cycle is open as the start and end groups are discrete.

[61] has explained how GTSP can be used as one versatile and elegant tool to model different classes of combinatorial optimization problems like covering tour problem, material flow system design, post-box collection, stochastic vehicle routing, and arc routing. It has prime relevance in location-based problems, urban planning, postal routing, logistics, microchips manufacturing, telecommunication problems, and railway track optimization. [8] and [87] have also discussed similar applications in detail. The complexity of GTSP has led to the advancement of various heuristic and metaheuristic algorithms like Ant Colony algorithm [161], Memetic algorithm

([10] and [38]), Variable Neighbourhood Search algorithm [53], Random Key Genetic algorithm [100], Reinforcing Ant Colony system, Efficient Composite heuristic [93], etc. However, because of bearing a mathematical origin, these algorithms are onerous to replicate by general users for TSP models representing GIS problems, primarily vehicle-routing.

In this study, the authors are primarily interested in E-GTSP to TSP transformation, which is a logical approach to solve E-GTSP for vehicle routing scenarios, as there already exists a large variety of exact and heuristic methods to solve TSP ([39], [65], [49]) like the state-of-the-art Lin-Kernighan-Helsgaun TSP solver [50]. Since the underlying model represents an urban street setup, authors have tried to think this transformation from an spatial point of view instead of mathematical which was being done by other researchers so far. Five different possible search algorithms are suggested and tested on real-world OpenStreetMap (OSM) street-network data to decide the best possible search criterion. Different group-counts (i.e. $|V^g| = 1,2,3,4,5$) are employed to test proposed algorithms at different complexity levels. Generated results are presented in graphical form and discussed in depth in the Results and Discussions Section 5.5.

To the best of authors' knowledge, this kind of spatial transformation of E-GTSP to TSP is first of its kind and no related work is available in any peer-reviewed online literature. It has opened-up future research possibilities to improve existing search algorithms or derive better one. In the following sections, the proposed search algorithms are explained and tested on OSM road data-set. Finally, a commentary on results are provided in the Conclusions and Future Work section.

4.3 Transformation of E-GTSP to TSP

The GTSP was introduced by [51], [101], and [97] through record balancing problems aroused in computer design. It is one of the few optimization problems that has been studied extensively ([63], [87]). Researchers ([22], [64], [87], [71], [9], [88]) have also elaborately attempted to transform it to standard TSP with many of them exploiting dynamic programming techniques ([28], [6]), which design to disintegrate complex problems to a set of relatively simple sub-problems and solve them independently. The shortcoming of this transformation, however, is that it increases the problem's

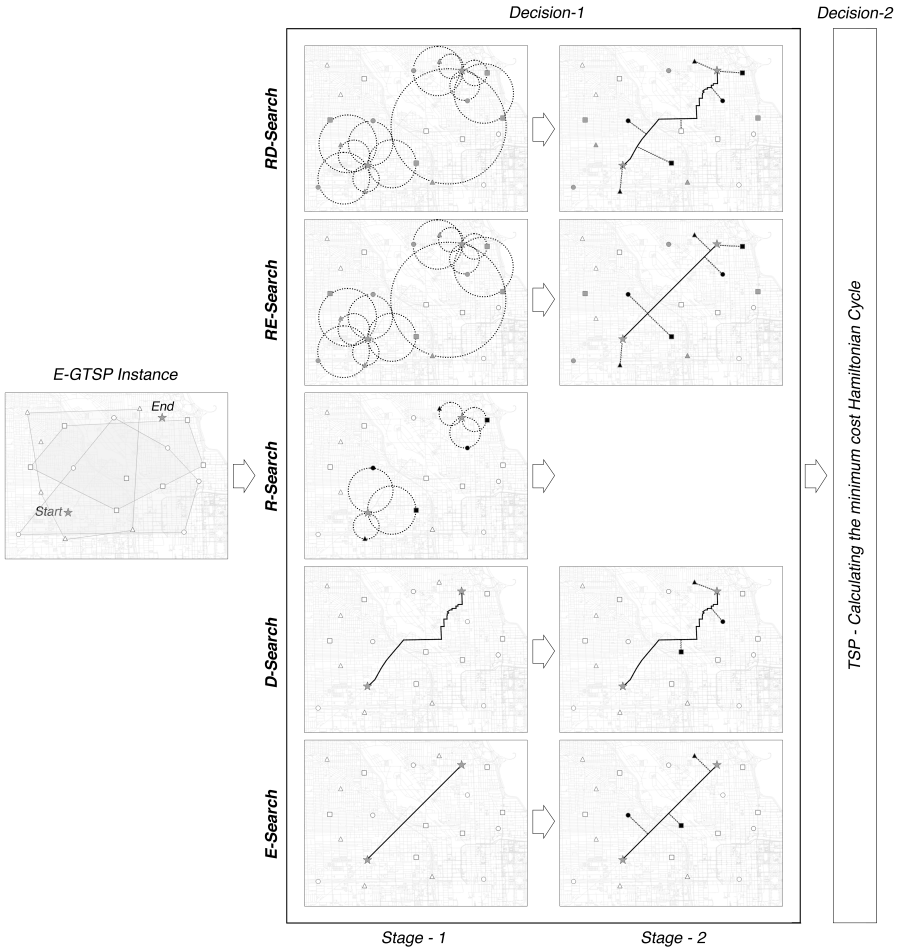


Figure 4.2 : Diagrammatic representation of 5 different proposed search algorithms.

dimension dramatically. Therefore, although theoretically it is possible to solve GTSP by converting to corresponding TSP, the new increased problem size ruins its practical feasibility. [160] have tried to unify both the problems into one uniform state by introducing generalized chromosome design. [28] have proposed a branch-and-cut algorithm to solve its *symmetric* version to optimality. However, these attempts are exclusively mathematical to apply and may intimidate amateur users to reproduce.

Dynamic programmatically, an E-GTSP's solution, representing a vehicle-routing model, is based upon deciding the following two **decisions** in written order:

- Selection of a vertex subset V^s, also termed as station, such that $V^s \subseteq V$ and $V^s \cap V_\alpha = 1$ for all $\alpha = 1,2,...,x$. *Note* that, $|V^s| = |V^g|$.
- Calculation of the minimum-cost *Hamiltonian* cycle in subgraph $G^s = (V^s, E^s)$ of G produced by V^s.

Since in our E-GTSP model the start and end vertices are different, the calculated *g-tour* is going to be an open one (Figure 4.1b). The presented search algorithms do not increase the problem's size by increasing vertex or edge count. Considering vertices' spatial distribution with respect to the start-end vertex, it is much easier to filter-out distant and possibly sub-optimal vertices from each group. In this article, *points* represent all possible vertices of V including start/end, *group* represents each set of vertex from V exhibiting one particular attribute, and *station* represents the selected point from each group (Figure 4.1).

4.4 Methodology

The main objective of this article is to answer **decision 1** (Section 4.3) for E-GTSP models which represent finding the shortest route from start to end point visiting exactly one point, i.e. station, from each mutually exclusive and exhaustive groups of points, like shops, offices, etc., within a city. Five different search criteria to select exactly one point from each group depending upon the start/end point are coined here (Figure 4.2). Although a number of such different algorithms are possible with little tweaks, authors believe selected ones to be mutually absolute and cover a whole range of search possibilities. Since this kind of spatial transformation is first of its kind, no other spatial approach is discussed in other literature. Primarily, they all consider the spatial spread of points in a given urban set-up, which is the first of its kind, to select the best possible one from each group for a given E-GTSP instance.

4.4.1 E-Search and D-Search

Euclidean-Search (E-Search) is the first search criterion based upon the euclidean line between start-end points in a given instance. As can be seen in Figure 4.2, the basic approach behind it is to connect both the start/end points by a straight line and select the closest point from each group with respect to this line. It involves two stages before reducing E-GTSP to TSP. It is $O(n)$, where $n = |V|$.

Dijkstra-Search (D-Search), on the other hand, involves calculating Dijkstra route, instead of euclidean line, between given start/end points before selecting the closest point from each group with respect to that route (Figure 4.2). Computationally it

is more expensive than E-Search, with $O(n^2)$. Algorithm 1 is an easy to grasp pseudo-code of the above two search algorithms.

4.4.2 R-Search

Radial-Search (R-Search) is the third methodology to select stations based upon their radial distance from both the start/end points for a particular E-GTSP instance. Closest point from each group is selected twice, making $|V^s| = 2 \times x$ (Section 4.3), which leads to 2^x different possible TSPs, making **decision 2** computationally expensive (Figure 4.2). There is only one stage in **decision 1**. It involves an O-complexity of $O(n)$ and Algorithm 2 represents its pseudo-code. It should be noted that, unlike E-Search and D-Search, it evaluates points lying outside the proximity of start-end region more efficiently (white region in Figure 4.7).

4.4.3 RE-Search and RD-Search

Finally, the last two search algorithms are a hybrid of R-Search and E-Search (RE-Search), and R-Search and D-Search (RD-Search). They first ask to find the radially closest two points from each group with respect to both the start/end points independently, making $|V^s| = 2 \times 2 \times x$. Subsequently, they reduce their V^s size to half by selecting closest stations from both the ends with respect to the euclidean line (for RE-Search) or Dijskstra route (for RD-Search), Figure 4.2. They have the complexities of $O(n)$ and $O(n^2)$, respectively. The second stage, thus, leads to the total of 2^x possible TSPs, similar to R-Search. Algorithm 3 represents their pseudo-codes.

Once estimated the best possible station, i.e. optimal V^s, from each group for minimum-cost open *Hamiltonian* route for a given instance, it is solved by Simulated Annealing [59] TSP Solver. Although authors personally have used this algorithm to solve reduced TSP (**decision 2**), one is free to select from among any available options (Section 5.2), as it will only going to affect all five search algorithms equally. Presented results in the Results and Discussion Section 5.5 are more relative in nature.

4.5 Study Area and Data-Set Used

In order to test all proposed search algorithms, real-world OSM street-network data were used. 15 cities belonging to five different fractal dimension (frac-D) bins were

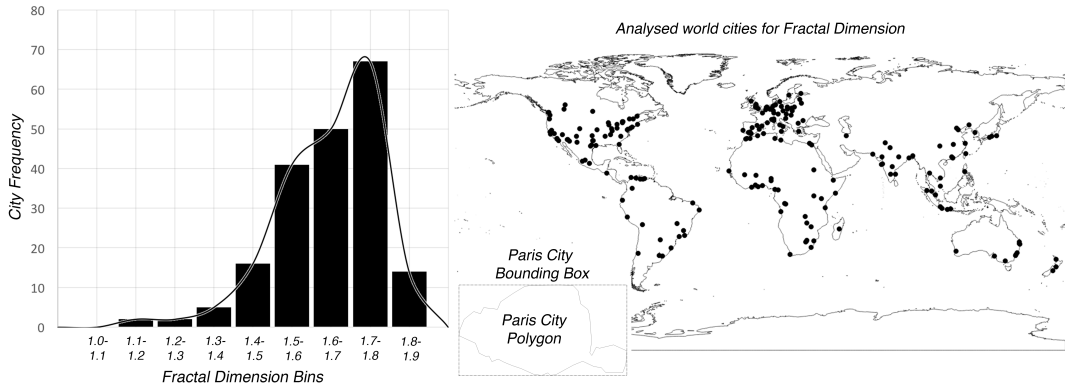


Figure 4.3 : Histogram showing the frequency of cities (marked on the world map *right*) for each Fractal-Dimension bin.

selected for analysis out of 210 cities worldwide, a sorted collection of which was downloaded from [142], (Figure 4.3). Increasing frac-D represents increasing road density within a given region, with 1-D represents area with only one road section and 2-D represents area completely filled with networks. Bin size in Figure 4.3 is intentionally kept small, i.e. 0.1, for finer fractally resolved classes. **F**ractal **D**imension **C**alculator is used to calculate the dimension of each city [143]. Table 5.1 contains all analysed cities sorted-out with increasing frac-D value along with corresponding decisive attributes derived from OSM vector data.

OSM data were downloaded from its OverpassAPI [122] covering the bounding box of each city, derived from [144]. All 105 start-end point pairs are randomly generated and 5 different kind of points for each group making one particular E-GTSP model are randomly selected from OSM point data to closely depict physical stops on the ground like shops, supermarkets, etc. In total, only 5 such groups are generated representing different group-counts as higher $|V^g|$ count dramatically increased brute-force processing time for optimal V^s selection for a given instance. The whole processing was done in python language. Instead of using sample instances from existing libraries, like GTSPLIB [140], which do not represent real-world scheme, authors have created their own E-GTSP instances from downloaded vector data. This provides an opportunity to test presented algorithms for better understanding and applicability for real-services. Authors also provide tested instances along with optimal V^s info, download-able from Github [145], for other researchers to carry out related studies in order to escape brute-force processing time which took months to finish. 5 E-GTSP instances with different $|V^g|$ value are used for each selected city to

Table 4.1 : General statistics of selected cities (Figure 4.3) for proposed search algorithm's test.

ID	City Country	Fractal Dim.	Area km^2	OSM # Vertex	OSM ρ Vertex	OSM # Edge	OSM ρ Edge	OSM $\sum$Edge
1	Hargeisa Somalia	1.400	42332	7267	0.17	10839	0.26	7782
2	Antananarivo Madagascar	1.413	42291	26955	0.64	33188	0.78	10455
3	La Paz Bolivia	1.432	14454	55239	3.82	83209	5.76	11012
4	Nairobi Kenya	1.511	84	3386	40.3	4469	53.2	482
5	Mexico City Mexico	1.517	1153	6839	5.93	7819	6.78	1241
6	Las Vegas USA	1.537	81440	154289	1.89	205164	2.52	40971
7	Seoul S-Korea	1.609	226	11764	52.0	16301	72.1	1394
8	Edmonton Canada	1.627	181262	95083	0.52	127203	0.70	56965
9	Calgary Canada	1.630	96779	130982	1.35	185201	1.91	49842
10	Amsterdam Holland	1.730	1152	62630	54.4	88977	77.2	8048
11	Brussels Belgium	1.745	943	54941	58.3	75833	80.4	7273
12	Delhi India	1.779	5078	198972	39.2	270302	53.2	27063
13	Dallas USA	1.800	4688	116034	24.7	166882	35.6	26376
14	Milan Italy	1.802	8043	265604	33.0	356506	44.3	36666
15	Munich Germany	1.811	1817	188387	103.7	253015	139.2	17380

run discussed algorithms for various complexities. Vital observations regarding best search algorithm and its behavior with increased complexity are discussed in Section 5.5.

4.6 Results and Discussions

In order to test the above five proposed search algorithms for efficient E-GTSP to TSP transformation, 15 cities are selected depending upon different levels of street-network pattern and density (quantified by frac-D, Table 5.1), for which data were obtained from OSM API. For each city, 5 different instances, i.e. $|V^s|$, modeling varied complexity of E-GTSP are created and tested for given algorithms. Figure 4.4a is a 3D-plot between *different search algorithms* used, *different number of stations to be visited*, i.e. $|V^g|$, and *different type of street-networks*, where each colored circle

represents the average fractional error (average of fractional errors coming out from all 105 analysed start-end pairs) in E-GTSP route-length estimated with respect to the optimal route (by brute-force). There are 375 (15`cities` $\times$ 5`group-counts` $\times$ 5`algorithms`) colored circles, with black circles representing errors more than 20%. It is clear from the horizontal color-shade shift that irrespective of the choice of a given search algorithm the percentage error in route length is bound to increase with higher $|V^s|$ instances (marked with big white arrow Figure 4.4a). It is an expected behavior in combinatorial optimization problem. It is, therefore, needed to slice the whole plot down into 5 different cross-sections for each group-count for astute observation. Figure 4.4b represents 5 graphs between *different search algorithms* and *different type of street-networks* belonging to different $|V^s|$ values. The black-boxed circles mark the lowest fractional error for that row. It can be seen that for lower $|V^s|$, the D-Search algorithm gives the lowest percentage error for maximum number of cities irrespective of their frac-Ds. It has drastically outperformed over other approaches for one group-count, with no win for RE-Search. However, this out-performance gets drifted towards R-Search for higher group-counts, with win of similar intensity at $|V^s| = 5$ as that for D-Search for $|V^s| = 1$. It is an interesting observation which suggests that for complex E-GTSP scenarios it is better to select radially close points from each group with respect to the start-end points than to find one closest to the corresponding Dijkstra-route. Another important observation is that in spite of bearing a more sophisticated algorithmic design for RE-Search and RD-Search, their performance was below average. A thought provoking notion which could be derived here is that computational complexity does not always mean better precision.

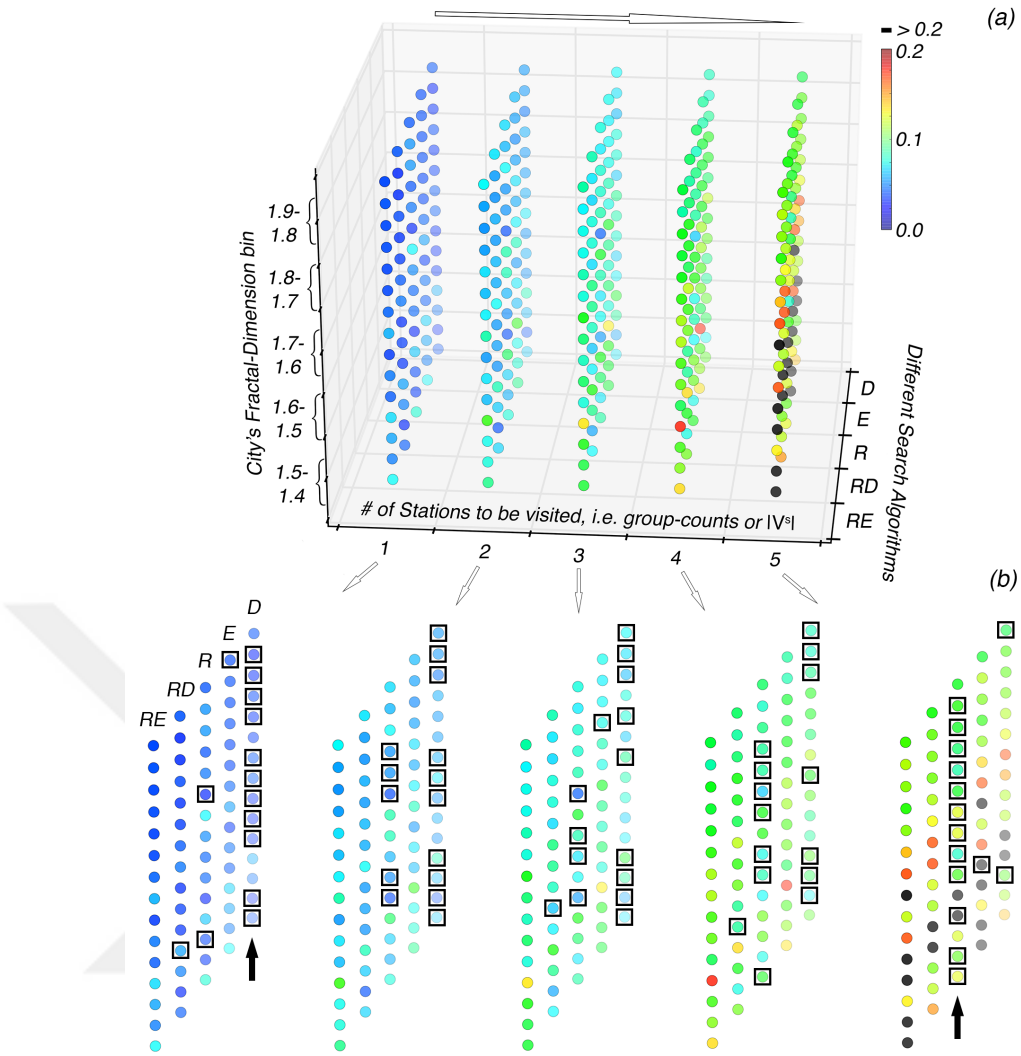


Figure 4.4 : 3D-graph between *Different Search Algorithms*, *Number of Stations to be visited*, i.e. $|V^s|$, and *City's Fractal-Dimension bin*.

Overall, out of 75 tests (15`cities` $\times$ 5`group-counts`) D-Search has outperformed 39 times, E-Search 3 times, R-Search 30, RE-Search 0, and RD-Search 3 times. Quantity-wise, D-Search is the best one for analyzed instances. However, it is important to realize that these wins are quantized and do not give any numerical idea of winner's gain over other runner-ups, making it necessary to compare results from a more absolute point of view, Figure 4.5.

Figure 4.5a contains 5 plots displaying *average fractional error* on y-axis for each city ordered according to Table 5.1 with each one for one particular $|V^s|$. Each colored circle represents one distinct search algorithm, mentioned in the legend. In order to make plots comparable, red (D-Search), and green (R-Search) circles for different cities are connected linearly. It can be seen that for $|V^s| = 1$ the red-line

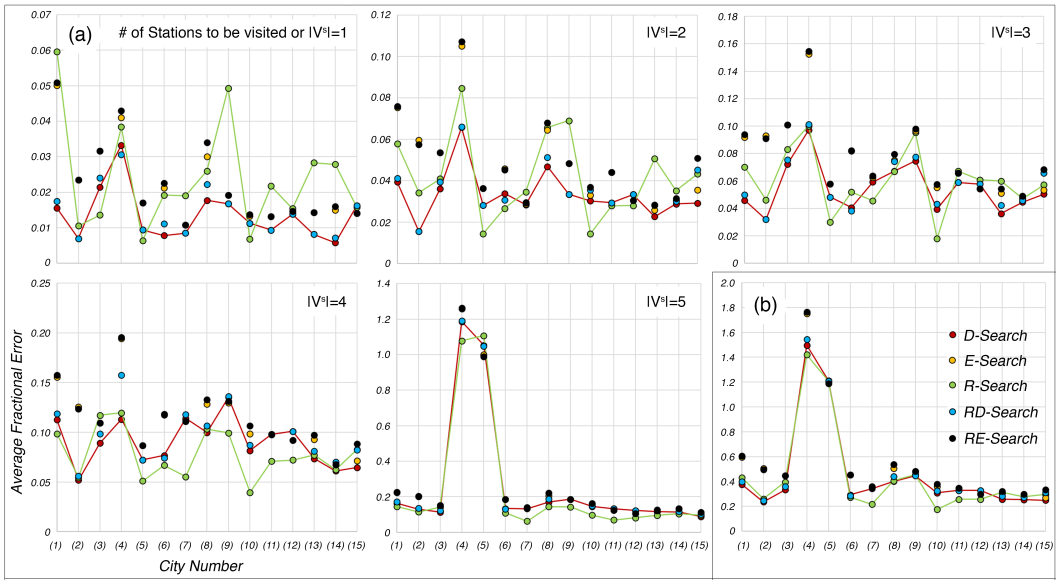


Figure 4.5 : (a) Scatter plot between the *Average Fractional Error* and *City Number* (representing city, Table 5.1) for all $|V^s|$ instances. (b) Y-axis, here, represents the summation of *Average Fractional Error* from all $|V^s|$ instances.

is almost always below the green-line (Figure 4.5a), which shows its out-performance. For higher $|V^s|$ this pattern gets reversed with green-line being below the red one. We have already mentioned this observation in Figure 4.4b. However, Figure 4.5a allows us to compare them absolutely. Figure 4.5b is a summation of *average fractional errors* from all $|V^s|$ instances for each city. It is clear that the green-line lies below the red one for most of the cities. This makes R-Search, accuracy-wise, the best possible overall search-algorithm for E-GTSP to TSP transformation although D-Search was quantity-wise better. In vehicle routing problems, absolute route length value, representing route-cost, acts as the most vital attribute quantity for any route selection process and therefore R-Search would be favoured over D-Search. Figure 4.6 gives the percentage route-length error one is supposed to score through presented search-algorithms for any given E-GTSP model for any number of group-counts (although only 5 different group-counts are tested over here). The R-Search gives an average error of 8.8%. However, because of being tested upon real-street data, it has brought-forth a better picture. It would be compelling to compare the performance gain of it with other approaches developed by [28] and [6] on common data-set.

Figure 4.7 gives one possible justification for R-Search win over D-Search for higher $|V^s|$ instances. The grey area is the proximity region of a given start-end points

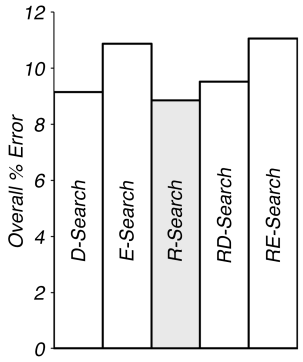


Figure 4.6 : Percentage error coming out of all proposed search algorithms.

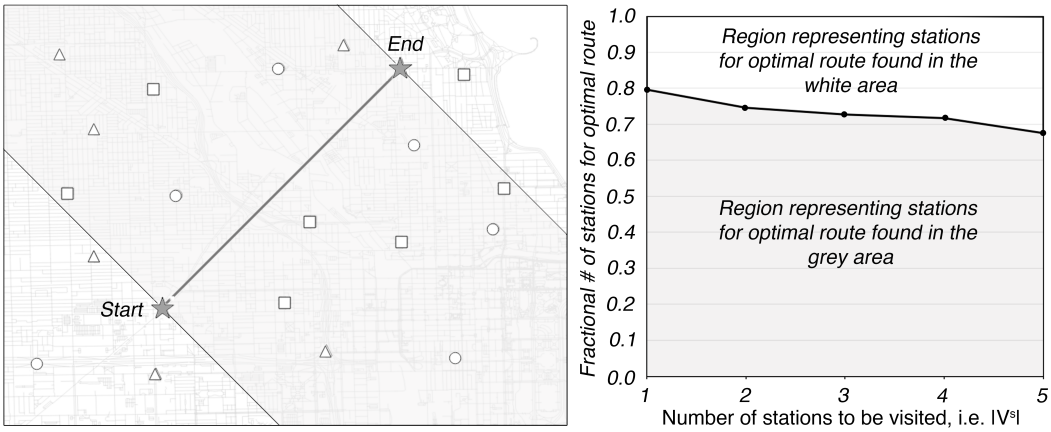


Figure 4.7 : Possible explanation of R-Search's win over D-Search for increased $|V^s|$ value.

pair, while the two white areas are outside this. Algorithmically, D-Search is quite efficient to select distant points, with respect to start-end point, lying close enough to the underlying Dijskstra-route. Since this route terminates at start-end points, points lying behind them (white region, Figure 4.7) unfavorably do not get selected during stage 2 (**decision 1**), refer Figure 4.2. On the other hand, as can be seen in Figure 4.2, R-Search remains unbiased toward points for their region of location (white/grey) during selection procedure. Although this leads to 2^x TSP routes, R-Search is a better bet for white-regioned points. Y-axis in Figure 4.7 represents the average fraction of optimal stations for each $|V^s|$ instance belonging to the white and grey areas, demarcated with an example on the left of the Figure. It should be noted that the amount of optimal stations falling into the grey-area gets lower with increase in complexity ($|V^s|$). This shows that with increase in group-counts more and more optimal stations come out of the white-area, thus, leading R-Search to win over D-Search, as it more efficiently inculcates behind stations too for **decision 2**. This is an interesting observation which shows the consideration of homogeneous spread

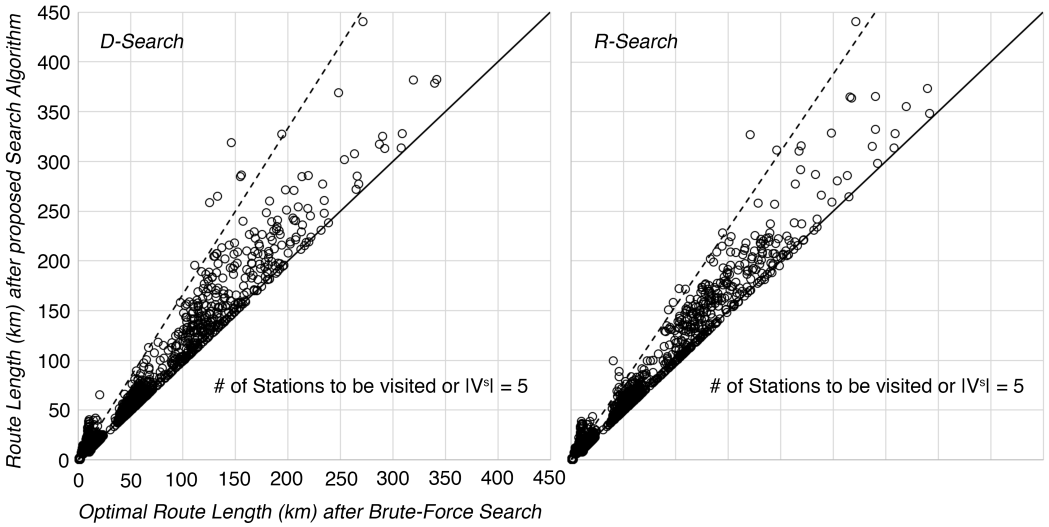


Figure 4.8 : Graph comparing the absolute route lengths coming after *Brute-Force* and *D-Search/R-Search* approaches.

of groups throughout the city-network, and continuous drop of curve value (Figure 4.7right) allows us to generalize the R-Search's performance for even higher $|V^s|$ instances too.

In order to observe the D-Search and R-Search performance for increasing start-end points pair distances, a plot is created between their estimated route length and optimal length (Figure 4.8). The graph represents all 105 routes from each 15 city for 5 group-counts. The key observation here is the fanning-out behavior of data-points with increase in optimal route length. As one increases the best route length, the estimated one by both the algorithms gets farther away from mean-line (solid) (Figure 4.8). Although only $|V^s| = 5$ scenario is presented here, similar spread is observed for other instances too. This behavior was expected as almost all routing algorithms get error-prone with distant start-end point pairs, nevertheless, it is always better to document all possible observations. Another point worth stating here is that in a given city road-network, for mere $|V^s| = 5$ instance, the optimal route length might get of the order of a couple of hundreds of kilometer. This shows the complexity of TSP in general and, therefore, a necessity to find its optimal solution as even half a percent gain may cause a drastic profit in time and money.

Finally, an interesting observation is extracted by plotting a graph between start-end points' displacement and corresponding optimal route length for each instance (Figure 4.9). Instead of plotting individual data-points on the graph authors have demarcated

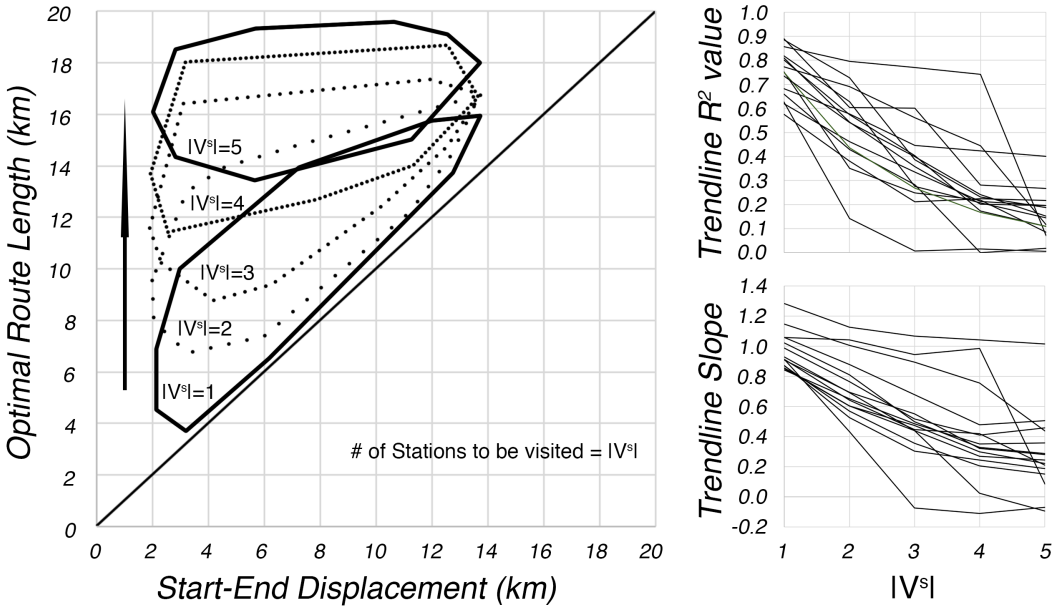


Figure 4.9 : Graph (*left*) between start-end points' displacement and corresponding optimal route length for all 5 group-counts (for Brussels), where only a polygon is plotted to show data-point's spread.

their spread-region with different styled lines, making rugged polygons, for better visualization and explanation. With increase in instance complexity ($|V^s|$) for a given city's E-GTSP model, Brussels (Belgium) in this case, corresponding polygon gets drifted (marked by solid arrow) away from the mean line towards higher y-value. On the right of Figure 4.9, there are two plots showing trendline's slope and R^2 value with respect to $|V^s|$. The trendline here belongs to each data-set which are demarcated by polygons (only one city is there on the left of the Figure). Each curve exemplifies one city model in Figure 4.9. Decreasing trendline's slope value with instance complexity shows the attainment of optimal route lengths saturation. It means that for larger group-counts for a given E-GTSP model the optimal route length becomes more independent to its start-end points euclidean separation. This observation is quite novel and lets us deduce the commentary that once an optimal route length for a given start-end pair for a complex instance is known it could be extrapolated to other pairs too. Like trendline's slope value, its R^2 value has also behaved similarly with decreasing trend. Since with increase in $|V^s|$ the points get more widely spreadout throughout the city, the decreased R^2 value pin-points the complex street-networking of real-world cities. For complex models, therefore, an amalgamation of heuristic approach to improve existing algorithms is advised, which is necessary also because

of the fact that a fresh route estimation every time for each scenario is computationally and realistically not feasible.

4.7 Conclusions and Future Work

The E-GTSP, which is an extension of famous TSP, has proven by plethora of researchers to model more realistically real-life combinatorial optimization problems where the task is to find a close/open *Hamiltonian* cycle visiting exactly one vertex from each group in a given graph. The recommended approach to solve it involves reducing it to corresponding TSP before solving to optimality, which is a dynamic programming approach and backed by many researchers. However, this transformation increases problem's dimension, making it practically unfeasible to solve. Nevertheless, there do exist a range of efficient transformation algorithms but with core mathematical origin. In this study, authors have presented 5 different search algorithms for a given E-GTSP to TSP model conversion that operate spatially, thus, not increasing TSP's vertex or edge count. They have been tested over 15 selected cities classified into road-network frac-Ds with 5 different $|V^s|$ instances each.

It is observed that with increase in instance complexity all algorithms get erroneous, thereby, giving longer routes which is independent of the choice of city street-pattern (Figure 4.4a). For $|V^s| = 1$ instances, D-Search is better among all presented one; however, as one increases the $|V^s|$ value R-Search gets better (Figure 4.4b). Figure 4.5b and Figure 4.6, additionally, refine this observation by showing that R-Search is in fact the best search approach for tested cities, which is a good sample of any city worldwide, by giving least overall route-length value. The win of R-Search in this study could be attributed by its ability to consider stations belonging to regions outside the start-end proximity (Figure 4.7). It has also been observed that with increase in the optimal route length value its corresponding estimated length error from D-Search or R-Search also gets bigger irrespective of the instance complexity (Figure 4.8). One final observation, which proves the convoluted networking of street-networks of world-cities, is done in Figure 4.9, where a deviation of data-points (between start-end displacement and their optimal routes for all instances) away from the mean value line towards a saturation zone is observed. It shows that for higher number of groups, i.e.

group-counts, in a given E-GTSP model, the optimal route length almost gets similar irrespective of the start-end points's euclidean displacement.

This study has brought forth a new search criterion of point/vertex selection from each group for an adequate E-GTSP to TSP transformation, tested on real-street data. This search criterion, i.e. R-Search, considers the radial spread of all points from each group with respect to the start and end point for best possible selection. It has widened up research possibilities in this pursuit; and other interested researchers are encouraged to use tested data provided by [145] as benchmark for subsequent studies. Future research might involve testing R-Search criterion with other algorithms discussed by researchers, like the state-of-the-art Lin-Kernighan-Helsgaun TSP solver. Additional work might involve developing heuristic-R-Search or ANN-R-Search algorithm.

Algorithm 1 ***E-Search*** & ***D-Search*** pseudo-code

Require: Terminal points S and F, $G = (V, E)$ representing urban street-network, & $V^g = \{V_1, V_2, ..., V_x\}$, i.e. points' groups set, where $V_i \subseteq V$ & $V_i \cap V_j = \varnothing \ \forall \ i, j = 1, 2, ..., x$ & $i \neq j$.

1: **procedure** E-GTSP $\Rightarrow$ TSP
2: **if** Algorithm 1 = ***E-Search*** **then** Draw SF **Euclidean** Line
3: **end if**
4: **if** Algorithm 1 = ***D-Search*** **then** Calculate SF **Dijkstra** Route
5: **end if**
6: Initialize an empty container $C1$
7: **for** $V_i \in V^g \ \forall \ i = 1, 2, .., x$ **do**
8: Initialize an empty container $C2$.
9: **for** $v_m \in V_i \ \forall \ m = 1, 2, .., |V_i|$ **do**
10: $C2$ <- $v_m - SF$ shortest Euclidean distance
11: **end for**
12: **return** $C2$
13: $C1$ <- $v = v_m - SF \in C2$, with the smallest Euclidean distance
14: **end for**
15: **return** $C1$
16: **end procedure**
17: **procedure** TSP
18: Calculate the minimum-cost *Hamiltonian* cycle induced by $C1$. *Note*: $C1 = V^s$.
19: **end procedure**

Algorithm 2 ***R-Search*** pseudo-code

Require: Terminal points S and F, $G = (V,E)$ representing urban street-network, & $V^g = \{V_1, V_2, ..., V_x\}$, i.e. points' groups set, where $V_i \subseteq V$ & $V_i \cap V_j = \varnothing \; \forall \; i, j = 1, 2, ..., x$ & $i \neq j$.

1: **procedure** E-GTSP $\Rightarrow$ TSP
2: Initialize an empty container $C1$
3: **for** $V_i \in V^g \;\; \forall \;\; i = 1, 2, .., x$ **do**
4: Initialize an empty container $C2$.
5: **for** $v_m \in V_i \;\; \forall \;\; m = 1, 2, .., |V_i|$ **do**
6: $C2$ <- $v_m - S$ Euclidean distance
7: $C2$ <- $v_m - F$ Euclidean distance
8: **end for**
9: **return** $C2$
10: $C1$ <- $v = v_m - S$ & $v_m - F \in C2$, with the smallest Euclidean distance
11: **end for**
12: **return** $C1$
13: **end procedure**
14: **procedure** TSP
15: Calculate the minimum-cost *Hamiltonian* cycle for all pairs (i.e. 2^x) induced by $C1$.
16: Select the pair with shortest minimum-cost cycle.
17: **end procedure**

Algorithm 3 ***RE-Search*** & ***RD-Search*** pseudo-code

Require: Terminal points S and F, $G = (V, E)$ representing urban street-network, & $V^g = \{V_1, V_2, ..., V_x\}$, i.e. points' groups set, where $V_i \subseteq V$ & $V_i \cap V_j = \varnothing \ \forall \ i, j = 1, 2, ..., x$ & $i \neq j$.

procedure E-GTSP $\Rightarrow$ TSP
    Initialize an empty container $C1$
    for $V_i \in V^g \ \forall \ i = 1, 2, .., x$ **do**
        Initialize an empty container $C2$.
        for $v_m \in V_i \ \forall \ m = 1, 2, .., |V_i|$ **do**
            $C2$ <- $v_m - S$ Euclidean distance
            $C2$ <- $v_m - F$ Euclidean distance
        end for
        return $C2$
        $C1$ <- Two $v = v_m - S$ & $v_m - F \in C2$, with least Euclidean distances
    end for
    return $C1$
    if Algorithm 3 = ***RE-Search*** **then** Run ***E-Search*** (Algorithm 1), treating $C1 \equiv V^g$
    end if
    if Algorithm 3 = ***RD-Search*** **then** Run ***D-Search*** (Algorithm 1), treating $C1 \equiv V^g$
    end if
    Estimate $C1^f$ (Filtered $C1$ after ***E-Search*** or ***D-Search***)
end procedure
procedure TSP
    Calculate the minimum-cost *Hamiltonian* cycle for all pairs (i.e. 2^x) induced by $C1^f$.
    Select the pair with shortest minimum-cost cycle.
end procedure

5. AN ATTEMPT TO REDUCE AN E-GTSP INSTANCE SIZE FOR GLKH SOLUTION

5.1 Abstract

The state-of-the-art approach to solve an E-GTSP instance involves its asymmetric transformation into TSP before using some TSP solver. This approach does not reduce instance size at any stage. For very large instances, this taxes considerable time and space resources. This study presents a custom cost of vertex, termed as *Cost Product*, in order to reduce the dimension of instance before solving it using GLKH solution. The shrinked matrices generated using this cost are compared with original matrices, obtained from GTSPLIB, in terms of cost error, time, and space. GLKH 1.0 is used to solve these two matrices. It is observed that for time and space, shrinked matrices are better than original ones of the order of 2^{nd} degree polynomial. It is also reported that percentage cost error is a function of average number of vertex per cluster and is bounded within certain range for different instances. The Cost Product is observed to be one custom cost that could be systematically used to reduce the size of any E-GTSP instance before solving it using the state-of-the-art solution.

5.2 Introduction

The Travelling Salesman Problem (TSP) is by far one of the well-known and extensively studied combinatorial optimization problem. It asks to find the shortest route (in terms of length, time, or any custom cost) that visits each vertex in a given set exactly once before returning to the starting vertex. Formally, it could be stated as follows: Given a directed/undirected graph $G = (V, E)$ with set of vertices V and set of weighted edges E, find the shortest route between start vertex s and end vertex e (e could be same as s for closed route) that visits each vertex in the set $V' \subseteq V - \{s, e\}$ exactly once. It is equivalent to finding the minimum-cost $Hamiltonian$ cycle in G [46], which is an NP-hard problem. The problem has numerous applications in areas like vehicle routing, communication networking, sequencing and scheduling etc. [67],

and therefore has always remained great source of attraction for various disciplines for decades. [91] has documented a detailed classification of different types of TSP and their possible solutions. In graph theory and also in this study, (route, tour, cycle, path), (node, vertex), and (cost, weightage) are loosely used synonymously.

The Generalized-TSP (GTSP) or Set-TSP or Travelling Politician Problem is an exemplary for selection and sequence related problem. It is an extension of TSP [91] and was introduced by [51]. Here, the set V is further segmented into m number of groups and the problem is to find a minimum-cost route, also known as g-tour, passing through at least one vertex from each group making an open or closed route. For many inherently hierarchical real-world problems, it offers a more precise model than TSP. The GTSP is mathematically defined as follows: Let $G = (V, E)$ be a graph where $V = \{v_1, v_2, ..., v_n\}$ is the set of vertex, $E = \{(v_i, v_j) \mid i \neq j; v_i, v_j \in V\}$ is the edge set, $W = \{w_{ij}\}$ is the non-negative cost defined on E, and V is partitioned into m mutually exclusive and exhaustive groups such that $V^g = \{V_1, V_2, ..., V_m\}$ and $V = V_1 \cup V_2 \cup ... \cup V_m$ with $V_\alpha \cap V_\beta = \varnothing$ for all $\alpha, \beta = 1, 2, ..., m$ and $\alpha \neq \beta$, determine the shortest *Hamiltonian* route passing through each group *at least once*. If E is undirected then direction is irrelevant and $(v_i, v_j) = (v_j, v_i)$. The *at least once* variant of GTSP was independently introduced by [51], [97], [101] and *exactly once* variant (also known as Equality-GTSP or E-GTSP) was introduced by [87] and [62]. If the cost-matrix W is symmetrical, i.e. $w_{ij} = w_{ji}$ for all $i, j = 1, 2, ..., n$ and $i \neq j$, the problem is prefixed with *symmetric*, otherwise *asymmetric*. The GTSP is an NP-hard problem [58] as it reduces down to equivalent TSP (also NP-hard) whenever $|V_\alpha| = 1 \; \forall \; \alpha = 1, 2, ..., m$, i.e. individual groups become singleton. In graph theory and also in this study, (route, tour, cycle, path), (node, vertex), and (cost, weightage) are loosely used synonymously.

It is well known that any E-GTSP instance can be asymmetrically transformed into TSP instance preserving the number of vertices [88]. The transformation allows one to solve the instance using the state-of-the-art LKH asymmetric TSP solver. LKH [151] is a local search heuristic that is based on the variable depth local search of Lin and Kernighan [72]. Figure 5.1a shows the state-of-the-art way to reduce a given E-GTSP instance to TSP before solving it using LKH solution. On the left, the matrix contains the cost of visiting between any two vertices of instance. Although, on one

hand, the state-of-the-art E-GTSP solution (i.e. GLKH [57]) guarantees to give optimal route in given time span [50], the approach does not reduce matrix size at any stage, i.e. at E-GTSP instance or Clustered TSP instance or TSP instance (Figure 5.1a). For instances with large V set and numerous clusters m, it consumes considerable computational space and time (especially when cost-matrices are generated on-the-fly). Figure 5.1b matrix represents a cost-matrix where the cost is unknown beforehand and is a function of time. Examples are cases where E-GTSP instance represents vehicle navigation scenario and the cost is defined as the time taken to navigate between two ground locations. Since time taken is a function of traffic (at least), cost-matrix is generated on-the-fly. One only need to calculate the time-dependent cost for grey cells in Figure 5.1b as only they connect vertices of different clusters. These kind of instances are not only empty but are also big in dimension and therefore cannot be fed into the state-of-the-art E-GTSP solution Figure 5.1a.

In this study, a new cost value is defined for each vertex in a given E-GTSP instance in order to reduce the overall cost-matrix size for fast and low-spaced computation. This new cost, termed as *Cost Product* or CP, is used to systematically reduce cluster size, keeping the probability of finding the best or optimal vertex in each cluster high. It has been argued that CP governs the above defined probability. Results in terms of time, space, and cost error are compared with results drawn from the state-of-the-art GLKH solution [150].

5.3 Cost Product (CP)

Graph is represented in the form of cost-matrix for better visualization. The i^{th} row in this matrix contains the cost of visiting each vertex from i^{th} vertex, Figure 5.1a. By far, there is no definite way to estimate the overall cost of i^{th} vertex in a graph. This overall cost is necessary to compare vertices of cluster to systematically delete those with high cost for fast solution. This cost, termed as *Cost Product* or CP, is defined as follows:

$$\begin{aligned} \mathtt{C}_p(V_{C_m i}) &= \mathtt{C}_a(V_{C_m i}C_1) \times \mathtt{C}_a(V_{C_m i}C_2) \ldots \mathtt{C}_a(V_{C_m i}C_{m-1}) \\ &= \prod_{x=1}^{m-1} \mathtt{C}_a(V_{C_m i}C_x) \end{aligned} \tag{5.1}$$

where, $\complement_p(V_{C_m i})$ i.e. $Cost_{prod.}(V_{C_m i})$ is the cost product of $V_{C_m i}$ vertex, $V_{C_m i}$ is the i^{th} vertex (V_i) of C_m cluster, C_x is the x^{th} cluster, and $\complement_a(V_{C_m i}C_x)$ i.e. $Cost_{avg.}(V_{C_m i}C_x)$ is the average cost between V_i vertex and all vertices of C_x cluster.

$$\begin{aligned}\complement_a(V_{C_m i}C_x) &= \frac{\complement(V_{C_m i}V_{C_x 1}) + \complement(V_{C_m i}V_{C_x 2}) \ldots \complement(V_{C_m i}V_{C_x |C_x|})}{|C_x|} \\ &= \frac{\sum_{y=1}^{|C_x|} \complement(V_{C_m i}V_{C_x y})}{|C_x|}\end{aligned} \tag{5.2}$$

where, $\complement(V_{C_m i}V_{C_x y})$ i.e. $Cost(V_{C_m i}V_{C_x y})$ is the non-negative cost between V_i of C_m cluster and y^{th} vertex of C_x cluster and $|C_x|$ is the dimension of C_x cluster. The definition of CP is derived from the nearest neighbour search concept as has also been done by [164]. It should be noted that one very small $Cost(V_{C_m i}V_{C_x y})$ value reduces the overall $Cost_{prod.}(V_{C_m i})$ value of the vertex, and therefore this vertex should be taken among the set of optimal vertices for E-GTSP solution.

Figure 5.2 is an illustration of the problem. Here each oval represents a cluster as defined in cost-matrix in Figure 5.1a. It has been argued that a low CP value of vertex in a cluster indicates its high possibility of being among the set of vertices for best or optimal route. In other words, when sorted in increasing CP value from left to right in a cluster, vertices on the left are optimal for minimum-cost route than vertices on the right. This filtering criteria on the basis of CP value dramatically reduces the overall cost-matrix size of Figure 5.1a, thus reducing the time and computational space consumption.

It rises to the question that how many vertices with low CP value (from the left of sorted CP list) should be taken from each cluster for shrinked cost-matrix. The shrinked cost-matrix is the matrix with same number of clusters, i.e. m, as in the original one but with less number of vertices. In other words, if for an original E-GTSP instance the total number of clusters are m and total number of vertices are n, then for shrinked E-GTSP instance, after vertex filtering using CP criteria, the total number of clusters would remain m, however the total number of vertices will get less than n. Fixed number of vertices for selection is not used for CP filtering as clusters vary in dimension within an instance and among different instances and rather CP percentage (%) is used. For example, if for a cluster the lowest and highest CP values are 1 and 101, respectively, then 5% will select all vertices with CP less than 6. This X% is a

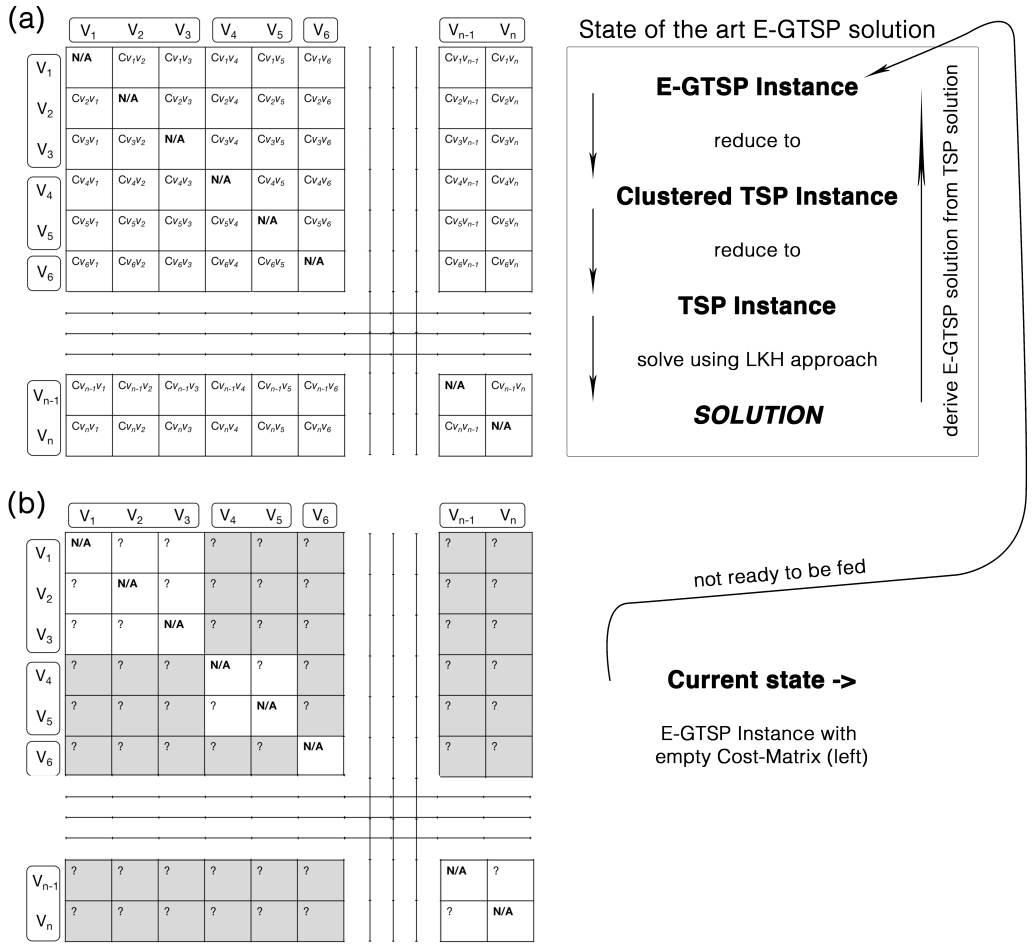


Figure 5.1 : The state-of-the-art GLKH solution for any E-GTSP instance.

matter of choice and in the results and discussion section it is shown how this % value governs the maximum possible error that can be introduced in route cost.

Figure 5.3 helps to visualise the CP % filtering approach. Vertices of each cluster are placed in increasing order of CP from left to right and over a number line for relative scaling. Solid oval represents a vertex for best or optimal route. As can be seen, clusters with high coefficient of variation miss the chances of finding solid oval within X%. The coefficient of variation (CV), also known as relative standard deviation (RSD), is a standardized measure of dispersion of data in relation to the mean of population. It is defined as the ratio of standard deviation and mean. For clusters with highly dispersed CP value, CV is high, otherwise low. Decreasing probability of finding solid oval within X% denotes increasing error.

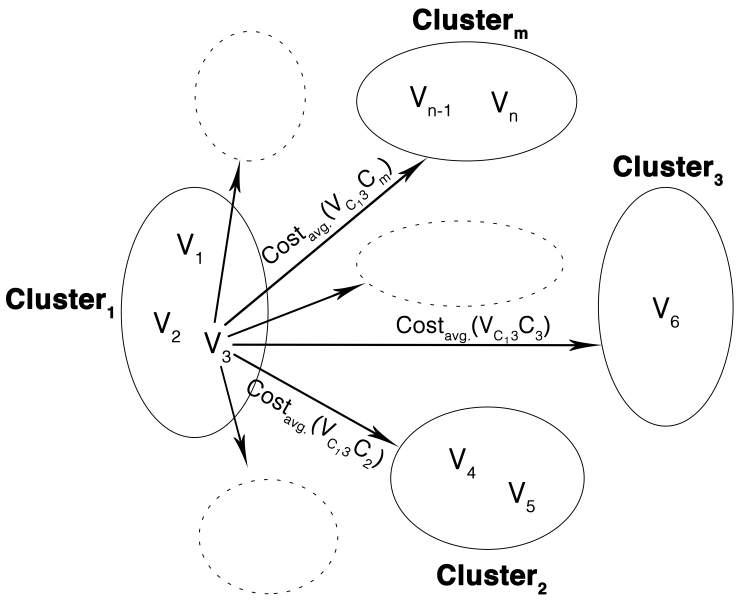


Figure 5.2 : An illustration of E-GTSP instance. In this example ovals representing each cluster do not overlap.

5.3.1 Upper bound of CV

Assume there is a cluster C_m in an E-GTSP instance that has k number of vertices. The non-negative CP of these vertices vary between the range $[0, M]$, where M is some very large value. The mean $\bar{x}$, standard deviation σ_x, and CV of CP for this cluster would be:

$$\bar{x} = \frac{\sum_{i=1}^{k} \complement_p(V_{C_m i})}{k} \tag{5.3}$$

$$\sigma_x = \sqrt{[\frac{\sum_{i=1}^{k} \{\complement_p(V_{C_m i}) - \bar{x}\}^2}{k}]} = \sqrt{[\frac{\sum_{i=1}^{k} \{\complement_p(V_{C_m i})\}^2}{k} - \bar{x}^2]} \tag{5.4}$$

$$CV = \frac{\sigma_x}{\bar{x}} \tag{5.5}$$

The CV would attain maximum value when $k-1$ vertices of cluster have $Cost_{prod.}$ value equal to 0 and remaining one vertex, considered as an outlier, has value M. This gives:

$$\bar{x} = \frac{M}{k}, \frac{\sum_{i=1}^{k} \{\complement_p(V_{C_m i})\}^2}{k} = \frac{M^2}{k}$$

$$\Rightarrow \sigma_x = \sqrt{[\frac{M^2}{k} - \frac{M^2}{k^2}]} = \frac{M}{k}\sqrt{(k-1)}$$

$$CV_{max.} = \frac{\sigma_x}{\bar{x}} = \frac{\frac{M}{k}\sqrt{(k-1)}}{\frac{M}{k}} = \sqrt{(k-1)} \tag{5.6}$$

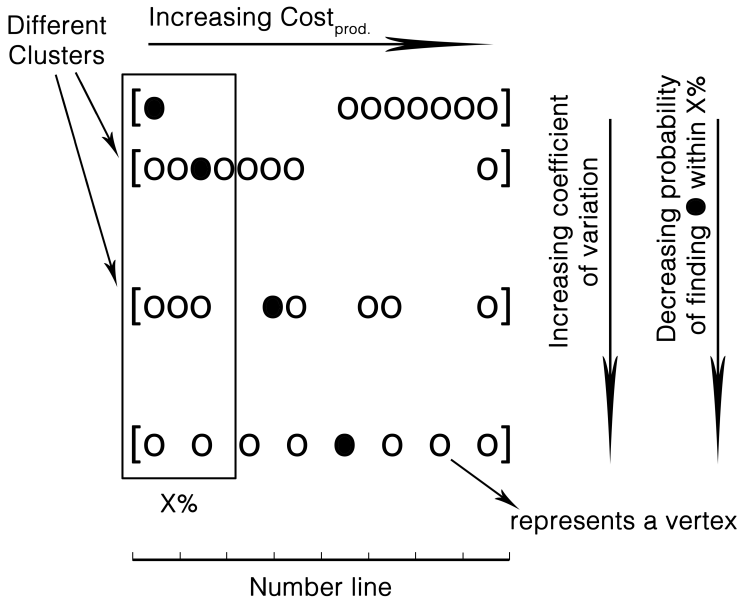


Figure 5.3 : Clusters with sorted vertices in increasing order of CP.

Eq.5.6 shows that the upper limit of CV of a cluster is a function of only its dimension. For an E-GTSP instance with average K number of vertices in cluster, the overall or average CV will not go beyond $\sqrt{(K-1)}$.

5.3.2 Probability as a function of CV and X%

Error induced by CP filtering in a given E-GTSP instance is not only a function of X% but also a function of average CV of all clusters, as shown in Figure 5.3. Mathematically the dependency can be explained as follows:

$$\frac{1}{CostError} \propto Probability(Figure5.3) = e^{(CV_f \times \ln(X_f))} \tag{5.7}$$

where, $f = fraction$. Figure 5.4 is a plot between $Probability$ and CV for different values of X%. The plot justifies eq.5.7 by making probability equal to 1 for X=100% and equal to 0 for X=0%. As for intermediate curves, it shows an asymptotic behavior by tending towards 0 probability for high CV values. It should be noted that for a given X% and average cluster dimension, the probability gets bounded within $[1, e^{constant}]$ (eq.5.6 and eq.5.7), irrespective of the $Cost_{prod.}$ or number of clusters m. For example, for an E-GTSP instance with average cluster dimension of 3601, $CV_{max.}$ becomes $\sqrt{(3601-1)} = 60$ and for a CP % filtering with X=98% the lowest possible probability of finding solid oval (Figure 5.3) becomes $e^{(60 \times \ln(0.98))} = 0.3$ (Figure 5.4).

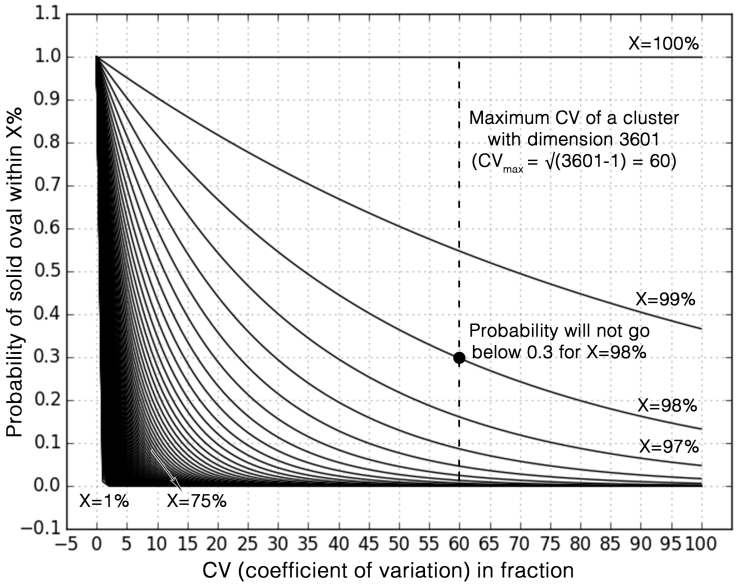


Figure 5.4 : A plot between *Probability* of finding solid oval within X% (Figure 5.3) and CV.

5.4 GTSPLIB sample instance library

Using $Cost_{prod.}$ value, the effort is to reduce the E-GTSP instance size for GLKH solution. It is necessary to compare the results drawn from the GLKH solution for both original and shrinked matrices for different instances. The GTSP instances for comparison are downloaded from [152]. This GTSP instance library was generated from TSPLIB using clustering procedure. TSPLIB offers a collection of sample instances for TSP and other related problems generated from various sources and of various types [153]. TSPLIB ([3] and [76]) and GTSPLIB [99] are primarily used by graph researchers as benchmark instances for development.

Small, large, and very large benchmark instances used in this study are documented in Table 5.1. This classification is based on the # of cluster in instances [50], with $[0,40]$ marked as small, $[40,100]$ as large, and $[>100]$ as very large. *Instance* is the name of the instance as written in the GTSPLIB library and $Avg.\#N/C$ is the average # of node in cluster in sample instances. The $Avg.\#N/C$ is nothing but K of Section 5.3.1. The STOP_AT_OPTIMUM = NO specifies whether the run to find an optimal route should stop when route length becomes equal to Optimum Value (provided in the library) or not (in the documentation of [154]). This setting was necessary as almost in all instances one does not know the Optimum Value beforehand. Also since the original and shrinked matrix represent two considerably different graphs, the YES

value of STOP_AT_OPTIMUM key will bias the results. The other key-values of runGLKH_EXP file of GLKH-1.0 are kept unaltered during the process.

Table 5.1 : Info. of small benchmark instances (STOP_AT_OPTIMUM = NO)

S.No.	Instance	Optimum	#Cluster	#Node	Avg.#Node/Cluster
1	3burma14	1805	3	14	4.7
2	4ulysses16		4	16	4.0
3	5ulysses22	5307	5	22	4.4
4	10att48	5394	10	48	4.8
5	11berlin52	4040	11	52	4.7
6	11eil51	174	11	51	4.6
7	14st70	316	14	70	5.0
8	16eil76	209	16	76	4.7
9	16pr76	64925	16	76	4.7
10	20gr96	29440	20	96	4.8
11	20kroA100	9711	20	100	5.0
12	20kroB100	10328	20	100	5.0
13	20kroC100	9554	20	100	5.0
14	20kroD100	9450	20	100	5.0
15	20kroE100	9523	20	100	5.0
16	20rd100	3650	20	100	5.0
17	21eil101	249	21	101	4.8
18	21lin105	8213	21	105	5.0
19	22pr107	27898	22	107	4.9
20	25pr124	36605	25	124	5.0
21	26bier127	72418	26	127	4.9
22	26ch130	2828	26	130	5.0
23	28gr137	36417	28	137	4.9
24	28pr136	42570	28	136	4.9
25	29pr144	45886	29	144	5.0
26	30ch150	2750	30	150	5.0
27	30kroA150	11018	30	150	5.0
28	30kroB150	12196	30	150	5.0
29	31pr152	51576	31	152	4.9
30	32u159	22664	32	159	5.0
31	39rat195	854	39	195	5.0

Table 5.2 : Info. of large benchmark instances (STOP_AT_OPTIMUM = NO)

S.No.	Instance	Optimum	#Cluster	#Node	Avg.#Node/Cluster
32	40d198	10557	40	198	5.0
33	40kroa200	13406	40	200	5.0
34	40krob200	13111	40	200	5.0
35	41gr202	23301	41	202	4.9
36	45ts225	68340	45	225	5.0
37	45tsp225	1612	45	225	5.0
38	46gr229	71972	46	229	5.0
39	46pr226	64007	46	226	4.9
40	53gil262	1013	53	262	4.9
41	53pr264	29549	53	264	5.0
42	56a280	1079	56	280	5.0
43	60pr299	22615	60	299	5.0
44	64lin318	20765	64	318	5.0
45	80rd400	6361	80	400	5.0
46	84fl417	9651	84	417	5.0
47	87gr431	101946	87	431	5.0
48	88pr439	60099	88	439	5.0
49	89pcb442	21657	89	442	5.0
50	99d493	20023	99	493	5.0

Table 5.3 : Info. of very large benchmark instances (STOP_AT_OPTIMUM = NO)

S.No.	Instance	Optimum	#Cluster	#Node	Avg.#Node/Cluster
51	200C1k.0	6375154	200	1000	5.0
52	200dsj1000	9187884	200	1000	5.0
53	200E1k.0	9662857	200	1000	5.0
54	201pr1002	114311	201	1002	5.0
55	212u1060	106007	212	1060	5.0
56	217vm1084	130704	217	1084	5.0
57	235pcb1173	23399	235	1173	5.0
58	259d1291	28400	259	1291	5.0
59	261rl1304	150468	261	1304	5.0
60	265rl1323	154023	265	1323	5.0
61	276nrw1379	20050	276	1379	5.0
62	280fl1400	15316	280	1400	5.0
63	287u1432	54482	287	1432	5.0
64	464u2319	65758	464	2319	5.0
65	479pr2392	169874	479	2392	5.0

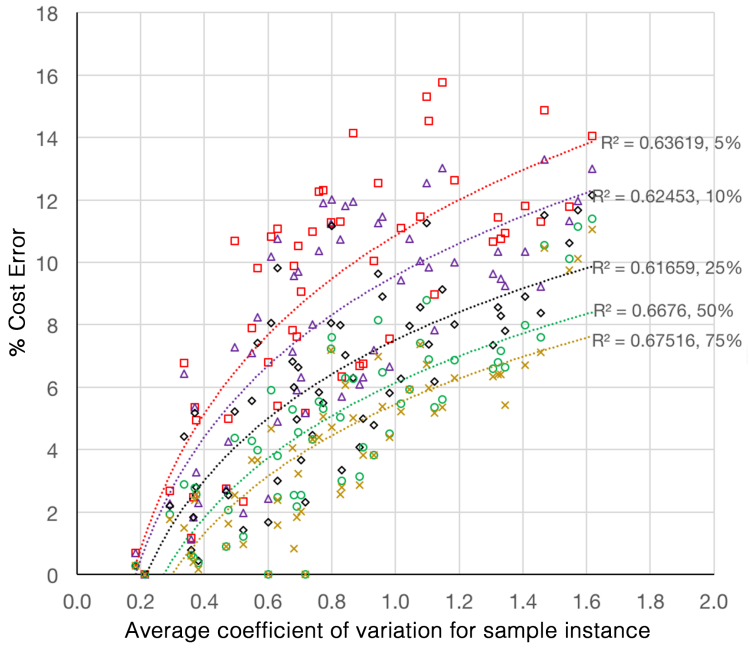


Figure 5.5 : A plot between *Average CV* and *% Cost Error* for different X% values.

5.5 Results and Discussions

Eq.5.7 is a mathematical relation between *Probability*, *CV* and *X%* as observed in this study. *Probability* is inversely proportional to *CostError* as increasing probability of finding solid oval (Figure 5.3) within X% decreases the cost error in optimal route of shrinked matrix. This inverse relationship is visible when compared Figure 5.4 and Figure 5.5, where one appears to be a mirror image of another along horizontal line parallel to x-axis. Figure 5.5 is a plot between average CV of GTSPLIB sample instances and Cost Error induced by applying GLKH solution on shrinked instances for different X%. R^2 value being greater than 0.6 shows good fit for each logarithmic curve. The Cost Error, instead of increasing exponentially, polynomially or linearly, increases logarithmically and appears to be tending towards some saturation value for high CV. This is because for given average cluster dimension and X% the probability of finding solid oval does not go below certain limit and keeps the Cost Error below certain plateau (Section 5.3.2). Author comments that more GTSP sample instances are required to compare *Probability* and *CostError*, as current GTSPLIB only offers 140 instances. However, it is clear that they follow an inverse relationship.

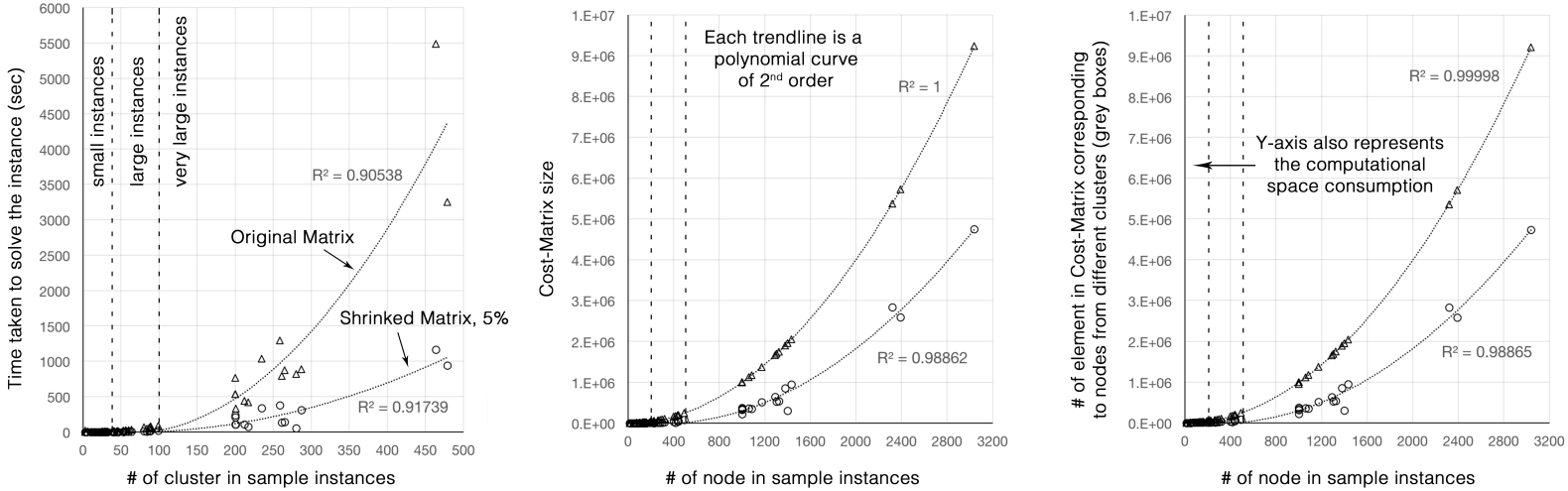


Figure 5.6 : Three plots between *TimeTaken*, *CostMatrixSize* and *InstanceDimension*.

By applying CP filtering on clusters of sample instances one alters the state of the graph and cost-matrix altogether, and therefore cost-errors induced by GLKH solution for original and shrinked matrix cannot be compared directly. Rather these errors should be studied individually with *Optimum* values provided in GLKH-1.0 (Table 5.1). It should be further noted that for instances with same # of node and cluster the CP filtering approach will give different results for different values of CV. However, the result will not get worsen than certain limit as eq.5.7 becomes constant. Studies like [50] only compare errors induced by different algorithms individually for different samples. This is because each instance in TSPLIB and GTSPLIB represents scenario of different source and type and a collective comparison is not possible.

In Section 5.3.1, it is showed that maximum value of CV is independent of $Cost_{prod.}$ and m and is only a function of K (eq.5.6). Since for all GTSPLIB instances the K value, i.e. $Avg.\#Node/Cluster$, is between 4.0 and 5.0 range (Table 5.1), the $CV_{max.}$ value of these instances remains within $[\sqrt{(4-1)}, \sqrt{(5-1)}]$ range, i.e. $[1.7, 2.0]$. This is clear in Figure 5.5 as x-value of all data-points does not go beyond 1.7. To test cases with different K values more sample instances in GTSPLIB with various number of vertices per cluster are required. Current sample collection does not provide this. The over cluttering of curves for X less than 75% in Figure 5.4 shows that probability only varies minutely for them. This behavior in Cost Error is also observed in Figure 5.5 where errors almost get similar in magnitude for 50% and 75%, and only gets double when stepping from 75% to 5%. More E-GTSP instances will help generalize these observations and help users find suitable combinations of X% for specific use-cases.

The time taken and resources consumed by GLKH solution for original and shrinked matrix are important to compare to test the suitability of CP filtering criteria for large and very large E-GTSP instances. Cost-matrix size and # of grey boxes count (Figure 5.1) are used as proxy of RAM consumption during the process since CP filtering is coded in Python in this study and GLKH was coded in C and the two cannot be compared directly. The *Time Taken* is the time taken by GLKH to solve the two matrices. Figure 5.6 is a collection of plots of $TimeTakenVs\#Cluster$, $CostMatrixSizeVs\#Node$, and $GreyBoxesVs\#Node$. These plots represent CP filtering with X equal to 5%. From all three plots it is clear that shrinked matrix takes less time and resource than original matrix, especially for large and very large instances.

The best fit curves are of 2^{nd} degree polynomial order with $R^2 = 1$ and 0.99998 in 2^{nd} and 3^{rd} plot since for them the equation of curve is $CostMatrixSize = n^2$ and $GreyBoxes = n^2 - \sum_{x=1}^{m} |C_x|^2$, respectively.

5.6 Conclusions and Future Work

This study has evaluated the performance of shrinked matrices generated by applying CP filtering criteria on GTSPLIB sample E-GTSP instances on error, time, and space scale. The filtering criteria defined in this study is one possible way to reduce any cost-matrix size for GLKH solution. In terms of time and space requirement, shrinked instances are way too faster and efficient than original instances. As for cost-error the study shows that the result cannot deviate from best solution beyond certain limit, which is a function of average # of vertex per cluster. The argued approach to reduce cost-matrix size is especially suitable for large and very large instances (Figure 5.6) where the gain in performance is of 2^{nd} degree polynomial order. Similar study with large number of varied E-GTSP instances will help to guesstimate beforehand the cost-error possible by CP filtering for new instances.

A possible future path for research would be to improve CP to better govern the probability of finding solid oval within X% (Figure 5.3). More number of GTSP instances would help support the findings of this study. One also need to better understand the relationship between $CostError$ and $Probability$ as defined in eq.5.7 by studying the structure, nature, and origin of used GTSPLIB instances.

6. CONCLUSIONS AND RECOMMENDATIONS

OpenStreetMap (OSM) has been demonstrated to be one valuable source of spatial data because of its big volume heterogeneity in context of many applications. However, concerns still exist regarding its suitability for specific use-cases. The study conducted during the presented doctoral thesis has attempted to understand how fit is it for E-GTSP (consult 4^{th} and 5^{th} chapters for detail) Vehicle Routing Services (VRS). Specialized VRS is getting more and more main stream to target specific users, like delivery van, transportation companies, or even general public with custom queries. Although in this thesis the main interest was over E-GTSP, many other similar queries are also possible. In order to interpret the big picture, the first important task is to see how the data-set has evolved with time, what are the governing factors for this evolution, how complete the data is for VRS and what is the future of it. Once done, it is necessary to see how existing algorithms could be improved for better road-length attribute calculation. Finally, an attempt is necessary to efficiently solve E-GTSP by converting it to TSP.

Although, the main attempt was to provide a general commentary about OSM time-series evolution, Turkish provincial boundary has been studied to act as one indicator for big picture. It has been observed that mapping activities drop down during winter season because of reduced outdoor activity and limited tourism. A spatial biasedness in mapped data is observed within the country, which is noticed to be a consequence of heterogeneous socio-economic factors, with maximum influence by population density and literacy level. Furthermore, a considerable mapping-participation inequality is observed, raising a question about data quality for VRS. Interestingly, it is found that major contributors sometime use other VGI projects as primary data-source for OSM. In a nutshell it could be said that, in spite of being a successful VGI project so far, OSM has a long way to go before it overtakes existing proprietary data-sets like Google Maps for advanced VRS. However, identification of better proxies to predict OSM node density evolution is required as next step. Along

with that, it is believed that a study on the interdependency of different VGI projects can help project developers and curators to better identify the source of the geo-data error.

Once the general attributes of OSM is understood in 2^{nd} chapter, an attempt has been made to improve its derived road length value. It was necessary to be fixed as one existing post-processing gap. Syntactically, it is not possible to estimate OSM's road length during data generation, except after data download which involves euclidean length estimation by existing tools. In spite of being accurate enough for gridded street-networks like Chicago (USA) or places with least round-abouts, this approach becomes erroneous for more curved sections by ignoring road curvature factor altogether. The attempt of 3^{rd} chapter was to derive and propose a piecewise cubic parametric polynomial curve fitting approach to consider road curviness during post-processing for improved road length calculation. The new algorithm is proven to be better for all four tested cities and is one potential development which should be incorporated into existing OSM handling tools. Unfortunately, because of having a varied mapping precisions in the data-set, road sections where the mapper has over-estimated curvature are not suitable for this curve-fitting approach, as it will bring more error than is already there because of euclidean formulation. This is the only known limitation of this methodology to become globally applicable. Future work requires the formulation of statistical analysis and machine learned tools to identify these overshot nodes, along with satellite imagery image analysis. A cross validation with other proprietary and government geo-data set is also required. It has raised a concern about how accurate our derived attributes are, like road length, polygon area, etcetra, and how they could be ameliorated.

In the 4^{th} chapter, an elaborate study has been done to understand E-GTSP problem and possible solution. So far, this problem has been understood as one graph theory problem, and a large number of researchers have attempted to provide a sub-optimal to optimal solution to this. However, because of all solutions being mathematical, a new approach to solve it is discussed in detail that considers the spatial spread of vertices in a graph into account (consult the chapter for detail). Out of five proposed search algorithms to select one vertex from each group in a given E-GTSP model, one turned out to be the best relative approach with route precision of 8.8%. This

approach, termed as radial search criterion, considers the radial separation of vertices with respect to the end destinations for filtering for sub-optimal E-GTSP to TSP transformation. Because of being tested on real street-data, derived from OSM, this observation is much reliable and provides a better picture of usability of different approaches. Although, the selected search criterion is quite precise, especially for low instances with few number of groups, there is room for possible improvisation by adoption of heuristic and neural network concepts. Depending upon the kind of street network the definition of R-Search should be tweaked to inculcate the effect of topology. Another key observation which has been deduced is that for models with large number of groups, the optimal route length almost gets saturated within a range in a given city irrespective of the location of end destinations. This has been used as a proxy to show the real street-network's complexity in an urban city. Presented approach has opened-up a possibility to structure complex routing-engines by thinking spatially, which is an easy to adopt approach in GIS realm.

Finally, the 5^{th} chapter is an extension of previous one where a Cost Product criteria is presented to reduce the matrix size of any E-GSTP instance for possible solution. The testing is done on GTSPLIB sample dataset, that is being used as benchmark for these kind of development. It has been observed that in terms of time and space, the shrinked matrix by cost product is way too faster and efficient than the one provided by GTSPLIB. For cost-error, the results show that there is a bounded bin where all values lie and has been observed mathematically. The presented cost product criteria is quite efficient for large and very large instances, causing a gain of 2^{nd} degree polynomial order. Detailed analysis has been provided in respective chapter and the explanation is beyond the scope in this section. Future suggestion is to improve the CP filtering criteria considering the probability of finding solid oval (consult the chapter). Study with other instances libraries with different average number of nodes per cluster is necessary as the current used library only provides instances with 4 and 5 average number of nodes per cluster. A comparison between Cost Error and Probability would help researchers to improve such kind of transformation, primarily by studying the structure, nature and origin of used instances.

This thesis is expected to be useful for scientific community, developer community, OSM system admins and almost anybody who deals with OSM, VRS, open-source

data and Free and Open Source Software for GeoSpatial (FOSS4G). Future research might include a more exhaustive approach to identify and fix data error in OSM, as it is proven to be a prime controlling factor for usage in geo-services. Researchers may also work on system designs for these kind of projects as existing platforms lack many functionalities and syntactic structuring. One promising work might involve satellite data usage to identify land features to fill data gaps, like erroneous and missing features. Although this current thesis was based on OSM project, there do exist some other less popular similar projects like wikimapia, which might also demand similar study for much wider commentary about VGI in general. Using these studies, a more advanced VRS is possible for specific use-cases. Since all major developed codes used in this thesis are published on-line with MIT License, they will assist future researchers and developers to pursue studies in this domain. For a more comprehensive topic-wise explanation and suggestions for further reading, one should consult each chapter's introduction and reference section.

Geo-data access has always remained a barrier in the past for successful Location-Based Services (LBS). There is a monopoly of few players in this sector, like Environmental Systems Research Institute (ESRI) or Governmental Agencies, because of their ability to obtain or collect these expensive data-sets. However, this trend is now changing with increase in awareness among end-users about open-source VGI projects, like OSM. This trend has caused this recent hype of OSM based services and is expected to get stronger in coming days. The open-source movement has overpassed its abstracted form and is really happening. It is believed to be the future of LBS and GIS.

REFERENCES

[1] **Amirian, P.**, **Basiri, A.**, **Gales, G.**, **Winstanley, A. and McDonald, J.** (2015). *OpenStreetMap in GIScience - The Next Generation of Navigational Services Using OpenStreetMap Data: The Integration of Augmented Reality and Graph Databases*, Springer.

[2] **Anderson, C.** (2008). *The Long Tail: Why the Future of Business is Selling Less of More*, Hyperion.

[3] **Angelelli, E.**, **Gendreau, M.**, **Mansini, R. and Vindigni, M.** (2017). The Traveling Purchaser Problem with time-dependent quantities, *Computers & Operations Research*, *82*, 15–26.

[4] **Azizan, M. H.**, **Lim, C. S.**, **Hatta, W. A. L. W. M. and Gan, L. C.** (2012). Application of OpenStreetMap Data in Ambulance Location Problem, *2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks*, 321–325.

[5] **Barron, C.**, **Neis, P. and Zipf, A.** (2014). A Comprehensive Framework for Intrinsic OpenStreetMap Quality Analysis, *Transactions in GIS*, *18*(6), 877–895.

[6] **Bellman, R.** (1962). Dynamic Programming Treatment of the Travelling Salesman Problem, *Journal of the ACM*, *9*(1), 61–63.

[7] **Ben-Arieh, D.**, **Chang, S.**, **Rys, M. and Zhang, G.** (2004). Geometric Modeling of Highways Using Global Positioning System Data and B-Spline Approximation, *Journal of Transportation Engineering*, *130*(5), 632–636.

[8] **Ben-Arieh, D.**, **Gutin, G.**, **Penn, M.**, **Yeo, A. and Zverovitch, A.** (2010). Process planning for rotational parts using the generalized travelling salesman problem, *International Journal of Production Research*, *41*(11), 2581–2596.

[9] **Ben-Arieh, D.**, **Gutin, G.**, **Penn, M.**, **Yeo, A. and Zverovitch, A.** (2003). Transformations of generalized ATSP into ATSP, *Operations Research Letters*, *31*, 357–365.

[10] **Bontoux, B.**, **Artigues, C. and Feillet, D.** (2010). A Memetic Algorithm with a large neighborhood crossover operator for the Generalized Traveling Salesman Problem, *Computers & Operations Research*, *37*(11), 1844–1852.

[11] **Budhathoki, N. R. and Haythornthwaite, C.** (2012). Motivation for Open Collaboration: Crowd and Community Models and the Case of OpenStreetMap, *American Behavioral Scientist*, 1–28.

[12] **Budhathoki, N. R.**, **Nedovic-Budic, Z. and Bruce, B.** (2010). An interdisciplinary frame for understanding volunteered geographic information, *Geomatica*, *64*(1), 11–26.

[13] **Cardillo, A.**, **Scellato, S.**, **Latora, V. and Porta, S.** (2006). Structural properties of planar graphs of urban street patterns, *Physical Review E*, *73*(6), 93–105.

[14] **Castro, M.**, **Iglesias, L.**, **Rodriguez-Solano, R. and Sanchez, J. A.** (2006). Geometric modelling of highways using global positioning system (GPS) data and spline approximation, *Transportation Research Part C*, *14*(4), 233–243.

[15] **Chen, B.**, **Sun, W. and Vodacek, A.** (2014). Improving Image-Based Characterization of Road Junctions, Widths, and Connectivity by Leveraging OpenStreetMap Vector Map, *IEEE IGRASS 2014*, 4958–4961.

[16] **Clarke, G. and Wright, J. W.** (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points, *Operations Research*, *12*(4), 568–581.

[17] **Coast, S.** (2011). *Web and Wireless Geographical Information Systems - How OpenStreetMap Is Changing the World*, Springer.

[18] **Corcoran, P.**, **Mooney, P. and Bertolotto, M.** (2013). Analysing the growth of OpenStreetMap networks, *Spatial Statistics*, *3*, 21–32.

[19] **Corcoran, P. and Mooney, P.** (2013). Characterising the metric and topological evolution of OpenStreetMap network representations, *The European Physical Journal*, *215*(1), 109–122.

[20] **Croes, G. A.** (1958). A Method for Solving Traveling-Salesman Problems, *Operations Research*, 791-812.

[21] **Curtin, K. M.**, **Voicu, G.**, **Rice, M. T. and Stefanidis, A.** (2014). A Comparative Analysis of Traveling Salesman Solutions from Geographic Information Systems, *Transactions in GIS*, *18*(2), 286–301.

[22] **Dimitrijevic, V. and Saric, Z.** (1997). An efficient transformation of the generalized traveling salesman problem into the traveling salesman problem on digraphs, *Information Sciences*, *102*(1-4), 105–110.

[23] **Dodge, M. and Kitchin, R.** (2013). Crowdsourced cartography: mapping experience and knowledge, *Environment and Planning A*, *45*, 19–36.

[24] **Douglas, D. H. and Peucker, T. K.**(1973). Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature, *Cartographica*, *10*(2), 112–122.

[25] **Drakopoulos, A. and Ornek, E.** (2000). Use of Vehicle-Collected Data to Calculate Existing Roadway Geometry, *Journal of Transportation Engineering*, *126*(2), 154–160.

[26] **Elwood, S.** (2008). Volunteered geographic information: future research directions motivated by critical, participatory, and feminist GIS, *GeoJournal: An International Journal of Geography*, *72*(3), 173–183.

[27] **Estima, J. and Painho, M.** (2013). Exploratory analysis of OpenStreetMap for land use classification, *GEOCROWD*, (39–46).

[28] **Fischetti, M.**, **Gonzalez, J. J. S. and Toth, P.** (1997). A Branch-and-Cut Algorithm for the Symmetric Generalized Traveling Salesman Problem, *Operations Research Letters*, *45*(3), 378–394.

[29] **Fritz, S.**, **McCallum, I.**, **Schill, C.**, **Perger, C.**, **Grillmayer, R.**, **Achard, F.**, **Kraxner, F. and Obersteiner, M.** (2009). Geo-Wiki.Org: The Use of Crowdsourcing to Improve Global Land Cover, *Remote Sensing*, *1*(3), 345–354.

[30] **Gersmehl, P. J.** (1991). *The Language of Maps. Pathway in Geography Series*, ERIC (Institute of Education Sciences).

[31] **Girres, J. F. and Touya, G.** (2010). Quality Assessment of the French OpenStreetMap Dataset, *Transactions in GIS*, *14*(4), 435–459.

[32] **Glover, F.** (1990). Tabu Search: A Tutorial, *Interfaces*, 74-94.

[33] **Goodchild, M. F.** (2007). Citizens as sensors: the world of volunteered geography, *GeoJournal: An International Journal of Geography*, *69*(4), 211–221.

[34] **Goodchild, M. F.** (2009). Geographic information systems and science: today and tomorrow, *Procedia Earth and Planetary Science*, *1*(1), 1037–1043.

[35] **Goodchild, M. F.** (2009). NeoGeography and the nature of geographic expertise, *NeoGeography*, *3*(2), 82–96.

[36] **Goodchild, M. F. and Li, L.** (2012). Assuring the quality of volunteered geographic information, *Spatial Statistics*, *1*, 110–120.

[37] **Goodrich, M. T. and Tamassia, R.** (2015). The Christofides Approximation Algorithm, *Algorithm Design and Applications*, *18*(1), 513–514.

[38] **Gutin, G. and Karapetyan, D.** (2010). A memetic algorithm for the generalized traveling salesman problem, *Natural Computing*, *9*(1), 47–60.

[39] **Gutin. G. and Punnen, A. P.** (2007). The Traveling Salesman Problem and Its Variations, *Combinatorial Optimization*, *12*(1), 830.

[40] **Gutin, G. and Yeo, A.** (2007). The Greedy Algorithm for the Symmetric TSP, *Algorithmic Operations Research*, *2*(1), 33–36.

[41] **Hagenauer, J. and Helbich, M.** (2012). Mining urban land-use patterns from volunteered geographic information by means of genetic algorithms and artificial neural networks, *International Journal of Geographical Information Science*, *26*(6), 963–982.

[42] **Haklay, M.**, **Basiouka, S.**, **Antoniou, V. and Ather, A.** (2010). How Many Volunteers Does it Take to Map an Area Well? The Validity of Linus' Law to Volunteered Geographic Information, *The Cartographic Journal*, *47*(4), 315–322.

[43] **Haklay, M.** (2009). How Good is Volunteered Geographical Information? A Comparative Study of OpenStreetMap and Ordnance Survey Datasets, *Environment and Planning B: Planning and Design*, *37*, 682–703.

[44] **Haklay, M.**, **Singleton, A. and Parker, C.** (2008). Web Mapping 2.0: The Neogeography of the GeoWeb, *Geography Compass*, *2*(6), 2011–2039.

[45] **Haklay, M. and Weber, P.** (2008). OpenStreetMap: User-Generated street Maps, *Pervasive Computing, IEEE*, *7*(4), 12–18.

[46] **Hamilton, W. R.** (1856). Memorandum respecting a new system of roots of unity, *Philosophical Magazine*, *12*, 446.

[47] **Heipke, C.** (2010). Crowdsourcing geospatial data, *ISPRS Journal of Photogrammetry and Remote Sensing*, *65*(6), 550–557.

[48] **Helsgaun, K.** (2000). An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic, *European Journal of Operational Research*, *126*, 106–130.

[49] **Helsgaun, K.** (2009). General k-opt submoves for the Lin-Kernighan TSP heuristic, *Mathematical Programming Computation*, *1*(2), 119–163.

[50] **Helsgaun, K.** (2015). Solving the equality generalized traveling salesman problem using the Lin-Kernighan-Helsgaun Algorithm, *Operations Research Letters*, *7*(3), 269–287.

[51] **Henry-Labor, A. L.** (1969). The record balancing problem: A dynamic programming solution of a genera salesman problem, *RAIRO - Operations Research*, *2*, 43–49.

[52] **Hochmair, H. H.**, **Zielstra, D. and Neis, P.** (2015). Assessing the Completeness of Bicycle Trail and Lane Features in OpenStreetMap for the United States, *Transactions in GIS*, *19*(1), 63–81.

[53] **Hu, B. and Raidl, G. R.** (2008). Effective Neighborhood Structures for the Generalized Traveling Salesman Problem, *Evolutionary Computation in Combinatorial Optimization*, *4972*, 36–47.

[54] **Ibrahim, N.**, **Ujang, U.**, **Desa, G. and Ariffin, A.**(2015). Analysing the Sustainability of Urban Development: A review on the Potential Use of Volunteered Geographic Information, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *2*(2).

[55] **Jia, T. and Jiang, B.** (2010). Measuring Urban Sprawl Based on Massive Street Nodes and the Novel Concept of Natural Cities, *Cornell University Library*.

[56] **Jimenez, F.**, **Aparicio, F. and Estrada, G.** (2009). Measurement uncertainty determination and curve-fitting algorithms for development of accurate digital maps for advanced driver assistance systems, *Transportation Research Part C*, *17*(3), 225–239.

[57] **Karapetyan, D. and Gutin, G.** (2011). Lin-Kernighan heuristic adaptations for the generalized traveling salesman problem, *European Journal of Operational Research*, *208*(3), 221–232.

[58] **Karp, R. M.** (2009). Reducibility among Combinatorial Problems, *50 Years of Integer Programming 1958-2008*, 3–31.

[59] **Kirkpatrick, S.**, **Gelatt, C. D. J. and Vecchi, M. P.** (1983). Optimization by Simulated Annealing, *Science*, *220*(4598), 671–680.

[60] **Kuhn, W.** (2007). Volunteered Geographic Information and GIScience, *NCGIA and Vespucci Workshop on Volunteered Geographic Information; Santa Barbara*, *4*, 86–97.

[61] **Laporte, G.**, **Asef-Vaziri, A. and Sriskandarajah, C.** (1996). Some Applications of the Generalized Travelling Salesman Problem, *The Journal of the Operational Research Society*, *47*(12), 1461–1467.

[62] **Laporte, G.**, **Mercure, H. and Nobert, Y.** (1983). Generalized travelling salesman problem through n sets of nodes: an integer programming approach, *INFOR: Information Systems and Operational Research*, *21*(1), 61–75.

[63] **Laporte, G.**, **Mercure, H. and Nobert, Y.** (1987). Generalized travelling salesman problem through n sets of nodes: the asymmetrical case, *Discrete Applied Mathematics*, *18*(2), 185–197.

[64] **Laporte, G. and Semet, F.** (2016). Computational Evaluation Of A Transformation Procedure For The Symmetric Generalized Traveling Salesman Problem, *INFOR: Information Systems and Operational Research*, *37*(2), 114–120.

[65] **Lawler, E. L.**, **Lenstra, J. K.**, **Kan, A. H. G. R. and Shmoys, D. B.** (1985). The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization, *Wiley*, 476.

[66] **Leeuw, J. D.**, **Said, M.**, **Ortegah, L.**, **Nagda, S.**, **Georgiadou, Y. and DeBlois, M.** (2011). An Assessment of the Accuracy of Volunteered Road Map Production in Western Kenya, *Remote Sensing*, *3*(2), 247–256.

[67] **Lenstra, J. K. and Kan, A. H. G. R.** (1975). Some Simple Applications of the Travelling Salesman Problem, *Operational Research Quarterly*, *26*(4), 717–733.

[68] **Lequiller, F. and Blades, D.** (2014). *Understanding National Accounts - Second Edition*, OECD Publishing.

[69] **Li, D. and Qian, X.** (2010). A brief introduction of data management for volunteered geographic information, *Wuhan Daxue Xuebao (Xinxi Kexue Ban) Geomatics and Information Science of Wuhan University*, *35*(4), 379–383.

[70] **Li, Q.**, **Fan, H.**, **Luan, X.**, **Yang, B. and Liu, L.** (2014). Polygon-based approach for extracting multilane roads from OpenStreetMap urban road networks, *International Journal of Geographical Information Science*, *28*(11), 2200–2219.

[71] **Lien, Y. N.**, **Ma, E. and Wah, B. W. S.** (1993). Transformation of the generalized traveling-salesman problem into the standard traveling-salesman problem, *Information Sciences*, *74*(1-2), 177–189.

[72] **Lin, S. and Kernighan, B. W.** (1973). An Effective Heuristic Algorithm for the Traveling-Salesman Problem, *Operations Research*, *21*(2), 498–516.

[73] **Lyons, G.** (2016). Getting smart about urban mobility - Aligning the paradigms of smart and sustainable, *Transportation Research Part A*.

[74] **Ma, D.**, **Sandberg, M. and Jiang, B.** (2015). Characterizing the Heterogeneity of the OpenStreetMap Data and Community, *ISPRS International Journal of Geo-Information*, *4*(2), 535–550.

[75] **Mackaness, W. A. and Ruas, A.** (2007). Chapter 5 - Evaluation in the Map Generalisation Process, *Generalisation of Geographic Information, Cartographic Modelling and Applications*, 89–111.

[76] **Manerba, D.**, **Mansini, R. and Riera-Ledesma, J.** (2017). The Traveling Purchaser Problem and its variants, *European Journal of Operational Research*, *259*(1), 1–18.

[77] **Marshall, W. E. and Garrick, N. W.**(2010). Street network types and road safety: A study of 24 California cities, *URBAN DESIGN International*, *15*(3), 133–147.

[78] **Mooney, P. and Corcoran, P.** (2014). Analysis of Interaction and Co-editing Patterns amongst OpenStreetMap Contributors, *Transactions in GIS*, *18*(5), 633–659.

[79] **Mooney, P. and Corcoran, P.** (2012). Characteristics of Heavily Edited Objects in OpenStreetMap, *Future Internet*, *4*, 285–305.

[80] **Mooney, P.**, **and Corcoran, P.** (2011). *Using OSM for LBS – An Analysis of Changes to Attributes of Spatial Objects : Advances in Location-Based Services*, Springer.

[81] **Neis, P.** (2015). Measuring the Reliability of Wheelchair User Route Planning based on Volunteered Geographic Information, *Transactions in GIS*, *19*(2), 188–201.

[82] **Neis, P.**, **Goetz, M. and Zipf, A.** (2012). Towards Automatic Vandalism Detection in OpenStreetMap, *ISPRS International Journal of Geo-Information*, *1*(3), 315–332.

[83] **Neis, P. and Zielstra, D.** (2014). Recent Developments and Future Trends in Volunteered Geographic Information Research: The Case of OpenStreetMap, *Future Internet*, *6*(1), 76–106.

[84] **Neis, P.**, **Zielstra, D. and Zipf, A.** (2013). Comparison of Volunteered Geographic Information Data Contributions and Community Development for Selected World Regions, *Future Internet*, *5*(2), 282–300.

[85] **Neis, P.**, **Zielstra, D. and Zipf, A.** (2011). The Street Network Evolution of Crowdsourced Maps: OpenStreetMap in Germany 2007-2011, *Future Internet*, *4*(1), 1–21.

[86] **Neis, P. and Zipf, A.** (2012). Analyzing the Contributor Activity of a Volunteered Geographic Information Project - The Case of OpenStreetMap, *ISPRS International Journal of Geo-Information*, *1*(2), 146–165.

[87] **Noon, C. E. and Bean, J. C.** (1991). A Lagrangian Based Approach for the Asymmetric Generalized Traveling Salesman Problem, *International Journal of Production Research*, *39*(4), 623–632.

[88] **Noon, C. E. and Bean, J. C.** (1993). An Efficient Transformation Of The Generalized Traveling Salesman Problem, *INFOR: Information Systems and Operational Research*, *31*(1), 39–44.

[89] **Over, M.**, **Schilling, A.**, **Neubauer, S. and Zipf, A.** (2010). Generating web-based 3D City Models from OpenStreetMap: The current situation in Germany, *Computers, Environment and Urban Systems*, *34*(6), 496–507.

[90] **Park, W. and Yu, K.** (2011). Hybrid line simplification for cartographic generalization, *Pattern Recognition Letters*, *32*(9), 1267–1273.

[91] **Psaraftis, H. N.**, **Wen, M. and Kontovas, C. A.** (2016). Dynamic vehicle routing problems: Three decades and counting, *Networks*, *67*(1), 3–31.

[92] **Raymond, E. S.** (2001). *The Cathedral and the Bazaar*, O'Reilly Media.

[93] **Renaud, J. and Boctor, F. F.** (1998). An efficient composite heuristic for the symmetric generalized traveling salesman problem, *European Journal of Operational Research*, *108*(3), 571–584.

[94] **Riera-Ledesma, J. and Salazar-Gonzaler, J. J.** (2005). A heuristic approach for the Travelling Purchaser Problem, *European Journal of Operational Research*, *162*(1), 142-152.

[95] **Rosenkrantz, D. J.**, **Stearns, R. E. and Lewis II, P. M.** (2009). An analysis of several heuristics for the traveling salesman problem, *Fundamental Problems in Computing*, *1*, 45–69.

[96] **Ruas, A.** (2008). Map Generalization, *Encyclopedia of GIS*, 631–632.

[97] **Saksena, J. P.** (1970). Mathematical model for scheduling clients through welfare agencies, *Canadian Operational Research Society*, *8*(3), 185.

[98] **Senaratne, H.**, **Mobasheri, A.**, **Ali, A. L.**, **Capineri, C. and Haklay, M.** (2017). A review of volunteered geographic information quality assessment methods, *International Journal of Geographical Information Science*, *31*(1), 139–167.

[99] **Smith, S. L. and Imenson, F.** (2017). GLNS: An effective large neighborhood search heuristic for the Generalized Traveling Salesman Problem, *Computers & Operations Research*, *87*, 1–19.

[100] **Snyder, L. V. and Daskin, M. S.** (2006). A random-key genetic algorithm for the generalized traveling salesman problem, *European Journal of Operational Research*, *171*(1), 38–53.

[101] **Srivastava, S. S.**, **Kumar, S.**, **Garg, R. C. and Sen, P.** (1969). Generalized travelling salesman problem through n sets of nodes, *Canadian Operational Research Society*, *7*(2), 97.

[102] **Stoter, J.**, **Burghardt, D.**, **Duchene, C.**, **Baella, B.**, **Bakker, N.**, **Blok, C.**, **Pla, M.**, **Regnauld, N.**, **Touya, G. and Schmid, S.** (2009). Methodology for evaluating automated map generalization in commercial software, *Computers, Environment and Urban Systems*, *35*(5), 311–324.

[103] **Strano, E.**, **Nicosia, V.**, **Latora, V.**, **Porta, S. and Barthelemy, M.** (2015). Elementary processes governing the evolution of road networks, *Scientic Reports*, *2*.

[104] **Url-1** <http://wiki.openstreetmap.org/wiki/Any_tags_you_like>, date retrieved: 09.02.2016.

[105] **Url-2** <https://github.com/Zia-/Turkey_OSM_Statistical_Analysis_Python-Scripts.git>, date retrieved: 03.02.2016.

[106] **Url-3** <http://wiki.openstreetmap.org/wiki/Import/Catalogue>, date retrieved: 27.01.2016.

[107] **Url-4** <http://www.kgm.gov.tr/Sayfalar/KGM/SiteTr/Istatistikler/DevletveIlYolEnvanteri.aspx>, date retrieved: 27.01.2016.

[108] **Url-5** <https://biruni.tuik.gov.tr/bolgeselistatistik/sorguSayfa.do?target=degisken>, date retrieved: 27.01.2016.

[109] **Url-6** <http://wiki.openstreetmap.org/wiki/Map_Features>, date retrieved: 26.01.2016.

[110] **Url-7** <https://josm.openstreetmap.de>, date retrieved: 26.01.2016.

[111] **Url-8** <http://www.openstreetmap.org>, date retrieved: 26.01.2016.

[112] **Url-9** <https://github.com/MaZderMind/osm-history-splitter>, date retrieved: 26.01.2016.

[113] **Url-10** <http://pgrouting.org/docs/tools/osm2pgrouting.html>, date retrieved: 26.01.2016.

[114] **Url-11** <http://wiki.openstreetmap.org/wiki/Osm2postgresql>, date retrieved: 26.01.2016.

[115] **Url-12** <http://wiki.openstreetmap.org/wiki/Osm2pgsql>, date retrieved: 26.01.2016.

[116] **Url-13** <http://osmcode.org/osmium>, date retrieved: 26.01.2016.

[117] **Url-14** <http://wiki.openstreetmap.org/wiki/Osmosis>, date retrieved: 26.01.2016.

[118] **Url-15** <http://wiki.openstreetmap.org/wiki/Open_Database_License>, date retrieved: 25.01.2016.

[119] **Url-16** <https://wiki.osmfoundation.org/wiki/License/We_Are_Changing_The_License>, date retrieved: 25.01.2016.

[120] **Url-17** <http://wiki.openstreetmap.org/wiki/API>, date retrieved: 25.01.2016.

[121] **Url-18** <http://download.geofabrik.de>, date retrieved: 25.01.2016.

[122] **Url-19** <http://overpass-api.de>, date retrieved: 25.01.2016.

[123] **Url-20** <http://wiki.openstreetmap.org/wiki/Elements>, date retrieved: 25.01.2016.

[124] **Url-21** <http://planet.openstreetmap.org/planet/full-history>, date retrieved: 25.01.2016.

[125] **Url-22** <http://archive09.linux.com/feature/125344>, date retrieved: 20.01.2016.

[126] **Url-23** <http://googlegeodevelopers.blogspot.com.tr/2011/10/introduction-of-usage-limits-to-maps.html>, date retrieved: 15.01.2016.

[127] **Url-24** <http://www.openstreetmap.org/stats/data_stats.html>, date retrieved: 15.01.2016.

[128] **Url-25** <http://www.longtail.com/about.html>, date retrieved: 14.01.2016.

[129] **Url-26** <http://hdr.undp.org/sites/default/files/hdr_2013_en_technotes.pdf>, date retrieved: 13.01.2016.

[130] **Url-27** <http://www.oreilly.com/pub/a//web2/archive/what-is-web-20.html>, date retrieved: 20.10.2016.

[131] **Url-28** <http://wiki.openstreetmap.org/wiki/List_of_OSM-based_services>, date retrieved: 20.10.2016.

[132] **Url-29** <http://wiki.openstreetmap.org/wiki/OSM_XML>, date retrieved: 20.10.2016.

[133] **Url-30** <http://tutorial.math.lamar.edu/Classes/CalcII/ArcLength.aspx>, date retrieved: 20.10.2016.

[134] **Url-31** <http://www.movable-type.co.uk/scripts/latlong.html>, date retrieved: 20.10.2016.

[135] **Url-32** <https://en.wikipedia.org/wiki/Curve_fitting>, date retrieved: 20.10.2016.

[136] **Url-33** <https://www.mathworks.com/help/matlab/math/polynomial-curve-fitting.html>, date retrieved: 20.10.2016.

[137] **Url-34** <http://mathworld.wolfram.com/SimpsonsRule.html>, date retrieved: 20.10.2016.

[138] **Url-35** <http://leafletjs.com>, date retrieved: 20.10.2016.

[139] **Url-36** <https://omerozyildirim.github.io/osmroadvis1>, date retrieved: 20.10.2016.

[140] **Url-37** <http://www.cs.rhul.ac.uk/home/zvero/GTSPLIB>, date retrieved: 15.11.2016.

[141] **Url-38** <https://en.wikipedia.org/wiki/Travelling_salesman_problem>, date retrieved: 15.11.2016.

[142] **Url-39** <https://munsonscity.com/2013/10/09/which-street-pattern-represents-your-continent>, date retrieved: 15.11.2016.

[143] **Url-40** <http://paulbourke.net/fractals/fracdim>, date retrieved: 15.11.2016.

[144] **Url-41** <http://www.gadm.org>, date retrieved: 15.11.2016.

[145] **Url-42** <https://github.com/Zia-/E_GTSP_to_TSP_tested_instances>, date retrieved: 15.11.2016.

[146] **Url-43** <http://munsonscity.com/2013/10/09/which-street-pattern-represents-your-continent>, date retrieved: 02.02.2016.

[147] **Url-44** <http://www.cut-the-knot.org/pythagoras/DistanceFormula.shtml>, date retrieved: 20.10.2016.

[148] **Url-45** <https://www.seas.gwu.edu/~simhaweb/champalg/tsp/tsp.html>, date retrieved: 15.11.2016.

[149] **Url-46** <https://agile-online.org/Conference_Paper/CDs/agile_2010/ShortPapers_PDF/142_DOC.pdf>, date retrieved: 20.10.2016.

[150] **Url-47** <http://www.akira.ruc.dk/~keld/research/GLKH/>, date retrieved: 20.10.2016.

[151] **Url-48** <http://webhotel4.ruc.dk/~keld/research/LKH/>, date retrieved: 20.10.2016.

[152] **Url-49** <http://www.cs.rhul.ac.uk/home/zvero/GTSPLIB/>, date retrieved: 20.10.2016.

[153] **Url-50** <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>, date retrieved: 20.10.2016.

[154] **Url-51** <http://www.akira.ruc.dk/~keld/research/GLKH/GLKH-1.0.tgz>, date retrieved: 20.10.2016.

[155] **Url-52** <https://esa.un.org/unpd/wup>, date retrieved: 05.01.2018.

[156] **Url-53** <https://www.weforum.org/agenda/2016/04/>, date retrieved: 05.01.2018.

[157] **Valenzuela, C. L. and Jones, A. J.** (1997). Estimating the Held-Karp lower bound for the geometric TSP, *European Journal of Operational Research*, *102*(1), 157–175.

[158] **Wang, Z. X. and Ouyang, J. H.**(2013). Curve length estimation based on cubic spline interpolation in gray-scale images, *Journal of Zhejiang University SCIENCE C*, *14*(10), 777–784.

[159] **Wilson, I. D.**, **Ware, J. M. and Ware, J. A.** (2003). A Genetic Algorithm approach to cartographic map generalisation, *Computers in Industry*, *52*(3), 291–304.

[160] **Wu, C.**, **Liang, Y.**, **Lee, H. P. and Lu, C.** (2004). Generalized chromosome genetic algorithm for generalized traveling salesman problems and its applications for machining, *PHYSICAL REVIEW E*, *70*(1).

[161] **Yanga, J.**, **Shia, X.**, **Marcheseb, M. and Lianga, Y.** (2008). An ant colony optimization method for generalized TSP problem, *Progress in Natural Science*, *8*(11), 1417–1422.

[162] **Zhang, Y.**, **Li, X.**, **Wang, A.**, **Bao, T. and Tian, S.** (2015). Density and diversity of OpenStreetMap road networks in China, *Journal of Urban Management*, *4*(2), 135–146.

[163] **Zhao, P.**, **Jia, T.**, **Qin, K.**, **Shan, J. and Jiao, C.** (2015). Statistical analysis on the evolution of OpenStreetMap road networks in Beijing, *Physica A: Statistical Mechanics and its Applications*, *420*, 59–72.

[164] **Zia, M.**, **Cakir, Z. and Seker, D. Z.** (2017). A New Spatial Approach for Efficient Transformation of Equality - Generalized TSP to TSP, *Free and Open Source Software for Geospatial (FOSS4G) Conference Proceedings*, *17*(5).

[165] **Zielstra, D. and Hochmair, H. H.** (2012). Using Free and Proprietary Data to Compare Shortest-Path Lengths for Effective Pedestrian Routing in Street Networks, *Transportation Research Record: Journal of the Transportation Research Board*, *2299*, 41–47.

[166] **Zielstra, D.**, **Hochmair, H. H. and Neis, P.** (2013). Assessing the Effect of Data Imports on the Completeness of OpenStreetMap - A United States Case Study, *Transactions in GIS*, *17*(3), 315–334.

[167] **Zielstra, D. and Zipf, A.** (2010). A Comparative Study of Proprietary Geodata and Volunteered Geographic Information for Germany, *13th AGILE International Conference on Geographic Information Science 2010, Guimarães, Portugal*.

CURRICULUM VITAE

Name Surname: Mohammed Zia

Place and Date of Birth: Visakhapatnam, A.P., India; 24 October 1989

E-Mail: mohammed.zia33@gmail.com

EDUCATION:

- **Integrated M. Tech. in Geological Technology:** 2008-2013; Indian Institute of Technology Roorkee, India; Department of Earth Sciences

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Zia, M.**, Cakir, Z., Seker, D.Z., 2017. An attempt to reduce an E-GTSP instance size for GLKH solution. *Operations Research Letters, Elsevier*, Under-Review. (5^{th} Chapter)
- **Zia, M.**, Ozyildirim, O., Seker, D.Z., Cakir, Z., 2017. Improving OpenStreetMap derived road length on a global scale using curve fitting approach. *Cartographica, University of Toronto Press*, In-Press. (3^{rd} Chapter)
- **Zia, M.**, Cakir, Z., Seker, D.Z., 2017. A new spatial approach for efficient transformation of Equality - Generalized TSP to TSP. *FOSS4G Conference Academic Proceedings*, 17(5). (4^{th} Chapter)
- **Zia, M.**, Cakir, Z., Seker, D.Z., 2017. A new spatial approach for efficient transformation of Equality - Generalized TSP to TSP. *International Conference for Free & Open Source Software for Geospatial, Boston, USA*, GeoForAll Award for 2nd Best Student Paper. (4^{th} Chapter, Travel Grant Winner)
- **Zia, M.**, Seker, D.Z., Cakir, Z., 2016. Spatial evolution of OpenStreetMap dataset in Turkey. *Geoadvances - ISPRS Workshop on Multi-Dim. & -Scale Spatial Data Modeling, Istanbul, Turkey*. (2^{nd} Chapter)
- **Zia, M.**, Ozyildirim, O., 2016. 3rd degree polynomial curve fitting approach to improve OSM derived curved road lengths. *FOSS4G (Free & Open Source Software for Geospatial) Belgium, Brussels, Belgium*. (3^{rd} Chapter)

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2013 - 2215 Graduate Scholarship Award by TUBITAK (The Scientific and Technological Research Council of Turkey) for PhD Thesis.

- 2012 - MHRD (Ministry of Human Resource Development, India) Scholarship Award for Master's Thesis.
- 2011 - Remote Sensing Researcher at The University of Western Ontario, Canada.
- 2008-2013 - Numerous other national level awards.

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- **Zia, M.**, 2017. An attempt to identify and visualise the ?Most-Used-Road-Segment? in a City. *International Conference for Free & Open Source Software for Geospatial, Boston, USA*. Travel Grant Winner.
- **Zia, M.**, Sharma, K., Saraf, A.K., Das, J.D., Baral, S.S., Das, M., 2013. Ground deformational studies using ALOS-PALSAR data between 2007 and 2010 of central Kutch area, Gujarat, India. *Natural Hazards, Springer*, 71(3), 1379-1388.
- **Zia, M.**, 2013. High altitude stationary clouds detection algorithm as an earthquake precursor using MODIS Terra/Aqua satellite data. *75th European Association of Geoscientists & Engineers Conference & Exhibition, London, UK*, Travel Grant Winner.
- Saraf, A.K., **Zia, M.**, Das, J.D., 2012. Satellite detection of thermal precursors of Yamnotri, Ravar, and Dalbandin earthquake. *Natural Hazards, Springer*, 61(2), 861-872.
- Saraf, A.K., **Zia, M.**, Das, J.D., Sharma, K., Rawat, V., 2011. False Topographic Perception Phenomena observed with the satellite images of Moon's surface. *International Journal of Remote Sensing, T&F*, 32(24), 9869-9877.
- **Zia, M.**, Irena Creed, 2011. Satellite Detection of Thermal Precursors of Yamnotri (India), Ravar (Iran) and Dalbandin (Pakistan) Earthquakes. *Gordon Resesarch Conference & Seminar on Catchment Sciences, Bates College, Maine, USA*
- **Zia, M.**, 2011. Qualitative assessment of Linear and Quadratic transformations in georeferencing of satellite imagery. *73rd European Association of Geoscientists & Engineers Conference & Exhibition, Vienna, Austria*, Travel Grant Winner.
- **Zia, M.**, 2010. Cloud cover pattern over Tehri dam (Uttarakhand, India) using MODIS-721-RGB image. *Cognizance (Technical Festival), Department of Earth Sciences, Indian Institute of Technology Roorkee, India*
- **Zia, M.**, 2009. False Topographic Perception Phenomena and its presence on Moon?s surface. *71st European Association of Geoscientists & Engineers Conference & Exhibition, Amsterdam, The Netherlands*, Travel Grant Winner.