USING HEURISTIC METHODS AND MACHINE LEARNING TO IMPROVE THE POSITIONAL ACCURACY OF HISTORICAL GEOSPATIAL DATASETS

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Pankaj M. Dahal

May 2022

DEDICATION

This dissertation is dedicated to my parents, who instilled in me the value of education and perseverance.

ACKNOWLEDGMENTS

There are many people I must thank for making the journey to graduate school so fruitful and memorable.

First of all, I want to express deep gratitude to my co-advisors, Dr. Lee D. Han and Dr. Hyun Kim. Dr. Han did not hesitate to accept me as his student despite my situation. I will forever be indebted to him - for his insights, guidance, and insistence on thinking like a researcher. Furthermore, his warmth, straightforwardness, and patience have played a decisive role in my tumultuous journey to graduation.

Dr. Kim mentored me and played a crucial role in my writing. He has been very flexible with his insights and generous with his time to help me. I am incredibly grateful to my other committee members: Dr. Russell Zaretzki and Dr. David Clarke, for accepting my request and improving the overall content of my dissertation.

I would also like to thank all of my fellow lab mates in JDT Room 311. I have truly enjoyed every moment we spent together, participated in different activities, attended conferences, and even been together through some difficult times. I have many memories with Hyeonsup Lim, Bumjoon Bae, Xiaobing Li, Yuandong Liu, Ali Boggs, Kwaku Boakye, Ranjit Khatri, Nirbesh Dhakal, Mohsen Kamrani, Behram Wali, and Mojdeh Azad. Also, thank you to the Department of Civil and Environmental Engineering faculty and staff for their assistance. Thank you to the Center of Transportation Research (CTR) for funding me for most of my Graduate School.

Also, I express my thanks to all of my friends in Knoxville who made my stay here memorable and have indirectly contributed to the dissertation.

ABSTRACT

The recent availability of multiple geospatial datasets can be attributed to advancements in location-based technologies. The merging of various datasets, commonly known as conflation, is essential to strengthening the content of existing datasets by integrating information from various sources. However, complexities arise when these merged datasets contain distinct representations of the same real-world entities with differing accuracy, projections, data structure, and consideration of details. Although conflation has been an interest to researchers for a long time, the existing methods do not cater to the specific needs of some datasets. For example, historical datasets have lower resolution with richer attribute information, while newer datasets are more positionally accurate but lack detailed or relevant attribute information. In addition, contemporary conflation algorithms usually demand manual review and verification for these datasets as the conflation process still lacks an efficient automated technique.

Through the work in this dissertation, both historical and modern datasets are investigated and explored using optimization and learning-based tools to aid in identifying corresponding and unifying elements in these datasets. The elements identified through these algorithms are validated against verified spatial information known as the ground truth dataset. Research results can inform practitioners with various tool options and notably reduce the time required to merge datasets compared to manual methods.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

VGI	Volunteer Geographic Information
OSM	OpenStreetMap
FRA	Federal Railroad Administration
NHPN	National Highway Planning Network
GIS	Geographic Information Systems
ESRI	Environmental Systems Research Institute

INTRODUCTION

Over the past 60 years, GIS has been an essential tool for capturing, storing, manipulating, and analyzing spatial data. With time, geographic data collected using different technologies for specific purposes have resulted in a plethora of geospatial information. However, integrating and aggregating these multiple datasets into one repository has always been challenging because of each dataset's independent creation and lack of a unifying primary key to identify corresponding elements. For example, historical datasets are less spatially accurate but have GIS attributes that have been tested and corrected through time. On the other hand, the rise of open-source datasets through VGI, which is geospatial content generated by non-professionals, such as OSM, is more detailed and positionally accurate than historical datasets. In order to perform comprehensive GIS analysis, both historical and modern datasets need to be integrated to obtain a new dataset with the best features from both datasets.

One issue with integration is that some historical datasets have proprietary attribute information rarely available in the more spatially accurate crowdsourced datasets. As a result, there is limited commonality available in both datasets to facilitate merging distinct datasets in such instances. Similarly, the differences in levels of details and accuracy make it challenging to locate corresponding elements.

This research aims to propose a systematic approach to identifying commonalities in disparate datasets, that is, to provide alternatives to the few existing methods to identify corresponding elements in two datasets. The evaluation and validation of the proposed methods are based on the comparison of results with the ground truth dataset. The framework aims to reduce the repetitive human element and develop systematic identification and categorizations. Finally, the goal is to compare and identify dataset inconsistencies and update the information in the historical maps by exploiting the positional accuracy of OSM. The major contributions can be summarized as follows.

- The proposed node-based heuristics method finds corresponding elements in two datasets with different levels of spatial accuracy. The approach compares the results from the combined algorithm with manual conflation conducted by an expert and obtains more than 88% accuracy. However, the methodology can only recognize the nodes with the minimum value of the objective function and not other nodes close to the minimum value, and thus only suitable for one-to-one mapping. Similarly, it incorporates only the length and not the shape of the connecting geographical line features when evaluating nodes. This limitation is assessed in the second study.

- The second study proposes a genetic algorithm-based optimization that relies upon five different spatial aspects of links to locate and isolate dissimilar links. A positive relationship between the links with lower dissimilarity scores and match rates from heuristic approaches is demonstrated. Further, the links in the historical network that have dissimilarity more than the threshold with the target dataset are flagged for manual examination.

The proposed machine learning approach is an alternative to the heuristic methodology, along with added spatial features like classifying nodes by k-core decomposition and graph bridge decomposition to identify corresponding elements. The approach also explores different matching strategies such as one-to-one node matching, one-to-many node matching, and one-to-cluster matching.

CHAPTER 1

USING NODE-BASED HEURISTIC APPROACHES IN GIS DATASETS FOR IMPROVING POSITIONAL ACCURACY

1.1 ABSTRACT

Rapid advancements in location-based technologies have contributed to the availability of geospatial datasets from multiple sources. However, these datasets contain distinct and varied representations of real-world entities due to various types of data structure needs and use that define data structure, level of representation, and prioritization and inclusion of unique details. Conflation, the process of synthesizing disparate geospatial datasets with different attributes and positional accuracies into a single dataset, has been a critical issue needed to improve the quality and conformity of information on spatial features. A crucial step of the conflation process is to locate the corresponding elements shared between different GIS datasets via feature matching. This research proposes a node-based heuristic approach that incorporates a semi-automated algorithm to improve feature matching of GIS datasets with different levels of detail and positional accuracy. The proposed method demonstrates the method's applicability with the example of two railroad networks in the United States with greater than 88% matching counterparts.

Keywords: conflation, network matching, levels of detail, positional accuracy, node-based

1.2 INTRODUCTION

Integrating or synthesizing information from different sources is inevitable when there is a need to enhance data quality. The process of improving information in one dataset by supplementing data is called conflation [1] . The practice of conflation is not limited to a particular field, and the use of conflation tools is ubiquitous for integrating different datasets. However, data disparity across domains presents obstacles in using the data for analysis or mapping.

Initially, conflation meant eliminating any spatial inconsistencies from different datasets to achieve the desired accuracy and ease the transfer of attributes. Recently, researchers have used conflation for data integration if different data resources exist and combined the information into a single reference. GIS is one of the domains where conflation is essential because of the increased scalability and availability of data sources. However, one critical issue highlighted is that these datasets bring many inconsistencies in the information collected from multiple sources.

Historical datasets, also called reference datasets, typically originate from different agencies at different time points, each with different scopes and objectives. These fragmented datasets characterize different aspects of the same object. For example, the network that models a traffic engineer's maximum speed and capacity would be different from a dataset that describes the physical condition modeled by a maintenance engineer. Therefore, historical datasets have accumulated key attribute information that is rigorously tested over time. On the other hand, target datasets such as OSM, an open-source and voluntarily gathered information dataset, often miss or lack relevant attributes, though greater accuracy and the latest information are ensured[2]. Therefore, transferring attributes to the target dataset from reference datasets is crucial to enhancing datasets' positional accuracy and completeness. The substantial benefits from the integrated

dataset include a deeper understanding of objects from several viewpoints, which enables conclusive geographic analysis [3] [4] or updating newer information in datasets [5] [6].

When matching two networks with different levels of detail, the possible feature (spatial object) relationships for matching are classified as unmatched (1-0 or 0-1), one-to-one relationships (1-1), many-to-many (N-M), and one-to-many relationships (1-N). First, the reference network does not have a corresponding counterpart to the target network (1-0 or 0-1). Second, one feature in the reference dataset matches exactly one feature in the target dataset (1-1). Third, many-to-many relationships are relatively rare in this type of dataset but might happen when the specifications are different. Finally, one-to-many cases (1-N) occur when a feature in the reference network matches multiple features in the target network. For example, a single line feature in historical datasets might describe multiple line features at junctions [7], or a node may be represented by many arcs connected to them [8]. This discrepancy in levels of detail causes non-systematic and sometimes extreme displacements of features in different parts of the historical map [9]. Since existing algorithms generally rely on proximity, these non-systematic displacements are prone to cause incorrect matching of features [10] [11] [12].

Similarly, the conflation of the datasets that have originated in different times (also known as non-contemporary datasets) entails several challenges: (1) the generalized or simplified geometrical features of historical datasets result in higher positional inaccuracy, (2) the lower level of spatial detail for these datasets compared to recent datasets (e.g., OSM), and (3) the occasional change of sub-regions with time. However, the previous studies on conflation have usually focused on examining the efficiency of updating existing data through newer data or evaluating data accuracy. Specifically, few studies address conflation issues when two non-contemporary datasets with different levels of details are integrated using only spatial elements.

Therefore, from a methodological perspective, there is a need for a conflation method that is efficient, reliable, and flexible. This paper presents a set of algorithms for automatically identifying anchor points from significant dissimilarities in position, length, or shape of spatial features between two datasets (target and reference). First, three different procedures were proposed for identifying corresponding elements, each relying on specific network property to obtain robust matching rates. Moreover, the proposed algorithm demonstrates improved reliability of feature matching in datasets with different levels of detail, using only spatial elements with minimal to no human intervention. Second, the performance of the proposed algorithm is assessed with a case study with two GIS datasets that have different levels of detail. Finally, the results and findings to draw future research direction are discussed.

1.3 LITERATURE REVIEW

The idea of conflation was initiated around the mid-1930s, but research work was scarce. However, in the late 1980s, conflation became mainstream due to the increased data available from different data sources. Therefore, the importance of having accurate and comprehensive digital spatial data sets via conflation has been underscored repeatedly by industry and academia [13] [14] [15] [16]. The research on conflation primarily focuses on improving existing maps with information available in different forms in different sources.

The earliest conflation method adopted by the US Census Bureau [17] [1] was based on rubber-sheeting. This method uses a bottom-up computational procedure. The process starts by matching the nodes with the counterpart points known as "anchors." It then recursively applies transformations locally in each region to make them more correlated with each other. Finally, the algorithm calculates discrepancy in each iteration and terminates once the difference is smaller than a set threshold. Many subsequent algorithms [18], [19] followed the same bottom-up approach from [1].

In another study, Filin, S. et al. [20] proposed a path-based conflation procedure that could additionally detect one-to-many matches. The methodology first identifies corresponding anchor nodes based on proximity. Then, the set of candidate anchor nodes is either confirmed or invalidated using the "round-trip walks" procedure. Similarly, another widely used conflation method is based on buffer analysis. A simple buffer method [21] quantifies the similarity of two links based on the percentage of the buffered area of one feature that falls into another. The k-closest pairs queries (KCPQ) find counterpart feature pairs in two networks that are most proximate to each other [22]. However, these methods make choices one after another in a series of steps such that each step increment makes the most significant improvement in the objective function. Furthermore, these algorithms do not consider the spatial context and match disparate features. As a result, there is little way to identify these false matches.

Li, L. et al. [11] and Lei, T. et al. [23] introduced an optimization-based approach, a significant separation from traditional procedure-based conflation methods. An assignment problem using similarity or discrepancy of features was employed and could automate conflation without manual inputs (for anchor points). Also, if a human expert is available, the external knowledge is easily incorporated, adding constraints. The changed model generated constraint-based optimal results compatible with the input. However, these approaches were only tested on datasets with similar aspatial and feature detail levels. Moreover, they do not consider the topological aspects of the network, which could help match networks with high positional differences.

This paper proposes a novel method because it incorporates proximity between corresponding nodes and connectivity to the neighbor nodes for the feature matching process. This allows the algorithm to access both spatial context and topology characteristics to identify corresponding features. Therefore, the proposed method uses only spatial elements of two datasets and requires minimum to no human intervention. However, the matching process relies on the data quality of the target dataset. Thus, an

assumption is made that one-to-many or one-to-one correspondence exists between each node from historical to the target network.

1.4 METHODOLOGY

Concepts and Definitions

First, as a conflation method, this study demonstrates how to integrate two GIS networks (a target network with higher positional accuracy and a reference network with richer attribute information) via a set of key elements for the feature matching process. A GIS network is defined as a system of point features (nodes) and interconnected line features (links) to represent edges. The seven key elements and their definitions for the GIS network are elaborated below to describe the matching process of the proposed method. The elements are Reference Network, Target Network, Search Buffer Distance (SBD), Candidate Node, Anchor Point, Degree of a Node, and Shortest Path Algorithm (SPA).

Reference Network:

The Reference Network, also known as the *historical* network, is a digital vector map composed of nodes/vertex, links/edges, and attributes. It is assumed that the reference network is free of connectivity issues. For simplicity, the symbol $G^r\ (N^r, E^r, W^r)$, is used to represent an undirected graph $G^r\ (N^r, E^r, W^r)$, where N^r is a set of nodes $\{n_1^r, n_2^r, n_3^r, \dots, n_i^r\}$, E^r is the set of edges $\{e_1^r, e_2^r, e_3^r, \dots, e_j^r\}$, and W^r is a set of attributes $\{w_1^r, w_2^r, w_3^r, \dots, w_l^r\}$.

Target Network:

The Target Network is a newer map with a higher level of detail and positional accuracy. It is represented as an undirected graph $G^t\ (N^t, E^t)$, where N^t is a set of nodes $\{n_1^t, n_2^t, n_3^t, \dots, n_k^t\}$ and E is the set of edges $\{e_1^t, e_2^t, e_3^t, \dots, e_l^t\}$.

Search Buffer Distance (SBD):

A buffer is a zone around a feature (node or link) encompassing all the areas within a specified distance. SBD is the distance from any node in N^r within which all the nodes in N^t are considered candidate nodes. Thus, SBD is the maximum euclidean distance between corresponding nodes on two GIS networks, G^r and G^t.

Candidate Node (V_i):

Candidate nodes, represented by $V_i = \{v_{i1}, v_{i2}, v_{i3}, \dots, v_{im}\}$ are the subset of N^t within SBD of any node n_i^r. Each reference node has multiple candidate nodes.

Anchor Point (a_i):

The candidate node in V_i with the minimum value of the objective function is considered an anchor point (a_i) for the reference node n_i^r.

Degree of a Node ($deg(n_i)$):

The degree of a node (n_i), represented by $deg(n_i)$ is the total number of links connecting node n_i.

Neighbor Nodes (B_ir, B_i^t):

Neighbor nodes of any reference node (n_i^r) are the nodes that are connected to the reference node by a single link. Let the neighbor nodes for node n_i^r is given by $B_i^r = \{b_{i1}^r, b_{i2}^r, b_{i3}^r, \dots, b_{im}^r\}$, where $m = deg(n_i^r)$. B_i^r is the set of nodes that are connected to node n_i^r with exactly one link (E). Similarly, $B_i^t = \{b_{i1}^t, b_{i2}^t, b_{i3}^t, \dots, b_{im}^t\}$, is the set of neighbor nodes in the target dataset for candidate nodes in V_i. When the exact neighbor nodes in the target dataset are not known, $B_i^{t\prime} = \{b_{i1}^{t\prime}, b_{i2}^{t\prime}, b_{i3}^{t\prime}, \dots, b_{im}^{t\prime}\}$, is approximated by evaluating the closest point in the closest edge in the target dataset.

Shortest Path Algorithm (SPA):

The length of the path connecting any candidate node (V_i) with its neighbor node (B_i^t) is identified using Dijkstra's SPA. The SPA solves the shortest-path problem in a weighted network with a single-source and single sink. When two endpoints of a link is known in the target dataset, the link connecting those endpoints is assumed to be the shortest path between them for subsequent analyses. Thus, the shortest path is assumed to be the best approximation for identifying links connecting two adjacent nodes. Since both the historical and target dataset attributes are unknown, no one-way restrictions, turn restrictions, or junction impedance constraints are considered. However, it should be noted that, in real-world situations, there could be restrictions on movement between two nodes.

The entire feature matching process consists of five steps. The first step, data preprocessing, ensures data consistency with projection, format conversion, and topological validation. A fixed SBD for the pair of networks is calculated in the second step. The third step identifies anchor points determined by three algorithms based on the path lengths to neighbor nodes. In the fourth step, the anchor points are verified through manual validation. Then, the verified anchor points are used to identify the corresponding links in the target dataset. Finally, the results are evaluated by comparing results from manual conflation. The steps are explained in detail below.

Data preprocessing

First, an appropriate coordinate system is chosen depending on the location and the map's extent. Then, both reference and target datasets are projected to the same coordinate system to reduce possible geometric errors. This ensures that both maps are on the same surface with the same geographical projection.

Second, both the datasets are filtered to include only the same mode of transportation to ensure that the shortest path lengths are calculated only for that mode. Finally, since the

node evaluation is based on its connectivity, it is ensured that each node in the target dataset has at least one route to the rest of the network. These requirements are crucial to avoid unnecessary processing in subsequent steps.

Third, any missing links in the target dataset or apparent differences between the two maps should be checked manually. This step is crucial because a high-resolution map, such as OSM, is not topologically correct in its standard form as it is often collected by untrained contributors mainly focused on mapping activities [23]. Moreover, due to a lack of standardization, the quality and coverage of OSM are not homogeneous [24].

Calculation of SBD

The next step is to calculate a justifiable distance to search for anchor points in the processed dataset based on proximity. The search for candidate nodes in the target dataset is based on two approaches, either (1) fixing the total number of candidate nodes or (2) fixing the buffer distance (threshold) varying candidate nodes. Since the nodes in both datasets are not distributed uniformly and have different detail levels, fixing the number of nodes would generate wasteful search distances in sparse areas while having minimal search distance in dense areas. This leads to evaluating inappropriate nodes and overlooking possible candidate nodes. On the other hand, the second approach assumes that all candidate nodes in the target network are always within a fixed buffer distance of the corresponding node in the reference network. Since the fixed buffer distance is based on the overall precision of the two networks[25], the second approach is preferable.

However, for a fixed buffer distance, a lower buffer distance may not consider all the possible nodes, while a higher value entails computational "inefficiency." The inefficiency is due to the presence of too many nodes for the algorithms to evaluate the objective function. Thus, nodes are randomly chosen from the Reference Network and evaluated to calculate a justifiable search distance. First, the candidate nodes at different buffer distances are evaluated for these nodes. Next, both procedures calculate the anchor

points for each buffer distance. Finally, the least buffer distance is chosen such that any increment in buffer distance would not change the predictions from both the procedures. This buffer distance is fixed for all the subsequent analyses.

Identification of Anchor Points

This chapter prescribes three different procedures to identify the anchor points for each node in the reference network. The first procedure is based on the minimum sum of path lengths to neighbor nodes. The second procedure is based on the difference in corresponding path lengths in the reference and target network. The third procedure is the normalized weighted combination of the first and second procedures. The first and second procedures quantify different aspects of the pair of networks to find a set of the best anchor points. The procedures of the three methods are described below.

Procedure 1: The Minimum Sum of Path Lengths to Neighbors (MSPL):

MSPL assumes that the anchor point is the candidate node such that minimum path distance has to be traversed to reach all its neighbor nodes. Thus, the objective function of this procedure is the sum of path lengths to the neighbor nodes in the target network. Since this procedure only considers the path lengths in the target network, any inaccuracy in the length of edges in the reference network does not influence MSPL. Mathematically, the anchor point (a_1^i) is expressed as equation (1).

$$a_1^i = \min\left(\gamma_1^i\right) \qquad (1)$$

Let $d(X, Y)$ be the path lengths between known points *X* and *Y*. Then, the sum of path lengths to neighbor nodes (b_{ij}^{t}) for candidate nodes (V_i) is calculated as follows:

$$\gamma_1^i = \sum_{1}^{m} d(V_i, b_{ij}^{\mathrm{t}}) \quad \forall j \qquad (2)$$

Figure 1-1 illustrates the MSPL with a sample network that consists of neighbor nodes (b_1^t, b_2^t, b_3^t) and candidate nodes $(v_1, v_2, v_3, v_4, v_5)$ for reference node n_i^r. The value of the objective function (γ_1) is calculated for each candidate node. For example, for the candidate node v_1, the shortest path lengths to its neighbors b_1^t, b_2^t, b_3^t are 10, 13, and 14, respectively. Thus, the sum of path lengths to neighbor nodes $\gamma_1 = 10 + 13 + 14 = 37$. Table 1-1 shows the calculation of γ_1 for each candidate node. Column γ_1 shows that v_3 has minimum value of γ_1. Thus, using MSPL, v_3 is the anchor point for reference node n_i^r.

However, there exists an issue of uncertainty (or approximation) in the coordinates of neighbor nodes. In earlier iterations of the algorithm, the anchor points for neighbor nodes are not yet evaluated. Thus, the neighbor node is approximated as the closest node within the SBD in the target network. Table 1-1 demonstrates that, assuming the approximate neighbor node $b_1^{t\prime}$ is 2 units length closer to all the candidate nodes, the estimation of anchor points remains unchanged using $b_1^{t\prime}$. This is because MSPL identifies the candidate node from which the minimum path must be traversed for all neighbor nodes. And, even though the sum of the path length changes, the candidate node with the lowest path length to neighbor nodes remains unchanged. Thus, using approximate neighbor nodes does not affect the predictions by MSPL.

Procedure 2: Minimum Residual Sum of Squares of Path Lengths to Neighbors (MSR2):

MSR2 assumes that the shortest path length to the neighbors from any reference node is the same for both reference and target networks. Thus, each point is computed as a minimum squared sum of differences in corresponding lengths in two networks. Mathematically, the a_2^i is expressed as equation (3).

$$a_2^i = \min(\gamma_2^i) \qquad (3)$$

The objective function (γ_2^i) is formulated as follows:

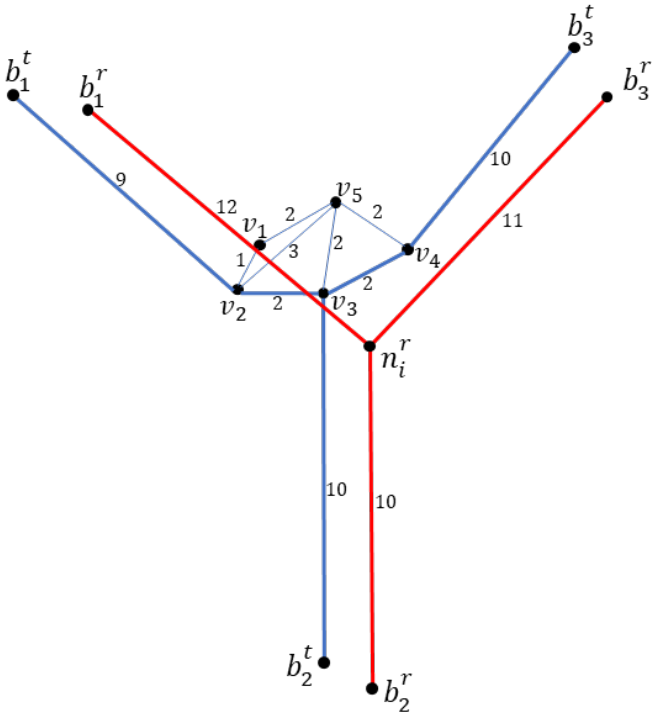


Figure 1-1 Demonstration of MSPL and MSR2

Table 1-1 Calculation of γ_1, γ_2 and demonstration of approximate node

		MSPL						MSR2					
		Neighbor nodes				γ_1	γ_1 with $b_1^{'t}$	Neighbor nodes				γ_2	γ_2 with $b_1^{'t}$
		b_1^t	$b_1^{'t}$	b_2^t	b_3^t			b_1^t	$b_1^{'t}$	b_2^t	b_3^t		
Candidate nodes	$\boldsymbol{v_1}$	10	8	13	14	37	35	2	4	-3	-3	4.69	5.83
	$\boldsymbol{v_2}$	9	7	12	14	35	33	3	5	-2	-3	4.69	6.16
	$\boldsymbol{v_3}$	11	9	10	12	33	31	1	3	0	-1	1.41	3.16
	$\boldsymbol{v_4}$	13	11	12	10	35	33	-1	1	-2	1	2.45	2.45
	$\boldsymbol{v_5}$	12	10	12	12	36	34	0	2	-2	-1	2.24	3.00

$$\gamma_2^i = \sqrt{\sum_{j=1}^{m} \left(d(V_i, b_{ij}^t) - d(n_i^r, b_{ij}^r) \right)^2} \quad \forall \text{ j} \qquad (4)$$

In the example network in Figure 1-1, γ_2 is calculated for each candidate node. For example, the candidate node v_1 has a distance of 10, 13, and 12 to neighbor nodes b_1^t, b_2^t and b_3^t, respectively. In contrast, the reference node (n_i^r) has a different distance of 12, 10, and 11 to b_1^r, b_2^r b_3^r respectively. Then, γ_2 is computed with $\sqrt{(2^2 + (-3)^2 + (-3)^2)}$ = 4.69. In a similar way, γ_2 for other candidate nodes is calculated and summarized in Table 1-1. Among five candidate nodes, $\boldsymbol{v_3}$ becomes the anchor point for n_i^r because it has the minimum value of γ_2 of 1.41.

Similarly, Table 1-1 shows the effect of using an approximate neighbor node for procedure 2. Assuming approximate neighbor node $b_1^{t\prime}$ is 2 unit length closer to all the candidate nodes, the estimation of anchor point changes using approximate neighbor nodes. Therefore, using approximate neighbor nodes affects the predictions from MSR2.

Procedure 3: Minimum Standardized combination of MSPL and MSR2 (MSPLR2):

Since MSPL and MSR2 capture different aspects of the pair of networks in finding anchor points, a combination of both could numerically assess both aspects simultaneously. However, the corresponding numerical values between these two procedures are not directly comparable. MSPLR2 is a compromising method that uses a weighting scheme for normalized values of γ_1 and γ_2 to find anchor points. The normalized values with weights are added to get a combined objective function. The minimum value of this function is the anchor point for each reference node for MSPLR2. Mathematically, the a_3^i is expressed as equation (5).

$$a_3^i = \min(\gamma_3^i) \qquad (5)$$

The combined objective function (γ_3^i) is given by,

$$\gamma_3^i = (1-\alpha) * z_1^i + \alpha * z_2^i \qquad (6)$$

where,

$$z_1^i = \frac{\gamma_1^i - \mu_1^i}{\sigma_1^i} \qquad (7)$$

$$z_2^i = \frac{\gamma_2^i - \mu_2^i}{\sigma_2^i} \qquad (8)$$

μ_1^i and μ_2^i are the means of the outcomes by equations (2) and (4), respectively, while σ_2^i and σ_2^i are the standard deviations of the outcomes by equations (2) and (4), respectively. The discriminatory weight (α) represents the proportion of the standardized objective function from MSPL, while ($1-\alpha$) represents the proportion of the standardized objective function from MSR2. The value of α is calculated based on a simulation where α is varied from 0 to 1.

Interactive Manual Validation

The interactive procedure allows an investigator to visually check and modify the assignments of anchor points from the algorithm. This is optional but critical to ensuring the anchor points generated by the automated algorithm are correct. Incorrect matches usually occur when target data has missing links, topologically incorrect links, or there exists a many-to-one relationship. When the target data does not meet these conditions, either there doesn't exist any node for evaluation within the SBD, or the path to at least one neighbor node is not found. This would lead to lower match rates. Thus, correction by interactive validation in an early stage of the matching process lowers the probability of errors in subsequent steps. This improves the performance and reliability of the matching result compared to the results using the algorithm alone.

Links Identification

When the anchor points are validated, each node in the reference network corresponds to a node in a target network. The next step is to identify the links in the target network. The endpoints of each link in the reference network are identified from the anchor points in the target dataset. Finally, for each link in the reference network, the links in the target dataset are identified as the shortest path within SBD between the anchor points.

In summary, identifying corresponding nodes begins by calculating the SBD for the entire dataset based on 30 randomly chosen nodes in the reference dataset. Next, candidate nodes for each reference node are evaluated. Next, the neighbor nodes in the reference dataset are calculated. If the anchor points for the neighbor nodes are not identified, then the approximate neighbor node is used. The algorithm stops when all the reference nodes are evaluated. The algorithm with the pseudo-code is summarized below.

Input: Reference and Target network

Output: Anchor Points A_1, A_2

1. $A_1, A_2 = \phi$ //initial anchor *points*
2. *let* N^r *= reference nodes sorted by distance to neighbor nodes*
3. *for* n *in* N^r:
4. $v_n = \phi$ *//initial candidate nodes for reference node n*
5. *for m in* N^t *within SBD of* N^r*:*
6. v_n*.append(m)*
7. b^t *= initial neighbor nodes in the target dataset*
8. $A_1[n]$ =*GetGamma1*(n, v_n, b^t)
9. $A_2[n]$ =*GetGamma2*(n, v_n, b^r, b^t)
10. A_1 = *min*$(A_1[n])$
11. A_2 = *min*$(A_2[n])$
12. *Export*

13. *End*

1.5 CASE STUDY

The algorithm was implemented on a personal computer with a python environment to examine the methodology's performance. The developed scripts can read networks in ESRI's shapefile, process it, and create a node-mapping table and shapefile as output. The mapped node can be compared to a ground truth dataset to evaluate the algorithm's performance.`

Data

This study tests the performance of the proposed algorithm with independent sources of GIS datasets using the Federal Railroad Administration (FRA) network as a reference network and an OSM as a target network. Both networks cover approximately the eastern half of the US. It is important to note that the FRA network has a lower level of detail, while the OSM has a higher detail level. The two datasets are described below.

FRA network:

FRA Railroad Network Database (FRARAIL) is the NHPN's railroad equivalent and can be used for analysis or national scale mapping. The FRA Network is a network that is a subset of the 1992 version of the FRARAIL that consists of links aggregated from the FRARAIL [25] show the spatial extent of the FRA network, which is used for analysis. The scale of the FRA varies from $1{:}\ 250{,}000 - 1{:}\ 2{,}000{,}000$. The FRA network with attributes is well tested for Network Flow Problems to predict traffic flows. It should be noted that the FRA network has simplified line features that are not positionally accurate. Due to this, the line features' lengths and shapes may not be reliable, and thus, it is pretty challenging to check the validity and update or modify the network manually.

OSM Rail Network:

The target dataset, OSM, is a collaborative project map to create freely available positionally accurate and detailed geographic data. OSM has a varying level of positional and geographic accuracy, which depends upon the location of the data, and the scale varies between 1:1,000 and 1:10,000. Since the FRARAIL reference network is a representation only of freight rail transportation, other modes such as motorways, bikeways, and footways are excluded from OSM. Based on the selection query on OSM coding conventions, OSM line elements of the rail network are extracted.

Figure 1-2 shows the differences between OSM and FRA Network. First, OSM Rail Network has more links and nodes, confirming the higher level of detail of the OSM Rail Network compared to the FRA network. Second, the average length of line features in OSM Rail Network is far lower than the FRA network, which indicates that one edge in the FRA network could correspond to multiple edges in the OSM Rail Network. The FRA network has 1,524 links and 1,103 nodes, while the OSM Rail Network has 233,226 links and 183,529 nodes. Similarly, the average length of links for the FRA Network is 30.1 miles, while the OSM Rail Network is 0.46 miles. Figure 1-3 highlights the differences between the two reference and target networks for two locations. The network densities are compared for Chicago, Illinois (a and b), and Corsicana, Texas (c and d). First, the complex rail lines in Chicago represent a highly dense network structure with significantly more links and nodes in the OSM Rail Network (a) than the FRA Network (b). Thus, each node in the FRA network could represent multiple nodes in the OSM rail network. On the other hand, Corsicana rail lines are sparse and have a simple structure for both the OSM (c) and FRA (d) Networks. Second, in terms of the geometry of the links, the FRA network is much coarser and less refined than the OSM network, which has a greater number of edges that represent more naturally occurring curved shapes. Finally, the OSM network is more detailed because it has additional links that do not correspond to any links in the FRA network.

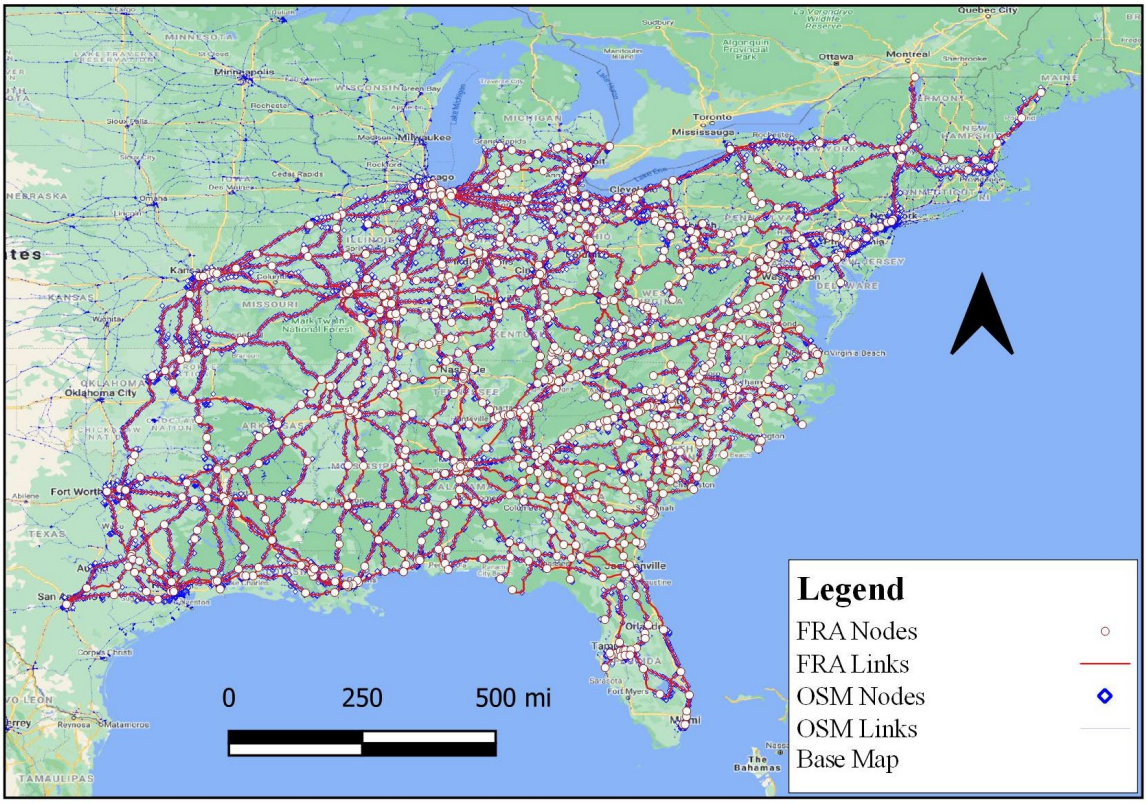
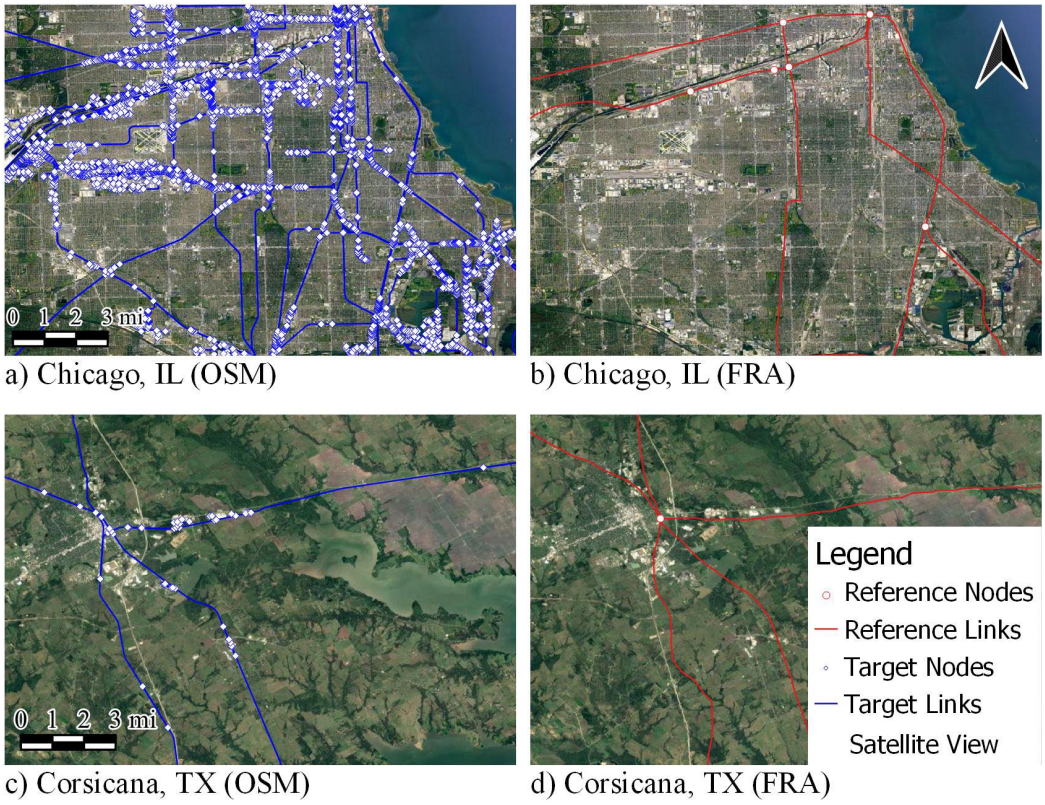


Figure 1-2 FRA Network overlayed on OSM Network



a) Chicago, IL (OSM)

b) Chicago, IL (FRA)

c) Corsicana, TX (OSM)

d) Corsicana, TX (FRA)

Figure 1-3 Comparison of FRA and OSM Networks

Zones of area for experiment:

For the experiment to compare the algorithm's performance by area characteristics, the entire study area was divided into zones with equal-sized map grids, which are a network of evenly spaced horizontal and vertical lines. However, since the nodes are not evenly distributed throughout the network, some zones have very few nodes. Due to having a small sample size for calculating match rates, the match rates for those zones could vary from 0% for all matched to 100% for all unmatched nodes. Thus, the zones with a small number of nodes were not considered to eliminate this possibility.

Evaluation Criterion

The anchor points obtained from the three procedures described above are compared to the actual pre-defined anchor points to verify correct and incorrect matches. A match occurs when the anchor point predicted by the algorithm is the same as the actual anchor point. A Ground Truth Dataset, which is a set of all the truly matched anchor points, is used to verify algorithm results.

Ground Truth Dataset (GTD)

The GTD is a manually identified set of nodes and links in the target network corresponding to the entire area in the reference network. First, it is created by overlaying the target OSM dataset on a raster image base map in ArcGIS. Then, each feature is compared and confirmed based on expert domain knowledge, shape comparison, and connectivity to obtain a geographically accurate dataset. It is important to note that, since the datasets have different levels of detail, one node in the reference dataset might correspond to multiple nodes in the target dataset. Therefore, each of these possible sets of nodes is manually identified in the GTD. The performance of the three procedures is tested by comparing it with the GTD based on the match rate criteria.

Match Rate, also known as True Match Rate, accounts for the rate of true matches

captured by the algorithm. Mathematically, the match rate is given by:

$$Match\ Rate\ = \frac{Number\ of\ Matches}{Total\ number\ of\ Predictions} \quad (7)$$

1.6 RESULTS

In order to assess the accuracy of the predictions made by the algorithm, corresponding elements for the FRA and OSM Networks were obtained, and the results were compared with the GTD.

First, an appropriate value of SBD was determined for the two datasets. A simulation with an increment of buffer distance up to 3.5 miles (see Figure 1-4) shows 30 randomly selected nodes from the reference network. The objective function γ_1 and γ_2 were calculated for these nodes. According to the result, increasing the buffer distance changes the predictions for up to 1.5 miles. However, changing the buffer distance beyond 2.5 miles from each node ($n_i{}^r$) does not affect the predictions by the two procedures. This could mean that most of the corresponding nodes in the target dataset lie within 2.5 miles distance of the reference dataset. Thus, an SBD of 2.5 miles was confirmed for all of the subsequent analyses.

Determination of Number of Zones

The entire dataset was divided into multiple zones to obtain the match rates for individual zones. Each zone had a distinct characteristic in terms of density, number of nodes, number of links, nodes ratio, and the average length of links. The match rate by all three different procedures for each zone was identified. However, the lack of criteria for sample size could highly influence the match rate for any zone. For example, in a zone with five nodes, with all five matched correctly, the match rate for that zone is 100% when in reality, the zone's characteristics may not explain such a high match rate.

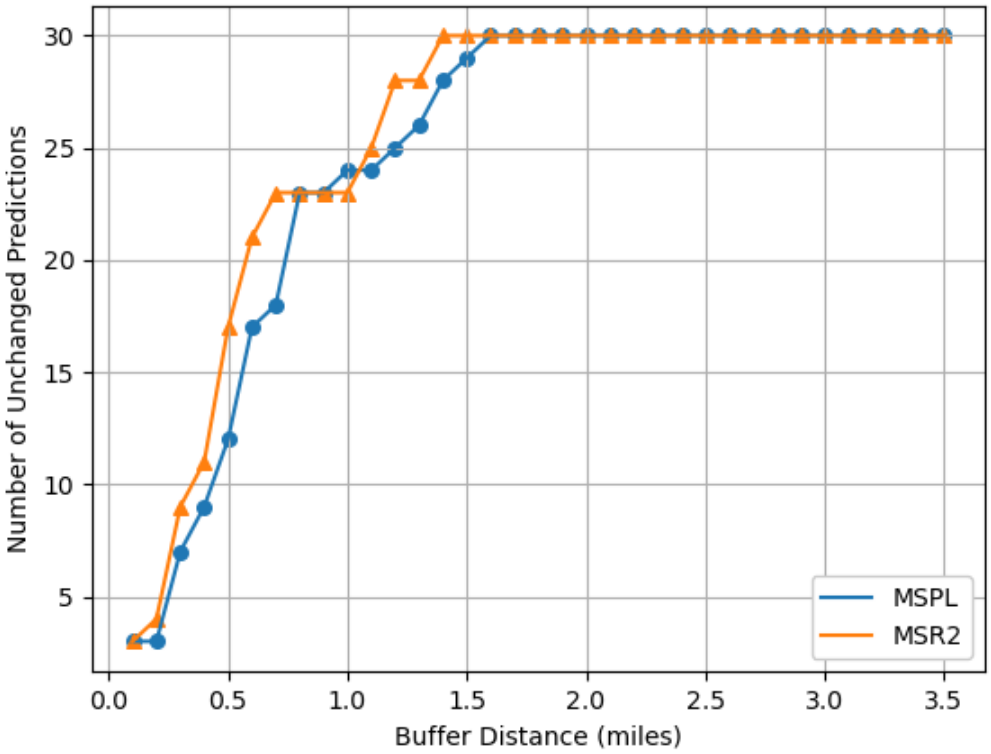


Figure 1-4 Buffer distance vs. Unchanged Predictions by MSPL and MSR2

Thus, each zone was required to have at least 30 nodes to be considered a valid zone. The choice of zones minimized the number of discarded nodes while maximizing the number of zones. Figure 1-5 shows the box plots for match rates for a different number of grids. Increasing the number of zones increases the number of discarded nodes while slightly increasing the number of match rates beyond the interquartile range (IQR), as shown in the figure. This could be because increasing the number of zones would increase zones with fewer nodes. Figure 1-6 shows how the grids divide the entire dataset. Although there are 6 grids in each direction, only 25 zones have nodes in them, out of which only 13 zones have nodes greater than 30.

Determination of Discriminatory Weight (α)

MSPLR2 is based on the mix of two procedures MSPL and MSR2, using a discriminatory weight ($\boldsymbol{\alpha}$). When $\boldsymbol{\alpha} = 0$, using equation (6), the results from MSLR2 imitate MSPL. While, when $\boldsymbol{\alpha} = 1$, the results imitate MSR2. Thus, any value of $\boldsymbol{\alpha}$ between 0 and 1 would perform a mix of procedures MSPL and MSR2. Figure 1-7 shows the match rates for different zones using different $\boldsymbol{\alpha}$ values. The lines show that a mix of the two procedures MSPL and MSR2 performs at least equal to or better than the individual procedures. Further, the black line shows the match rate for all zones, and it is evident that, as alpha increases, the match rate first increases up to $\alpha = 0.5$, then the match rate gradually decreases and finally drops sharply at $\alpha = 1$. This concludes that the match rate is the highest when we take an even mix of MSPL and MSR2.

Match Rates by Zones

Using (7), the match rates using three procedures, MSPL, MSR2, and MSPLR2, are 78.57%, 78.84%, and 88.12%, respectively. MSPLR2 has the highest match rates, followed by MSPL and MSR2.

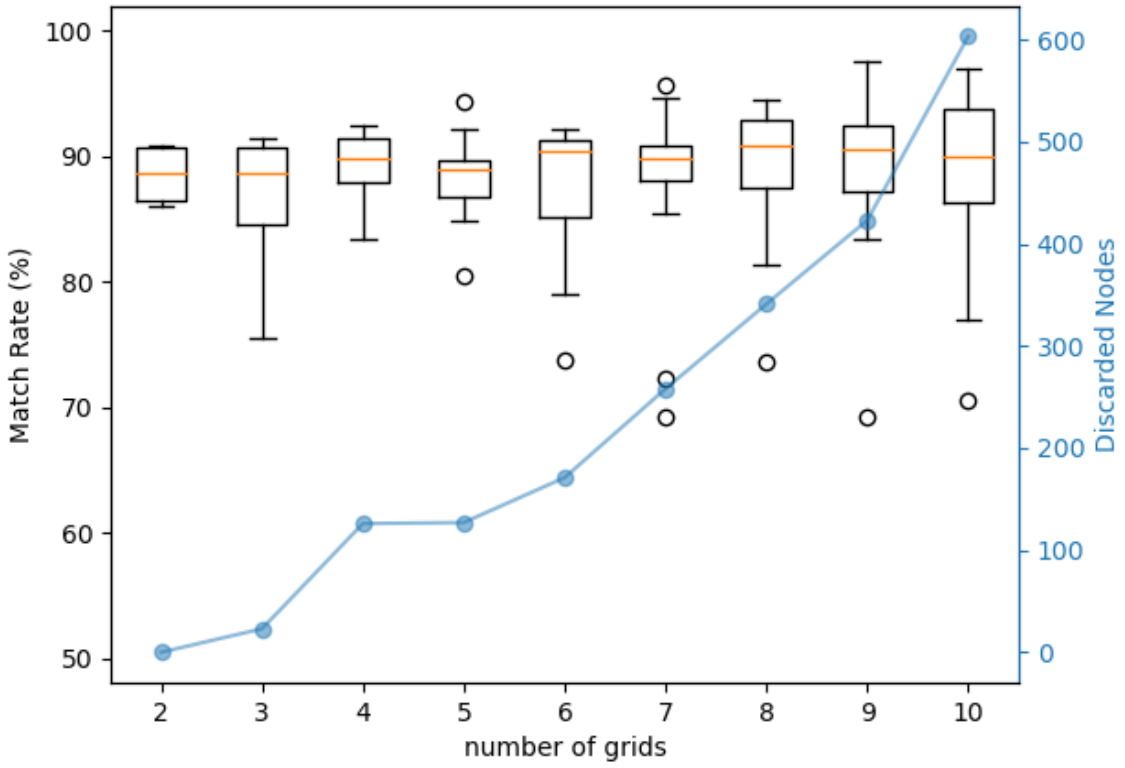


Figure 1-5 Determination of Number of Grids

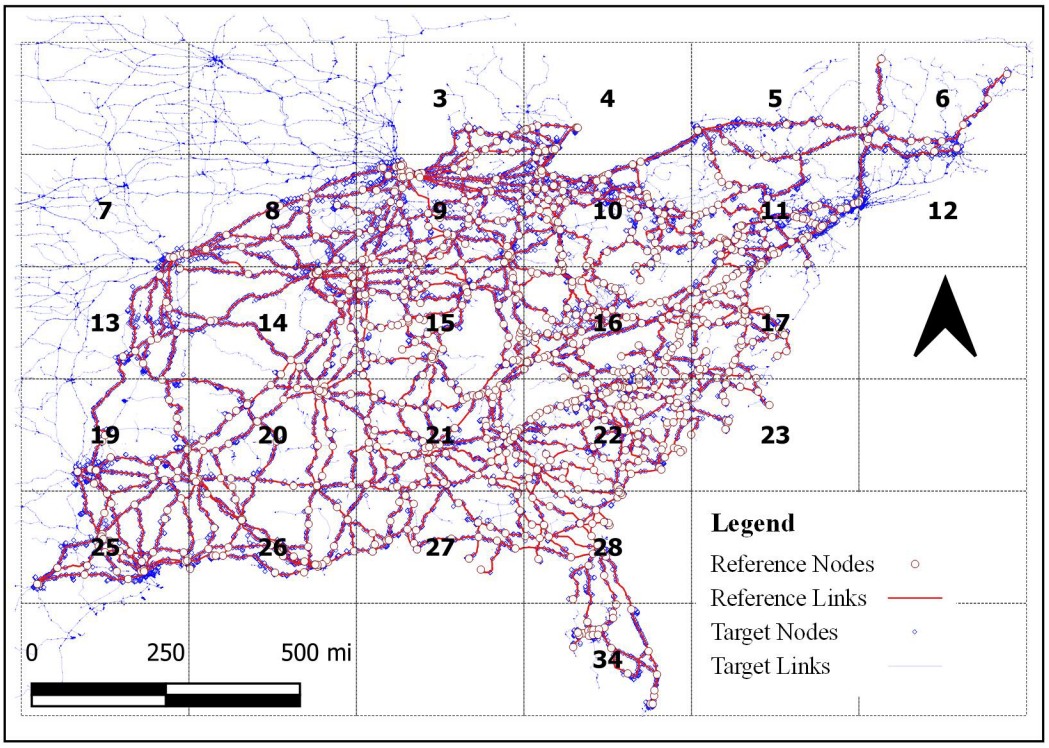


Figure 1-6 Division into Zones

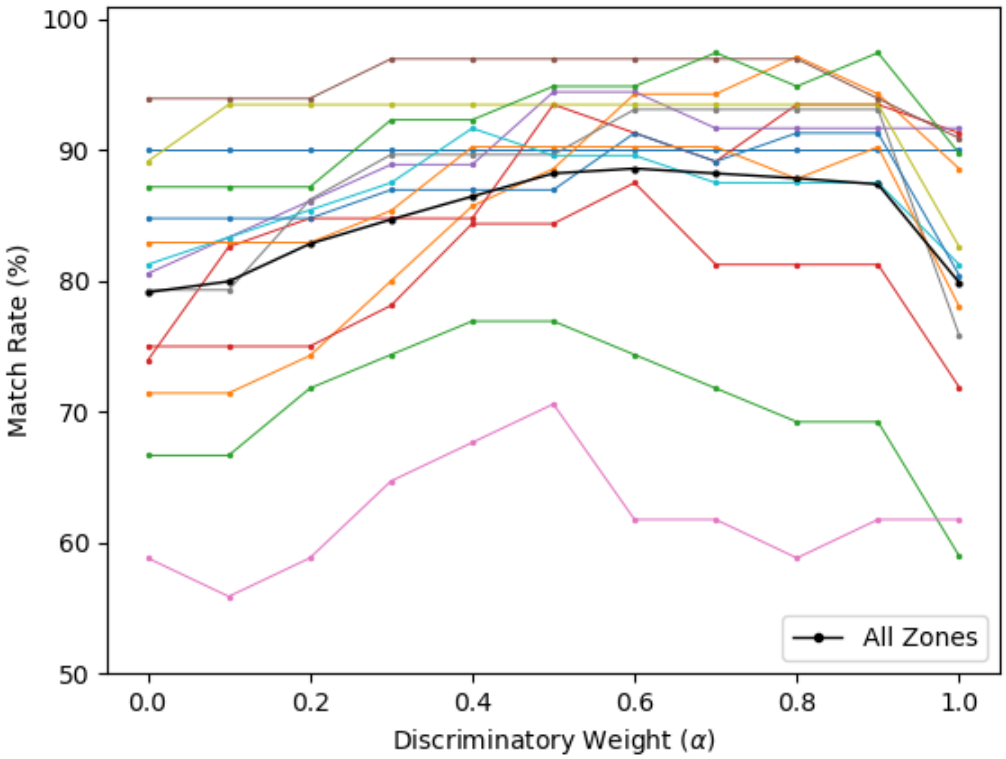


Figure 1-7 Determination of Discriminatory Weight

Figure 1-8 and Figure 1-9 show the match rates for the three procedures in different zones. The figure shows that MSPLR2 outperforms MSPL and MSR2 in all scenarios, indicating that MSPLR2 captures spatial information more than MSPL and MSR2. Of note is a slight negative relationship between the node ratio and MSPLR2, implying that the algorithm performs slightly better for areas with lower nodes ratios, such as that found in Corsicana, Texas. The match rates from MSR2 are dependent on the similarities in the length of edges connecting corresponding nodes. Thus, zones with a lower match rate from MSR2 could be due to the significant dissimilarities in the edge lengths of the OSM and FRA datasets.

Match Rates by the Average length to Neighbor nodes

Figure 1-8 shows the Match Rates by all procedures across all zones. The x-axis shows the average length to neighbor nodes, while the y-axis shows the match rate. The figure shows that the match rate by MSPLR2 is greater than both MSPL in all scenarios. However, we do not see any distinct relationship between the match rate and the average path length to neighbor nodes. This could be because the path length is insufficient to explain the difference in match rates. Furthermore, two nodes with the same path lengths may have very different characteristics because of the difference in connectivity.

Match Rates by nodes ratio

Similarly, Figure 1-9, upon observation, seems to have a slight negative correlation between nodes ratio and match rate from all three procedures. However, this is not the case. For example, MSPLR2 has a *R-squared* of 0.4 and a *p-value* of 0.17 suggests that the model does not explain the data variation and is not significant. Therefore, the nodes ratio does not explain the variation in the match rates.

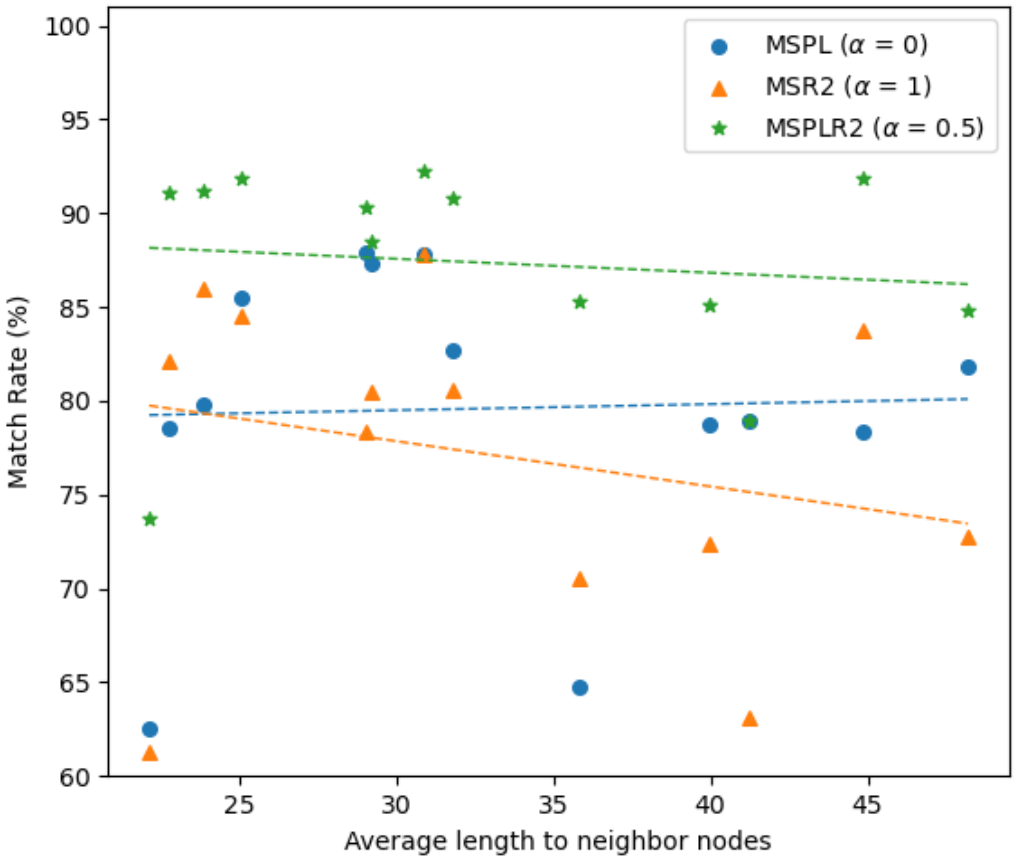


Figure 1-8 Match Rates by Average lengths to neighbor nodes

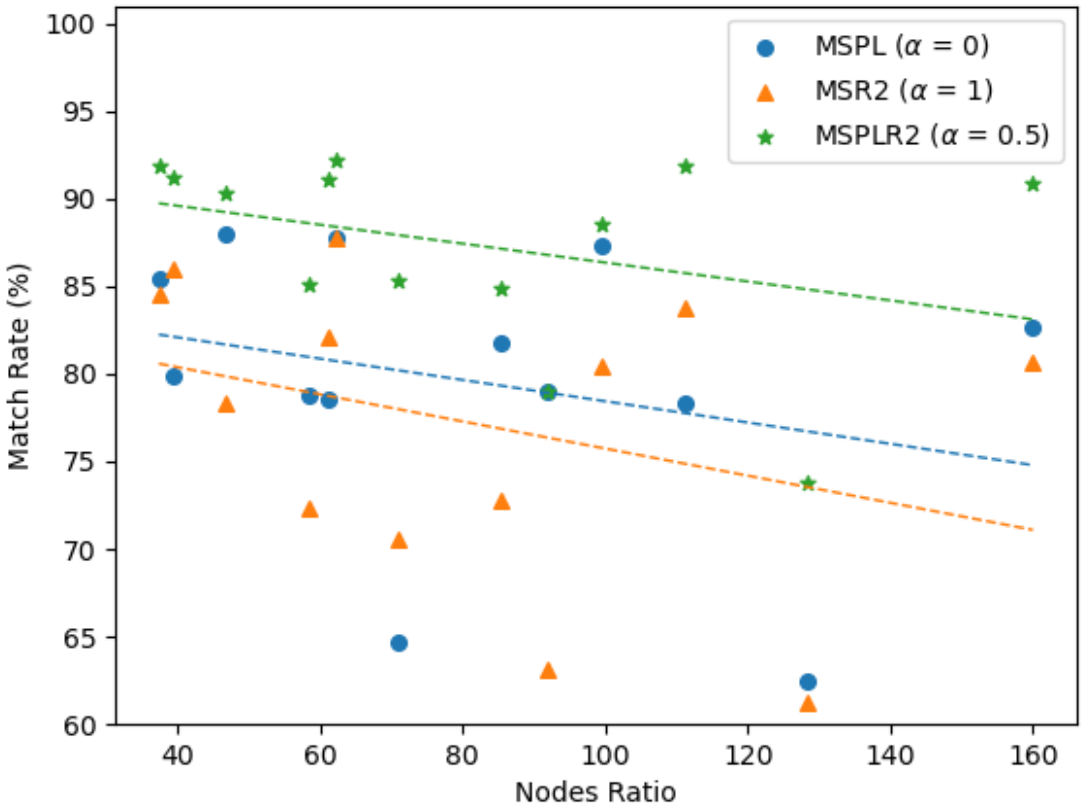


Figure 1-9 Match Rates by Nodes Ratio

1.7 DISCUSSION

The proposed algorithms MSPL and MSR2 demonstrated a substantial percentage of the conflation of point features correctly. A third weighted procedure, MSPLR2, was introduced to maximize the match rate by identifying the optimum mix of MSPL and MSR2. MSPLR2 performed consistently better than MSPL and MSPLR2.

Several things are worth noting. First, the algorithms performed inferiorly, especially for areas characterized by a higher nodes ratio and a smaller average length to neighbor nodes. Next, using a combined z-score provides new insights into data and overcomes problems in interpreting the data due to various tradeoffs. However, further research needs to be done to characterize situations where the summed z-score approach works the best and where they do not. Next, oversimplification in some areas of the historical network was observed that could make it problematic for the identification of anchor points. However, room to improve the algorithms was found, and three directions are suggested to address these issues to improve the algorithms for future research.

First, the use of approximate neighbor nodes could be elaborated via using iteration-based updating. For example, each iteration would update all the neighbor nodes simultaneously until the algorithm reaches a threshold or there is no change in predictions of anchor points. Next, a separate process to test the apparent differences of features in reference and target datasets. The process would check for any significant dissimilarities in links or nodes between datasets. Next, an SBD for the entire dataset could be replaced by a local SBD. This can be obtained either by identifying the search distance using clustering methods or by calculation of SBD for a smaller zone. Finally, in areas with multiple routes connecting a candidate node and its neighbor node, the route distance could be based not on the shortest path distance but on the similarity of links connecting them.

CHAPTER 2

ACCESSING THE QUALITY AND SPATIAL DIFFERENCE OF GEOGRAPHIC LINE FEATURES

2.1 ABSTRACT

The presence of geo-referenced data is prevalent throughout the world. The data is created with different formats or types of representations for the same real-world entities, which are not necessarily consistent with each other. This can undermine the validity of any GIS analysis that requires the inclusion of multiple data sources. Therefore, it is imperative to identify any relevant inconsistencies in the multiple datasets and to reconcile mapped discrepancies. The approach to comparing data contained in multiple maps is often greatly influenced by the purpose of the maps and the purpose for combining them. In particular, it is critical to identify differences in topology, shapes, or lengths of geographical line features if the maps are evaluated for usability in transportation routing efforts. For any VGI to be considered fit for routing, any apparent difference with the historical map should be minimized. This chapter presents a Genetic Algorithm (GA)-based optimization to identify the geographical line features that are significantly different between datasets with different levels of scale and spatial detail. The key procedure in the algorithm for comparison includes confirming similar features in different datasets that represent the same real-world entities and minimizing the discrepancies between links. The process is based on unweighted scores defined by five different aspects (indicators) of the link. Lower scores correspond to a more significant match between maps' key indicators. The result shows that the proposed algorithm provides a range of scenarios for decision-makers to identify, adjust, and correct the discrepancies in links from multiple maps.

2.2 INTRODUCTION

The volume of geo-referenced data generated, transferred, and utilized has increased in recent years. This has led to inconsistencies in the type and format of spatial attributes contained within datasets resulting in difficulties merging these attributes into a single repository. In addition, each datasets' ubiquitous availability, accessibility permissions, and applications have highlighted the need for continuity in feature classification and data attribute quality as geographic data is often used to support decision-making based on dataset prioritization and the best available data.

The recent increase in the number of online spatial datasets and the possibility of interactivity between users and web pages through online collaboration has been possible with Web 2.0[26]. For example, VGI is a new growing data source where hundreds of people can participate simultaneously in data collection, analysis, and dissemination[27, 28]. The rapid growth of VGI can be attributed to overwhelming volunteer interest, participation, and frequent updating. This leads to a self-enhancing and positionally accurate dataset; however, the collaborative nature of VGI by non-professionals is concerning. For example, allowing volunteers with no filter for expertise has contributed to 75% of the participants having no "professional geographic experience" [29]. As a result, VGI data could have a lower database quality than other professionally developed sources, like national, regional, and local mapping agencies, produced by those with experience and expertise.

Similarly, VGI is inherently heterogeneous, and the quality of data varies with the contributor's level of interest and knowledge regarding a specific geographic area. As a result, there is usually a spatial bias of information, with more data collected in urban compared to rural areas[30, 31] and bias toward a specific element of the network, usually affected by the interests of the volunteers[32]. Furthermore, the details in geographic areas with a higher number of volunteers are more robust than those in other

areas[33] [34]. Additionally, these biases could further be influenced by the knowledge of digital resources, the language of VGI application, differences in culture, and time of data submission[35]

Similarly, VGI lacks definite specifications compared to authoritative geographic datasets, resulting in data reliability and credibility[36]. While collaborative mapping improves data quality to a certain extent[24], frequent updates in the same features can sometimes lower the overall quality and usability of the data. Further, the lack of standard and varying specifications means that data quality varies over space and time.

In contrast, reference networks (also known as authoritative networks) are usually produced by (or under the control of) a national, regional, or local mapping agency that conforms to well-defined guidelines and is thus reliable[37]. However, their usability in analyses that require high positional accuracy and adaptability to changes can be lacking compared to VGIs. Thus, one could benefit from comparing, contrasting, and differentiating VGI and authoritarian datasets.

VGI has received much attention and has been subject to substantial research, particularly concerning OSM. For example, several studies have tried to assess the quality of OSM by comparing it with authoritative datasets [38-40]. The comparisons usually assume that the authoritative data have an acceptable quality as they are created through rigorous specifications and can act as reference networks to evaluate the quality of VGI datasets. These studies focus on matching data while considering different data quality elements, such as positional accuracy or completeness. However, it is laborious to find corresponding feature attribute elements when the two datasets significantly differ in positional accuracy, connectivity, and levels of detail. Thus, automating the process of finding differences in two non-contemporary maps is crucial.

This chapter provides a novel approach to assessing the differences between two datasets. The methodology uses several quality measures and indicators to assess data quality

differences. The objective is to identify the critical geographic line features in VGI that are not "similar" to those contained in authoritative datasets. The similarity is score-based, with the higher scores representing higher differences in relevance metrics.

2.3 LITERATURE REVIEW

As a representative stream of VGI, OSM is a trendy project that collects and provides timely and detailed geographic information through the web and distributes it for free. As a result, OSM has become an extensive database and has been successfully applied in different fields and in projects such as 3D City Models [41], pedestrian navigation [42] [43], robot navigation [44], spatial queries [45] [46], and disaster logistics [47].

However, unlike authoritative geographic information, it carries no data quality assurance [48]. Therefore, it is crucial to understand and evaluate the quality of OSM data before applying it to specific research. There have been several attempts to assess the quality of OSM data against other authoritative sources. For example, [49] compared the positional accuracy, completeness, and users' density of OSM and OS Meridian data. [50] extended the work of [49] and demonstrated the same results with more extensive spatial data elements, such as point, line, and polygon to access the geometric accuracy, semantic accuracy, attribute accuracy, completeness, temporal accuracy, logical consistency, lineage, and usage. Researchers found that the OSM data was flexible and responsive but heterogeneous and could have possible limitations in its applicability. [51] compared OSM and TeleAtlas data in Germany and found discrepancies in the total length of roads. [52] chose five cities and towns to analyze the quality of OSM in Ireland against Google Maps and Bing Maps. However, the study found no straightforward preferred dataset among the three mapping platforms.

Positional accuracy is usually measured by comparing a test dataset with a reference dataset that is assumed to represent the ground truth[53]. Similarly, completeness

assesses the quality of the VGI based on the errors of "omission" or “commission” for datasets that represent the absence or presence of excessive data[54, 55].

Quality is always relative and often calculated by comparing a dataset to a "ground truth" dataset. The approach in this chapter compares the newer datasets such as OSM to a much older historical dataset. The aspects of the two datasets that highlight their apparent differences are compared such that the differences could be corrected manually. The differences include the endpoint differences (quantifies how far the corresponding endpoints are), angular differences (assesses the angle between corresponding links), euclidean distance (evaluates the similarity in Euclidean distance between corresponding endpoints of a link), path (assesses the similarity in path lengths), and edge differences (assesses the differences in curvature and shape of corresponding links).

Despite the continuous voluntary updating of the OSM dataset for the past 15 years, the OSM dataset still lacks coverage in areas where volunteers' knowledge and interest are low. Furthermore, the issue of locating missing or significantly different features is essential when one of the datasets has proprietary attribute information and thus cannot be used for comparison.

The following section illustrates the computation of a metric developed to identify the differences in geographical line features of two datasets. The metric is used to highlight features in datasets where the routes between any two points in the two maps are different. The two points are in this chapter are the endpoints of links in the reference network. Section 4 results compare the lowest scores with the matches in Chapter 1 and identify the links with the topological issues in corresponding maps.

2.4 METHODOLOGY

This paper proposes an optimization-based approach for identifying spatial differences in GIS datasets with different levels of detail. Linear object matching is considered an

optimization problem to iteratively investigate the geometric criteria (denoted by m) using similarity evaluation criteria. The proposed approach uses five different geometric criteria: Normalized Endpoints' Displacement, Normalized Angle, Normalized Euclidean Distance Difference, Normalized Path Difference, and Normalized Shape Score. Each criterion is detailed in Section 3.2.

Concepts and Definitions

First, this study demonstrates the comparison of two GIS networks (a target network with higher positional accuracy and a reference network with richer attribute information) via a set of key elements. A GIS network is defined as point features (nodes) and interconnected line features (links) representing edges. The four elements and their definitions for the GIS network process are elaborated on below. The elements are Reference Network, Target Network, Search Buffer Distance (SBD), and Shortest Path Algorithm (SPA).

Reference Network:

The Reference Network, also known as the historical network, is a digital vector map composed of nodes/vertex, links/edges, and attributes. It provides numerous resources for the target network to be matched. It is assumed that the reference network is free of connectivity issues. For simplicity, the symbol $G^r\ (N^r, E^r)$ is used to represent an undirected graph where N^r is a set of nodes $\{n_1^r, n_2^r, n_3^r, \dots, n_i^r\}$ and E^r is the set of edges $\{e_1^r, e_2^r, e_3^r, \dots, e_j^r\}$.

Target Network:

The Target Network is a newer map with a higher level of detail and positional accuracy. It is represented as an undirected graph $G^t\ (N^t, E^t)$, where N^t is a set of nodes $\{n_1^t, n_2^t, n_3^t, \dots, n_k^t\}$ and E^t is the set of edges $\{e_1^t, e_2^t, e_3^t, \dots, e_l^t\}$.

Search Buffer Distance (SBD):

A buffer is a zone around a feature (node or link) encompassing all the areas within a specified distance. SBD is the distance from any node in N^r within which all the nodes in N^{t} are considered candidate nodes. Thus, SBD is the maximum distance between corresponding nodes on two GIS networks, G^r and G^t.

Shortest Path Algorithm (SPA):

The shortest path between any two nodes in a network is the shortest network distance between the two nodes. Therefore, SPA is assumed to be the best approximation for identifying links or the actual flow between two adjacent nodes. For simplicity, no one-way restrictions, turn restrictions, or junction impedance constraints are considered. However, it should be noted that, in real-world situations, there could be restrictions on movement between two nodes.

Calculation of Search Buffer Distance (SBD, s)

A buffer in GIS is a robust tool that identifies the proximate features around a feature. A buffer is employed with criteria to identify apparent differences in the two maps on multiple occasions. When searching for corresponding features in the target map, the algorithm explores all the features within a specified distance to identify candidate nodes in the target network. In the selection process of SBD, a smaller distance buffer value leads to a less exhaustive search and could miss potential candidate nodes. In contrast, a more significant distance leads to extensive searches with a higher computational expense for the algorithm's execution to complete the search.

The cumulative distribution of links with different buffer distances is explored to determine an appropriate SBD. As the target network has a higher level of detail, new links are encountered with an increase in buffer distance. Usually, after a certain distance, the increase in the number of links with an increase in buffer distance stabilizes to a fixed

number. However, sometimes, the number of links increases even at more considerable distances. This is because of the presence of links in the target dataset that do not have corresponding links in the reference networks. Therefore, an SBD that encompasses only the relevant links is chosen in these instances.

Criteria for Geometric Differences (m)

Comparing corresponding geographical line features in two datasets with different measures is a well-known strategy for assessing the geometric differences in two datasets. To accomplish this, the endpoints of any line feature in the reference dataset are located. Then, the corresponding endpoints in the target dataset are located using the endpoints. Next, for every two endpoints in the target dataset, the corresponding link in the target network is approximated to be the shortest path between endpoints. Finally, the "best" shortest path is determined based on a minimization criterion of five different scores representing different aspects of the two links being compared. These scores are made based on Normalized Endpoints' Displacement, Normalized Angle, Normalized Euclidean Distance Difference, Normalized Path Lengths Difference, and Normalized Shape Difference. Each of the scores' values ranges between 0 and 1, with a lower value representing a stronger match.

Normalized Endpoints' Displacement score (m_1)*:*

This score (m_1) quantifies the proximity of the corresponding candidate nodes in the target network for each node in the reference network. The score is calculated as the sum of the Euclidean distances between endpoints in the reference and target network divided by the maximum possible search distance, which is equal to SBD (*s)*. Figure 2-1 shows *A* and *B* are two nodes in reference dataset with their corresponding nodes being *A'*, *B'* in the target dataset. Using the figure, m_1 is given by,

$$m_1 = \frac{d_{AA'} + d_{BB'}}{2s} \qquad (1)$$

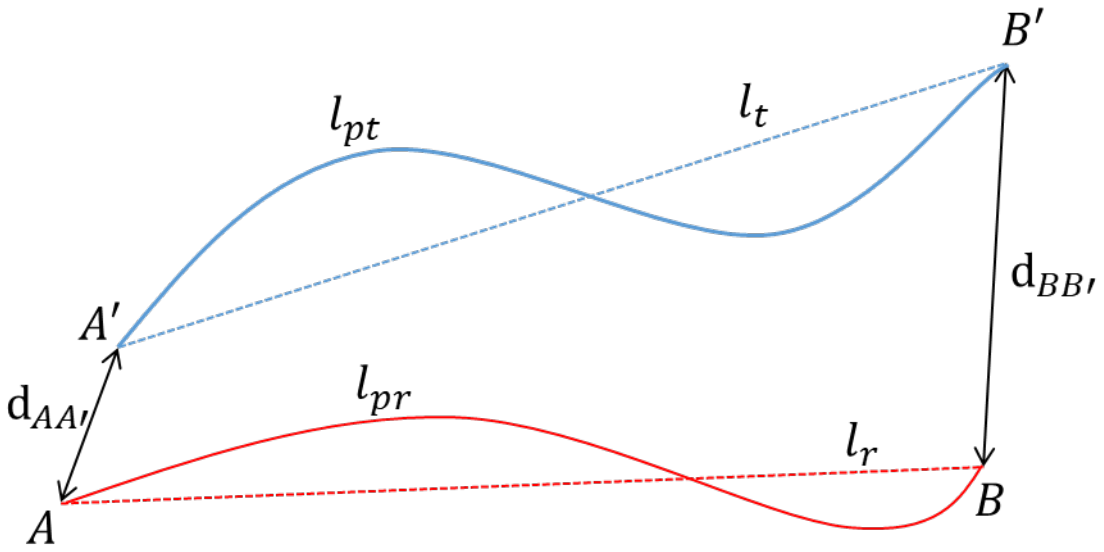


Figure 2-1 Schematic Diagram of corresponding links in two datasets

Normalized Angle score (m_2)*:*

This score (m_2) quantifies how dissimilar the two links in the target and reference network are in terms of their angle. The angle score confirms the selection of candidate nodes in the target network with the best angles to the endpoints in the reference network. The maximum possible angle (θ_{max}) is calculated by equation (2) to ensure that the angle is normalized. Using Figure 2-2,

$$\theta_{max} = \begin{cases} \sin^{-1}\left(\frac{2s}{l_r}\right), \ if \ l_r \geq 2s \\ 180, otherwise \end{cases} \quad (2)$$

Thus, the Angle score (m_2) is the angle ratio between corresponding endpoints between reference and target network to the maximum possible angle between them.

$$m_2 = \frac{\theta}{\theta_{max}} \quad (3)$$

Normalized Euclidean Distance Difference (m_3*):*

One of the essential aspects of similar links in the target network is that the distance between endpoints of each link should be around the same length. Thus, the candidate nodes for the link's endpoints in the reference network are used to calculate the Euclidean distance in the target network. This length is then compared to the original length in the reference network. Figure 2-1 shows the Euclidean distance between nodes A,B and A',B' to be l_r and l_t respectively. The length score is calculated as

$$m_3 = \begin{cases} \frac{2\mathrm{l_t}}{l_r + l_t}, \ if \ l_r \geq l_t \\ \frac{2\mathrm{l_r}}{l_r + l_t}, \ if \ l_t > l_r \end{cases} \quad (4)$$

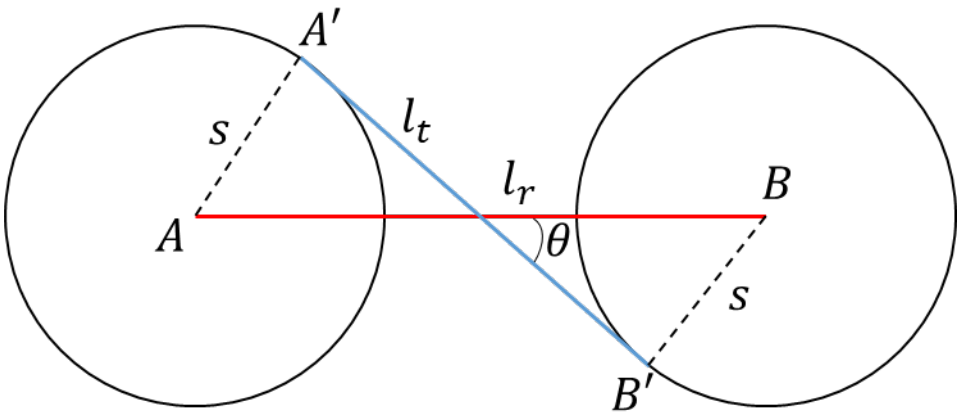


Figure 2-2 Demonstration of angle between corresponding links in two datasets

Normalized Route Length Difference (m_4):

As route length can be used to measure dissimilarities, any two endpoints constituting a route as evaluated by the SPA in the target network are used to calculate the route similarity as path length. Figure 2-1 shows the path distance between nodes A,B and A',B' to be l_{pr} and l_{pt} respectively. The path lengths in reference and target networks are compared as follows.

$$m_4 = \begin{cases} \frac{2\mathrm{l_{pt}}}{l_{pr} + l_{pt}}, \ if \ l_{pr} \geq l_{pt} \\ \frac{2\mathrm{l_{pr}}}{l_{pr} + l_{pt}}, \ if \ l_{pt} > l_{pr} \end{cases} \qquad (5)$$

Normalized Shape score (m_5):

An essential criterion for two links in different networks to be considered matched is to measure the similarity of shape between corresponding links. As illustrated in Figure 2-3, let AB_s and $A'B'_s$ be the shape of the polygon within s distance from the path AB and $A'B'$, respectively. Then, the normalized shape score (m_5) is given by:

$$m_5 = \frac{(AB_s \cap \mathrm{A'B'_s})}{(AB_s \cup \mathrm{A'B'_s}) - (AB_s \cap \mathrm{A'B'_s})} \qquad (6)$$

Objective Function of matching score:

The objective function of assessing the dissimilarities between corresponding links is to minimize the sum of the five abovementioned scores, identifying the "best" links that minimize five different geometric differences. The search for a link that minimizes the objective function ensures a similar path for the target network for each link in the reference network. The objective function is calculated as equation (7)

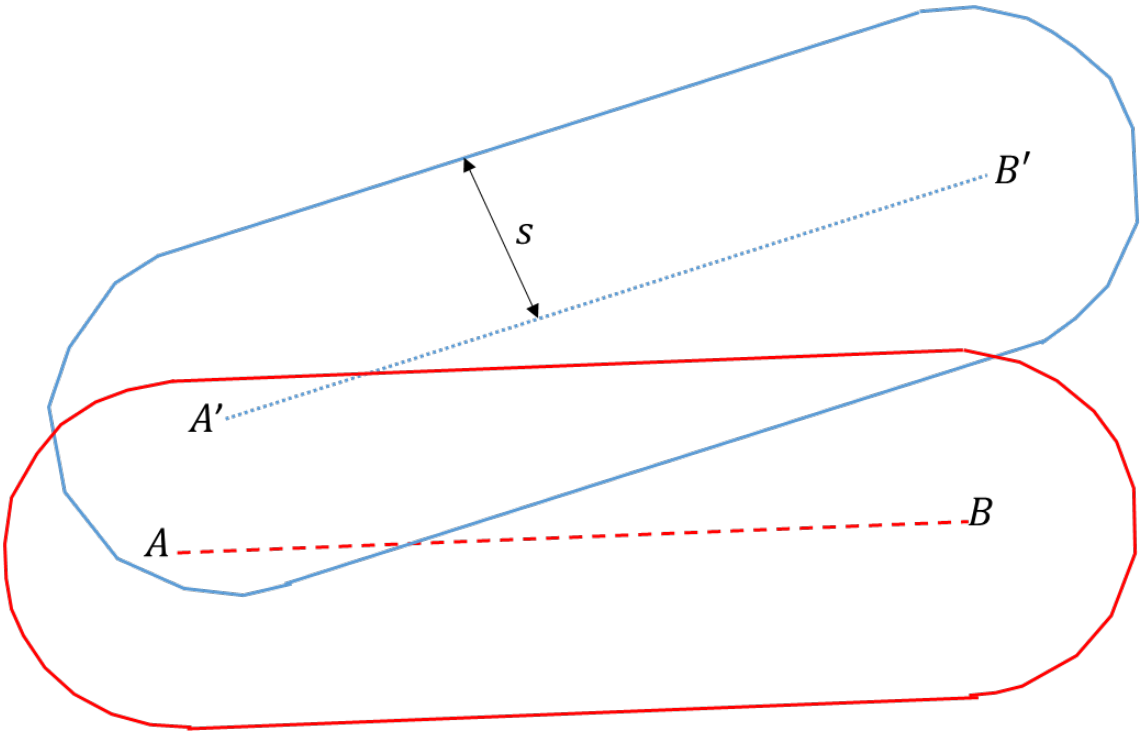


Figure 2-3 Demonstration of Shape Scores (m_5)

$$\psi = \sum_{i=1}^{n} \sum_{j=1}^{5} m_{ji} \quad \forall i \qquad (7)$$

where n is the total number of links in the reference network.

However, when endpoints exist, especially with networks with higher levels of detail, the process entails a time-intensive exhaustive search, which requires efficient search strategies. A need for an efficient search strategy is justified because of the computational burden of calculating the scores of each possible individual route in the target dataset. Thus, GA is implemented to search for the best solution.

GA is a search heuristic that finds the maximum or minimum solutions to a particular function[56]. The optimization problems use techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover, to solve for the "fittest" solution [57]. Like in evolution, GA allows setting both the level of randomization and control [57]. Thus, compared to random search or exhaustive search algorithms, GA is more powerful and efficient with the proper choice of parameters[58]. GA has demonstrated its efficiency in solving optimization problems in GIScience [59-62]. The objective of GA is to maximize or minimize the fitness function[56]. The iterative process of initialization, evaluation, selection, crossover, and mutation are described below.

In the GA structure, the parameters that code the independent variables are called chromosome structures. Every chromosome represents a possible solution to the problem that GA is trying to solve[56]. A GA starts with a randomly selected combination of parameters from chromosomes, which acts as the first generation. Next, in the evaluation step, a fitness value for each chromosome in the generation is calculated. After calculating the fitness function for all chromosomes, the best values (the lowest sum of all weights) are selected for the next generation.

The selected chromosomes from the previous step are known as parents. The increase in the number of parents decreases the convergence rate to the optimal solution while increasing the dissimilarity between child and parent. Therefore, two parents with the best fitness score from the generation were selected. The child chromosomes are produced by recombining parent chromosomes based on a crossover operator, which resembles the chromosomes' biological crossing over and recombination. The operator switches a sequence from the parent chromosomes to create offspring.

The mutation operator randomly flips individual bits in the new chromosome to prevent the algorithm from being stuck in the local optima before finding the global optima. The bit is usually selected based on the low probability known as mutation rate. While the selection and crossover preserve chromosomes' genetic information with high fitness scores, the mutation operator increases the diversity of chromosomes. This prevents the algorithm from converging too quickly and losing any " useful genetic material" [57].

The cycle of selection, crossover, and mutation repeats, storing the best fitness score in each generation. Usually, GAs are iterated until the fitness scores stabilize and do not change for many generations. In this study, the GA stops when the fitness score remains unchanged after 250 iterations.

Therefore, the computational burden of evaluating each possible link between two points could be reduced using GA. The score from equation (7) represents the difference between any two links and could be used for identifying the link with the least differences for each link in the reference dataset. The link with the least difference is the link with the highest fitness score.

2.5 CASE STUDY

A programming script is implemented to execute the methodology and perform the aforementioned operations. The program can read networks in ESRI's shapefile, process

it, and create a node-mapping table and shapefile as output.

Data

For convenience, two publicly available geographical line features that encompass US areas are used for the matching process.

Federal Railroad Administration (FRA) network:

FRA Railroad Network Database (FRARAIL) is the NHPN's railroad equivalent and can be used for analysis or national scale mapping. The FRA Network is a network that is a subset of the 1992 version of the FRARAIL that consists of links aggregated from the FRARAIL [25]. The scale of the FRA varies from $1{:}\ 250{,}000 - 1{:}\ 2{,}000{,}000$. The FRA network with attributes is well tested for Network Flow Problems to predict traffic flows. However, it should also be noted that the FRA network has simplified line features that are not positionally accurate. Due to this, the line features' lengths and shapes may not be reliable, and thus, it is challenging to check the validity and update or modify the network manually.

OSM Rail Network:

The target network, OSM, is a collaborative project map to create freely available positionally accurate and detailed geographic data. OSM has a varying level of positional and geographic accuracy, which depends upon the location of the data, and the scale varies between 1:1,000 and 1:10,000. Since the reference network segments are only intercity Rail, other modes such as motorways, bikeways, and footways are excluded from OSM. Based on the select query on OSM coding conventions, OSM line elements of the rail network are extracted.

Division into Zones

For the experiment to compare the algorithm's performance by region, the entire study area was divided into equal-sized zones. The division of zones was based on the equal-

sized grids in both North-South and East-West directions. However, the network does not cover the entire area in each grid (Figure 1-6). The characteristics of zones are summarized in Table 2-1. The Table shows the variation in the number of links and nodes across zones. For example, Zone 12 has the lowest number of links and nodes, while Zone 16 has the most.

2.6 Calculation of Search Buffer Distance (SBD)

Figure 2-4 visually illustrates the total number of target links within an SBD series of 0.1-mile increments of reference links. Since the target network has a higher level of detail, the increase in SBD predictably increases the number of contained target links, which is followed by a subsequent leveling off by reaching a steady rate. However, for some datasets, the number of links with the increase of buffer distance increases monotonously with a diminishing number of links for higher SBDs. For example, Figure 2-4 shows that the cumulative number of links increases sharply and stabilizes around three miles. In such instances, an SBD based on a sharp increase in the number of links is done. Therefore, an SBD of three miles is confirmed for further analysis.

2.1 Demonstration of minimum scores by m_n

Each score (m_n) represents a difference in a specific spatial property of the network. The sum of scores for each link represents the overall difference between the reference and target networks. The spatial differences in the two maps are demonstrated using Radar Plots (see Figure 2-5). A radar plot is a graphical method of displaying multivariate data in 2 dimensions. To reflect the magnitude of each criterion m_n, vertices are created radially with a minimum value of 0 and a maximum value of 0.6. The area of the shaded region represents the dissimilarity between the two maps.

Figure 2-5 and Figure 2-6 show four different cases of links with different scores in each category. First, Figure 2-5 (a) and Figure 2-6 (a) show significantly different endpoint

Table 2-1 Summary of Zones

Zones	No. of Nodes	No. of Links	Nodes Ratio	Average Link Lengths
3	11	13	85.91	34.55
4	15	21	167.80	30.24
5	9	12	151.00	59.10
6	16	17	103.31	56.39
7	6	8	430.50	41.11
8	17	30	69.65	58.91
9	*98*	*141*	*160.04*	*31.79*
10	*88*	*124*	*99.56*	*29.19*
11	*81*	*108*	*128.25*	*22.13*
12	1	1	608.00	91.35
13	10	16	29.00	72.66
14	*37*	*55*	*111.27*	*44.87*
15	*88*	*128*	*46.72*	*29.01*
16	*118*	*152*	*39.44*	*23.88*
17	*63*	*81*	*61.25*	*22.77*
19	20	32	71.95	52.55
20	*47*	*68*	*58.47*	*39.97*
21	*91*	*136*	*62.32*	*30.86*
22	*113*	*150*	*37.37*	*25.05*
23	13	15	56.54	29.37
25	*33*	*47*	*85.33*	*48.19*
26	*38*	*56*	*91.92*	*41.26*
27	28	39	71.29	39.41
28	*35*	*52*	*71.09*	*35.82*
34	26	31	85.23	33.73

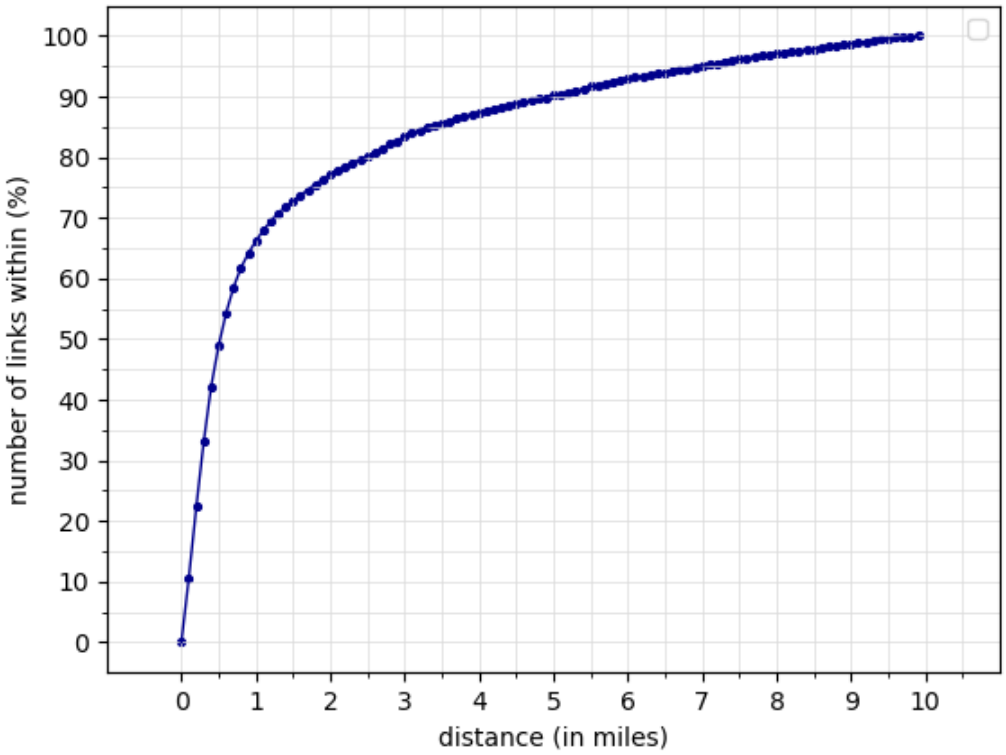


Figure 2-4 Cumulative number of links within the distance (in miles) and calculation of SBD

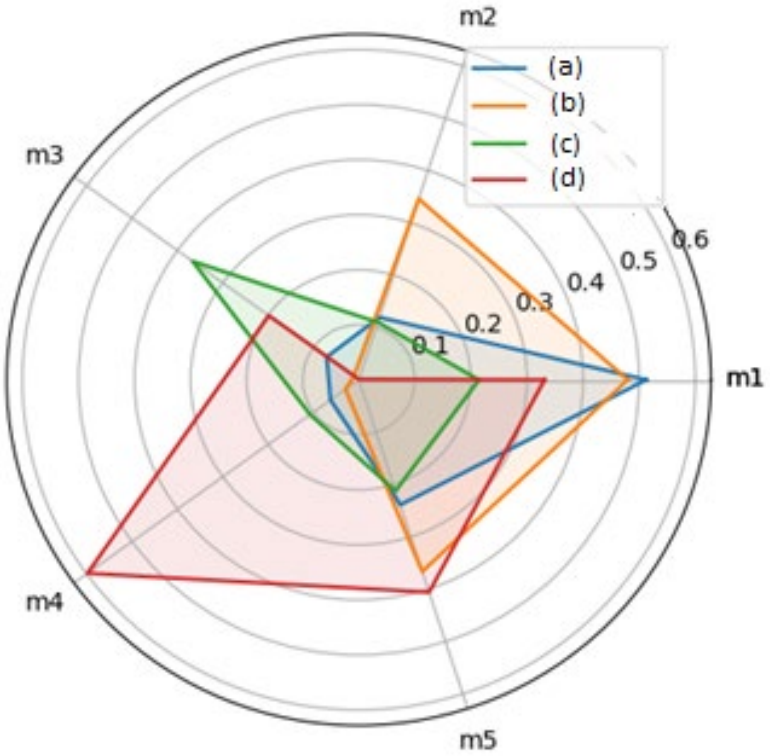


Figure 2-5 Radar plots of links (a), (b), (c), and (d) demonstrating m_n

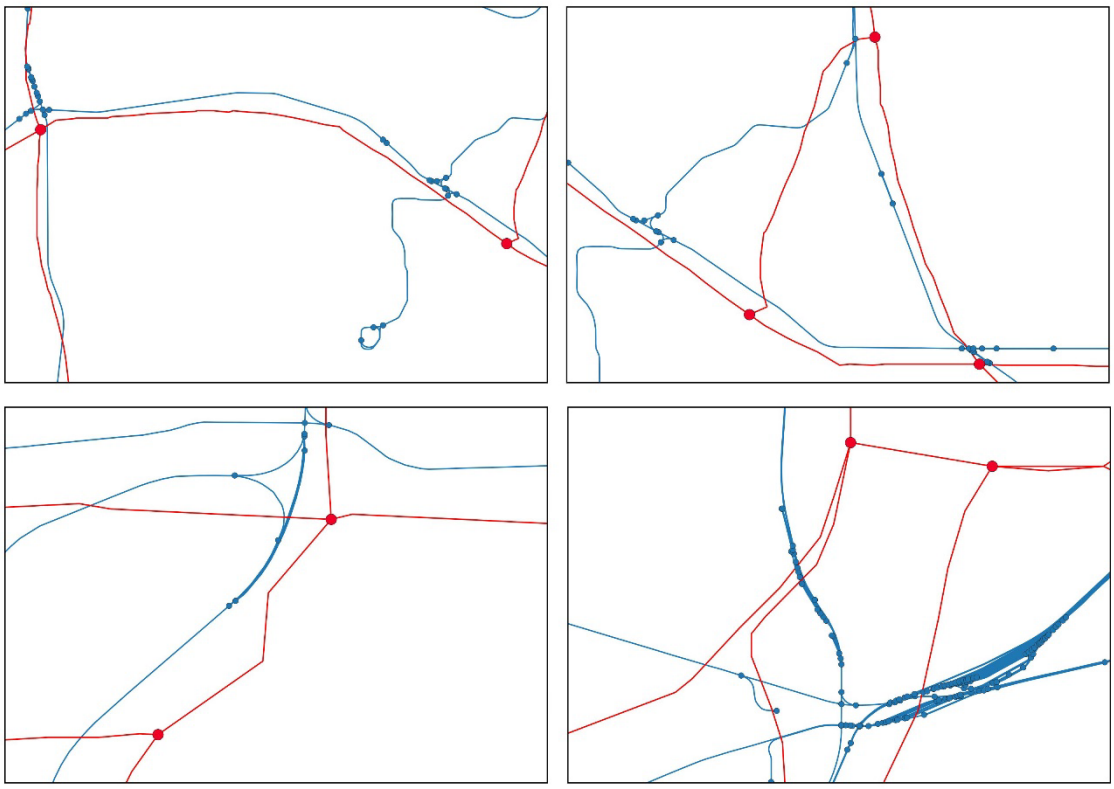


Figure 2-6 Demonstrating Radar Plots plots for links (a), (b), (c), (d)

positions for a link in two networks; thus, the links have a higher m_1 score. The differences in the endpoints might be due to the change in the real-world entity or differences in the accuracy of the two networks. Next, Figure 2-5 (b) and Figure 2-6 (b) demonstrate corresponding links with different angles (m_2) and endpoints (m_1). Similarly, Figure 2-5 (c) and Figure 2-6(c) have a high m_3 score compared to other scores. This is because it has a high difference in distance between endpoints. Finally, Figure 2-5 (d) and Figure 2-6d) show a high value for multiple scores, m_4 and m_5. The figure shows a high difference in the route length and shapes of the corresponding link.

Application of GA

To search for the link for the minimum sum of scores, GA is used. After each epoch (iteration), the averaged SBD for each zone is calculated and stored. The stored SBD is recursively used for every subsequent iteration to calculate the SBD for a link within that zone. Thus, the updating and recalculating of the objective function based on the SBD occurs until the objective function remains unchanged between iterations. In this analysis, a maximum of 30 epochs is used.

All possible links in the target dataset within the SBD are examined for a corresponding link in the reference network. Then the five scores for 5 of those possible links are calculated, and each link's fitness score is calculated. For the crossover step, to retain the information from previous iterations, two distance bands are used. The first is the local distance band, which is 10% more than and 10% less than the SBD of the link with the highest fitness score. The second is the global distance band, which is 10% more than and 10% less than the global SBD within the zone. A choice of a higher distance band would allow the GA for a more exhaustive search but is more computationally intensive. Next, continuing the crossover operation, the start coordinate of the link with the highest fitness score is paired with the end coordinate of the newly searched links within SBD and vice versa. Thus, any reliable solutions are re-used by pairing with another coordinate within a

search distance band. Similarly, a mutation operation is applied with a probability of 1 in50. However, for mutation, five links are randomly chosen within the SBD. The steps above are repeated until the condition that the fitness function is unchanged for 250 iterations is satisfied. Then, the average SBD is calculated as the global SBD for each zone for each epoch. The steps are summarized below.

Input: Target network $G^t = \{N^t, E^t\}$, Reference network $G^r = \{N^r, E^r\}$,

Output: k-core and bridge classification

1. *Set epochs = 30*
2. *for iteration in 1 to epoch:*
3. *for each link $e_i^r \in E^r$:*
4. *initialize links randomly from E^t within SBD*
5. *do unless fitness score is unchanged for 250 iterations*
6. *calculate fitness score*
7. *apply crossover and mutation*

Predicting incorrect matches

The scores for each with the least differences obtained using equation (7) are transformed to obtain scores on each node. Since every node is an endpoint to at least one link, its score is calculated as the mean score of links connecting to it. The node score represents the similarity between two corresponding nodes in reference and target datasets based on the similarity of links connecting its neighbors. Since our node matching algorithms are based on spatial features, especially a node's connectivity to its neighbors, nodes scores can predict an incorrect match for automated algorithms. The node score's ability to predict incorrect matches is assessed using the matches in the first chapter.

Identifying dissimilar links

Similarly, the obtained node scores are used to predict dissimilar links. Identifying dissimilar links can reduce the time and effort made by decision-makers when manually

adjusting links in reference or target networks before applying any automated conflation algorithms. Dissimilar links are usually the result of a topological issue or change in any physical entity with time. The methodology outlined in this chapter can be used to identify these dissimilar links and allows for correction if necessary.

2.2 RESULTS

By calculating dissimilarity scores, the GA exposes each link's dissimilarity in the reference network with the most similar link in the target dataset. First, the scores thus obtained are used to predict the incorrect matches from chapter 1. The nodes with high scores correspond to high apparent differences in connecting links and identified matches and unmatches with more than 76% accuracy. Similarly, in the second part of the results, a threshold for which the links are significantly "different" was chosen. All the links with scores above the threshold were visually checked for any topological issues or absence of links. They could be manually corrected if necessary.

Identification of possible false matches

Figure 2-7 shows the Match Rate by three different methodologies divided into deciles. A decile is a quantitative method of splitting ordered data into ten equal subsections. First, the minimum link scores from GA are converted to node scores. The node score for any node is the mean of the scores of all the links connecting to that node. Then, the dataset is ranked in ascending order of scores, such that each category contains all the nodes with scores within that category.

For example, a decile rank of 1 includes all the nodes with a mean score between 0% to 10%. Similarly, the highest decile rank of 10 has all the nodes with a mean score from 90% to 100%. From the Figure, as moving from left to right, the decile value increases, meaning that the dissimilarity scores of nodes increase and, consequently, the match rates are lower. Thus, from the Figure, it can be concluded that higher minimum dissimilarity

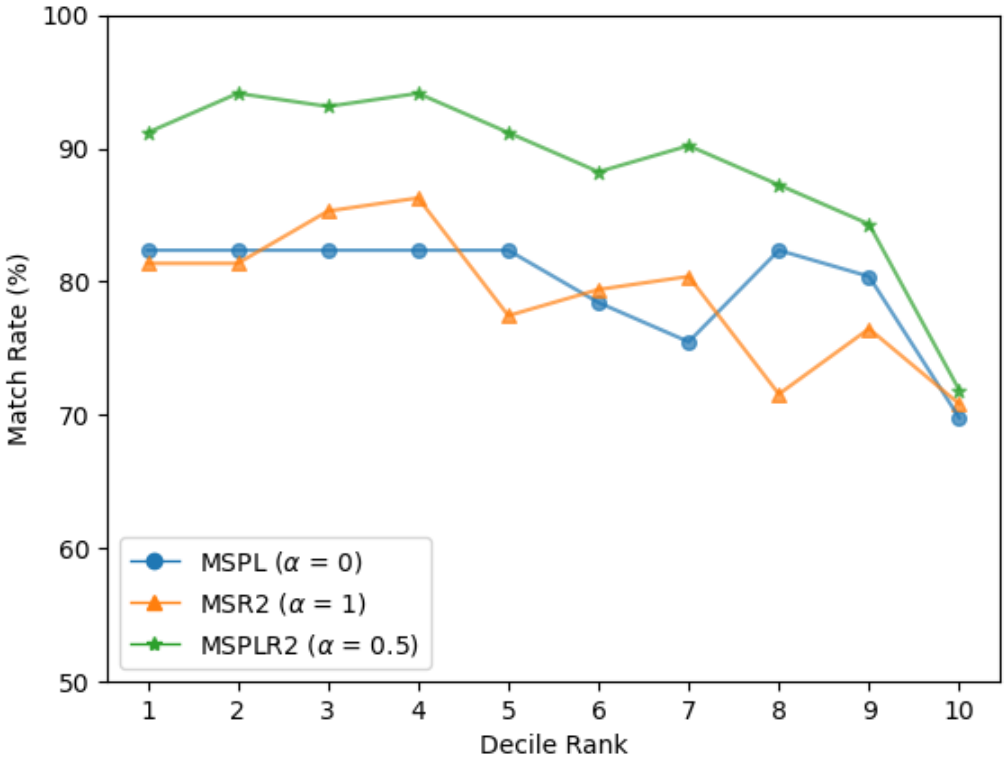


Figure 2-7 Match Rate by decile rank

scores correspond to lower match rates for all three procedures. Figure 2-7 shows that MSPL decreases with a slope of -0.91 and *R-squared* of 0.65. Similarly, MSR2 decreases with a slope of -1.29 and *R-squared* of 0.76. Finally, MSPLR2 decreases with a slope of -1.73 and *R-squared* of 0.79. Thus, the match rates from all three procedures decrease increase in the decile rank. This indicates that the scores can reasonably predict the match rates from all three procedures. Specifically, for MSPLR2, the change in the match rate is most significant as it decreases by an average of 1.73% with each increase in the decile rank.

Identification of dissimilar links

The minimum combined scores from GA can simply be used to assess the dissimilar links between two datasets. The dissimilar links can be separated and manually inspected for any necessary corrections. There might be several reasons for links to be dissimilar in two datasets. First, since the two datasets are usually non-contemporary, construction or abandonment of tracks could change the network's connectivity. Next, any crowdsourced dataset could have areas where the volunteers might not have enough knowledge or might not be interested in mapping. This would result in incorrect mapping in the target dataset. Next, any slight diversion of traffic bypassing towns could change the curvatures or connectivity of links.

The minimum combined score for each link quantifies how different is the least dissimilar corresponding link compared to the reference link. For example, each link in the reference dataset could have several possible links in the target dataset. However, the one with the minimum score has the least dissimilarities to that link.

All the links in the target dataset with the minimum scores have a distribution, with links very similar to the reference dataset to different links. A range of thresholds can be used to separate these "similar" and "different" links. However, a lower threshold value would discard many links, and a higher threshold value would include links with higher

differences.

The reference and target networks' links with 4 kinds of topological differences were identified using the threshold, as shown in Figure 2-8. First, Figure 2-8(a) shows links in the target network having an Undershooting topology. This disconnection from undershooting can range from a few feet to several miles. If the undershooting topology is higher, it can be detected manually; however, this is difficult for smaller undershooting topology. Second, Figure 2-8(b) shows that links in the target network have an overshooting topology. In this case, there is no connection in the point of contact of the two lines because the two points are very close to each other. These cases could go easily undetected by the human eye. Third, Figure 2-8(c) shows that the target network has missing connectivity or corresponding nodes. Thus, the presence of similar nodes does not warrant a match as they are not connected to the rest of the network in topologically similar ways. Finally, Figure 2-8(d) shows that the reference and target networks are significantly dissimilar. This could be because the positional accuracy of the reference network is too low, or there is a change in the physical characteristics of the links.

2.3 DISCUSSION

The proposed methodology demonstrated the detection of dissimilar links by minimizing the differences. However, the methodology compares each link in the reference dataset with possible links in the target dataset. Thus, it does not directly identify links in the target network with high apparent differences from the reference network. This is because it is assumed that the target dataset has a higher level of detail in all areas. Despite correctly identifying links in the target dataset with high apparent differences for each link in the reference dataset, there was room for improvement in detection.

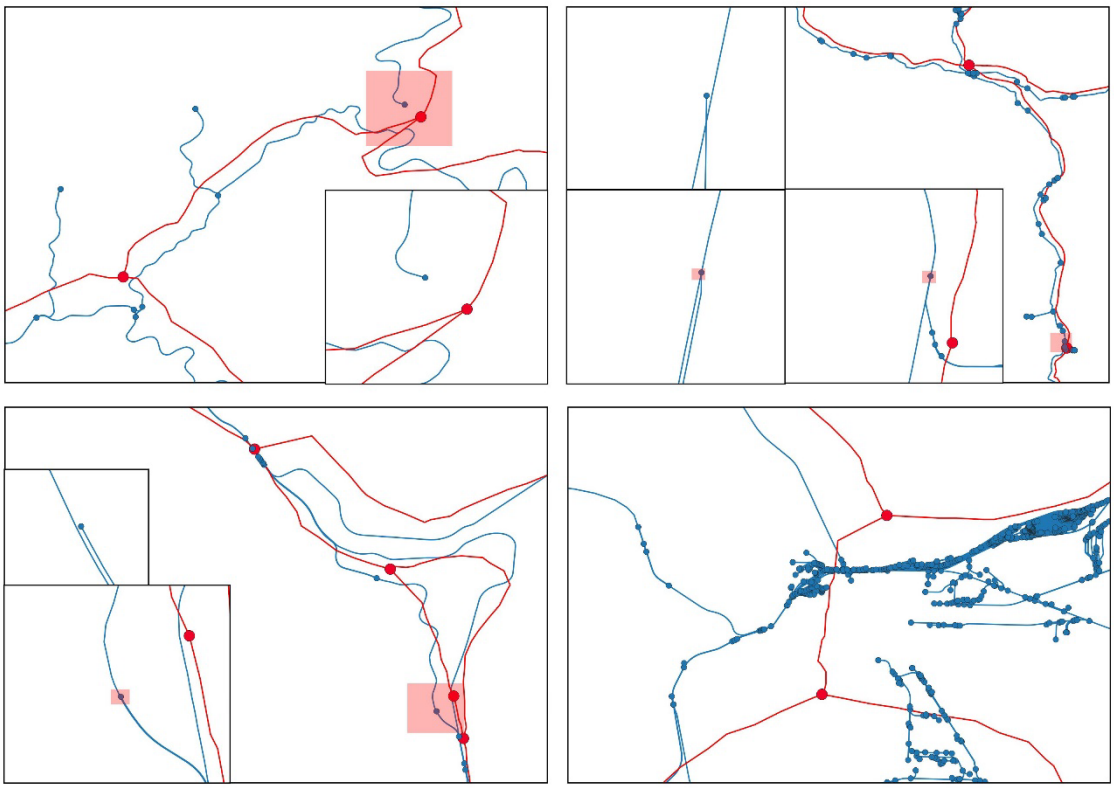


Figure 2-8 Cases of mismatching links with high minimum sum of scores (a) Undershooting Topology (b) Overshooting Topology (c) Missing Connectivity (d) Highly dissimilar links

First, using the shortest path algorithm to identify the links with the least dissimilarities in the target network is problematic when multiple parallel links connect the neighbor node. SPA can wrongly evaluate the links scores, choosing the incorrect routes in the target dataset in those cases. This could be improved by identifying the route within a variable SBD. The SBD would go from low to high unless a link with a difference less than a threshold is found. Next, the weights of the scores are assumed to be equal in this chapter. Even though each of the individual scores lies between 0 and 1, the value of the individual score might be differently contributing to the actual difference. A multivariate regression model can be used to see what weights can explain the variation of differences to address this. Finally, the node scores are assumed to be the mean scores of the connecting links. This does not take into consideration the length of the connecting links. A more scientific way of incorporating the connecting links' length could be explored.

CHAPTER 3
USING NEURAL NETWORKS IN GEOSPATIAL DATASETS TO IMPROVE POSITIONAL ACCURACY

3.1 ABSTRACT

GIS has been a valuable tool for capturing, storing, manipulating, and analyzing spatial data. However, due to the rapid advancements in location-based technologies, several datasets have many viewpoints of the same real-world entities. This study aims to develop a framework to locate corresponding elements in two datasets using an artificial neural network. The historical dataset, also known as the reference dataset, is less positionally accurate and has a lower level of detail, while the target dataset is a positionally accurate dataset with a higher level of detail. Any geometric criteria of the network that could potentially influence the identification of corresponding elements are extracted and used as an input in the neural network. Two node classification algorithms, k-core decomposition, and graph bridge decomposition were introduced to classify the nodes in the target dataset. A ground truth dataset is used to train the neural network, and a back-propagation neural network is employed. Unlike traditional optimization problems, the neural network determines the weight of each measure through learning. Three different strategies that demonstrate different matching scenarios were presented. The verification results show a satisfactory agreement between the predicted and actual values.

Keywords: conflation, neural network, levels of detail, positional accuracy, node-based

3.2 INTRODUCTION

The availability of multiple techniques to integrate and synthesize the growing size of geospatial datasets from different sources is well established. For example, among many sources, crowdsourced datasets such as VGI [63] have been popular in recent years. In addition, many independent spatial data providers result in a separate representation of the same entity. Thus, manual matching has become impractical due to the increasing number of these datasets.

Spatial data matching is the process of integrating different data sources with a certain degree of similarity in a systematic way[64]. The matching process includes attributes, proximity, similarity, or dissimilarity metrics to find corresponding elements. However, the lack of attribute information in datasets results in lower match rates using available methods. This study explores more spatial features and proposes using node classifications such as k-core decomposition and graph bridge decomposition to find corresponding features.

The k-core decomposition method is a powerful network classification tool that measures nodes' importance and connectedness. Recently, the k-core decomposition method has been widely used in application domains such as in-network visualization [65], protein function prediction [66], and graph clustering [67] and has established itself as a standard tool in node classification. By implementing k-core, any node’s importance in the network can be accessed based on its connection to the network.

Similarly, bridge identification is a concept for graph decomposition for physical railway/traffic networks in areas with networks' relatively low connectivity/degeneracy. A bridge of a network refers to the links in a network whose removal can split one network into two subnetworks. In other words, any traveler on a network must use the bridge to travel from one subnetwork to another subnetwork. A bridge usually divides

any network into "looped" and dead-end components. The "looped" component forms the vast majority of the network for which there exist at least two paths to travel from any two nodes within the network. In contrast, the dead-end components are separate from the “looped” network and are connected to it through a bridge.

The entire network can be decomposed to a pruned major network or “looped” network connected to many smaller networks through bridges by implementing bridge decomposition. The smaller networks refer to the isolated stations, destinations, and terminal facilities in the major railway, while the major network represents the main lines.

An Artificial Neural Network, inspired by the structure and function of the biological neural network, is a computational mechanism that can acquire, represent, and compute a mapping from one multivariate information space to another[68]. A neural network learns from training input and output to create a model that correctly maps any unseen input to output. The back-propagation training algorithm maps the input to output by determining the weight of each factor through back-propagation, as described below.

The Back-Propagation Neural Network (BPNN) comprises a forward and a backward phase. In the forward phase, the values of each node are calculated using the current weights. Next, the computed output node values are compared with the actual values in the second phase. The difference between the computed and actual values are treated as errors. Finally, the weights in the previous layers are adjusted to minimize the errors. This represents one epoch in the back-propagation cycle. This process continues iteratively until the total error in the system is within a pre-specified level. The weights thus calculated are tested through classification data that the neural network has not seen before.

This paper proposes a BPNN to identify corresponding elements in two networks (target and reference) with significant similarities in position, length, or shape of spatial features.

Information such as connectivity, distance to neighbors, the difference in lengths to neighbor nodes, and graph decomposition metrics are extracted and fed to the neural network as input with ground truth values as outcomes. The neural network learns the effect of each predictor via backward propagation to automatically evaluate the effects for each predictor. Thus, effectively, the neural network predicts the probability of each node being a corresponding node in the reference network. The performance of the proposed algorithm is assessed with a case study with two GIS datasets with different levels of detail. Finally, the results and findings to draw future research direction are discussed.

3.2 LITERATURE REVIEW

The idea of conflation was introduced around the mid-1930s, but research work was scarce for many years afterward. However, after the late 1980s, conflation became mainstream due to the increased data availability from multiple sources. Since then, the importance of having accurate and comprehensive digital spatial data sets via conflation has been underscored repeatedly by industry and academia [13] [14] [15] [16]. The research on conflation primarily focuses on improving existing maps with information available in different forms in different sources.

Traditional conflation methods incorporate a simple buffer distance [21] that quantifies the similarity of two links based on the percentage of the buffered area of one feature that falls onto another. For example, K-closest pairs queries (KCPQ) search for counterpart feature pairs most proximate to each other in two different networks[22]. Similarly, path-based conflation procedures[20] first identify candidate nodes based on proximity and keep or discard them based on the "round-trip walks" procedure. However, these algorithms incorporate only proximity and "round trip distance" that do not entirely obtain the spatial context of the network.

The optimization-based approach introduced by Li, L. et al. [11] and Lei, T. et al. [69] is a significant departure from traditional procedure-based conflation methods is. It employs an assignment problem using similarity or discrepancy of features and can automate conflation without manual inputs (for anchor points). Also, external knowledge is easily incorporated if a human expert is available, adding constraints. As a result, the changed model generates constraint-based optimal results compatible with the input. However, the approaches do not consider the topological aspects of the network, which could help match networks with high positional differences.

Wang [70] employed a neural network to determine the weight of each similarity measure using probability-based feature matching methods. However, he does not incorporate the spatial context and topological relations in the feature matching process. Similarly, [62] [71] uses the similarity of shapes and spatial distances to match linear objects. However, they do not mention the situations where the positional accuracy of datasets is significantly different.

Similarly, out of all the studies dedicated to matching crowdsourced datasets (such as VGI) with historical datasets, little attention has been paid to the area of matching datasets with only spatial attributes. This paper proposes a learning-based method that incorporates several geometric criteria such as connectivity, proximity, and graph decomposition for the feature matching process. Thus, the proposed neural network-based learning accesses the spatial context, characteristics of topology, and node classification to identify corresponding features.

3.3 METHODOLOGY

Concepts and definitions

This paper prescribes different elements of reference and target networks to be considered for inputs for the neural network. MSPL and MSR2 are two different

procedures to capture various aspects of the pair of networks. However, the maximum distance to which the corresponding node is searched has to be set beforehand. The choice of SBD is based on interpreting the results from MSPL and MSR2 for different search distances. First, nodes were randomly chosen from the Reference Network, and MSPL and MSR2 were evaluated for each node. Next, both procedures calculate the anchor points for each buffer distance. Finally, the least buffer distance such that any increment in buffer distance would not change the predictions from both the procedures is chosen. This buffer distance is fixed for all the subsequent analyses.

Procedure 1: The Minimum Sum of Path Lengths to Neighbors $(MSPL)$*:*

MSPL assumes that the anchor point is the candidate node such that the minimum path distance has to be traversed to reach all its neighbor nodes. Thus, the objective function (γ_1) of MSPL is the minimization of the path lengths to the neighbor nodes in the target network. Since the procedure only considers the path lengths in the target network, any inaccuracy in the length of edges in the reference network would not influence MSPL. Mathematically, the anchor point (a_1^i) is expressed as equation (1).

$$a_1^i = \min\left(\gamma_1^i\right) \qquad (1)$$

Let $d(X, Y)$ be the path lengths between known points *X* and *Y*. Then, the sum of path lengths to neighbor nodes (b_{ij}^{t}) for candidate nodes (V_i) is calculated as follows:

$$\gamma_1^i = \sum_{1}^{m} d(V_i, b_{ij}^{\mathrm{t}}) \quad \forall j \qquad (2)$$

Procedure 2: Residual Sum of Squares of Path Lengths to Neighbors ($MSR2$):

MSR2 assumes that the shortest path's length to neighbors from a reference node is the same for both reference and target networks. Thus, the anchor point is computed as a minimum squared sum of differences in corresponding lengths in two networks. Mathematically, the a_2^i is expressed as equation (3).

$$a_2^i = \min\left(\gamma_2^i\right) \tag{3}$$

The γ_2^i formulated as follows:

$$\gamma_2^i = \sqrt{\sum_{j=1}^{m}\left(d\left(V_i, b_{ij}^t\right) - d\left(n_i^r, b_{ij}^r\right)\right)^2} \quad \forall \text{ j} \tag{4}$$

Degree of a Node ($deg(n_i^r), deg(n_i^t)$):

The degree of a node, $deg(n_i)$ represents the number of links connecting to n_i.

K-core decomposition:

Let G be a graph and G' a subgraph of G with a set of nodes N. Then, G' is the k-core of G, if all the nodes have a degree of at least k. Each k-core is a unique subgraph of G, and it is not necessarily connected. The subgraph G' contains a vertex of degree at most k. The k-core sub-graphs are identified by repeatedly deleting all vertices of degrees less than k.

Bridge Decomposition:

Applying k-core decomposition alone would not be able to extract the information about the importance of the nodes within the network. A bridge is a link whose removal can split one connected graph into two subgraphs in graph theory. The two ending nodes of the bridge are called banks. On observation of any network, the network can be viewed as

two components: the major "looped network" and "tree structures" attached to the major network. Specifically, the "looped network" forms the majority of traveling routes, while the tree structure denotes the isolated stations, dead-end tracks, or classification yards. A "tree structure" is always connected to the "looped network" using a single edge, known as the bridge. In this way, the major "looped network" can be extracted from the entire network by identifying the bridges.

Figure 3-1 shows a schematic network diagram with a main "looped network" and "tree structures" separated by bridges. However, only the bridges with at least one connecting node with a k-core greater than 1 separate the "looped network" and the "tree structures" (shown as a double arrowed line in Figure 3-1). Such bridges connect the "tree structures" to the main "looped network" through a "root node". Thus, identifying the "root node" can meaningfully decompose the entire network into subgraphs and further aid in the node classification and, consequently, node matching algorithms.

Let $f(G)$ and $g(G)$ denote the k-core algorithm and bridge identification algorithm, respectively. The pseudocode of the new algorithm using the k-core analysis algorithm and bridge identification algorithm is given below.

Input: Target network $G^t = \{N^t, E^t\}$.
Output: k-core and bridge classification

1. *Identify k-core (k_i) for each node $n_i^t \in N^t$ in G using $f(G^t)$*
2. *Identify all bridge links $e_j^t \in E^t$ using algorithm $g(G^t)$.*
3. *For each e_j^t, identify its ending nodes (n_{j1}, n_{j2}) connecting to bridge e_j^t.*
4. *For (n_{j1}, n_{j2}), check their k-cores (k_{j1}, k_{j2}).*
5. *If at least one k-cores are greater than 1, then the bridge link meaningfully decomposes the network.*

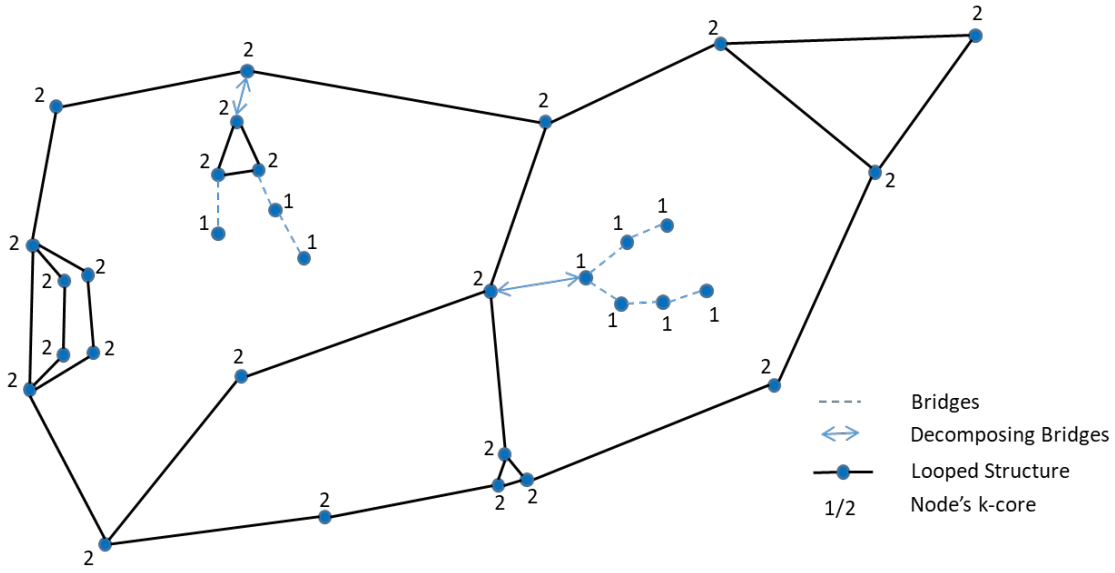


Figure 3-1 Demonstration of K-core Decomposition and Bridge Decomposition

Notice that since both the k-core and bridge identification algorithms have a linear time complexity to nodes and links (i.e., $O(V + E)$), the proposed algorithm also solves the whole task in linear time.

Link Match Scores:

Link Match Scores quantify the similarity of a link in the reference dataset with a link in the target dataset that has the least differences. The link is identified by minimizing the sum of 5 different criteria explained in detail in section 2.4.

DBSCAN Clustering:

One of the most widely applied point clustering techniques is the DBSCAN algorithm. The DBSCAN algorithm requires two parameters, the minimum number of points ($minPts$) and the searching radius (Eps). The points are classified into one cluster if at least $minPts$ neighboring nodes for any point within the cluster. A core point is a point if it has more than $minPts$ within Eps. Next, a border point has fewer than $MinPts$ within Eps but is in the neighborhood of a core point. Similarly, a noise point is a point that is not a part of the core point or border point. Thus, if any two core points are within Eps, they are put in the same cluster.

There are several advantages of using DBSCAN clustering. First, the total number of clusters is not a parameter for clustering. Next, it can form arbitrary-shaped clusters. Next, it only requires two parameters and is primarily insensitive to ordering the points in the database. Similarly, DBSCAN identifies outliers as noise, unlike other algorithms that force potential outlier points to the same cluster despite having different characteristics. Finally, for clustering, since the network distance is more representative of the proximity of nodes in the network, for this work, Network distance instead of Euclidean distance is used.

However, DBSCAN clustering requires Eps and $minPts$ to be specified before

clustering, and there's no systematic way of identifying that. For example, in choosing the value of Eps, for a smaller value, a large part of the data will not be clustered. But, for a high value, most of the data will be in the same cluster. Similarly, since the size of the cluster is unknown in a network, there could be the presence of arbitrary-sized clusters. Thus, sometimes the clusters grow very large in areas such as classification yards, and in several of those instances, the clusters include the actual anchor point. This results in mapping one node in training with many nodes in the target dataset. This skews the predictions of nodes in the target dataset towards being an anchor point.

Data preprocessing

First, an appropriate coordinate system is chosen depending on the location and the map's extent. Then, both reference and target datasets are projected to the same coordinate system to reduce possible geometric errors. This ensures that both maps are on the same surface with the same geographical projection.

Since corresponding elements in datasets with different levels of details are being identified, it is reasonable to assume that the classification of the dataset is imbalanced. This is because there would be fewer matches than mismatches in the target dataset for each node in the reference dataset. Thus, three different methods were tested to overcome the imbalanced classification, and the best one that works for the datasets was chosen.

First, Random Oversampling is a method where the minority sample data is drawn repeatedly with replacement unless the classification is balanced. Second, Synthetic Minority Oversampling Technique (SMOTE) selects cases close in the feature space and linearly interpolates the feature space. Thus, a new "synthetic" dataset is created in the process that did not exist in the original dataset. However, for any classification problem, it is crucial to focus on areas that are boundaries to a different classification to classify the dataset correctly. So thirdly, Borderline SMOTE is tested. In addition to linearly interpolating in the feature space, this method chooses samples close to areas with

borderline classification.

Backward Propagation Neural Network (BPNN)

The mapping function of the neural network $u = G(x)$ is input with a training set where the network learns to associate input patterns x to output patterns correctly U. As shown in Figure 3-2, the inputs fed into the model through the input layer are multiplied by interconnection weights and summed up as the data moves from the input to the output layer through hidden layers.

There are infinite ways to structure the network for a given dataset that learns to map the input to output correctly. However, the choice of the number of layers and neurons affects the complexity and learning time of the neural network. Any network with two hidden layers can approximate any arbitrary non-linear function and generate any complex decision for any classification problem. Therefore, trial and error were used to correctly determine the network structure based on classifying the test dataset.

The neural network used in this study consisted of 4 layers. The first layer is the input layer, where the number of nodes equals the number of independent variables. The second and third layers are the internal or "hidden" layers, and the fourth is the output layer. Each node in the hidden layer was interconnected to nodes in the preceding and following layers by weighted connections. Similarly, each neuron transforms many of the input connections and a single output. The activation function in a neural network defines how the weighted sum of the input is transformed into the next hidden layer or output. Therefore, the choice of activation function in any layer influences the capability and performance of a neural network. The choice of activation function is described below.

Figure 3-3 shows the activation functions used in this study. First, a linear activation function is a linear function that is directly proportional to the input. The increase in the input results in the increase of the output. Second, Rectified Linear Unit (ReLU) is an

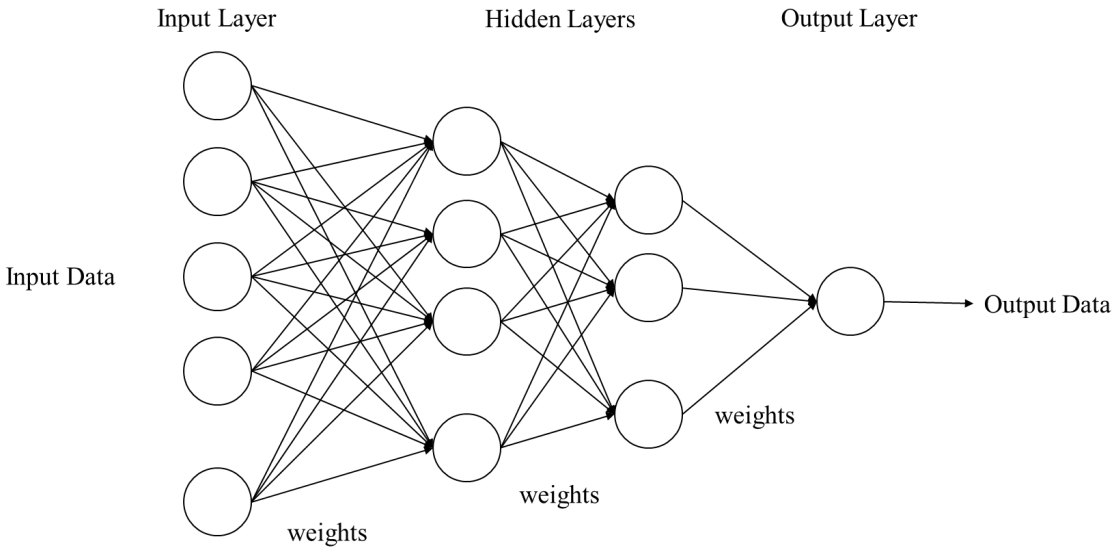


Figure 3-2 Schematic Diagram of Back-propagation Neural Network

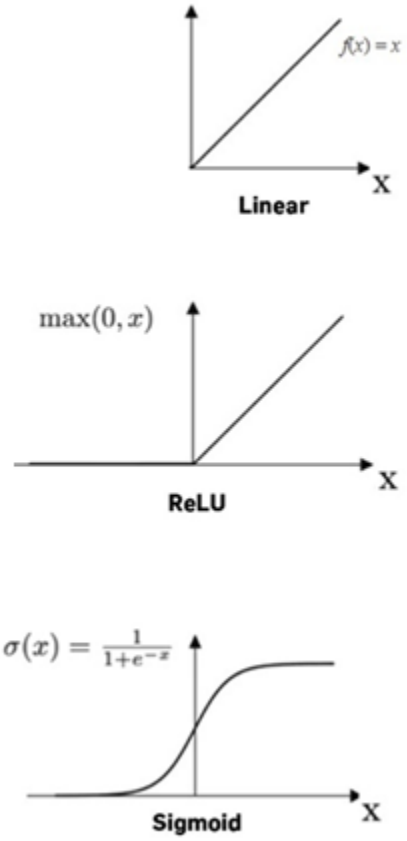


Figure 3-3 Visualization of Linear, ReLU, and Sigmoid Activation Functions

activation function in which only the inputs with a positive value are activated, so it filters information that propagates forward through the network. In other words, for any input of a value less than 0, the output is 0, while anything other is passed as it is. ReLU is a general activation function and is used in most cases because of its computational simplicity. Finally, the output layer is a sigmoid activation function, a non-linear activation function that transforms any value to a range 0 to 1.

The back-propagation algorithm is applied to calculate the weights between the input and the hidden layers, and between the hidden layers and the output layers, by modifying the number of hidden layers and learning rate.

The objective of a neural network is to minimize the loss function. The sigmoid function in the output layer transforms the values from the previous layer between [0, 1]. However, the actual values for classification are 1 or 0. So, to overcome this, a binary cross-entropy loss function is implemented. Entropy is the level of uncertainty inherent in the variable's possible outcome. The greater the entropy, the higher the uncertainty with the distribution. The binary cross-entropy is the average cross-entropy across all data samples.

3.4 CASE STUDY

The values of objective functions and the graph decompositions are fed into a neural network and are trained using the ground truth dataset. The trained model is tested through unseen data to calculate the methodology’s performance.

Data

This study tests the performance of the proposed algorithm with independent GIS datasets: a Federal Railroad Administration (FRA) network as a reference network and an OSM as a target network. Both networks cover approximately the eastern half of the US. It is important to note that the FRA network has a lower level of detail than the OSM

Network. The two datasets are described below.

Federal Railroad Administration (FRA) network:

The FRA Network is a historical network, a subset of the 1992 version of the FRA Railroad Network Database (FRARAIL). FRARAIL is the National Highway Planning Network's (NHPM) railroad equivalent and can be used for analysis or national scale mapping. Figure 1-2 shows the spatial extent of the FRA network, which is used for analysis. The scale of the FRA varies from $1{:}\ 250{,}000 - 1{:}\ 2{,}000{,}000$. It should be noted that the FRA network has simplified line features that are not positionally accurate. Due to this, the line features' lengths and shapes may not be reliable, and thus, it is quite difficult to check the validity and update or modify the network manually.

OSM Rail Network:

The target dataset, OSM, is a collaborative project map to create freely available positionally accurate and detailed geographic data. OSM has a varying level of positional and geographic accuracy, which depends upon the location of the data, and the scale varies between 1:1,000 and 1:10,000.

Figure 1-3 shows the differences between OSM and FRA Network. First, the OSM Rail Network has more links and nodes, confirming the higher level of detail of the OSM Rail Network than the FRA network. Second, the average length of links in the OSM Rail Network is far smaller than in the FRA network, indicating a one-to-many relationship between FRA and OSM Rail Network.

Table 3-1 shows the descriptive statistics of the dataset. A few things worth noting are that the mean and standard deviation of MSPL is 91.865 and 70.304, respectively, indicating that the dataset has a low spread. On the other hand, for the same sets of links, the mean and standard deviation of MSR2 are 4.92 and 14.626, respectively, showing higher spread-out values. The degree of node of both reference and target dataset is

Table 3-1 Descriptive Statistics of input variables (N = 29,355)

Variable	N	Mean	Std. Dev	Min	Max
MSPL (γ_1)	29,355	91.865	70.303	0.192	488.066
MSR2 (γ_2)	29,355	4.924	14.626	0.0008	312.284
MSPL (Z_1)	29,355	0	1	-5.323	7.690
MSR2 (Z_2)	29,355	0	1	-4.873	7.559
Degree of Node (reference)	29,355	3.004	1.061	1	7
Degree of Node (target)	29,355	2.956	0.467	1	6
No. of target Nodes within SBD	29,355	52.099	26.063	1	
k-core degeneracy	29,355	1.969	0.173	1	2
Bridge decomposition	29,355	0.969	0.173	0	1

similar; however, the number of target nodes is almost 25 times more than the reference nodes.

Evaluation Criterion

It is interesting to note that matching networks with different levels of detail entails a philosophical problem of whether the node from two networks refers to the same entity. For example, a node FRA network could refer to a whole terminal or town, whereas a node in the OSM network is detailed enough to refer to a physical track switch. This leads to a representative gap between the nodes of the two networks. Thus, three different matching schemes are proposed. The reason to prescribe the schemes is that real-life situations rarely fall entirely under a single scheme. Therefore, evaluating the performance of the matching algorithm under different schemes gives insights into how the algorithm performs under different conditions.

Scheme 1. One to one node matching

Scheme 1 assumes that one node in the reference dataset is represented in the target dataset as precisely one node. In other words, one node in the target dataset can describe the physical facility, station, or town. Moreover, among all the possible nodes in the target dataset, it is assumed that there should be one node with more representative power than any other node. This scenario is usually true in rural areas where one node in the target dataset can represent one node in the reference dataset. Therefore, the matching in this scheme has a node-to-node bijective relation.

Scheme 2. One to many node matching

Scheme 2 assumes that several nodes might have similar representative power to correctly represent a node in the reference network. This is usually true in urban areas where the reference dataset lacks enough information to correctly identify one best node representation in the target dataset. However, due to the lack of enough information

extracted only through spatial attributes, it is not easy to identify all the possible nodes in the target dataset with similar representative power.

Scheme 3. Node to cluster matching

Similarly, scheme 3 assumes that the most representative nodes in the target dataset usually form a cluster, and any node in the cluster can represent the reference node equally. A node to cluster matching has more possible matches per node than one to many node matching. This is usually true in rural and urban areas, with exceptions where two nodes are very close to each other in the reference dataset. Thus, the matching problem is translated to a node to cluster matching problem.

Before matching, it is crucial to identify the clusters to represent stations or other units (such as intersections or terminals) in the railway system. Therefore, the DBSCAN clustering method is implemented in this study for simplicity and convenience.

3.5 RESULTS

The three schemes were tested to identify corresponding elements in the OSM Network and FRA Network. A neural network was trained with the zone with the highest number of nodes and tested on the remaining zones. Zone 16 has 118 nodes and 152 links.

Scheme 1. One to one matching

Assuming each node in the historical network could be represented by a single node in the target dataset, a neural network was implemented. The probability of each node being a reference node was obtained. However, only the highest probability nodes were considered a match. The results indicate an overall match rate of 89.42%. In addition, the match rate was greater than MSPLR2 for each zone and thus greater than MSPL and MSR2. Figure 3-4 shows the relationship between average link scores in a particular zone and the match rate. As anyone would expect, the plot shows a negative relationship

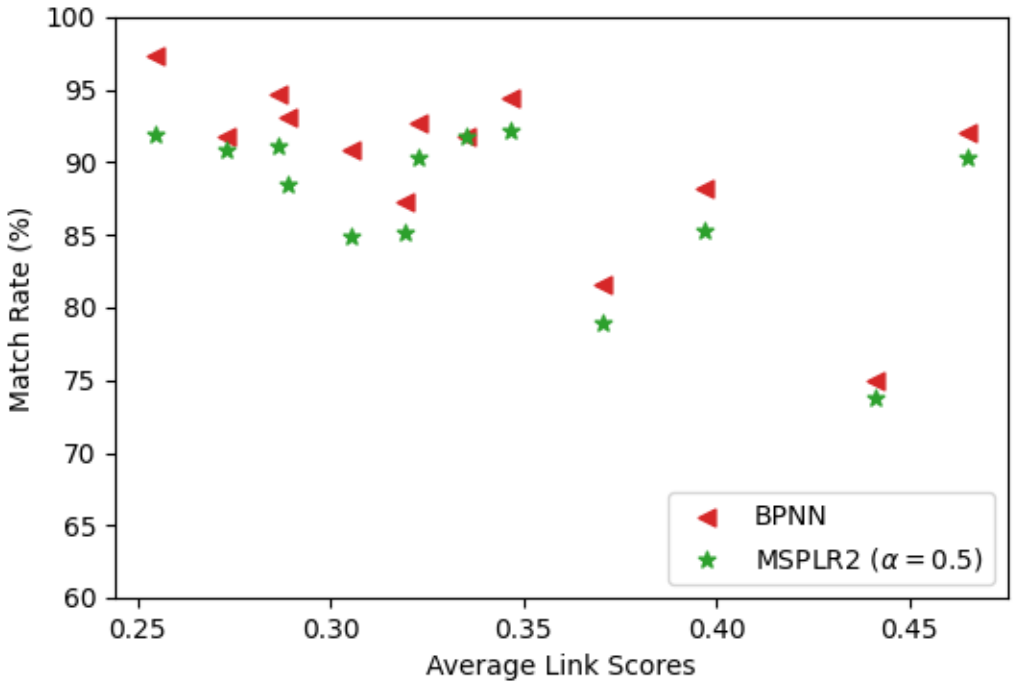


Figure 3-4 Match Rates by MSPLR2 and *BPNN*

between average link scores and match rates with an *R-squared* of 0.65. This implies that the difference in the shape of the corresponding links connecting a node and its neighbor is an indicator of the match rate.

Scheme 2. One to many node matching

The same neural network as scheme 1 was used, but a match was considered when the probability of matching was greater than 0.5. Assuming that several nodes could represent one node in the historical dataset equally, the match rates for all zones are shown in Figure 3-5. The match rates is much lower than one to many matching. Further, there is no correlation between match rate and link scores, suggesting that the similarity of connecting links in reference and target dataset is not a predictor for the number of possible matches and their likeliness of representing the same entity.

Some of the possible reasons behind the lower match rates are as follows. First, when the neural network is tested, it sees the dataset that it has never seen before. The presence of certain peculiarities for any tested zone that the neural network may have never seen in the training dataset decreases the match rate. Similarly, the predictors used in the neural network might still be insufficient and thus cannot predict all the possible nodes in the target dataset. Therefore, more predictors could be added to categorize the unexplained mismatches correctly. Next, for some situations, the detection of all possible matching nodes in the target dataset, the information from spatial elements alone might not be enough, or the unique feature and purpose of the node placement in the reference dataset has to be known.

Scheme 3. Node to cluster matching

Each node in the historical network was assumed to match with a cluster in the target network, using DBSCAN clustering technique. The choice of the DBSCAN parameters $minPts$ and Eps was based on trial and error and the characteristics of the target dataset. Thus, the entire nodes in the target network were grouped into clusters based on

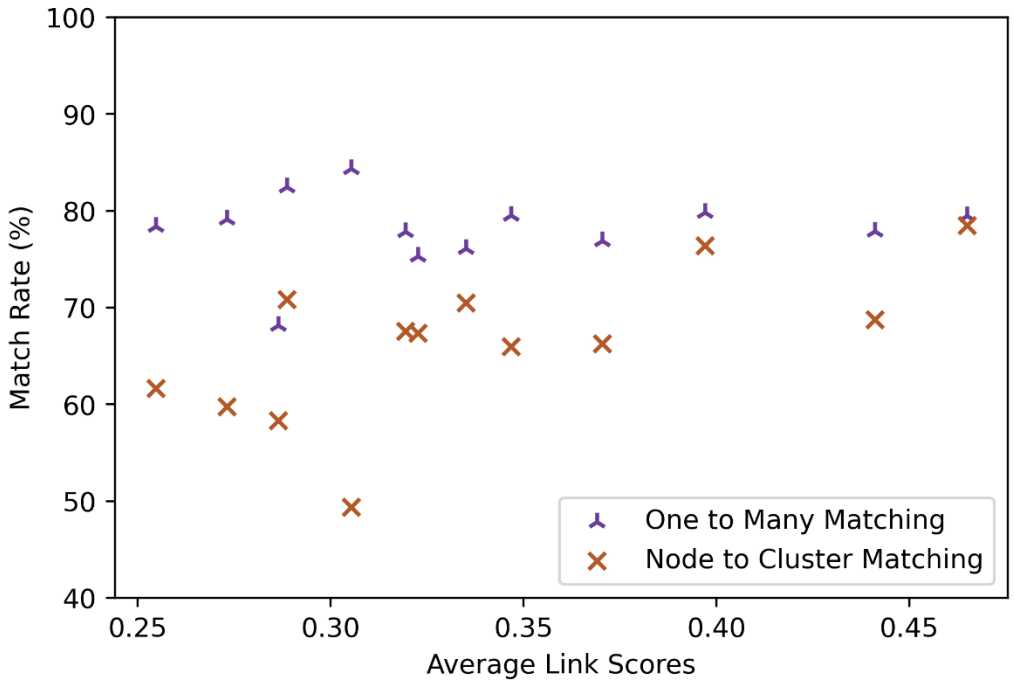


Figure 3-5 Match Rates for one to many and node to cluster matching

$minPts = 3$ and $Eps = 0.2$ miles was used. Figure 3-5 shows a lower match rate compared to one-to-many matching. The reasons for lower match rates are described below.

Because of the nature of DBSCAN clustering, the formation of clusters may not necessarily represent a singular real-world facility. For example, two clusters within Eps might merge into a single cluster when reference nodes are very close to each other. Similarly, the size of the clusters is based on the density and proximity of nodes in the target dataset; however, some instances, such as urban areas, where reference nodes are proximate to each other, require the size of the clusters to be smaller in the target dataset when, in fact, they are larger when more than $minPts$ nodes are present within Eps.

3.6 DISCUSSION

The proposed algorithms demonstrate an improved match rate of point features compared to algorithm-based matching. In addition, because node classification methods were added, the neural network could characterize critical nodes. The match rates were compared against the average link scores and found to be positively related in one-to-one matching. Thus, indicating that the link scores were a good predictor for identifying one node in the target dataset that best represents a node in the reference dataset. But, match rates were not found to be related to the average link scores for one to many and one to cluster matching. Several reasons were identified for a low match rate in all three scenarios. Some of them are listed below.

First, issues inherent to the network, such as oversimplification and inadequate information in some areas of the historical network, led to lower match rates. Similarly, the presence of nodes in reference with different characteristics such as intersections, terminals, source, or sink that are unique to a specific location made it challenging for the neural network to identify nodes in the target dataset. Despite that, room to improve the

match rates was found, and several directions are suggested to address these issues and improve the algorithms for future research.

First, the choice of SBD could be enhanced by dynamically adjusting the SBD based on the clustering of nodes in the target dataset. This would speed up the process of identification of nodes in the target dataset and filter the nodes that do not require evaluation. Next, instead of using zone-based training that mixes nodes with different characteristics, nodes with similar properties could be randomly selected and trained. This would lead to enhanced match rates as the training datasets have similar characteristics. Next, the use of static approximate neighbor nodes could be eliminated by recursively updating the approximate neighbor nodes with identified nodes with higher match probabilities. Finally, MSPL and MSR2 rely on SPA, limiting the neural network's ability to identify nodes in the areas with multiple routes connecting two nodes in the reference network.

CONCLUSION

This dissertation has compiled a series of methods that assist in integrating two datasets with different levels of detail, exploring different ways to conflate GIS datasets. The goal of identifying corresponding elements in two datasets was to improve the positional accuracy of the dataset with lower positional accuracy. The primary concern of this research is the use only of spatial elements to improve match rates.

When integrating two datasets, the lack of quality on at least one dataset is a primary concern with each transportation mode with a specific problem. For example, the railroad dataset does not change much with time but also, the volunteers' interest and contribution are lower. Similarly, the highly-dense areas in railroads are much lesser in density than roadways. The dissertation addressed these peculiarities by proposing several procedures to integrate any two rail datasets. The approaches used in this dissertation were tested on OSM Rail and FRARAIL networks.

First, two heuristic approaches were used to identify corresponding elements between two networks. The first approach (MSPL) minimizes the sum of path distance to neighbor nodes. Similarly, the second approach (MSR2) minimizes the squared root of the sum of the squared difference in corresponding path lengths. Finally, a third approach is a weighted sum of the standardized values from the first and second methods. In all cases, the discriminatory weight of 0.5 performed better matching than MSPL and MSR2. However, the results did not show any strong relationship between match rates in zones with different nodes ratio or average length to neighbor nodes.

Second, a genetic algorithm-based optimization approach proposed a method to detect apparent differences in two maps. The method applies five different spatial elements: endpoints' proximity, endpoints' angle, the displacement between endpoints, path distance between endpoints, and shape similarities to identify differences. The method identified a positive relationship between lower apparent differences scores and match rates using the

heuristic methods. Furthermore, the method detected links with high apparent differences and flagged them for manual intervention.

Lastly, a neural network-based approach was proposed, which introduces the k-core decomposition and bridge decomposition in addition to the parameters incorporated in the heuristic methods. The neural network learns from the ground truth dataset to obtain the probability of each node in the target dataset being the corresponding node. For the matching process, three different scenarios are proposed. The one-to-one match rates were higher than heuristic approaches and negatively correlated with the average link scores. However, no relationship was observed for one to many and one to cluster matching.

Overall, the dissertation provides multiple analysis frameworks and tools for identifying corresponding elements. The methodology was tested on OSM Rail Dataset and FRARAIL. The chapters start with one-to-one node matching. The second chapter uses the links' shapes to facilitate the matching process. Finally, the third chapter explores how different scenarios could result in different match rates when matching networks with varying levels of detail.

FUTURE WORK

This section lists the areas where further improvements could be made that were left out for future work.

The results from the third chapter indicate that training and testing on zones with different characteristics could potentially be the reason for lower match rates. Thus, a study could be done on how the training based on randomly selected nodes from national-level data and testing on smaller sub-regions might affect the testing accuracy.

Since the matching procedures only rely on spatial aspects of the network, it could be interesting to add different spatial noises such as distortions in shapes to the reference links and explore their effects on match rate. This could generate specific cases not present in the available FRA dataset and lead to more comprehensive training. However, the ground truth for the distorted shape has to be identified manually to compare against automated procedures.

Similarly, the methodology was only tested on the FRA Rail network and OSM Rail Network due to lack of availability. However, other networks such as waterways, airways, and pipelines could be explored.

REFERENCES

1. Saalfeld, A.J.I.J.o.G.I.S., *Conflation automated map compilation.* 1988. 2(3): p. 217-228.

2. Haklay, M. and P.J.I.P.c. Weber, *Openstreetmap: User-generated street maps.* 2008. 7(4): p. 12-18.

3. Timpf, S., et al., *A conceptual model of wayfinding using multiple levels of abstraction*, in *Theories and methods of spatio-temporal reasoning in geographic space*. 1992, Springer. p. 348-367.

4. Beller, A., Y. Doytsher, and E.J.P.o.A.A. Shimbersky, *Practical Linear Conflation in an innovative software environment.* 1997.

5. Haunert, J.-H. *Link based conflation of geographic datasets*. in *ICA Workshop on Generalisation and Multiple Representation*. 2005.

6. Yuan, S. and C.J.P.o.G. Tao, *Development of conflation components.* 1999. 99: p. 1-13.

7. Bolstad, P., *GIS fundamentals: A first text on geographic information systems*. 2016: Eider (PressMinnesota).

8. Walter, V. and D. Fritsch, *Matching spatial data sets: a statistical approach.* International Journal of geographical information science, 1999. 13(5): p. 445-473.

9. Church, R., et al. *Positional distortion in geographic data sets as a barrier to interoperation*. in *Proceedings, ACSM Annual Convention, Baltimore*. 1998.

10. Xavier, E.M., F.J. Ariza-López, and M.A.J.A.C.S. Urena-Camara, *A survey of measures and methods for matching geospatial vector datasets.* 2016. 49(2): p. 1-34.

11. Li, L., M.F.J.I.J.o.I. Goodchild, and D. Fusion, *An optimisation model for linear feature matching in geographical data conflation.* 2011. 2(4): p. 309-328.

12. Tong, X., D. Liang, and Y.J.I.J.o.G.I.S. Jin, *A linear road object matching method for conflation based on optimization and logistic regression.* 2014. 28(4): p. 824-846.

13. Bruce, E.J.G.f.c.z.m., *Spatial uncertainty in marine and coastal GIS.* 2004: p. 51-

62.

14. Chrisman, N.R.J.G.i.s., *The error component in spatial data.* 1991. 1(12): p. 165-174.

15. Tveite, H.J.I.j.o.g.i.s., *An accuracy assessment method for geographical line data sets based on buffering.* 1999. 13(1): p. 27-47.

16. Devogele, T., C. Parent, and S.J.I.J.o.G.I.S. Spaccapietra, *On spatial database integration.* 1998. 12(4): p. 335-352.

17. Rosen, B. and A. Saalfeld. *Match criteria for automatic alignment*. in *Proceedings of 7th international symposium on computer-assisted cartography (Auto-Carto 7)*. 1985.

18. Cobb, M.A., et al., *A rule-based approach for the conflation of attributed vector data.* 1998. 2(1): p. 7-35.

19. Pendyala, R.M., *Development of GIS-based conflation tools for data integration and matching*. 2002: Florida Department of Transportation Tallahassee, FL.

20. Filin, S., Y.J.S. Doytsher, and l.i. systems, *Detection of corresponding objects in linear-based map conflation.* 2000. 60(2): p. 117-128.

21. Goodchild, M.F. and G.J.J.I.j.o.g.i.s. Hunter, *A simple positional accuracy measure for linear features.* 1997. 11(3): p. 299-306.

22. Ahmadi, E. and M.A. Nascimento. *k-Optimal meeting points based on preferred paths*. in *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 2016.

23. Schmitz, S., A. Zipf, and P. Neis. *New applications based on collaborative geodata—the case of routing*. in *Proceedings of XXVIII INCA international congress on collaborative mapping and space technology*. 2008.

24. Haklay, M.J.E., p.B. Planning, and design, *How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets.* 2010. 37(4): p. 682-703.

25. Clarke, D.B., *An examination of railroad capacity and its implications for rail-highway intermodal transportation*. 1995, The University of Tennessee.

26. O'Reilly, T., *What is web 2.0*. 2009: " O'Reilly Media, Inc.".

27. Goodchild, M.F.J.G., *Citizens as sensors: the world of volunteered geography.* 2007. 69(4): p. 211-221.

28. Haklay, M.J.C.g.k., *Citizen science and volunteered geographic information: Overview and typology of participation.* 2013: p. 105-122.

29. Budhathoki, N.R. and C.J.A.B.S. Haythornthwaite, *Motivation for open collaboration: Crowd and community models and the case of OpenStreetMap.* 2013. 57(5): p. 548-575.

30. Fonte, C.C., et al., *Assessing VGI data quality.* 2017: p. 137-163.

31. Fonte, C.C., et al., *Usability of VGI for validation of land cover maps.* 2015. 29(7): p. 1269-1291.

32. Bégin, D., R. Devillers, and S. Roche, *Assessing volunteered geographic information (VGI) quality based on contributors' mapping behaviours.* Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci, 2013. 2013: p. 149-154.

33. Antoniou, V. and C. Schlieder, *Participation patterns, VGI and gamification.* Proceedings of AGILE 2014. Presented at the AGILE, 2014: p. 3-6.

34. Estima, J., C.C. Fonte, and M. Painho, *Comparative study of Land Use/Cover classification using Flickr photos, satellite imagery and Corine Land Cover database.* 2014.

35. Zook, M.A. and M. Graham, *Mapping DigiPlace: geocoded Internet data and the representation of place.* Environment and Planning B: Planning and Design, 2007. 34(3): p. 466-482.

36. Zielstra, D. and H.H. Hochmair, *Using free and proprietary data to compare shortest-path lengths for effective pedestrian routing in street networks.* Transportation Research Record, 2012. 2299(1): p. 41-47.

37. Goodchild, M.F., *Geographic information systems and science: today and tomorrow.* Annals of GIS, 2009. 15(1): p. 3-9.

38. Girres, J.F. and G.J.T.i.G. Touya, *Quality assessment of the French OpenStreetMap dataset.* 2010. 14(4): p. 435-459.

39. Haklay, M., et al., *How many volunteers does it take to map an area well? The validity of Linus' law to volunteered geographic information.* 2010. 47(4): p. 315-322.

40. Estima, J. and M. Painho. *Exploratory analysis of OpenStreetMap for land use classification*. in *Proceedings of the second ACM SIGSPATIAL international workshop on crowdsourced and volunteered geographic information*. 2013.

41. Over, M., et al., *Generating web-based 3D City Models from OpenStreetMap: The current situation in Germany.* Computers, Environment and Urban Systems, 2010. 34(6): p. 496-507.

42. Zheng, J., et al., *Reliable Optimal Path for Pedestrian Navigation*, in *ICCTP 2010: Integrated Transportation Systems: Green, Intelligent, Reliable*. 2010. p. 1785-1796.

43. Jacob, R., et al., *Pedestrian navigation using the sense of touch.* Computers, Environment and Urban Systems, 2012. 36(6): p. 513-525.

44. Hentschel, M. and B. Wagner. *Autonomous robot navigation based on openstreetmap geodata*. in *13th International IEEE Conference on Intelligent Transportation Systems*. 2010. IEEE.

45. Yin, J. and J.D. Carswell. *Touch2Query enabled mobile devices: a case study using OpenStreetMap and iPhone*. in *International Symposium on Web and Wireless Geographical Information Systems*. 2011. Springer.

46. Codescu, M., et al., *Osmonto-an ontology of openstreetmap tags.* State of the map Europe (SOTM-EU), 2011. 2011: p. 23-24.

47. Neis, P., P. Singler, and A. Zipf, *Collaborative mapping and emergency routing for disaster logistics–case studies from the haiti earthquake and the UN Portal for Afrika*. 2010: na.

48. Goodchild, M.F. and L. Li, *Assuring the quality of volunteered geographic information.* Spatial statistics, 2012. 1: p. 110-120.

49. Haklay, M., *How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets.* Environment and planning B: Planning and design, 2010. 37(4): p. 682-703.

50. Girres, J.F. and G. Touya, *Quality assessment of the French OpenStreetMap dataset.* Transactions in GIS, 2010. 14(4): p. 435-459.

51. Zielstra, D. and A. Zipf. *A comparative study of proprietary geodata and volunteered geographic information for Germany*. in *13th AGILE international conference on geographic information science*. 2010.

52. Cipeluch, B., et al. *Comparison of the accuracy of OpenStreetMap for Ireland with Google Maps and Bing Maps*. in *Proceedings of the Ninth International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Sciences*. 2010. Citeseer.

53. Goodchild, M.F., L. Anselin, and U. Deichmann, *A framework for the areal interpolation of socioeconomic data.* Environment and planning A, 1993. 25(3): p. 383-397.

54. van Oort, P.A., *Spatial data quality: from description to application*. 2006: Wageningen Universiteit.

55. Devillers, R. and R. Jeansoulin, *Fundamentals of spatial data quality*. 2006: ISTE Publishing Company.

56. Mitchell, M. *Genetic algorithms: An overview*. in *Complex.* 1995. Citeseer.

57. Goldberg, D.E., B. Korb, and K. Deb, *Messy genetic algorithms: Motivation, analysis, and first results.* Complex systems, 1989. 3(5): p. 493-530.

58. Kinnear, K.E., et al., *Advances in genetic programming*. Vol. 3. 1994: MIT press.

59. Demetriou, D., L. See, and J. Stillwell, *A spatial genetic algorithm for automating land partitioning.* International Journal of Geographical Information Science, 2013. 27(12): p. 2391-2409.

60. Vinh, N.N. and B. Le. *Incremental spatial clustering in data mining using genetic algorithm and R-tree*. in *Asia-Pacific Conference on Simulated Evolution and Learning*. 2012. Springer.

61. Yücenur, G.N. and N.Ç. Demirel, *A new geometric shape-based genetic clustering algorithm for the multi-depot vehicle routing problem.* Expert Systems with Applications, 2011. 38(9): p. 11859-11865.

62. Chehreghan, A. and R. Ali Abbaspour, *A geometric-based approach for road matching on multi-scale datasets using a genetic algorithm.* Cartography and Geographic Information Science, 2018. 45(3): p. 255-269.

63. Goodchild, M.F., *Citizens as sensors: the world of volunteered geography.* GeoJournal, 2007. 69(4): p. 211-221.

64. Zhang, M. and L. Meng, *An iterative road-matching approach for the integration of postal data.* Computers, Environment and Urban Systems, 2007. 31(5): p. 597-

615.

65. Alvarez-Hamelin, J.I., et al. *Large scale networks fingerprinting and visualization using the k-core decomposition*. in *Advances in neural information processing systems*. 2006.

66. Wuchty, S. and E. Almaas, *Peeling the yeast protein network.* Proteomics, 2005. 5(2): p. 444-449.

67. Giatsidis, C., et al. *Corecluster: A degeneracy based graph clustering framework*. in *Twenty-Eighth AAAI Conference on Artificial Intelligence*. 2014.

68. Paola, J.D. and R.A. Schowengerdt, *A detailed comparison of backpropagation neural network and maximum-likelihood classifiers for urban land use classification.* IEEE Transactions on Geoscience and remote sensing, 1995. 33(4): p. 981-996.

69. Lei, T. and Z.J.T.i.G. Lei, *Optimal spatial data matching for conflation: A network flow-based approach.* 2019. 23(5): p. 1152-1176.

70. Wang, Y., et al., *A back-propagation neural network-based approach for multi-represented feature matching in update propagation.* Transactions in GIS, 2015. 19(6): p. 964-993.

71. Chehreghan, A. and R. Ali Abbaspour, *An assessment of the efficiency of spatial distances in linear object matching on multi-scale, multi-source maps.* International Journal of Image and Data Fusion, 2018. 9(2): p. 95-114.

VITA

Pankaj was born in a small village named Sakhuwathan, in the south-eastern part of Nepal. He was raised in Biratnagar, where he received his education up to the high school level. After completing his Bachelor's degree in Civil Engineering from Tribhuvan University, he worked at a private consulting firm GOEC, Nepal. Next, he started his graduate study at the University of Tennessee, Knoxville, in August 2015. In 2017, he obtained his M.S in Civil Engineering. As a graduate student, he worked at the Center of Transportation Research (CTR). He is expected to get his PhD. in May 2022 with a concentration in Transportation Engineering and an M.S in Statistics.