A Study of the Railway Network Critical Evaluation by Multiple Criteria: Case Studies of Inter-city and Urban Railway Networks in Japan

著者	プロムスワン ポンウィット
著者別表示	PROMSUWAN PONGWIT
journal or publication title	博士論文本文Full
学位授与番号	13301甲第5591号
学位名	博士（学術）
学位授与年月日	2022-09-26
URL	http://hdl.handle.net/2297/00068850

doi: https://doi.org/10.1016/j.eastsj.2022.100074

Dissertation

A Study of the Railway Network Critical Evaluation by Multiple Criteria: Case Studies of Inter-city and Urban Railway Networks in Japan

Graduate School of
Natural Science & Technology
Kanazawa University

Division of Environmental Design

Student ID No. 1924052010
Name: PROMSUWAN Pongwit
Chief advisor: Professor NAKAYAMA Shoichiro
Date of Submission: 18 August 2022

Abstract

This research aimed to study the nodes or links of the railway network's topology by multiple criteria, mainly composed of centrality, vulnerability, and stochastic block model analyses. The testing was conducted on both inter-city and urban railway network case studies of Kyushu, Tokyo, and Osaka. The main purpose is to find the critical nodes or links that extensively affect the operation if it is disrupted or cut off. Each node and link represent the station and railway section in the network, respectively. The results are expected to help railway operators scope the stations or sections that need the priority for preventive planning, such as inspecting, maintaining, and repairing under the limit of their resources and budget. Any rail section, considered vulnerable from multiple criteria views and has a very critical station within, can be managed as a priority. In addition, each type of centrality and vulnerability were studied and then compared the performance of each other. This comparative analysis was conducted based on the criteria's purpose and the network's topology.

Contents

Chapter 1 Introduction........10

1.1 Background........10
1.2 Research Purpose........12
1.3 Research Planning Process........12

Chapter 2 The Previous Research Studies........17

2.1 Overview........17
2.2 Railway Network Centrality Analysis........19
2.3 Railway Network Efficiency Analysis........22
2.4 Algebraic Connectivity for Railway Network Analysis........23
2.5 Stochastic Block Model........24
2.6 The Position of the Research Study........26

Chapter 3 Basic Theories of Network Analysis........34

3.1 Graph Theory........34
3.2 Eigenvalue Theory........35
3.3 Centrality Theories........36
3.3.1 Degree centrality........36
3.3.2 Closeness centrality........37
3.3.3 Eigenvector centrality........37
3.3.4 Betweenness centrality........38
3.3.5 Information centrality........39
3.4 Algebraic Connectivity........40
3.5 Global Efficiency........41
3.6 Stochastic Block Model Basic Theory........41

Chapter 4 Case Study Railway Networks........45

4.1 Case Study of Kyushu Railway Network........45
4.2 Case Study of Tokyo Subway Network........60

4.3 Case Study of Osaka Subway Network 66

Chapter 5 Network Centrality Analysis for the Railway Network..................................72

5.1 Overview of Centrality Analysis 72
5.2 Square Grid Network Testing for All Five Centralities...72
5.2.1 The calculation time testing of centrality analyses 72
5.2.2 Comparison of the centrality of square grid networks 73
5.3 Centralities Testing Results with a Case Study of the Kyushu Railway Network 75
5.4 Centralities Testing Results with a Case Study of the Tokyo Subway Network...87
5.5 Centralities Testing Results with a Case Study of the Osaka Subway Network ...92
5.6 Comparison of Centrality Results from All Case Study Networks 94
5.6.1 Degree centrality result...95
5.6.2 Closeness centrality result 97
5.6.3 Eigenvector centrality result...99
5.6.4 Betweenness centrality result 101
5.6.5 Information centrality result..103
5.7 Correlation Analysis 105
5.8 Summary of the Centrality Analyses 106

Chapter 6 Algebraic Connectivity- and Global Efficiency-based Vulnerability Analyses 109

6.1 Overview of Vulnerability Analysis 109
6.1.1 Equations and algorithms for vulnerability analysis 109
6.1.2 The attack testing...111
6.2 Result of Comparison Between Algebraic Connectivity- and Global Efficiency-based Vulnerability Analyses 111
6.2.1 The calculation time testing of vulnerability analyses 111
6.2.2 Comparison of the vulnerability of the square grid network.........................112
6.2.3 Case studies vulnerability results 113
1) Vulnerable sections of the algebraic connectivity-based vulnerability analysis 137
2) Vulnerable sections of the global efficiency-based vulnerability analysis ...139
3) Critical sections of the edge betweenness centrality analysis 142

6.2.4 Case studies correlation analysis 144

6.3 Summary of the Topology-based Comparison Between Algebraic Connectivity- and Global Efficiency-based Vulnerability Analyses 150

6.4 Algebraic Connectivity-based Vulnerability Analysis with the Passenger-weighted Link in the Kyushu Railway Network 151

Chapter 7 Stochastic Block Model Analysis 157

7.1 Overview of Weighted Stochastic Block Model (WSBM) 157

7.2 The Simple Result of Weighted Stochastic Block Model for Case Studies 157

7.2.1 The weighted stochastic block model results of the Kyushu railway network 167

7.2.2 The weighted stochastic block model results of the Tokyo subway network 171

7.2.3 The weighted stochastic block model results of the Osaka subway network173

Chapter 8 Discussions and Conclusions 177

8.1 Discussions 177

8.2 Conclusions 182

8.2.1 Railway network centrality analysis 182

8.2.2 The algebraic connectivity- and global efficiency-based vulnerability analyses for the railway network 182

8.2.3 The stochastic block model analysis for the railway network 183

8.3 Future Works 183

Acknowledgments 184

Appendixes 186

Appendix 1: MATLAB code for the algebraic connectivity-based vulnerability analysis 186

Appendix 2: MATLAB code for the global efficiency-based vulnerability analysis 188

Appendix 3: MATLAB code for the edge betweenness centrality analysis 190

Appendix 4: MATLAB code for the degree, closeness, betweenness, eigenvector, and information centralities 194

Appendix 5: MATLAB code for the weighted stochastic block model (WSBM) 196

List of Figures

Figure 1. Research planning process flow chart....13

Figure 2. An example of an infinite relational model for network data with 4 clusters .24

Figure 3. The example of clustering from the stochastic block model with the probability of each pair of edge blocks 25

Figure 4. The simple example of a labeled graph, degree matrix, adjacency matrix, and Laplacian matrix....34

Figure 5. The example of the information centrality measuring of node 3, the value can obtain by the difference in global efficiency of the entire network ($E_{glob}(G)$) and global efficiency after removing every link that connects node 3 ($E_{glob}(G'_3)$) 39

Figure 6. A case study railway network in Kyushu, which compose of JR Kyushu, Nishitetsu railway, and Fukuoka city subway network in yellow highlight 46

Figure 7. Railway network of JR Kyushu, Nishitetsu Railway, and Fukuoka City Subway, including the sections connected to the JR West network....47

Figure 8. Subway network in Tokyo 61

Figure 9. Subway network in Osaka....67

Figure 10. Calculation time results of centralities on the six different sizes of square grid networks 73

Figure 11. Comparison of the critical nodes from all five centralities on the example of a square grid network 74

Figure 12. Degree centrality result of the Kyushu railway network 95

Figure 13. Degree centrality result of the Tokyo subway network 96

Figure 14. Degree centrality result of the Osaka subway network....96

Figure 15. Closeness centrality result of the Kyushu railway network 97

Figure 16. Closeness centrality result of the Tokyo subway network....98

Figure 17. Closeness centrality result of the Osaka subway network 98

Figure 18. Eigenvector centrality result of the Kyushu railway network....99

Figure 19. Eigenvector centrality result of the Tokyo subway network 100

Figure 20. Eigenvector centrality result of the Osaka subway network....100

Figure 21. Betweenness centrality result of the Kyushu railway network....101

Figure 22. Betweenness centrality result of the Tokyo subway network....102

Figure 23. Betweenness centrality result of the Osaka subway network 102

Figure 24. Information centrality result of the Kyushu railway network....103

Figure 25. Information centrality result of the Tokyo subway network....104

Figure 26. Information centrality result of the Osaka subway network 104
Figure 27. Example of an attack scenario on the link between nodes 6 and 7 110
Figure 28. Calculation time results on the six different sizes of square grid networks 112
Figure 29. Comparison between the algebraic connectivity-based vulnerability (left), global efficiency-based vulnerability (center), and edge betweenness centrality (right) 113
Figure 30. The algebraic connectivity-based vulnerability of the Kyushu railway network 137
Figure 31. The algebraic connectivity-based vulnerability of the Tokyo subway network 138
Figure 32. The algebraic connectivity-based vulnerability of the Osaka subway network 139
Figure 33. The global efficiency-based vulnerability of the Kyushu railway network .140
Figure 34. The global efficiency-based vulnerability of the Tokyo subway network...141
Figure 35. The global efficiency-based vulnerability of the Osaka subway network ...141
Figure 36. Edge betweenness centrality of the Kyushu railway network 142
Figure 37. Edge betweenness centrality of the Tokyo subway network 143
Figure 38. Edge betweenness centrality of the Osaka subway network........................144
Figure 39. Correlation between the algebraic connectivity- and global efficiency-based vulnerabilities of the Kyushu railway network 145
Figure 40. Correlation between the algebraic connectivity-based vulnerability and edge betweenness centrality of the Kyushu railway network 146
Figure 41. Correlation between the global efficiency-based vulnerability and edge betweenness centrality of the Kyushu railway network 146
Figure 42. Correlation between the algebraic connectivity- and global efficiency-based vulnerabilities of the Tokyo subway network 147
Figure 43. Correlation between the algebraic connectivity-based vulnerability and edge betweenness centrality of the Tokyo subway network 147
Figure 44. Correlation between the global efficiency-based vulnerability and edge betweenness centrality of the Tokyo subway network 148
Figure 45. Correlation between the algebraic connectivity- and global efficiency-based vulnerabilities of the Osaka subway network...148
Figure 46. Correlation between the algebraic connectivity-based vulnerability and edge betweenness centrality of the Osaka subway network 149
Figure 47. Correlation between the global efficiency-based vulnerability and edge betweenness centrality of the Osaka subway network 149

Figure 48. The simple JR Kyushu railway network 151

Figure 49. The algebraic connectivity-based vulnerability with passenger-weighted link of the simple JR Kyushu railway network 155

Figure 50. The weighted stochastic block model clustering of the Kyushu railway network with 3 clusters (k = 3)..168

Figure 51. The weighted stochastic block model clustering of the Kyushu railway network with 4 clusters (k = 4)..169

Figure 52. The location of cluster group 3 (in yellow) divided the core clusters 1 and 2 after analyzing the weighted stochastic block model with the three-cluster case of the Kyushu railway network..170

Figure 53. The weighted stochastic block model clustering of the Tokyo subway network with 3 clusters (k = 3)..171

Figure 54. The weighted stochastic block model clustering of the Tokyo subway network with 4 clusters (k = 4)..172

Figure 55. The location of cluster group 1 (in red) divided the core clusters 2 and 3 after analyzing the weighted stochastic block model with the three-cluster case of the Tokyo subway network..173

Figure 56. The weighted stochastic block model clustering of the Osaka subway network with 3 clusters (k = 3)..174

Figure 57. The weighted stochastic block model clustering of the Osaka subway network with 4 clusters (k = 4)..174

Figure 58. The location of cluster group 2 (in green) divided the core clusters 1 and 3 after analyzing the weighted stochastic block model with the three-cluster case of the Osaka subway network..175

Figure 59. The vulnerability of the section between Shin-Tosu and Hizen-Yamaguchi stations of the JR Kyushu Nagasaki Main Line, Kyushu railway network..................180

Figure 60. The very high centrality area at the Shin-Tosu, Kurume, and Chikugo-Funagoya stations when considering the closeness centrality, betweenness centrality, and information centrality 180

List of Tables

Table 1. Characteristics of each type of network centrality 20
Table 2. Node number and station name of the Kyushu railway network 48
Table 3. Link number and pair of nodes of the Kyushu railway network.......................55
Table 4. Node number and station name of the Tokyo subway network 62
Table 5. Link number and pair of nodes of the Tokyo subway network.........................64
Table 6. Node number and station name of the Osaka subway network.........................67
Table 7. Link number and pair of nodes of the Osaka subway network 69
Table 8. The average processing time of the centrality analyses on various sizes of square grid networks..72
Table 9. All five node-based centralities of the Kyushu railway network 75
Table 10. The top 10 ranks the highest centrality of the Kyushu railway network.........87
Table 11. All five node-based centralities of the Tokyo subway network 87
Table 12. The top 10 ranks the highest centrality of the Tokyo subway network 91
Table 13. All five node-based centralities of the Osaka subway network 92
Table 14. The top 10 ranks the highest centrality of the Osaka subway network 94
Table 15. Centrality correlation result of the Kyushu railway network 105
Table 16. Centrality correlation result of the Tokyo subway network..........................106
Table 17. Centrality correlation result of the Osaka subway network 106
Table 18. Summary of centrality's performance by each type.....................................107
Table 19. The algorithm for link-based vulnerability evaluation.................................110
Table 20. The average processing time of the vulnerabilities and edge betweenness centrality analyses on various sizes of square grid networks 112
Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network..113
Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network...128
Table 23. The results of vulnerabilities and edge betweenness centrality by the link of the Osaka subway network 134
Table 24. Node number and station name of the simple JR Kyushu network 152
Table 25. The results of algebraic connectivity-based vulnerability with passenger-weighted edges 153
Table 26. Clusters group of the Kyushu railway network WSBM analysis.................158

Table 27. Clusters group of the Tokyo subway network WSBM analysis....................164
Table 28. Clusters group of the Osaka subway network WSBM analysis 166

Chapter 1

Introduction

1.1 Background

In recent years, the rail transport system had an extensive role in developing the economy and life quality. This transport mode rules the solution for traveling and transporting both people and freights for nearly two centuries, and it also has a positive effect on world society in several dimensions. For example, commuter services solve road traffic problems, reduce the carbon emission from road vehicles and develop the logistic system to help to improve the supply chain (Pyrgidis, 2016; Carbonbrief, 2019).

However, the railway systems are facing higher and more pressure to compete with more developed highway systems and more advanced technology from air transport. For these reasons, many railway operators try to develop new technologies and strategies to motivate people and business sectors to use rail transport mode. One of the important reliability strategies is the network's critical analysis, which purposes to evaluate and solve for the guarantee that the network can still operate or recover fastest if natural or man-made incidents disrupt it. Nevertheless, most of the operators have limited resources to operate and maintain the system, such as budget, equipment, labor, or infrastructure, so it is nearly impossible to manage or inspect all of the railway's sections in the same period. To keep the railway network still reliable, the priority management needs to consider solving these constrain by focusing on the most critical or vulnerable point or section as the top priority that needs management as a preventive strategy.

At present, we have several criteria to evaluate and identify the most important railway node or section. The centrality analysis is one of the most popular methods to identify these nodes or links. The centrality analysis aims to evaluate any node within the network by calculating the graph in matrix form. If any node or link has a high value of centrality, it means this node or link has a strong influence or is critical, which probably has an extensive effect on the network if it is disrupted or fails (Boudin, 2013; Rodrigues, 2019). For this reason, many research used the centrality analysis to identify the important stations or railway sections as the top priority for strategic or preventive planning. In this thesis, we intend to use the main centrality criteria to analyze the railway network, composed of eigenvector centrality, degree centrality, betweenness centrality, closeness centrality, and information centrality (Boudin, 2013; Rodrigues, 2019; Amrit and ter Maat, 2018) to compare results from each case after analyzing case study networks.

Another criterion for analyzing railway network reliability and critical is vulnerability analysis. This method's primary purpose is to identify the most sensitive edge or vertex, which is usually represented as the station, junction, city, or railway section. The node or edge, which has a high vulnerability volume, is considered vulnerable or sensitive to disrupt traffic in extensive areas if attacked or cut off. To keep the network still reliable, the operator can analyze the vulnerable section and then consider managing priority for checking, inspecting, or repairing for preventive maintenance planning. In this thesis, we studied two types of vulnerability analysis, the first is developed algebraic connectivity-based vulnerability analysis, and another is the existing global efficiency-based vulnerability analysis.

The global efficiency analysis is one of the common criteria that can measure the network's efficiency by calculating the average shortest path from every pair of nodes in inverse form. The value of efficiency can show the connectivity of the network. If its value is high, this network is highly connected (Ek et al., 2015). For evaluation of the vulnerability, it can calculate by measuring the change in global efficiency entire the network before the attacking simulation and the efficiency of remain network after attacking (Sun and Guan, 2016; Candelieri et al., 2019; Saadat et al., 2020; Noguchi and Fuse, 2020; Gu et al., 2020). If any section or station has a high vulnerability, that means it is very vulnerable. The global efficiency-based vulnerability analysis not only analyzes the topology characteristic but also applies to the passenger flow, traffic flow, and freight flow. In addition, the global efficiency-based vulnerability can also scope to identify the vital section or area alongside the centrality analysis, such as betweenness centrality.

The following criterion, which usually measures the strength of the network, is the algebraic connectivity analysis. This indicator is based on the second smallest eigenvector analysis, which aims to measure the network's connectivity level, which depends on the network's strong or density (Fiedler, 1973; Xu et al., 2020; De Abreu, 2007). In other words, if the network has a high algebraic connectivity value, it means this network has robust connectivity. However, if the links within the network are removed or cut off, the algebraic connectivity will decrease. From this property, it is possible to apply these criteria to analyze the vulnerability by measuring the change of algebraic connectivity both before and after removing the link. This concept can develop into the algebraic connectivity-based vulnerability analysis, which can expect to illustrate the property and result if testing with a case study of the railway network.

The addition criterion is the stochastic block model (SBM), which purpose to identify or classify the group of node’s communities or clusters (Aicher et al., 2015). The measuring can obtain by analyzing the probabilistic of pairwise interactions between nodes. If any probability of any pair of nodes has a high value, that means both nodes come from the same group. If pairs of nodes come from different groups, the probability will be low (Lee and Wilkinson, 2019). Any edge connected between different groups of clusters can be considered a vulnerable link.

From all criteria considered for evaluating the critical of the railway network, this thesis intends to analyze the topology of the railway network to study the performance, and its result, then compare to identify both advantages and disadvantages. The study result can be expected to analyze the passenger flow or traffic flow as future tasks, which has benefit to the management of railway operation, preventive planning, and definition of strategic planning to improve operation and service.

1.2 Research Purpose

Due to the research purpose of railway network critical evaluation, the process for the purpose of this research is illustrated by the following.

1.2.1 Study the characteristic of the case study railway networks, both inter-city railway networks and mass rapid transit networks (subway and urban railway), in which each case are considered by the city or region with an extensive network. According to the graph theories, these networks will be plotted into the graph network, then transformed into the adjacency matrix, which is the basic matrix for analyzing the centrality, global efficiency, algebraic connectivity, and stochastic block model.

1.2.2 Evaluating each node or link of the case study network by centrality analysis with various types, including degree centrality, eigenvector centrality, closeness centrality, betweenness centrality, or information centrality. The results of each network or condition will be compared to each other to analyze the characteristics, critical stations, or sections.

1.2.3 Evaluating the developed algebraic connectivity-based vulnerability of the case study railway network, then studying the vulnerable section from this criterion.

1.2.4 Evaluating the global efficiency and global efficiency-based vulnerability of the case study railway network, then comparing the performance with algebraic connectivity-based vulnerability to study the similarities or differences between both types of analysis.

1.2.5 Evaluating the stochastic block model of the case study railway network to identify the group of node blocks or clusters, then identify the link that connected between blocks or clusters as the vulnerable link.

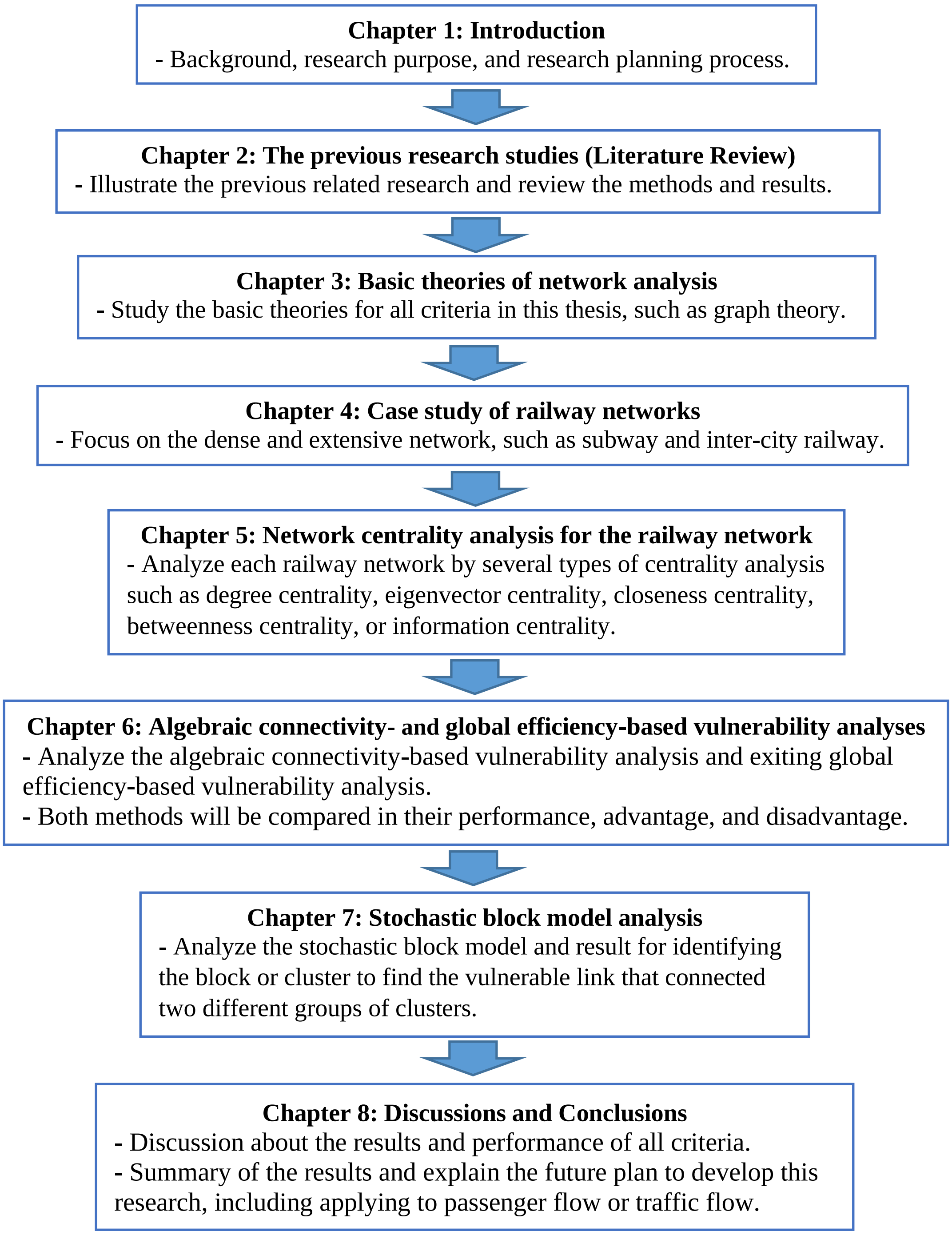
1.3 Research Planning Process

The planning process will be conducted and illustrated in the flow chart in Fig. 1. Each chapter in this thesis is followed the research process.

1.3.1 Study background of the railway operation situation, including operation management, reliability, and critical evaluation. After analyzing the situation, the next step is defining the purposes of the research, including the planning of the research's methodologies and processes.

1.3.2 Study the previous related research for analyzing the different or similar purposes, methodologies, processes, and results, obtained from several railway network case studies. All of this information will be summarized in the literature review in chapter 2 of this report (Shown in Fig. 1).

1.3.3 As in chapter 3, we plan to study the related theories, including graph theory, eigenvalue analysis, centralities, algebraic connectivity, global efficiency, and stochastic block model.



Chapter 1: Introduction
- Background, research purpose, and research planning process.

Chapter 2: The previous research studies (Literature Review)
- Illustrate the previous related research and review the methods and results.

Chapter 3: Basic theories of network analysis
- Study the basic theories for all criteria in this thesis, such as graph theory.

Chapter 4: Case study of railway networks
- Focus on the dense and extensive network, such as subway and inter-city railway.

Chapter 5: Network centrality analysis for the railway network
- Analyze each railway network by several types of centrality analysis such as degree centrality, eigenvector centrality, closeness centrality, betweenness centrality, or information centrality.

Chapter 6: Algebraic connectivity- and global efficiency-based vulnerability analyses
- Analyze the algebraic connectivity-based vulnerability analysis and exiting global efficiency-based vulnerability analysis.
- Both methods will be compared in their performance, advantage, and disadvantage.

Chapter 7: Stochastic block model analysis
- Analyze the stochastic block model and result for identifying the block or cluster to find the vulnerable link that connected two different groups of clusters.

Chapter 8: Discussions and Conclusions
- Discussion about the results and performance of all criteria.
- Summary of the results and explain the future plan to develop this research, including applying to passenger flow or traffic flow.

Figure 1. Research planning process flow chart

1.3.4 Each railway network will be selected as a case study for testing with several critical analyses to evaluate and compare the performances and results. This research focuses on the dense and extensive network to compare the various criteria.

1.3.5 Begin to use the centrality analysis to evaluate and analyze the selected railway networks. Each network will be analyzed with several types of centralities, such as degree centrality, eigenvector centrality, closeness centrality, betweenness centrality, and information centrality. The result of each condition will be comparing the characteristics, performances, and applications for analyzing the critical of the network.

1.3.6 Study the edge-based vulnerability as one of the major criteria for analyzing the critical of the railways. The main vulnerability indicator in this research is the algebraic connectivity-based vulnerability analysis that intends to evaluate each link of the railway network as the track section evaluation. The algebraic connectivity-based vulnerability can be obtained by the second smallest eigenvector from the network, which can measure the condition of the network's connectivity. The results and performances of this study will be compared with the exiting global efficiency-based vulnerability analysis, which is more frequently for evaluating the edge robust and is based on the shortest paths measuring. Moreover, both methods are planned to compare the performance with edge betweenness centrality, which has been used to evaluate railway sections alongside the global efficiency analysis.

1.3.7 Study and analyze the stochastic block model (SBM) for dividing and finding the group of nodes block or clusters. The result can apply to identify the vulnerable link within the network.

1.3.8 Discussion of the result from centrality analyses, edge-based vulnerability analyses, and stochastic block model, including the performances, advantages and disadvantages, the usage and application as future tasks, and the conclusion of this research.

All testing for this research used MATLAB program in a computer with Intel(R) Core(TM) i5-4570 CPU @ 3.20GHz, and 16.0 GB of RAM due to more convenience for linking data with spreadsheet software and ease for the testing algorithm.

References

[1] Aicher, C., Jacobs, A. Z., & Clauset, A. (2015). Learning latent block structure in weighted networks. *Journal of Complex Networks*, 3, 221–248. https://doi.org/10.1093/comnet/cnu026

[2] Amrit, C., & ter Maat, J. (2018). Understanding Information Centrality Metric: A Simulation Approach. (http://arxiv.org/abs/1812.01292; Accessed December 1, 2020)

[3] Boudin, F. (2013). A Comparison of Centrality Measures for Graph-Based Keyphrase Extraction. *International Joint Conference on Natural Language Processing*, 834–838.

[4] Candelieri, A., Galuzzi, B. G., Giordani, I., & Archetti, F. (2019) Vulnerability of public transportation networks against directed attacks and cascading failures. *Public Transport*, 11(1), 27–49. https://doi.org/10.1007/s12469-018-00193-7

[5] Carbonbrief (2019) Eight charts show how ‘aggressive’ railway expansion could cut emissions. (https://www.carbonbrief.org/eight-charts-show-how-aggressive-railway-expansion-could-cut-emissions; Accessed November 17, 2020)

[6] De Abreu, N. M. M. (2007) Old and new results on algebraic connectivity of graphs. *Linear Algebra and Its Applications*, 423(1), 53–73. https://doi.org/10.1016/j.laa.2006.08.017

[7] Ek, B., VerSchneider, C., & Narayan, D. A. (2015). Global efficiency of graphs. *AKCE International Journal of Graphs and Combinatorics*, 12(1), 1–13. https://doi.org/10.1016/j.akcej.2015.06.001

[8] Fiedler, M. (1973). Algebraic connectivity of graphs, with applications. *Czechoslovak Mathematical*, 23(November), 298–305.

[9] Gu, Y., Fu, X., Liu, Z., Xu, X., & Chen, A. (2020) Performance of transportation network under perturbations: Reliability, vulnerability, and resilience. *Transportation Research Part E: Logistics and Transportation Review*, 133, 101809. https://doi.org/10.1016/j.tre.2019.11.003

[10] Lee, C., & Wilkinson, D. J. (2019). A review of stochastic block models and extensions for graph clustering. *Applied Network Science*, 4. https://doi.org/10.1007/s41109-019-0232-2

[11] Noguchi, H., & Fuse, M. (2020) Rethinking critical node problem for railway networks from the perspective of turn-back operation. *Physica A: Statistical Mechanics and Its Applications*, 558, 124950. https://doi.org/10.1016/j.physa.2020.124950

[12] Pyrgidis, C. N. (2016) *Railway Transportation Systems.* CRC Press Taylor & Francis Group, Florida.

[13] Rodrigues, F. A. (2019) Network centrality: an introduction, *A Mathematical Modeling Approach from Nonlinear Dynamics to Complex Systems.* Vol.22, New York: Springer.

[14] Saadat, Y., Ayyub, B. M., Zhang, Y., Zhang, D., & Huang, H. (2020) Resilience-based strategies for topology enhancement and recovery of metrorail transit networks. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering,* 6(2), 04020017. https://doi.org/10.1061/ajrua6.0001057

[15] Sun, D. J., & Guan, S. (2016) Measuring vulnerability of urban metro network from line operation perspective. *Transportation Research Part A: Policy and Practice*, 94, 348–359. https://doi.org/10.1016/j.tra.2016.09.024

[16] Xu, Z., Zhang, Q., Chen, D., & He, Y. (2020) Characterizing the connectivity of railway networks. *IEEE Transactions on Intelligent Transportation Systems*, 21(4), 1491–1502. https://doi.org/10.1109/TITS.2019.2909120

Chapter 2

The Previous Research Studies

2.1 Overview

The reliability of railway transportation has been studied and described for decades in many issues, and all have the main purpose of improving both efficiency and safety of railway service (Vromans, 2005; Landex, 2012; Pyrgidis, 2016). In the case of a large or dense railway network management, the reliability analysis has an important role in network evaluation for identifying the critical point, section, or link that affects the railway operation and infrastructure if it fails or disrupts. For identifying a critical node or link, the node and link-based analysis can be evaluated and performed by various methods (Bıyıkolu et al., 2007; Boudin, 2013; Kaye et al., 2014; Parajuli and Haynes, 2018; Zhang et al., 2019; Du et al., 2020).

One of the patterns for evaluating the reliability of railway networks is critical evaluation, which is conducted using the graph network topological analysis, especially at the macroscopic level. The macroscopic graph illustrates the nodes or links that connect each other and usually represent stations, junctions, terminals, and rail sections, making it easy to analyze and manage in the overview. For example, Schlechte et al. (2011) analyzed the critical node at the macroscopic level for easier study and analyzing all of the factors that affect the entire network. Ye and Kim (2019) also studied the entire Amtrak rail network and used an algorithm-based approach for solving shortest path network problems to identify and evaluate the critical node if a disruption happens in the network. In addition, the node and link analysis on the macroscopic network not only applies to the transportation network but also has been applied to an infrastructure network, such as water and gas supply system (Shuang et al., 2014; Chandra et al., 2019), social network (Kaye et al., 2014) and telecommunication network (Santos et al., 2018; Barbosa et al., 2018).

The first critical evaluation method in the rail transportation field is the centrality analysis, which uses to identify the most important node that needs to protect or preventive plan (Boudin, 2013; Rodrigues, 2019).

Another evaluation method is simulated attacking by removing the nodes or links within the study network. This method purposes to simulate either target attack or random attack scenario if the station or railway sections are disrupted or suspended by a natural disaster, system malfunction, accident, or even terrorist attack (Zhu et al.; 2018). The change of any indicator after the attacking scenario can be illustrated as vulnerability. In the example, Rodríguez-Núñez and García-Palomares (2014) evaluated the Madrid metro system by measuring the change in travel time after the link disruption scenario, as illustrated in Eq. 2.1 that

$$V_i = \frac{\sum_{a \in A_W} \bar{T}_{ia} - \bar{T}_i}{N_w} \tag{2.1}$$

where V_i is the exposer in station *i*, N_w is the number of links in network A_w, $\bar{T}_i$ is the average traveling time for station *i*, and $\bar{T}_{ia}$ is the average traveling time for station *i* after the disruption of link *a* under the assumption that the average traveling time will increase.

Another example is the case of Lu et al. (2022), who created the multiple criteria vulnerability model of urban rail transit under cascading failures. The model is composed of the number of failed node ratios, change of passengers, and change of average travel time after attacking scenarios. All three criteria also definite the weight of the corresponding rate of change. Their model of vulnerability (*V*) is similar to the following equation.

$$V = \lambda_1 \left(\frac{N_T}{N_O}\right) + \lambda_2 \left(1 - \frac{|F_O - F_T|}{F_O}\right) + \lambda_3 \left(\frac{T_T - T_O}{T_O}\right) \tag{2.2}$$

where N_T and N_O are the number of disrupted stations and the total number of stations on the network, respectively. F_O and F_T are the number of passengers on the network before station failures and the sum of all affected passengers after failures, respectively. T_O and T_T are the average travel time before and after failures, respectively. λ_1, λ_2 and λ_3 represent the weight of the corresponding rate of change, which $\lambda_1, \lambda_2, \lambda_3 \in (0, 1)$ and $\lambda_1 + \lambda_2 + \lambda_3 = 1$. However, this research distributed the weight of the corresponding rate of change to all three variables equally, which means $\lambda_1, \lambda_2, \lambda_3$ = 0.33333.

Moreover, there are several other research works that evaluated critical nodes or links by measuring the change of topology and indicator after a node or link attack. For example, Santos et al. (2018) used the critical node detection (CND) problem, which is linear programming to identify the critical node within a telecommunication network. If the critical node is removed, it will minimize or restrict a given metric of network connectivity. Sohn (2006) studied the risk of highway network disruption from flooding. The research showed the accessibility index, which is composed of several factors and can evaluate deterioration after the link disruption. Chandra et al. (2019) evaluated the critical gas pipeline crossings for truck routes by using the dangerous index, which is based on the weighted number of road-pipeline crossings to evaluate truck tonnage impacts. Another criterion is the detour index, which is obtained from the change of shortest route of the pipeline after disruption of the original section and needs to supply gas via an alternate route. Shuang et al. (2014) evaluated the water supply network vulnerability by identifying the critical nodes that could make an extensive network performance reduction. They also used the minimum cut set to study the failure of nodes with the effect on supply pressure in the network. Khademi et al. (2021) measured the change or impact of the multi-criterion-based vulnerability of each link of the Iranian railway network if it is disrupted. The vulnerability's criteria are composed of the passenger unsatisfied demand, the additional cost of transporting passengers, freight shipment unsatisfied demand, the additional cost of transporting freight, and reconstruction cost. Divya et al. (2022) measured the change of average geodesic after simulating simultaneous and sequential attacks on the network, which is considered three types of network centrality.

In addition, network evaluation is usually applied to specific scenario simulations for analyzing the network, and critical points, such as Günneç and Salman (2011) studied the reliability and vulnerability of the Istanbul highway network using the Monte Carlo simulation algorithm under earthquake risk. Liu et al. (2017) analyzed Beijing-Tianjin-Hebei Region (BTHR) rail transit network vulnerability to evaluate connectivity and reliability. The change in network performance measures the vulnerability after a node attack, especially network efficiency and the largest component size, which consider the average shortest path and number of nodes, respectively. The connectivity reliability measure based on the Monte Carlo simulation also was used in this research.

For this reason, this thesis will focus on the methods that are frequently considered to analyze the critical of the railway's network, especially the centrality analysis and attacking scenario vulnerability analysis, which applies to several indicators such as the network efficiency and algebraic connectivity analysis. These methods have been used in several previous research due to the change of topology after attacking and illustrating the methodology to apply with the case study networks for analyzing the reliability and critical. In addition, the network efficiency-based vulnerability, which measures the change of average shortest path both before and after a node or link attack, was usually selected to evaluate the vulnerability of the network (Sun and Guan; 2016, Noguchi and Fuse, 2020; Saadat et al., 2020), while the algebraic connectivity analysis can be considered to develop as the vulnerability criteria. In the example illustration, the related case study review is described by the following topics.

2.2 Railway Network Centrality Analysis

The centrality analysis is a well-known method for analyzing the network. The main purpose of this indicator is to evaluate the sensitivity of the node or link within the network and find the most critical or influential node or link, which has an extensive influence or effect if it is attacked or disrupted (Boudin, 2013; Rodrigues, 2019). Centrality analysis can divide into several methods, which depend on the objective and characteristics of the network. For example, degree centrality, closeness centrality, betweenness centrality, edge betweenness centrality, eigenvector centrality, and information centrality, as shown in Table 1 (Boudin, 2013; Rodrigues, 2019; Amrit and ter Maat, 2018). However, according to Rodrigues (2019), the degree of centrality directly depends on the number of nodes or links that connect the measured node, so this method can only illustrate the local centrality, not the real central node of the entire network. In the example, the network has at least two nodes with the same maximum degree of centrality.

Table 1. Characteristics of each type of network centrality

Centrality Type	**Characteristic**
Degree centrality	- Calculate the number of links that connect the measured node. - Simple method, but not accurate because it does not consider other factors, such as the distance of the shortest route or the influence of the neighbor node. - Show only the local influence node.
Closeness centrality	- Calculate from the shortest path between the measured node and all of the other nodes. - Can applies to flow analysis such as traffic and passenger flow. - Has a very narrow range of centrality value.
Betweenness centrality	- Consider by the shortest path between pair of nodes via measured link (or measured edge in the case of edge betweenness centrality). - Not calculate the distance of the shortest path directly but calculate the number of shortest paths. - Can applies to analyze the network flow easily with the assumption that most of the flow follows the shortest paths. For example, traffic flow and passenger flow. - Has been found in several transportation-related research works.
Eigenvector centrality	- Consider the number of links and also consider the influence of neighbor nodes. - Calculate by eigenvector-based equation. - It is usually used to analyze both transportation and social networks.
Information centrality	- Consider both the number of paths through the measured node and the lengths of the new path. - Calculate by the change in the global efficiency after removing every link that connects the measured node. - The global efficiency obtained by the average value of the shortage path in reciprocal form (Ek et al., 2015).

In recent years, many researchers used centrality analysis to evaluate the railway network. In the first example, Zhang et al. (2019) studied the metro network in Greater London to find critical nodes by using centrality indicators alongside with Node-place-design model, which analyzes the interaction between land use, transportation, and the walking friendliness around the station areas by multiple indexes. The network centrality helps to analyze the role and influence of critical stations after dividing all stations into the cluster groups by the model (classified by indexes value). Du et al. (2020) studied the passenger flow of the metro network in Shenzhen, China, at different times and on different specific days. The critical nodes on the special days, weekdays, and weekends were identified by in-degree centrality, out-degree centrality, weighted closeness centrality, weighted betweenness centrality, and eigenvector centrality. Liu et al. (2020) used degree centrality, betweenness centrality, and closeness centrality methods to identify the important station, then simulated attacking these stations to analyze the relative number of tolerable travel paths, relative travel efficiency, and passenger tolerability coefficient.

For link-based centrality analysis, Ando et al. (2020)'s research is a good example of link-based eigenvector centrality by using the value of the capacity-weighted eigenvector centrality equation between the pair of nodes from the measured link.

However, one of the most popular centrality analyses for railway networks is the betweenness centrality, which was first introduced by Freeman (1977) because it measures the number of the shortest path between two nodes via any measured node under the assumption that most flows follow the shortest distance (Boudin, 2013; Rodrigues, 2019). For this reason, several researchers applied betweenness centrality to analyze the networks with passenger flow, traffic flow, and freight flow (To, 2015; Sun and Guan, 2016; Mukherjee, 2017; Li et al., 2017; Zhu et al., 2018). For example, To (2015) used the betweenness centrality alongside closeness centrality, degree centrality, eigenvector centrality, and PageRank centrality (consider probability distribution that represents the likelihood of the node) to analyze Hong Kong's urban rail system. The conclusion illustrated that the betweenness centrality is the most important indicator because it refers to the extent of the node that considers the topologically center for the flow between other nodes and can therefore facilitate or impede the transmission of traffic or passenger. Mukherjee (2017) used the betweenness centrality to analyze the American Amtrack railway traffic. He explained that the betweenness centrality has a very important role when the rail traffic congestion is reaching its peak. Calzada-Infante et al. (2020) used closeness centrality alongside betweenness centrality to analyze the European international railway network and passenger transfers. After analyzing the geographical household, the mean closeness centrality directly depends on the distance threshold due to its definition based on the shortest distance. However, the betweenness centrality is still low value at any distance because its definition is based on route corridors and links between all the nodes in a corridor. Zhu et al. (2018) also used the betweenness centrality as one of the parameters to analyze the metro networks of five major cities. Li et al. (2017) used the betweenness centrality analysis based on the optimum path alongside other parameters to create the model regarding passenger flow in the Beijing Subway networks. Sun and Guan (2016) used the betweenness centrality and passenger betweenness centrality (PBC) measuring with passenger flow, purposed to find the most important node (station) within the Shanghai metro system. Its results help to scope the most important line that needs to consider and analyzed by the vulnerability specifically. Liu et al. (2017) used betweenness centrality alongside the degree centrality and connectivity reliability to evaluate rail network vulnerability. For the link-based centrality evaluation, the betweenness centrality-based method to analyze the link as the edge betweenness centrality was illustrated by Girvan and Newman (2002), who considered the edges that are least central for network communities can be the most between edge (Cuzzocrea et al., 2012).

In addition, the betweenness centrality has an important role in the highway network analysis, such as Mahajan and Kim (2020), who analyzed the vulnerability assessment of Alberta's provincial highway network by using betweenness alongside the remoteness index and accessibility index for evaluating the communities along the highway.

Another interesting centrality method is the information centrality analysis, which is based on the network efficiency analysis and evaluated by measuring the change of efficiency after removing every link that connected the measured node. This criterion was used by Amrit and ter Maat (2018) to study the information flow of people's communication and the relation with the other type of centralities such as betweenness, closeness, and eigenvector types. From the result, they explained that information centrality has properties more similar to eigenvector centrality. However, Latora and Machiori (2007) explained that information centrality is closely related to or corresponds with closeness centrality. In addition, Crucitti et al. (2006) used information centrality to analyze the urban street network and compared it with the other centrality analysis. Their result explained that the information centrality strongly corresponds with the betweenness centrality analysis.

This thesis aims to analyze multiple centralities alongside the other indicator to point to the most influential station in the network, and that station is also located in the critical rail section, then show the critical vulnerable area of the railway network in multiple views. From case studies, each centrality has different characteristics and can apply to various criteria or methods for transportation analysis. However, the optimal method depends on the user objective and research purpose.

2.3 Railway Network Efficiency Analysis

Network efficiency is analyzing the efficiency of communication or traffic within the network. If the distance between two nodes is higher, the communication efficiency between these nodes decreases. The efficiency can be obtained by the average value of the shortage path in invert form (Ek et al., 2015; Wu et al., 2016) and can be divided into two main subtypes. The global efficiency calculates the average value of the shortage path in invert form for the entire network and measures total efficiency (Wu et al., 2016). Another is the local efficiency that considers the neighborhood of the node and is related to the clustering coefficient (Wu et al., 2016). However, this thesis aims to analyze the critical link or node entire the network, so global efficiency is considered used for analysis in this research, especially the edge-based evaluation. In the rail transport field, this criterion is usually applied to evaluate network vulnerability by measuring global efficiency changes after an attack or by removing some of the specific nodes or links. The first example is the research of Chen et al. (2021), who evaluated the vulnerability assessment of transit systems, which is composed of the loss of transport capacity equation that is comprised of the transport efficiency and rate of the connected OD pairs. Both criteria components measure the change in global efficiency and the number of OD pairs after attacking the network. Sun et al. (2015) analyzed the Shanghai Metro network by using the global efficiency alongside the station vulnerability, which was also evaluated by measuring the change of global efficiency after attacking by removing nodes. The result showed the top ten critical stations when considering the reduction of the network serviceability, which was obtained from station vulnerability and other criteria, such as the platform passenger flow per hour. Zhang et al. (2018) also used global efficiency to evaluate the connectivity vulnerability of metro networks in Shanghai, Beijing, and Guangzhou. The result showed that the global efficiency of all three networks decreased if they removed more nodes in the network. Xiao et al. (2019) evaluated the travel efficiency vulnerability of the Beijing subway network, which considers the global efficiency change after the vertex or edge failure. Saadat et al. (2020) used the global efficiency-based vulnerability to analyze a metro network in Washington, D.C., if the important sections failed or cutoff. Noguchi and Fuse (2020) used global efficiency to analyze the critical station with or without turn-back operation of the Nagoya metro network and also measured its impact from disruption, and Jiao et al. (2020) applied the global efficiency variant to be service frequency-based network efficiency of the high-speed rail network. They evaluated the percentage decrease in network efficiency by measuring the change in efficiency after the failure. The trend showed that efficiency would decrease if the disruption duration increases when considered over 24 hours.

Many pieces of research also used centrality analysis, especially betweenness centrality, to help evaluate the network efficiency. The primary examples are Sun and Guan (2016), who used global efficiency combined with the average path length and passenger flow to evaluate the critical line and section, which is considered a risk or very sensitive to disruption. This research also used the passenger betweenness centrality to point to the critical line that needs to evaluate as the first priority too. Yin et al. (2016) used global efficiency alongside shortest paths and passenger flow-based betweenness centrality analysis to evaluate the subway's disruption. However, the main purpose of centrality analysis was to define the attack strategy by centrality rank. Shi et al. (2019) also used global efficiency-based vulnerability analysis alongside betweenness centrality and other indicators with a similar purpose. The result of both Yin et al. (2016) and Shi et al. (2019) was that if removing the node by the rank of betweenness centrality, the network efficiency will significantly decrease than random attack scenarios with the same number of removed nodes. Zhang and Thomas Ng (2022) analyzed the robustness of the urban railway network with the passenger flow. The robustness can be measured by the change of the connected component and operational efficiency, which is also based on global efficiency. For the attack simulation, several criteria, including closeness-based failure and betweenness-based failure, were used in the scenario. The result showed that the cascading failure during peak time in the morning and evening has more effect on the change of the connected component and operational efficiency than another period.

2.4 Algebraic Connectivity for Railway Network Analysis

The algebraic connectivity analysis is the indicator for evaluating the network connectivity level, which can define as the second smallest eigenvalue (Fiedler, 1973). The property of algebraic connectivity is sensitive to the nodes or links that failed, so its value will decrease if it removes some node or link. In recent years, some researchers studied the relation between algebraic connectivity and network topology change, such as Galvan and Agarwal (2020) tested the ten-node network example with several indicators, including algebraic connectivity. The result showed the star shape network, in which every node connected to the center node, had the highest value of algebraic connectivity, but the bat shape network, which had very few links to connect to the center node, had the lowest value. Yazdani et al. (2011) studied the relation between the algebraic connectivity and the percentage of added pipe length from various expansion scenarios. In the transportation field, algebraic connectivity has a role in analyzing air transportation networks. For example, Wei and Sun (2011) used the weighted algebraic connectivity applied to the flight route adjustment problems. In the rail transport analysis, there are some researchers who used algebraic connectivity analysis to evaluate the railway network. For instance, Wang et al. (2017) used algebraic connectivity analysis as a part of a multi-criteria robustness analysis to evaluate 33 metro networks from several major cities and then analyze the correlation between the criteria. Xu et al. (2020) also used algebraic connectivity alongside several other indicators to evaluate the connectivity of the Chinese high-speed railway network. The result showed that removing some edges would have an impact on the decrease of algebraic connectivity.

Because the sensitivity of the topological change is similar to network efficiency analysis, the algebraic connectivity can be developed into the algebraic connectivity-based vulnerability analysis, which purposes to evaluate the critical railway section that will have an extensive effect if it fails or is cut off. In addition, the algebraic connectivity-based vulnerability can also use the same algorithm as the network efficiency-based vulnerability if it needs a link or node-based attack scenario.

2.5 Stochastic Block Model

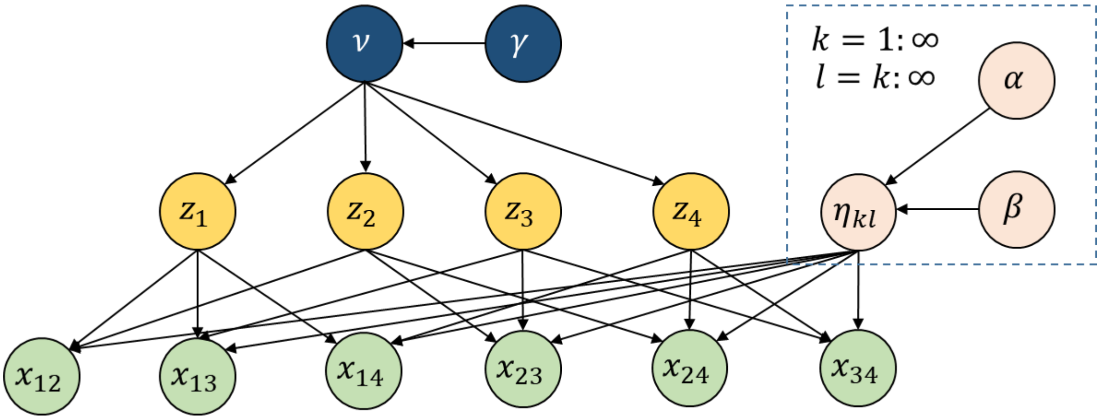


Figure 2. An example of an infinite relational model for network data with 4 clusters (Konishi et al., 2015)

The node-based cluster analysis helps to classify the group of nodes that have a relationship with each other or with nodes from other groups. One of the basic criteria is the infinite relational model (IRM). This model is a probabilistic model under the assumption that an object has owned cluster assignment and observed relations are generated on the basis of the assignment (Konishi et al., 2015). the concept example was illustrated by Konishi et al. (2015), who studied the cluster by infinite relational model, in which link probabilities are generated by the Beta distribution, as shown in Fig. 2.

From Fig. 2, η_{kl} is the link probability between cluster *k* and *l* that is obtained from the Beta distribution with α and β parameter, z_i is a vector of object i where only one element corresponding to a cluster is 1 and the other 0, β is the concentration parameter, and x_{ij} is a binary variable, which whether objects *i* and *j* link or not.

However, this research aims to analyze with the stochastic block model (SBM), which is a more advanced model that can analyze and identify the group of clusters or communities on a large scale, such as the group of railway stations in the network. According to Aicher et al. (2015), the measuring can obtain by analyzing the probabilistic of pairwise interactions vertices (node), while each vertex belongs to any cluster group or block. If any probability of any pair of nodes has a high value, that means both nodes come from the same group. Suppose pair of nodes come from different groups. In that case, the probability will be low value (Lee and Wilkinson, 2019) like the example shown in Fig. 3. In addition, Aicher et al. (2015) also used an advanced version, the weighted stochastic block model (WSBM), to analyze the node clusters with the edge weights obtained from any exponential family distribution.

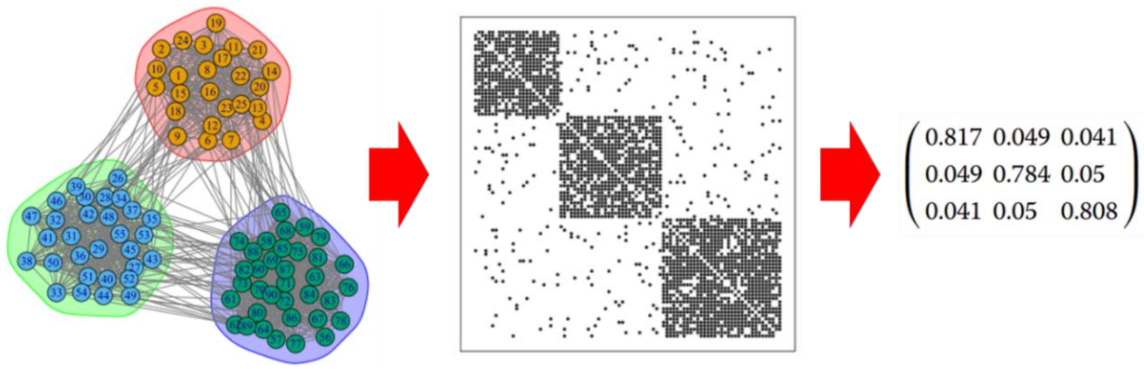


Figure 3. The example of clustering from the stochastic block model with the probability of each pair of edge blocks (Lee and Wilkinson, 2019)

In transportation field analysis, several examples applied with SBM analysis, such as Carlen et al. (2019), who used the time-dependent mixed-membership stochastic block model (TDMM-SBM) and time-dependent discrete stochastic block model (TDD-SBM) to analyze the role of bicycle sharing station and analyze the traffic both within or between a block of bicycle-sharing stations in the networks. The model was applied to the bicycle-sharing networks in Los Angeles, San Francisco, and New York; the result illustrated the role of riders both home and worked roles on each period. In addition, their model obtained the concepts from the work of Karrer and Newman (2011), who used a degree-corrected stochastic block model for analyzing the community structure.

In the network critical analysis, the stochastic block model can be used for analyzing the vulnerable railway section by considering the border between the cluster of station groups, which may be vulnerable to failure or cut-off.

2.6 The Position of the Research Study

This research studies the critical node or link of the railway network, which has an important role in network management, especially preventive planning under the risk of any failures on railway sections. The research objective is inspired by the rail transport disruption problem by natural disasters, man-made disasters, and technical errors such as earthquakes, flooding, derailing, rampage attack, and malfunction during operation. This issue extensively affected public transport, which passengers must change to travel via another route or use another transport mode. To countermeasure planning, it is necessary to evaluate the network critical before creating the preventive strategy. Moreover, the operators are facing the constraint of operation resources, such as materials, budget, labor, or equipment, that make it difficult to inspect, maintain or repair the entire network in the same period. In this thesis, several parameters are considered to help the operator manage priority to inspect, maintain or repair the track section or point on each station that is considered very important, critical, and has an extensive effect on the network.

One of the most frequently used basic criteria, the centrality analysis, is selected to find the critical node that can be conducted using several methods such as closeness centrality, eigenvector centrality, or betweenness centrality. However, the betweenness centrality is considered popular for analyzing railway networks due to its characteristics that fit for analyzing the flow under the concept that most flows follow the shortest path. For these reasons, this criterion is easy to apply to passenger or traffic flow as a future task. For the edge evaluation, the global efficiency and algebraic connectivity analyses are selected for evaluating the vulnerability by measuring the change of global efficiency or algebraic connectivity if it removes or attacks some links. In addition, the stochastic block model can use to analyze the vulnerable section by focusing on the section that connected two different clusters or communities in the network.

In this research, the main objective is to use multiple indicators for evaluating the railway network for analyzing the critical, especially the vulnerability of the railway network on multiple views, to scope the most important or vulnerable sections that need to classify as the high-priority to manage. The testing aims to be conducted on the three railway networks, the Kyushu railway, Tokyo subway, and Osaka subway networks. The Kyushu railway network, which is composed of the subway, commuter railway, and inter-city railway, was selected for three main reasons. First, the network is not too large like the Honshu railway network under the research time limit and not too spare compared to the Hokkaido and Shikoku railway network. Second is the high annual ridership from main operators in the region. Another reason, this network has been disrupted by natural disaster events in the past several years. The subway networks in Tokyo are considered for studying critical or vulnerable stations and sections of urban railway networks, which have much smaller sizes but are stronger robust, and denser, as well as the Osaka subway network, which has similar topology characteristics.

All three networks were studied in their topology in the past several years. For example, Wang et al. (2017) used algebraic connectivity and average efficiency analysis to evaluate 33 metro networks, including the Osaka and Tokyo metro networks. Murayama (1994) studied the change in the Japanese railway accessibility of cities, which is based on the minimum travel time between pairs of cities in the shortest path. The results showed that Fukuoka city in Kyushu has benefited from the expansion of the Shinkansen to reduce the accessibility difference. In another example, Wu et al. (2018) evaluated the performance of six metro networks, including the Tokyo metro network, by the node occupying probability and another criterion, including global efficiency. The node occupying probability is the centrality analysis based on the probability that a path from any pair of nodes passes through the measured node. After the attack simulation, the Tokyo subway network showed that it has the most robust under random attack simulation.

All railway network analysis purposes evaluating the critical nodes and links on both mainline and branch lines help the operator decide to manage the preventive strategies, especially the section with high vulnerability and connect the most centrality station to keep the network reliable.

References

[1] Aicher, C., Jacobs, A. Z., & Clauset, A. (2015). Learning latent block structure in weighted networks. *Journal of Complex Networks*, 3, 221–248. https://doi.org/10.1093/comnet/cnu026

[2] Amrit, C., & ter Maat, J. (2018). Understanding Information Centrality Metric: A Simulation Approach. (http://arxiv.org/abs/1812.01292; Accessed December 1, 2020)

[3] Ando, H., Bell, M., Kurauchi, F., Wong, K. I., & Cheung, K. F. (2020). Connectivity evaluation of large road network by capacity-weighted eigenvector centrality analysis. *Transportmetrica A: Transport Science*, 0(0), 1–27. https://doi.org/10.1080/23249935.2020.1804480

[4] Barbosa, F., de Sousa, A., & Agra, A. (2018). The Design of Transparent Optical Networks Minimizing the Impact of Critical Nodes. *Electronic Notes in Discrete Mathematics,* 64, 165–174. https://doi.org/10.1016/j.endm.2018.01.018

[5] Bıyıkoğlu, T., Leydold, J., & Stadler, P. F. (2007). *Laplacian Eigenvectors of Graphs*. New York: Springer.

[6] Boudin, F. (2013). A Comparison of Centrality Measures for Graph-Based Keyphrase Extraction. *International Joint Conference on Natural Language Processing*, 834–838.

[7] Calzada-Infante, L., Adenso-Díaz, B., & García Carbajal, S. (2020). Analysis of the European international railway network and passenger transfers. *Chaos, Solitons and Fractals,* 141, 110357. https://doi.org/10.1016/j.chaos.2020.110357

[8] Carlen, J., Pont, J. de D., Mentus, C., Chang, S.-S., Wang, S., & Porter, M. A. (2019). Role Detection in Bicycle-Sharing Networks Using Multilayer Stochastic Block Models. *arXiv,* (http://arxiv.org/abs/1908.09440 Accessed September 20, 2020)

[9] Chandra, S., Nguyen, H., & Nguyen, A. (2019). Evaluating critical gas pipeline crossings for freight truck routes. *Case Studies on Transport Policy, 7*, 680-688. https://doi.org/10.1016/j.cstp.2019.10.005

[10] Chen, H., Zhang, L., Liu, Q., Wang, H. & Dai, X. (2021). Simulation-based vulnerability assessment in transit systems with cascade failures. *Journal of Cleaner Production,* 295, 126441. https://doi.org/10.1016/j.jclepro.2021.126441

[11] Crucitti, P., Latora, V., & Porta, S. (2006). Centrality in networks of urban streets. *Chaos*, 16. https://doi.org/10.1063/1.2150162

[12] Cuzzocrea, A., Papadimitriou, A., Katsaros, D., & Manolopoulos, Y. (2012) Edge betweenness centrality: A novel algorithm for QoS-based topology control over wireless sensor networks. *Journal of Network and Computer Applications*, 35(4), 1210–1217. https://doi.org/10.1016/j.jnca.2011.06.001

[13] Divya, P. B., Lekha, D. S., Johnson, T. P., & Balakrishnan, K. (2022). Vulnerability of link-weighted complex networks in central attacks and fallback strategy. *Physica A: Statistical Mechanics and Its Applications,* 590. https://doi.org/10.1016/j.physa.2021.126667

[14] Du, Z., Tang, J., Qi, Y., Wang, Y., Han, C., & Yang, Y. (2020). Identifying critical nodes in metro network considering topological potential: A case study in Shenzhen city-China. *Physica A: Statistical Mechanics and Its Applications*, 539. https://doi.org/10.1016/j.physa.2019.122926

[15] Ek, B., VerSchneider, C., & Narayan, D. A. (2015). Global efficiency of graphs. *AKCE International Journal of Graphs and Combinatorics*, 12(1), 1–13. https://doi.org/10.1016/j.akcej.2015.06.001

[16] Fiedler, M. (1973). Algebraic connectivity of graphs, with applications. *Czechoslovak Mathematical*, 23(November), 298–305.

[17] Freeman, L. C. (1977). A Set of Measures of Centrality Based on Betweenness. Sociometry, 40(1), 35-41. https://doi.org/10.2307/3033543

[18] Galvan, G., & Agarwal, J. (2020). Assessing the vulnerability of infrastructure networks based on distribution measures. *Reliability Engineering and System Safety*, 196(November 2019), 106743. https://doi.org/10.1016/j.ress.2019.106743

[19] Girvan, M., & Newman, M. E. J. (2002) Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99(12), 7821–7826. https://doi.org/10.1073/pnas.122653799

[20] Günneç, D., & Salman, F. S. (2011). Assessing the reliability and the expected performance of a network under disaster risk. *OR Spectrum*, *33*(3), 499–523. https://doi.org/10.1007/s00291-011-0250-7

[21] Jiao, J., Zhang, F., & Liu, J. (2020). A spatiotemporal analysis of the robustness of high-speed rail network in China. *Transportation Research Part D: Transport and Environment*, 89, 102584. https://doi.org/10.1016/j.trd.2020.102584

[22] Karrer, B., & Newman, M. E. J. (2011). Stochastic blockmodels and community structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 83, 016107. https://doi.org/10.1103/PhysRevE.83.016107

[23] Kaye, T., Khatami, D., Metz, D., & Proulx, E., (2014). Quantifying and Comparing Centrality Measures for Network Individuals as Applied to the Enron Corpus. *SIAM Undergraduate Research Online*, 7. https://doi.org/10.1137/14s013202

[24] Khademi, N., Bababeik, M., & Fani, A. (2021). Sparse rail network robustness analysis: Functional vulnerability levels of accidents resulting from human errors.

Journal of Safety Science and Resilience, 2, 111–123. https://doi.org/10.1016/j.jnlssr.2021.07.001

[25] Konishi, T., Kubo, T., Watanabe, K., & Ikeda, K. (2015). Variational Bayesian Inference Algorithm for Infinite Relational Model of Network Data. IEEE Transaction on Neural Networks and learning System, 26(9), 2176-2181.

[26] Landex, A. (2012) Reliability of Railway Operation, *Annual Transport Conference at Aalborg University*, Aalborg, Denmark.

[27] Latora, V., & Marchiori, M. (2007). A measure of centrality based on network efficiency. *New Journal of Physics*, 9. https://doi.org/10.1088/1367-2630/9/6/188

[28] Lee, C., & Wilkinson, D. J. (2019). A review of stochastic block models and extensions for graph clustering. *Applied Network Science*, 4. https://doi.org/10.1007/s41109-019-0232-2

[29] Li, M., Wang, Y., & Jia, L. (2017) The modeling of attraction characteristics regarding passenger flow in urban rail transit network based on field theory. *PLoS ONE*, 12(9), e0184131. https://doi. org/10.1371/journal.pone.0184131

[30] Liu, J., Lu, H., Ma, H., & Liu, W. (2017). Network vulnerability analysis of rail transit plans in Beijing-Tianjin-Hebei region considering connectivity reliability. *Sustainability (Switzerland)*, 9, 1479. https://doi.org/10.3390/su9081479

[31] Liu, J., Peng, Q., Chen, J., & Yin, Y. (2020). Connectivity Reliability on an Urban Rail Transit Network from the Perspective of Passenger Travel. *Urban Rail Transit*, 6(1), 1-14. https://doi.org/10.1007/s40864-019-00117-z

[32] Lu, Q. C., Zhang, L., Xu, P. C., Cui, X., & Li, J. (2022). Modeling network vulnerability of urban rail transit under cascading failures: A Coupled Map Lattices approach. *Reliability Engineering and System Safety*, 221(August 2021), 108320. https://doi.org/10.1016/j.ress.2022.108320

[33] Mahajan, K., & Kim, A. M. (2020). Vulnerability assessment of Alberta’s provincial highway network. *Transportation Research Interdisciplinary Perspectives*, 6, 100171. https://doi.org/10.1016/j.trip.2020.100171

[34] Mukherjee, S. (2017) How good are network centrality measures? Longitudinal analysis of traffic in a railway network in the United States. *Indian Academy of Sciences–Conference Series*, 1(1), 1–8. https://doi.org/10.29195/iascs.01.01.0001

[35] Murayama, Y. (1994). The impact of railways on accessibility in the Japanese urban system. *Journal of Transport Geography*, 2(2), 87–100. https://doi.org/10.1016/0966-6923(94)90015-9

[36] Noguchi, H., & Fuse, M. (2020) Rethinking critical node problem for railway networks from the perspective of turn-back operation. *Physica A: Statistical Mechanics and Its Applications*, 558, 124950. https://doi.org/10.1016/j.physa.2020.124950

[37] Parajuli, J., & Haynes, K. E. (2018). Transportation network analysis in Nepal: a step toward critical infrastructure protection. *Journal of Transportation Security*, 11, 101–116. https://doi.org/10.1007/s12198-018-0194-0

[38] Pyrgidis, C. N. (2016) *Railway Transportation Systems*, Florida: CRC Press Taylor & Francis Group.

[39] Rodrigues, F. A. (2019) Network centrality: an introduction, *A Mathematical Modeling Approach from Nonlinear Dynamics to Complex Systems.* Vol.22, New York: Springer.

[40] Rodríguez-Núñez, E., & García-Palomares, J. C. (2014). Measuring the vulnerability of public transport networks. *Journal of Transport Geography*, Vol., 50–63. https://doi.org/10.1016/j.jtrangeo.2014.01.008

[41] Saadat, Y., Ayyub, B. M., Zhang, Y., Zhang, D., & Huang, H. (2020) Resilience-based strategies for topology enhancement and recovery of metrorail transit networks. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering,* 6(2), 04020017. https://doi.org/10.1061/ajrua6.0001057

[42] Santos, D., de Sousa, A., & Monteiro, P. (2018). Compact Models for Critical Node Detection in Telecommunication Networks. *Electronic Notes in Discrete Mathematics*, *64*, 325–334. https://doi.org/10.1016/j.endm.2018.02.007

[43] Schlechte, T., Borndörfer, R., Erol, B., Graffagnino, T., & Swarat, E. (2011). Micro-macro transformation of railway networks. *Journal of Rail Transport Planning and Management*, *1*(1), 38–48. https://doi.org/10.1016/j.jrtpm.2011.09.001

[44] Shi, J., Wen, S., Zhao, X., & Wu, G. (2019). Sustainable development of urban rail transit networks: A vulnerability perspective. *Sustainability (Switzerland),* 11, 1335. https://doi.org/10.3390/su11051335

[45] Shuang, Q., Zhang, M., & Yuan, Y. (2014). Node vulnerability of water distribution networks under cascading failures. *Reliability Engineering and System Safety*, 124, 132–141. https://doi.org/10.1016/j.ress.2013.12.002

[46] Sohn, J. (2006). Evaluating the significance of highway network links under the flood damage: An accessibility approach. *Transportation Research Part A: Policy and Practice*, 40, 491–506. https://doi.org/10.1016/j.tra.2005.08.006

[47] Sun, D. J., & Guan, S. (2016) Measuring vulnerability of urban metro network from line operation perspective. *Transportation Research Part A: Policy and Practice*, 94, 348–359. https://doi.org/10.1016/j.tra.2016.09.024

[48] Sun, D., Zhao, Y., & LU, Q. C. (2015). Vulnerability analysis of urban rail transit networks: A case study of Shanghai, China. *Sustainability (Switzerland),* 7, 6919–6936. https://doi.org/10.3390/su7066919

[49] To, W. M. (2015) Centrality of an urban rail system. *Urban Rail Transit*, 1(4), 249–256. https://doi.org/10.1007/s40864-016-0031-3

[50] Vromans, M. (2005) *Reliability of Railway Systems*, ERIM Ph.D series Research in Management 62, Erasmus Research Institute of Management (ERIM), Erasmus University, Rotterdam, Netherlands.

[51] Wang, X., Koç, Y., Derrible, S., Ahmad, S. N., Pino, W. J.A., & Kooij, R. E. (2017) Multi-criteria robustness analysis of metro networks. *Physica A: Statistical Mechanics and Its Applications*, 474, 19–31. https://doi.org/10.1016/j.physa.2017.01.072

[52] Wei, P., & Sun, D. (2011). Weighted algebraic connectivity: An application to airport transportation network. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 44, 13864–13869. https://doi.org/10.3182/20110828-6-IT-1002.00486

[53] Wu, X., Dong, H., Tse, C. K., Ho, I. W. H., & Lau, F. C. M. (2018). Analysis of metro network performance from a complex network perspective. *Physica A: Statistical Mechanics and Its Applications,* 492, 553–563. https://doi.org/10.1016/j.physa.2017.08.074

[54] Wu, X., Tse, C. K., Dong, H., Ho, I. W. H., & Lau, F. C. M. (2016). A Network Analysis of World's Metro Systems. *2016 International Symposium on Nonlinear Theory and Its Applications,* 606-609.

[55] Xiao, X., Jia, L., Wang, Y., & Zhang, C. (2019) Topological characteristics of metro networks based on transfer constraint. *Physica A: Statistical Mechanics and Its Applications*, 532, 121811. https://doi.org/10.1016/j.physa.2019.121811

[56] Xu, Z., Zhang, Q., Chen, D., & He, Y. (2020) Characterizing the connectivity of railway networks. *IEEE Transactions on Intelligent Transportation Systems*, 21(4), 1491–1502. https://doi.org/10.1109/TITS.2019.2909120

[57] Yazdani, A., Otoo, R. A., & Jeffrey, P. (2011). Resilience enhancing expansion strategies for water distribution systems: A network theory approach. *Environmental Modelling and Software*, 26, 1574–1582. https://doi.org/10.1016/j.envsoft.2011.07.016

[58] Ye, Q., & Kim, H. (2019). Assessing network vulnerability using shortest path network problems. *Journal of Transportation Safety and Security*, 13(1), 1-25. https://doi.org/10.1080/19439962.2019.1600626

[59] Yin, H., Han, B., Li, D., & Wang, Y. (2016). Evaluating Disruption in Rail Transit Network: A Case Study of Beijing Subway. *Procedia Engineering*, 137, 49–58. https://doi.org/10.1016/j.proeng.2016.01.233

[60] Zhang, J., Wang, S., & Wang, X. (2018) Comparison analysis on vulnerability of metro networks based on complex network. *Physica A: Statistical Mechanics and Its Applications*, 496, 72–78. https://doi.org/10.1016/j.physa.2017.12.094

[61] Zhang, Y., & Thomas Ng, S. (2022**).** Robustness of urban railway networks against the cascading failures induced by the fluctuation of passenger flow. *Reliability Engineering and System Safety,* 219, 108277. https://doi.org/10.1016/j.ress.2021.108227

[62] Zhang, Y., Marshall, S., & Manley, E. (2019). Network criticality and the node-place-design model: Classifying metro station areas in Greater London. *Journal of Transport Geography*, *79*(January), 102485. https://doi.org/10.1016/j.jtrangeo.2019.102485

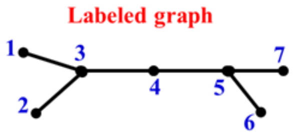
Chapter 3

Basic Theories of Network Analysis

The main basic theories for this research are the graph theory, eigenvector theories, centralities, algebraic connectivity, global efficiency, and stochastic block model. The graph theory is the basic theory necessary for calculating every criterion in this research. The eigenvector theory is also important in this research due to the basis of finding the value of eigenvector centrality and algebraic connectivity. In addition, the stochastic block model can help to support the vulnerability analysis by identifying the link located on the border between blocks or clusters.

3.1 Graph Theory

The graph Laplacians is the part of graph theory that studies the topology of graphs, which are mathematic structures and purpose to model the pairs of two connected objects and then model networks on a larger scale. The graph composes of several nodes (also called vertexes or points) that are linked or connected by links (also called edges or lines).



Degree matrix

$$\begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&0&0&0\\ 0&0&3&0&0&0&0\\ 0&0&0&2&0&0&0\\ 0&0&0&0&3&0&0\\ 0&0&0&0&0&1&0\\ 0&0&0&0&0&0&1 \end{bmatrix}$$

Adjacency matrix

$$\begin{bmatrix} 0&0&1&0&0&0&0\\ 0&0&1&0&0&0&0\\ 1&1&0&1&0&0&0\\ 0&0&1&0&1&0&0\\ 0&0&0&1&0&1&1\\ 0&0&0&0&1&0&0\\ 0&0&0&0&1&0&0 \end{bmatrix}$$

Laplacian matrix

$$\begin{bmatrix} 1&0&-1&0&0&0&0\\ 0&1&-1&0&0&0&0\\ -1&-1&3&-1&0&0&0\\ 0&0&-1&2&-1&0&0\\ 0&0&0&-1&3&-1&-1\\ 0&0&0&0&-1&1&0\\ 0&0&0&0&-1&0&1 \end{bmatrix}$$

Figure 4. The simple example of a labeled graph, degree matrix, adjacency matrix, and Laplacian matrix

The graph Laplacians represent a graph in square matrix form, which has the benefit of applying and analyzing several issues, such as calculating the algebraic connectivity or the number of spanning trees in Kirchhoff's theorem. On the basis property, the graph is usually represented as the ordered pair of $G(V, E)$ with node set $V = \{1, \dots, n\}$, link set $E = \{1, \dots, m\}$, and G as a graph with n nodes and m links (Bıyıkoğlu, et al, 2007; Keller, 2019).

The Laplacian of G can illustrate by the matrix, as illustrated in Fig. 4 that

$$L(G) = D(G) - A(G) \tag{3.1}$$

where $L(G)$ is the Laplacian of graph G, $D(G)$ is the diagonal degree matrix whose entries are the degrees of the vertices, which show the number of edges that connected each node, and $A(G)$ denotes the adjacency matrix of G, which identify the linked neighbors of each node by a_{ij} = 1 if node v_i is linked to node v_j, and a_{ij} = 0 otherwise.

In the Laplacian matrix $L(G)$, the diagonal elements between two nodes l_{ij} of L are equal to the degree of node v_i, and off-diagonal elements, which are obtained from the adjacency matrix, l_{ij} are -1 if node v_i is adjacent to v_j and 0 otherwise, so it can conclude by

$$l_{ij} = \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is connect to } v_j \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

where deg (v_i) is the degree of node i.

3.2 Eigenvalue Theory

The eigenvalue is the nonzero linear vector, which can be changed by the scalar factor and has an important role in dynamic problems. According to Smith (1985), the basic definition can be illustrated by the equation

$$Ax = \lambda x \tag{3.3}$$

When x denotes the *eigenvector* of the square matrix A, while λ denotes the *eigenvalue*. If we give A^n, n = 2, 3, …, the eigenvalue will be exponent as λ^n, which means $A^n x = \lambda^n x$ for n = 2, 3, …

From Eq. 3.3, if we need the basic solution of eigenvalue and eigenvector with N different eigenvalues, the example can be illustrated by

$$Ax_i = \lambda_i x_i, \quad i = 1, 2, \dots, N \tag{3.4}$$

then written in matrix form as

$$\begin{aligned} A[x_1, x_2, \dots, x_N] &= [Ax_1, Ax_2, \dots, Ax_N] \\ &= [\lambda_1 x_1, \lambda_2 x_2, \dots, \lambda_N x_N] \\ &= [x_1, x_2, \dots, x_N] \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_N \end{bmatrix} \end{aligned}$$

From the matrix equation, the eigenvector can be represented as $[x_1, x_2, \dots, x_N] = X$, and usually called the model matrix, then

$$\begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_N \end{bmatrix} = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_N] = \lambda_D$$

In conclusion, the solution can be illustrated in a compact equation as

$$X^{-1}AX = \lambda_D \tag{3.5}$$

then $X^{-1}A^nX = \lambda_D^n$= diag$[\lambda_1^n, \lambda_2^n, \ldots, \lambda_N^n]$, n = 2, 3, …

In addition, the eigenvalue λ can be a solution to the characteristic equation (Stoica and Moses, 1997) of

$$|A - \lambda I| = 0 \tag{3.6}$$

In this research, the eigenvalue equation, as shown in Eq. 3.3 – 3.5 can be applied to analyze the eigenvalue centrality to find the most influential node in the network. It can also use to calculate the algebraic connectivity, which measures the level of the network connectivity and then is applied to evaluate the network vulnerability.

3.3 Centrality Theories

The centrality analysis is the criteria to evaluate the node or link within the network for measuring the importance or influence. Suppose any node or link has a high value of centrality. In that case, it means that the node or link is very important or critical, which has an extensive effect on the traffic or communication within the network if it is removed or disrupted from any disasters or emergency cases.

This research aims to compare several variants of centrality analysis to study the results, characteristics, and performance of each type with railway network case studies. The centrality methods for this study are composed of degree centrality, closeness centrality, eigenvector centrality, betweenness centrality, and information centrality (Boudin, 2013; Rodrigues, 2019; Amrit and ter Maat, 2018).

3.3.1 Degree centrality

Degree centrality is a simple method to measure the network's centrality, which defines by the number of links or connections of each node (Rodrigues, 2019). From the adjacency matrix of the graph theory, degree centrality (C_D) can illustrate the basis equation by

$$C_D(i) = \deg(a_{ij}) \quad , i \neq j;\ i, j \in G. \tag{3.7}$$

where $\deg(a_{ij})$ is the degree of each node (number of links of each node), which can also calculate and obtained from the degree matrix. However, this method cannot efficiently identify specifics or critical nodes because it is a high possibility to have several nodes with the same degree in the network, including nodes that have the highest degree (Rodrigues, 2019).

3.3.2 Closeness centrality

Closeness centrality measures the weight of node important by calculating the distance or length of the shortest paths between the measured node and all the other nodes in the network. If any node has a lower total distance to all other nodes, it can consider a more centrality node (Boudin, 2013; Rodrigues, 2019). This method is easy to analyze the flow or traveling through the shortest path and analyze the hub of network traveling. However, the weak point of this criteria is it has a very narrow range of variation due to the small diameter of networks (Rodrigues, 2019).

The basic equation of closeness centrality (C_C) can define by

$$C_C(i) = \frac{n-1}{\sum_i^n \sum_j^n d_{ij}} \qquad , i \neq j;\ i,j \in G. \tag{3.8}$$

where d_{ij} is the shortest distance between nodes i and j, which is calculated by the number of nodes in the patch connecting both of them, and n is the number of nodes in the graph G (Rodrigues, 2019).

3.3.3 Eigenvector centrality

Eigenvector centrality can measure the network centrality by considering the score or weight of every node, which depends on both the number of linked neighbors and the quality of its connections. To identify the critical node and check the importance of all other nodes within the network, if any node has a connection to higher score or weight nodes, it can consider more important or influence nodes (Boudin, 2013; Rodrigues, 2019).

The algorithm of the eigenvector centrality can describe by the following processes.

1) Beginning by assigning a centrality score of 1 for all nodes (Kaye et al., 2014).

2) Calculate eigenvector centrality score (C_E) Every node by an equation of

$$C_{Ek}(i) = \frac{1}{\lambda}\sum_{j\in N(i)} w_{ij} C_{Ek+1}(i) = \frac{1}{\lambda}\sum_{j\in N(i)} a_{ij} C_{Ek+1}(i) \tag{3.9}$$

where w_{ij} is the weight of links between nodes i and j, a_{ij} is the element of the adjacency matrix that links nodes i and j, $N(i)$ is the set of nodes connected to j, λ is the proportionally constant, and k is the iteration number (Boudin, 2013). In addition, a small rearrangement can represent a vector form of the eigenvector equation $A(G)C_{Ek+1} = \lambda C_E$

while $A(G)$ is the adjacency matrix of graph G, and C_{E1} is obtained from the beginning centrality score of 1 from 1).

However, to keep entry values in the eigenvector non-negative, the Perron–Frobenius theorem will also use for keeping this condition. This theorem is definite by letting A is a non-negative and irreducible matrix. Then there exists a simple eigenvalue $\lambda > 0$, which has an associated positive eigenvector (Borobia and Trias, 1992).

3) For defining an absolute score, the normalization C_E will be considered by calculating with the Power-iteration method. The calculating begin by starts with an eigenvector C_{Ek+1} of a network graph at the end of the $(k+1)^{\text{th}}$ iteration is given by

$$C_{Ek+1} = \frac{A(G)C_{Ek}}{\|A(G)C_{Ek}\|} \tag{3.10}$$

where $\|A(G)C_{Ek}\|$ is the normalized value.

4) From 3), the initial value of C_{Ek} has a column vector of all 1 at the beginning. This process needs to continue interaction repeatedly until the normalized value of $\|A(G)C_{Ek+1}\| = \|A(G)C_{Ek}\|$. In other words, the normalized value converges to the previous normalized value. The final value of the eigenvector can conclude as the eigenvector centrality of the graph.

3.3.4 Betweenness centrality

Betweenness centrality can identify the most important node or link by measuring the number of the shortest path between two nodes via any measured node under the assumption that most flow follows the shortest distance, such as passengers or traffic flow (To, 2015). This criterion can easily analyze any flow within the network compared with degree centrality and eigenvalue centrality, which mainly focus on the influence among the nodes (Boudin, 2013; Rodrigues, 2019). The node-based betweenness centrality (C_B) equation can be illustrated by (Kaye et al., 2014).

$$C_B(i) = \sum_{i \neq j \neq k \in G} \frac{\sigma_{jk(i)}}{\sigma_{jk}} \tag{3.11}$$

where σ_{jk} is the number of shortest paths from node j to node k, and $\sigma_{jk(i)}$ define as the number of shortest paths between node j and node k via measured node i.

The edge betweenness centrality ($C_{EB}(e)$) equation of edge e can be illustrated by (Cuzzocrea et al., 2012; Golbeck, 2015)

$$C_{EB}(e) = \sum_{i \neq j \in G} \frac{\sigma_{ij(e)}}{\sigma_{ij}} \qquad , e \in E, \tag{3.12}$$

where σ_{ij} is the number of shortest paths from node *i* to node *j*, and $\sigma_{ij(e)}$ is the number of shortest paths between nodes *i* and *j* via edge *e*.

3.3.5 Information centrality

Information centrality is the criteria that purpose for measuring the most important or influential node. The concept is that flow of information or traffic along the network paths is considered important (Amrit and ter Maat, 2018). The nodal evaluation can be obtained by measuring the network efficiency (global efficiency) from the node deactivation or removal, as shown in Fig. 5. In other words, the centrality can be measured by the change in global efficiency after removing every edge that connects the measured node (Latora and Machiori, 2007; Fortunato et al., 2004; Wang et al., 2008; Porta et al., 2010).

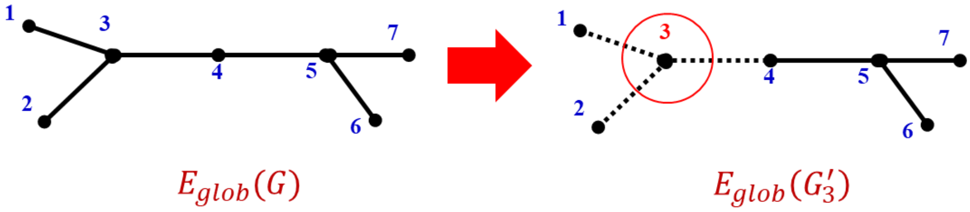


Figure 5. The example of the information centrality measuring of node 3, the value can obtain by the difference in global efficiency of the entire network ($E_{glob}(G)$) and global efficiency after removing every link that connects node 3 ($E_{glob}(G_3')$)

The global efficiency (E_{glob}), which evaluates the communication within the network, can calculate by the average value of the shortage path in reciprocal form (Ek et al., 2015) as the following equation.

$$E_{glob} = \frac{1}{n(n-1)} \sum_i^n \sum_j^n \frac{1}{d_{ij}} \quad , i \neq j;\ i, j \in G. \tag{3.13}$$

where *G* is the graph of the network, which is composed of *n* nodes, and d_{ij} is the shortest path between nodes *i* and *j*.

For evaluating the information centrality ($C_I(i)$) of node *i*, the basic equation can be concluded by

$$C_I(i) = \frac{E_{glob}(G) - E_{glob}(G_i')}{E_{glob}(G)} \tag{3.14}$$

where $E_{glob}(G)$ is the global efficiency of the entire network before removing any link, and $E_{glob}(G_i')$ is the global efficiency after removing every link that connected node *i*.

From Eq. 3.14, if the information centrality of any node is higher, it means the node is more important or has a stronger influence on the network.

Its definition and calculating method show that information centrality depends not only on the number of geodesics passing node i but also on the lengths of the new geodesics (Latora and Machiori, 2007).

3.4 Algebraic Connectivity

The algebraic connectivity is an eigenvalue-based graph connectivity indicator, which was first introduced by Miroslav Fiedler in 1973 (Fiedler, 1973) and purposed to measure the volume of connection or robustness of the overall graph network. The algebraic connectivity value can be obtained by the second smallest eigenvalue, which corresponds normalized eigenvector known as the Fiedler vector. The basic definition is illustrated by letting $G(V, E)$ be the graph with node set $V = \{1, \dots, n\}$ and link set $E = \{1, \dots, m\}$. $A(G)$ is the adjacency matrix of graph G where a_{ij} = 1 if node v_i is linked to node v_j, and a_{ij} = 0 otherwise, as has been described in section 3.1. In addition, $L(G)$ is the Laplacian matrix that is obtained by $D(G) - A(G)$, where $D(G)$ is the degree matrix.

According to Spiers et al. (2012) and Wei et al. (2014), if the element $e = (1, \dots, 1) \in \mathbb{R}^n$ and W is a set of column eigenvectors x that $W = \{x \in \mathbb{R}^n | \|x\| = 1, e^T x = 0\}$, the basic equation of the second smallest eigenvalue (λ_2) can be obtained as

$$\lambda_2 = \min_{x \in W} x^T L(G) x. \tag{3.15}$$

From Eq. 3.15, the algebraic connectivity can be denoted as $\lambda_2(G)$. If we measure the connectivity of graph G, the result can be divided the condition into two cases

$$\lambda_2(G) = 0 \quad \text{if and only if } G \text{ is not connected (Lal et al., 2011)} \tag{3.16}$$
$$\lambda_2(G) = n \quad \text{if and only if } G \text{ is connected} \tag{3.17}$$

From the condition of Eq. 3.17, if the network still connected every node to each other as an undirect graph, the algebraic connectivity still has a positive value and may be higher if the connected edges are increased and make the network denser. However, algebraic connectivity can be a negative value in a general directed graph. If the edge is cut off and makes the network is separated at least into two parts, the algebraic connectivity will drop to zero as the condition in Eq. 3.16. Each condition shows that the algebraic connectivity is sensitive to a separated or weak connected network.

In the case of the network with passenger flow analysis, we can illustrate each weight-link, which represents the number of passengers/day between each pair of stations (w_{ij}), into the weighed element of the adjacency matrix, then calculate the algebraic connectivity to analyze the vulnerability (Wei and Sun, 2011).

The weighted elements adjacency matrix ($a_{w,ij}$) can be described by

$$a_{w,ij} = \begin{cases} w_{ij} & \text{if node } i \text{ and } j \text{ connected with the passenger flow between both nodes} \\ 0 & \text{if node } i \text{ and } j \text{ are not connected} \end{cases} \tag{3.18}$$

then weighed Laplacian matrix element ($l_{w,ij}$) can describe by

$$l_{w,ij} = \begin{cases} -a_{w,ij} & \text{if } i \neq j \text{ and node } i \text{ and } j \text{ are connected} \\ \sum_{i=1}^{n} a_{w,ij} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{3.19}$$

From Eq.3.15 and 3.19, if we definite $L_w(G)$ as the passenger-weighted Laplacian matrix, the passenger-weighted algebraic connectivity ($\lambda_{2,w}$) can calculate by

$$\lambda_{2,w} = \min_{x \in W} x^T L_w(G) x. \tag{3.20}$$

However, because of the difficulty of obtaining the passenger flow data, especially from the inter-operator interchange stations, this research will be testing the passenger-weight algebraic connectivity only on the simple version of the JR Kyushu network.

3.5 Global Efficiency

Global efficiency is the part of network efficiency, which is based on the average shortest path in inverse from (Ek et al., 2015). These criteria are used to analyze the network's topological efficiency, which affects communication or traffic. The basic equation can be illustrated by

$$E_{glob} = \frac{1}{n(n-1)} \sum_i^n \sum_j^n \frac{1}{d_{ij}} \quad , i \neq j;\ i, j \in G. \tag{3.21}$$

where G is the graph of the network, which is composed of n nodes, and d_{ij} is the shortest path between nodes i and j.

If the efficiency has a high value, it means the network has strong robustness. However, if the edge has been removed, the efficiency will be changed following the change of the network topology. For this reason, this research chooses global efficiency to compare with algebraic connectivity for the vulnerability evaluation role.

3.6 Stochastic Block Model Basic Theory

The Stochastic Block Model (SBM) purpose is to model the community structure within the networks, especially the group of communities and the interaction between these groups. According to Aicher et al., 2015, the measuring can obtain by analyzing the probabilistic of pairwise interactions n vertices (node), while each vertex i belonging to any K cluster group or block can denote as. The vertices connection of the group's element between i and j can measure by the probability of the link, which is also obtained from the element of the adjacency matrix A between i and j (a_{ij}). From the $K \times K$ cluster matrix in the example of Fig. 3, if any probability of any pair of nodes has a high value, that means both nodes come from the same group. If pair of nodes come from different groups, the probability will be low value (Lee and Wilkinson, 2019).

The basic SBM likelihood equation of the group memberships of vertices i and j can be definite by

$$\mathrm{P}(A|z,\theta) = \prod_{ij} \theta_{z_i z_j}^{a_{ij}} \left(1 - \theta_{z_i z_j}\right)^{1\,-\,a_{ij}} \tag{3.22}$$

which can rewrite as

$$\mathrm{P}(A|z,\theta) = \prod_{ij} exp\left(a_{ij}\log\left(\frac{\theta_{z_i z_j}}{1-\theta_{z_i z_j}}\right) + \log\left(1 - \theta_{z_i z_j}\right)\right) \tag{3.23}$$

Given the vector obtained from the cluster of each vertex $z_i \in \{1, \dots, K\}$ and existence probability θ can illustrate the likelihood function that is based on the assumption of edges being Bernoulli distributed conditional on the group membership.

References

[1] Aicher, C., Jacobs, A. Z., & Clauset, A. (2015). Learning latent block structure in weighted networks. *Journal of Complex Networks*, 3, 221–248. https://doi.org/10.1093/comnet/cnu026

[2] Amrit, C., & ter Maat, J. (2018). Understanding Information Centrality Metric: A Simulation Approach. (http://arxiv.org/abs/1812.01292; Accessed November 2, 2020)

[3] Bıyıkoğlu, T., Leydold, J., & Stadler, P. F. (2007). *Laplacian Eigenvectors of Graphs.* Springer, New York.

[4] Borobia, B. & Trias, U. R. (1992). A Geometrie Proof of the Perron-Frobenius Theorem. *REVISTA MATEMATICA de la*, Volumen 5. Número 1.

[5] Boudin, F. (2013). A Comparison of Centrality Measures for Graph-Based Keyphrase Extraction. *International Joint Conference on Natural Language Processing*, pp. 834–838.

[6] Cuzzocrea, A., Papadimitriou, A., Katsaros, D., & Manolopoulos, Y. (2012) Edge betweenness centrality: A novel algorithm for QoS-based topology control over wireless sensor networks. *Journal of Network and Computer Applications*, 35(4), 1210–1217. https://doi.org/10.1016/j.jnca.2011.06.001

[7] Ek, B., VerSchneider, C., & Narayan, D. A. (2015). Global efficiency of graphs. *AKCE International Journal of Graphs and Combinatorics*, 12(1), 1–13. https://doi.org/10.1016/j.akcej.2015.06.001

[8] Fiedler, M. (1973). Algebraic connectivity of graphs, with applications. *Czechoslovak Mathematical*, 23(November), 298–305.

[9] Fortunato, S., Latora, V., & Marchiori, M. (2004). Method to find community structures based on information centrality. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, Vol. 70, p. 13. https://doi.org/10.1103/PhysRevE.70.056104

[10] Golbeck, J. (2015) *Introduction to Social Media Investigation*. Syngress, Massachusetts.

[11] Kaye, T., Khatami, D., Metz, D., & Proulx, E. (2014). Quantifying and Comparing Centrality Measures for Network Individuals as Applied to the Enron Corpus. *SIAM Undergraduate Research Online*, 1–20. (https://www.siam.org/students/siuro/vol7/S01320.pdf Accessed December 2, 2020)

[12] Keller, O. (2019). LP min-cut max-flow presentation. (http://u.cs.biu.ac.il; Accessed July 17, 2019)

[13] Lal, A. K., Patra, K. L., & Sahoo, B. K. (2011). Algebraic Connectivity of Connected Graphs with Fixed Number of Pendant Vertices. Graphs and Combinatorics, 27, 215–229. https://doi.org/10.1007/s00373-010-0975-0

[14] Latora, V., & Marchiori, M. (2007). A measure of centrality based on network efficiency. *New Journal of Physics*, 9. https://doi.org/10.1088/1367-2630/9/6/188

[15] Lee, C., & Wilkinson, D. J. (2019). A review of stochastic block models and extensions for graph clustering. *Applied Network Science*, 4. https://doi.org/10.1007/s41109-019-0232-2

[16] Porta, S., Latora, V. & Strano, E. (2010). *Networks Science*, Springer, London.

[17] Rodrigues, F. A. (2019) Network centrality: an introduction, *A Mathematical Modeling Approach from Nonlinear Dynamics to Complex Systems.* Vol.22, New York: Springer.

[18] Smith, G. D. (1985). *Numerical solution of partial differential equations: finite difference methods.* 3rd Ed. Oxford University Press, New York. p132-133.

[19] Spiers, G., Wei, P., & Sun, D. (2012). Algebraic connectivity optimization of large scale and directed air transportation network. *IFAC Proceedings Volumes (IFAC-Papers Online)*, 45(24), 103–109. https://doi.org/10.3182/20120912-3-BG-2031.00019

[20] Stoica, P., & Moses, R. L. (1997). *Introduction to Spectral Analysis.* Prentice Hall, New Jersey.

[21] To, W. M. (2015) Centrality of an urban rail system. *Urban Rail Transit*, 1(4), 249–256. https://doi.org/10.1007/s40864-016-0031-3

[22] Wang, X., Shen, Y. & Luan, E. (2008). A measure of centrality based on modularity matrix. *Progress in Natural Science*, 18. https://doi.org/10.1016/j.pnsc.2008.03.015

[23] Wei, P., & Sun, D. (2011). Weighted algebraic connectivity: An application to airport transportation network. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 44, 13864–13869. https://doi.org/10.3182/20110828-6-IT-1002.00486

[24] Wei, P., Chen, L., & Sun, D. (2014). Algebraic connectivity maximization of an air transportation network: The flight routes' addition/deletion problem. *Transportation Research Part E: Logistics and Transportation Review*, 61, 13–27. https://doi.org/10.1016/j.tre.2013.10.008

Chapter 4

Case Study Railway Networks

In the research planning, the network will transform into the graph networks for analysis by definite under the following condition.

1) Two or more stations, which passengers can interchange, will be assumed to combine as one node and represent the junction.

2) Two or more different operator stations, which passengers can interchange with each other, will also be combined as the same node.

3) Some railway lines which share the same track section will be considered as the same edge except high-speed railway lines.

4) The station on the branch section, which is connected to other railway networks that exclude a case study, will be assumed as the last station on that line.

This research conducted three main case studies. First is the Kyushu railway network, which covers inter-city and urban railway networks. The second is the Tokyo subway network for analyzing the denser network. Furthermore, another case study is the Osaka subway network.

4.1 Case Study of Kyushu Railway Network

A case study of Kyushu railway networks was selected to analyze the centralities, vulnerabilities, and stochastic block model for three main reasons. First, the network is not large for consume research time too much, but not too spare when compared with the Hokkaido or Shikoku railway networks. The second reason is the railway network in Kyushu is composed of three main operators, Kyushu Railway Company or JR Kyushu, Nishitetsu railway, and Fukuoka city subway, each operator has an annual ridership of approximately 352 million (in 2019) (JR Kyushu Railway Company, 2020), 106 million (in 2019) (Nishitetsu, 2020), and 173 million (in 2019) (Fukuoka City Subway, 2020) passengers respectively. Another reason is that the railways in Kyushu have been cut off or disrupted by several natural disasters event. For example, the Kumamoto earthquake in 2016 caused a landslide and cut off the railway section of the JR Kyushu Hōhi Main Line (Kiyasu, 2017), and the flooding in 2020 destroyed the railroad bridge on the JR Kyushu Kyudai Main Line (The Asahi Shimbun, 2020).

The network comprises several rail transport functions, such as subway, commuter railway, and inter-city railway, including the Shinkansen line that has several interchange stations expected to show the most influence stations and vulnerable sections. From Fig. 6, a case study network from all three operators covers more than 2,400 km (JR Kyushu Railway Company, 2020; Nishitetsu, 2020; Fukuoka City Subway, 2020). The railway, which all three operators own, was highlighted in the yellow line.

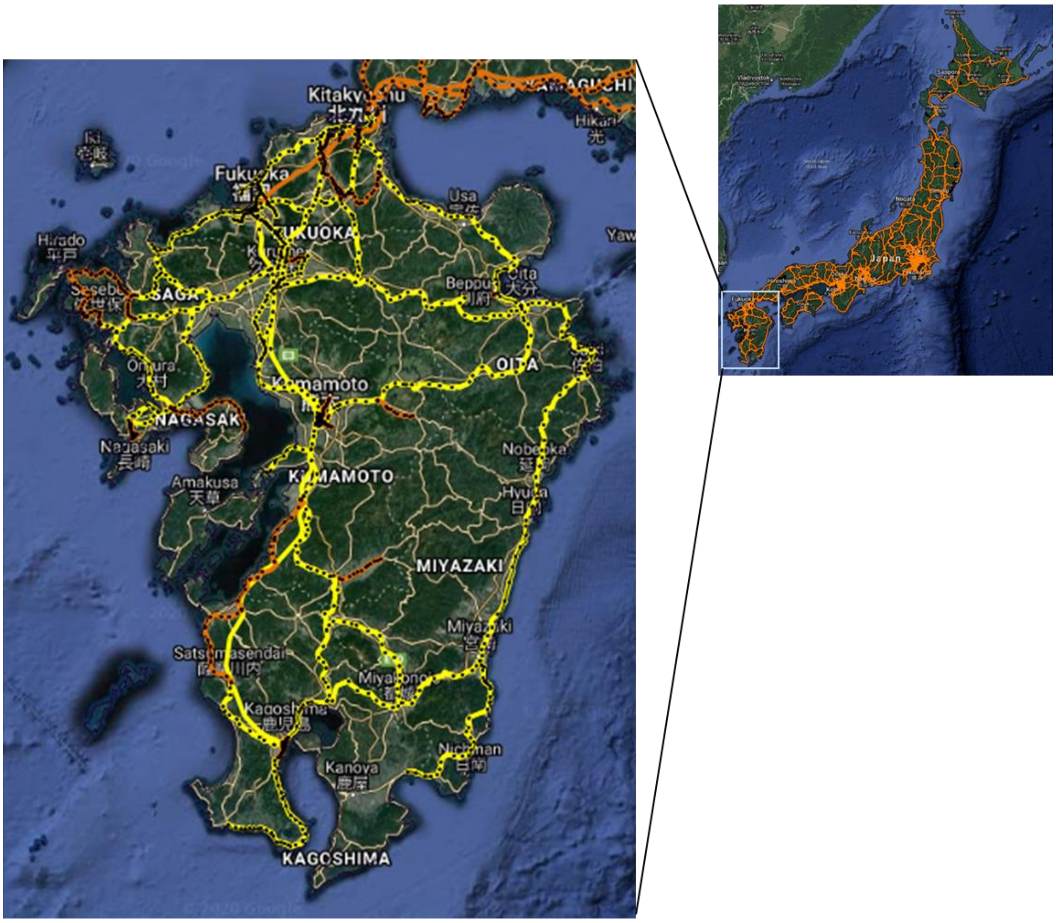


Figure 6. A case study railway network in Kyushu, which compose of JR Kyushu, Nishitetsu railway, and Fukuoka city subway network in yellow highlight (Google Maps, 2020; National Land Information Division, National Spatial Planning and Regional Policy Bureau, MLIT of Japan, 2020)

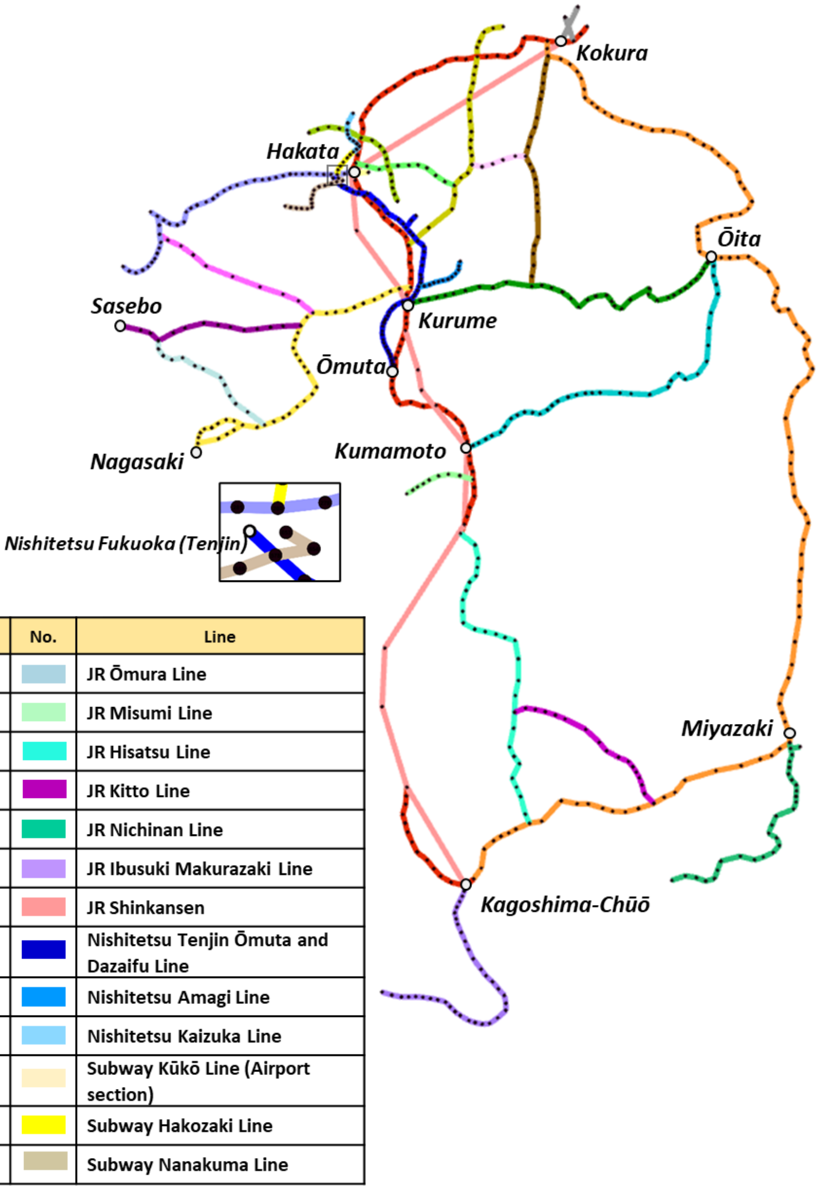
To apply to the graph form, we add some detail to the graph network, illustrated by the following.

1) The Shinkansen section between the Hakata and Kokura stations, which are operated by JR West, and the sections that connect the JR West network (gray line) on the Sanyo Main Line were included.

2) The Fukuoka City Subway Kūkō Line shares the most section with the JR Kyushu Chikuhi Line between the Hakata and Meinohama stations. This section was assumed to be the same section.

3) Nishitetsu Fukuoka (Tenjin) station, i.e., the terminal station of the Nishitetsu Tenjin Ōmuta Line, was assumed to be still separated and not directly connected to the JR Kyushu Chikuhi Line.

After using Gephi (Bastian et al., 2009) version 0.9.2, open-source software to write and create the graph network, the model of the Kyushu railway network shows the component of 26 lines from all three operators with 671 nodes and 692 links as illustrated in Fig. 7.



Note:

No.	Line	No.	Line
🟥	JR Kagoshima Main Line	🟦	JR Ōmura Line
🟧	JR Nippō Main Line	🟩	JR Misumi Line
🟩	JR Chikuhō Main Line	🟦	JR Hisatsu Line
🟨	JR Nagasaki Main Line	🟪	JR Kitto Line
🟩	JR Kyūdai Main Line	🟩	JR Nichinan Line
🟦	JR Hōhi Main Line	🟪	JR Ibusuki Makurazaki Line
🟩	JR Sasaguri Line	🟥	JR Shinkansen
🟩	JR Kashii Line	🟦	Nishitetsu Tenjin Ōmuta and Dazaifu Line
🟫	JR Hitahikosan Line	🟦	Nishitetsu Amagi Line
🟪	JR Gotōji Line	🟦	Nishitetsu Kaizuka Line
🟪	JR Chikuhi Line	🟨	Subway Kūkō Line (Airport section)
🟪	JR Karatsu Line	🟨	Subway Hakozaki Line
🟪	JR Sasebo Line	🟫	Subway Nanakuma Line

Figure 7. Railway network of JR Kyushu, Nishitetsu Railway, and Fukuoka City Subway, including the sections connected to the JR West network

From Fig. 7, railway lines by operators are

1) JR Kyushu
    1.1) Kagoshima Main Line
    1.2) Nippo Main Line
    1.3) Chikuhō Main Line
    1.4) Nagasaki Main Line

1.5) Kyūdai Main Line
1.6) Hōhi Main Line
1.7) Sasaguri Line
1.8) Kashii Line
1.9) Hitahikosan Line
1.10) Gotōji Line
1.11) Chikuhi Line
1.12) Karatsu Line
1.13) Sasebo Line
1.14) Ōmura Line
1.15) Misumi Line
1.16) Hisatsu Line
1.17) Kitto Line
1.18) Nichinan Line (Include the Miyazaki Kūkō Line, the branch line that connects Miyazaki airport)
1.19) Ibusuki Makurazaki Line
1.20) Shinkansen Line
2) Nishitetsu Railway
    2.1) Tenjin Ōmuta and Dazaifu Line
    2.2) Amagi Line
    2.3) Kaizuka Line
3) Fukuoka City Subway
    3.1) Kūkō Line
    3.2) Hakozaki Line
    3.3) Nanakuma Line

From the Kyushu railway network, this research defined the node number, which identifies stations, and the link number, which identifies the section between stations, in Tables 2 and 3.

Table 2. Node number and station name of the Kyushu railway network

No.	Station Name	Line	No.	Station Name	Line
1	Mojikō	Kagoshima Main Line	19	Tōgō	Kagoshima Main Line
2	Komorie	Kagoshima Main Line	20	Higashi-Fukuma	Kagoshima Main Line
3	Moji	Kagoshima Main Line	21	Fukuma	Kagoshima Main Line
4	Kokura	Kagoshima Main Line	22	Chidori	Kagoshima Main Line
5	Nishi-Kokura	Kagoshima Main Line	23	Koga	Kagoshima Main Line
6	Kyūshūkōdaimae	Kagoshima Main Line	24	Shishibu	Kagoshima Main Line
7	Tobata	Kagoshima Main Line	25	Shingū-Chūō	Kagoshima Main Line
8	Edamitsu	Kagoshima Main Line	26	Fukkōdaimae	Kagoshima Main Line
9	Space World	Kagoshima Main Line	27	Kyūsandaimae	Kagoshima Main Line
10	Yahata	Kagoshima Main Line	28	Kashii	Kagoshima Main Line
11	Kurosaki	Kagoshima Main Line	29	Chihaya	Kagoshima Main Line
12	Jinnoharu	Kagoshima Main Line	30	Hakozaki	Kagoshima Main Line
13	Orio	Kagoshima Main Line	31	Yoshizuka	Kagoshima Main Line
14	Mizumaki	Kagoshima Main Line	32	Hakata	Kagoshima Main Line
15	Ongagawa	Kagoshima Main Line	33	Takeshita	Kagoshima Main Line
16	Ebitsu	Kagoshima Main Line	34	Sasabaru	Kagoshima Main Line
17	Kyōikudaimae	Kagoshima Main Line	35	Minami-Fukuoka	Kagoshima Main Line
18	Akama	Kagoshima Main Line	36	Kasuga	Kagoshima Main Line

Table 2. Node number and station name of the Kyushu railway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
37	Ōnojō	Kagoshima Main Line	102	Kusami	Nippō Main Line
38	Mizuki	Kagoshima Main Line	103	Kanda	Nippō Main Line
39	Tofurōminami	Kagoshima Main Line	104	Obase Nishikōdai-mae	Nippō Main Line
40	Futsukaichi	Kagoshima Main Line	105	Yukuhashi	Nippō Main Line
41	Tenpaizan	Kagoshima Main Line	106	Minami-Yukuhashi	Nippō Main Line
42	Haruda	Kagoshima Main Line	107	Shindenbaru	Nippō Main Line
43	Keyakidai	Kagoshima Main Line	108	Tsuiki	Nippō Main Line
44	Kiyama	Kagoshima Main Line	109	Shiida	Nippō Main Line
45	Yayoigaoka	Kagoshima Main Line	110	Buzen-Shōe	Nippō Main Line
46	Tashiro	Kagoshima Main Line	111	Unoshima	Nippō Main Line
47	Tosu	Kagoshima Main Line	112	Mikekado	Nippō Main Line
48	Hizen-Asahi	Kagoshima Main Line	113	Yoshitomi	Nippō Main Line
49	Kurume	Kagoshima Main Line	114	Nakatsu	Nippō Main Line
50	Araki	Kagoshima Main Line	115	Higashi-Nakatsu	Nippō Main Line
51	Nishimuta	Kagoshima Main Line	116	Imazu	Nippō Main Line
52	Hainuzuka	Kagoshima Main Line	117	Amatsu	Nippō Main Line
53	Chikugo-Funagoya	Kagoshima Main Line	118	Buzen-Zenkōji	Nippō Main Line
54	Setaka	Kagoshima Main Line	119	Yanagigaura	Nippō Main Line
55	Minami-Setaka	Kagoshima Main Line	120	Buzen-Nagasu	Nippō Main Line
56	Wataze	Kagoshima Main Line	121	Usa	Nippō Main Line
57	Yoshino	Kagoshima Main Line	122	Nishi-Yashiki	Nippō Main Line
58	Ginsui	Kagoshima Main Line	123	Tateishi	Nippō Main Line
59	Ōmuta	Kagoshima Main Line	124	Naka-Yamaga	Nippō Main Line
60	Arao	Kagoshima Main Line	125	Kitsuki	Nippō Main Line
61	Minami-Arao	Kagoshima Main Line	126	Ōga	Nippō Main Line
62	Nagasu	Kagoshima Main Line	127	Hiji	Nippō Main Line
63	Ōnoshimo	Kagoshima Main Line	128	Yōkoku	Nippō Main Line
64	Tamana	Kagoshima Main Line	129	Bungo-Toyooka	Nippō Main Line
65	Higo-Ikura	Kagoshima Main Line	130	Kamegawa	Nippō Main Line
66	Konoha	Kagoshima Main Line	131	Beppu-Daigaku	Nippō Main Line
67	Tabaruzaka	Kagoshima Main Line	132	Beppu	Nippō Main Line
68	Ueki	Kagoshima Main Line	133	Higashi-Beppu	Nippō Main Line
69	Nishisato	Kagoshima Main Line	134	Nishi-Ōita	Nippō Main Line
70	Sōjōdaigakumae	Kagoshima Main Line	135	Ōita	Nippō Main Line
71	Kami-Kumamoto	Kagoshima Main Line	136	Maki	Nippō Main Line
72	Kumamoto	Kagoshima Main Line	137	Takajō	Nippō Main Line
73	Nishi-Kumamoto	Kagoshima Main Line	138	Tsurusaki	Nippō Main Line
74	Kawashiri	Kagoshima Main Line	139	Ōzai	Nippō Main Line
75	Tomiai	Kagoshima Main Line	140	Sakanoichi	Nippō Main Line
76	Uto	Kagoshima Main Line	141	Kōzaki	Nippō Main Line
77	Matsubase	Kagoshima Main Line	142	Sashiu	Nippō Main Line
78	Ogawa	Kagoshima Main Line	143	Shitanoe	Nippō Main Line
79	Arisa	Kagoshima Main Line	144	Kumasaki	Nippō Main Line
80	Senchō	Kagoshima Main Line	145	Kami-Usuki	Nippō Main Line
81	Shin-Yatsushiro	Kagoshima Main Line	146	Usuki	Nippō Main Line
82	Yatsushiro	Kagoshima Main Line	147	Tsukumi	Nippō Main Line
83	Sendai	Kagoshima Main Line	148	Hishiro	Nippō Main Line
84	Kumanojō	Kagoshima Main Line	149	Azamui	Nippō Main Line
85	Kobanchaya	Kagoshima Main Line	150	Kariu	Nippō Main Line
86	Kushikino	Kagoshima Main Line	151	Kaizaki	Nippō Main Line
87	Kamimuragakuenmae	Kagoshima Main Line	152	Saiki	Nippō Main Line
88	Ichiki	Kagoshima Main Line	153	Kamioka	Nippō Main Line
89	Yunomoto	Kagoshima Main Line	154	Naomi	Nippō Main Line
90	Higashi-Ichiki	Kagoshima Main Line	155	Naokawa	Nippō Main Line
91	Ijūin	Kagoshima Main Line	156	Shigeoka	Nippō Main Line
92	Satsuma-Matsumoto	Kagoshima Main Line	157	Sōtarō	Nippō Main Line
93	Kami-Ijūin	Kagoshima Main Line	158	Ichitana	Nippō Main Line
94	Hiroki	Kagoshima Main Line	159	Kitagawa	Nippō Main Line
95	Kagoshima-Chūō	Kagoshima Main Line	160	Hyūga-Nagai	Nippō Main Line
96	Kagoshima	Kagoshima Main Line	161	Kita-Nobeoka	Nippō Main Line
97	Shimonoseki	Sanyo Main Line	162	Nobeoka	Nippō Main Line
98	Minami-Kokura	Nippō Main Line	163	Minami-Nobeoka	Nippō Main Line
99	Jōno	Nippō Main Line	164	Asahigaoka	Nippō Main Line
100	Abeyamakōen	Nippō Main Line	165	Totoro	Nippō Main Line
101	Shimosone	Nippō Main Line	166	Kadogawa	Nippō Main Line

Table 2. Node number and station name of the Kyushu railway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
167	Hyūgashi	Nippō Main Line	232	Kadomatsu	Sasaguri Line
168	Zaikōji	Nippō Main Line	233	Sasaguri	Sasaguri Line
169	Minami-Hyūga	Nippō Main Line	234	Chikuzen-Yamate	Sasaguri Line
170	Mimitsu	Nippō Main Line	235	Kido Nanzōin-mae	Sasaguri Line
171	Higashi-Tsuno	Nippō Main Line	236	Kurōbaru	Sasaguri Line
172	Tsuno	Nippō Main Line	237	Chikuzen-Daibu	Sasaguri Line
173	Kawaminami	Nippō Main Line	238	Saitozaki	Kashii Line
174	Takanabe	Nippō Main Line	239	Umi-no-Nakamichi	Kashii Line
175	Hyūga-Shintomi	Nippō Main Line	240	Gannosu	Kashii Line
176	Sadowara	Nippō Main Line	241	Nata	Kashii Line
177	Hyūga-Sumiyoshi	Nippō Main Line	242	Wajiro	Kashii Line
178	Hasugaike	Nippō Main Line	243	Kashii-Jingū	Kashii Line
179	Miyazaki-Jingū	Nippō Main Line	244	Maimatsubara	Kashii Line
180	Miyazaki	Nippō Main Line	245	Doi	Kashii Line
181	Minami-Miyazaki	Nippō Main Line	246	Iga	Kashii Line
182	Kanō	Nippō Main Line	247	Sakado	Kashii Line
183	Kiyotake	Nippō Main Line	248	Sue	Kashii Line
184	Hyūga-Kutsukake	Nippō Main Line	249	Sue-Chūō	Kashii Line
185	Tano	Nippō Main Line	250	Shinbaru	Kashii Line
186	Aoidake	Nippō Main Line	251	Umi	Kashii Line
187	Yamanokuchi	Nippō Main Line	252	Ishida	Hitahikosan Line
188	Mochibaru	Nippō Main Line	253	Shii-Kōen	Hitahikosan Line
189	Mimata	Nippō Main Line	254	Shii	Hitahikosan Line
190	Miyakonojō	Nippō Main Line	255	Ishiharamachi	Hitahikosan Line
191	Nishi-Miyakonojō	Nippō Main Line	256	Yobuno	Hitahikosan Line
192	Isoichi	Nippō Main Line	257	Saidōsho	Hitahikosan Line
193	Takarabe	Nippō Main Line	258	Kawara	Hitahikosan Line
194	Kitamata	Nippō Main Line	259	Ipponmatsu	Hitahikosan Line
195	Ōsumi-Ōkawara	Nippō Main Line	260	Tagawa-Ita	Hitahikosan Line
196	Kita-Naganoda	Nippō Main Line	261	Tagawa-Gotōji	Hitahikosan Line
197	Kirishima-Jingū	Nippō Main Line	262	Ikejiri	Hitahikosan Line
198	Kokubu	Nippō Main Line	263	Buzen-Kawasaki	Hitahikosan Line
199	Hayato	Nippō Main Line	264	Nishi-Soeda	Hitahikosan Line
200	Kajiki	Nippō Main Line	265	Soeda	Hitahikosan Line
201	Kinkō	Nippō Main Line	266	Kanyūsha-Hikosan	Hitahikosan Line
202	Chōsa	Nippō Main Line	267	Buzen-Masuda	Hitahikosan Line
203	Aira	Nippō Main Line	268	Hikosan	Hitahikosan Line
204	Shigetomi	Nippō Main Line	269	Chikuzen-Iwaya	Hitahikosan Line
205	Ryūgamizu	Nippō Main Line	270	Daigyōji	Hitahikosan Line
206	Wakamatsu	Chikuhō Main Line	271	Hōshuyama	Hitahikosan Line
207	Fujinoki	Chikuhō Main Line	272	Ōtsuru	Hitahikosan Line
208	Okudōkai	Chikuhō Main Line	273	Imayama	Hitahikosan Line
209	Futajima	Chikuhō Main Line	274	Yoake	Hitahikosan Line
210	Honjō	Chikuhō Main Line	275	Kami-Mio	Gotōji Line
211	Higashi-Mizumaki	Chikuhō Main Line	276	Shimo-Kamoo	Gotōji Line
212	Nakama	Chikuhō Main Line	277	Chikuzen-Shōnai	Gotōji Line
213	Chikuzen-Habu	Chikuhō Main Line	278	Funao	Gotōji Line
214	Kurate	Chikuhō Main Line	279	Gion	Chikuhi Line
215	Chikuzen-Ueki	Chikuhō Main Line	280	Nakasu-Kawabata	Chikuhi Line
216	Shinnyū	Chikuhō Main Line	281	Tenjin	Chikuhi Line
217	Nōgata	Chikuhō Main Line	282	Akasaka	Chikuhi Line
218	Katsuno	Chikuhō Main Line	283	Ōhorikōen	Chikuhi Line
219	Kotake	Chikuhō Main Line	284	Tōjinmachi	Chikuhi Line
220	Namazuta	Chikuhō Main Line	285	Nishijin	Chikuhi Line
221	Urata	Chikuhō Main Line	286	Fujisaki	Chikuhi Line
222	Shin Iizuka	Chikuhō Main Line	287	Muromi	Chikuhi Line
223	Iizuka	Chikuhō Main Line	288	Meinohama	Chikuhi Line
224	Tentō	Chikuhō Main Line	289	Shimoyamato	Chikuhi Line
225	Keisen	Chikuhō Main Line	290	Imajuku	Chikuhi Line
226	Kami Honami	Chikuhō Main Line	291	Kyūdai-Gakkentoshi	Chikuhi Line
227	Chikuzen Uchino	Chikuhō Main Line	292	Susenji	Chikuhi Line
228	Chikuzen Yamae	Chikuhō Main Line	293	Hatae	Chikuhi Line
229	Yusu	Sasaguri Line	294	Itoshima-Kokomae	Chikuhi Line
230	Harumachi	Sasaguri Line	295	Chikuzen-Maebaru	Chikuhi Line
231	Chojabaru	Sasaguri Line	296	Misakigaoka	Chikuhi Line

Table 2. Node number and station name of the Kyushu railway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
297	Kafuri	Chikuhi Line	356	Ōkusa	Nagasaki Main Line (old)
298	Ikisan	Chikuhi Line	357	Honkawachi	Nagasaki Main Line (old)
299	Chikuzen-Fukae	Chikuhi Line	358	Nagayo	Nagasaki Main Line (old)
300	Dainyū	Chikuhi Line	359	Kōda	Nagasaki Main Line (old)
301	Fukuyoshi	Chikuhi Line	360	Michinoo	Nagasaki Main Line (old)
302	Shikaka	Chikuhi Line	361	Nishi-Urakami	Nagasaki Main Line (old)
303	Hamasaki	Chikuhi Line	362	Ogi	Karatsu Line
304	Nijinomatsubara	Chikuhi Line	363	Higashi-Taku	Karatsu Line
305	Higashi-Karatsu	Chikuhi Line	364	Naka-Taku	Karatsu Line
306	Watada	Chikuhi Line	365	Taku	Karatsu Line
307	Karatsu	Chikuhi Line	366	Kyūragi	Karatsu Line
308	Nishi-Karatsu	Chikuhi Line	367	Iwaya	Karatsu Line
309	Onizuka	Chikuhi Line	368	Ōchi	Karatsu Line
310	Yamamoto	Chikuhi Line	369	Honmutabe	Karatsu Line
311	Hizen-Kubo	Chikuhi Line	370	Ōmachi	Sasebo Line
312	Nishi-Ōchi	Chikuhi Line	371	Kitagata	Sasebo Line
313	Sari	Chikuhi Line	372	Takahashi	Sasebo Line
314	Komanaki	Chikuhi Line	373	Takeo-Onsen	Sasebo Line
315	Ōkawano	Chikuhi Line	374	Nagao	Sasebo Line
316	Hizen-Nagano	Chikuhi Line	375	Mimasaka	Sasebo Line
317	Momonokawa	Chikuhi Line	376	Kami-Arita	Sasebo Line
318	Kanaishihara	Chikuhi Line	377	Arita	Sasebo Line
319	Kami-Imari	Chikuhi Line	378	Mikawachi	Sasebo Line
320	Imari	Chikuhi Line	379	Haiki	Sasebo Line
321	Shin-Tosu	Nagasaki Main Line	380	Daitō	Sasebo Line
322	Hizen-Fumoto	Nagasaki Main Line	381	Hiu	Sasebo Line
323	Nakabaru	Nagasaki Main Line	382	Sasebo	Sasebo Line
324	Yoshinogari-Kōen	Nagasaki Main Line	383	Huis Ten Bosch	Ōmura Line
325	Kanzaki	Nagasaki Main Line	384	Haenosaki	Ōmura Line
326	Igaya	Nagasaki Main Line	385	Ogushigō	Ōmura Line
327	Saga	Nagasaki Main Line	386	Kawatana	Ōmura Line
328	Nabeshima	Nagasaki Main Line	387	Sonogi	Ōmura Line
329	Balloon Saga (seasonal)	Nagasaki Main Line	388	Chiwata	Ōmura Line
330	Kubota	Nagasaki Main Line	389	Matsubara	Ōmura Line
331	Ushizu	Nagasaki Main Line	390	Takematsu	Ōmura Line
332	Hizen-Yamaguchi	Nagasaki Main Line	391	Suwa	Ōmura Line
333	Hizen-Shiroishi	Nagasaki Main Line	392	Ōmura	Ōmura Line
334	Hizen-Ryūō	Nagasaki Main Line	393	Iwamatsu	Ōmura Line
335	Hizen-Kashima	Nagasaki Main Line	394	Kurume-Kōkōmae	Kyūdai Main Line
336	Hizen-Hama	Nagasaki Main Line	395	Minami-Kurume	Kyūdai Main Line
337	Hizen-Nanaura	Nagasaki Main Line	396	Kurume-Daigakumae	Kyūdai Main Line
338	Hizen-Iida	Nagasaki Main Line	397	Mii	Kyūdai Main Line
339	Tara	Nagasaki Main Line	398	Zendōji	Kyūdai Main Line
340	Hizen-Ōura	Nagasaki Main Line	399	Chikugo-Kusano	Kyūdai Main Line
341	Konagai	Nagasaki Main Line	400	Tanushimaru	Kyūdai Main Line
342	Nagasato	Nagasaki Main Line	401	Chikugo-Yoshii	Kyūdai Main Line
343	Yue	Nagasaki Main Line	402	Ukiha	Kyūdai Main Line
344	Oe	Nagasaki Main Line	403	Chikugo-Ōishi	Kyūdai Main Line
345	Hizen-Nagata	Nagasaki Main Line	404	Teruoka	Kyūdai Main Line
346	Higashi-Isahaya	Nagasaki Main Line	405	Hita	Kyūdai Main Line
347	Isahaya	Nagasaki Main Line	406	Bungo-Miyoshi	Kyūdai Main Line
348	Nishi-Isahaya	Nagasaki Main Line	407	Bungo-Nakagawa	Kyūdai Main Line
349	Kikitsu	Nagasaki Main Line	408	Amagase	Kyūdai Main Line
350	Ichinuno	Nagasaki Main Line	409	Sugikawachi	Kyūdai Main Line
351	Hizen-Koga	Nagasaki Main Line	410	Kita-Yamada	Kyūdai Main Line
352	Utsutsugawa	Nagasaki Main Line	411	Bungo-Mori	Kyūdai Main Line
353	Urakami	Nagasaki Main Line	412	Era	Kyūdai Main Line
354	Nagasaki	Nagasaki Main Line	413	Hikiji	Kyūdai Main Line
355	Higashisono	Nagasaki Main Line (old)	414	Bungo-Nakamura	Kyūdai Main Line

Table 2. Node number and station name of the Kyushu railway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
415	Noya	Kyūdai Main Line	480	Isshōchi	Hisatsu Line
416	Yufuin	Kyūdai Main Line	481	Naraguchi	Hisatsu Line
417	Minami-Yufu	Kyūdai Main Line	482	Watari	Hisatsu Line
418	Yunohira	Kyūdai Main Line	483	Nishi Hitoyoshi	Hisatsu Line
419	Shōnai	Kyūdai Main Line	484	Hitoyoshi	Hisatsu Line
420	Tenjinyama	Kyūdai Main Line	485	Okoba	Hisatsu Line
421	Onoya	Kyūdai Main Line	486	Yatake	Hisatsu Line
422	Onigase	Kyūdai Main Line	487	Masaki	Hisatsu Line
423	Mukainoharu	Kyūdai Main Line	488	Yoshimatsu	Hisatsu Line
424	Bungo-Kokubu	Kyūdai Main Line	489	Kurino	Hisatsu Line
425	Kaku	Kyūdai Main Line	490	Ōsumi-Yokogawa	Hisatsu Line
426	Minami-Ōita	Kyūdai Main Line	491	Uemura	Hisatsu Line
427	Furugō	Kyūdai Main Line	492	Kirishima Onsen	Hisatsu Line
428	Heisei	Hōhi Main Line	493	Kareigawa	Hisatsu Line
429	Minami-Kumamoto	Hōhi Main Line	494	Naka-fukura	Hisatsu Line
430	Shin-Suizenji	Hōhi Main Line	495	Hyōkiyama	Hisatsu Line
431	Suizenji	Hōhi Main Line	496	Hinatayama	Hisatsu Line
432	Tōkai-Gakuen-mae	Hōhi Main Line	497	Tsurumaru	Kitto Line
433	Tatsutaguchi	Hōhi Main Line	498	Kyōmachi Onsen	Kitto Line
434	Musashizuka	Hōhi Main Line	499	Ebino	Kitto Line
435	Hikari no Mori	Hōhi Main Line	500	Ebino Uwae	Kitto Line
436	Sanrigi	Hōhi Main Line	501	Ebino Iino	Kitto Line
437	Haramizu	Hōhi Main Line	502	Nishi Kobayashi	Kitto Line
438	Higo-Ōzu	Hōhi Main Line	503	Kobayashi	Kitto Line
439	Seta	Hōhi Main Line	504	Hirowara	Kitto Line
440	Tateno	Hōhi Main Line	505	Takaharu	Kitto Line
441	Akamizu	Hōhi Main Line	506	Hyūga Maeda	Kitto Line
442	Ichinokawa	Hōhi Main Line	507	Takasaki Shinden	Kitto Line
443	Uchinomaki	Hōhi Main Line	508	Higashi Takasaki	Kitto Line
444	Aso	Hōhi Main Line	509	Mangatsuka	Kitto Line
445	Ikoi-no-Mura	Hōhi Main Line	510	Tanigashira	Kitto Line
446	Miyaji	Hōhi Main Line	511	Hyūga Shōnai	Kitto Line
447	Namino	Hōhi Main Line	512	Tayoshi	Nichinan Line
448	Takimizu	Hōhi Main Line	513	Minamikata	Nichinan Line
449	Bungo-Ogi	Hōhi Main Line	514	Kibana	Nichinan Line
450	Tamarai	Hōhi Main Line	515	Undōkōen	Nichinan Line
451	Bungo-Taketa	Hōhi Main Line	516	Sosanji	Nichinan Line
452	Asaji	Hōhi Main Line	517	Kodomonokuni	Nichinan Line
453	Ogata	Hōhi Main Line	518	Aoshima	Nichinan Line
454	Bungo-Kiyokawa	Hōhi Main Line	519	Oryūzako	Nichinan Line
455	Miemachi	Hōhi Main Line	520	Uchiumi	Nichinan Line
456	Sugao	Hōhi Main Line	521	Kouchiumi	Nichinan Line
457	Inukai	Hōhi Main Line	522	Ibii	Nichinan Line
458	Takenaka	Hōhi Main Line	523	Kitagō	Nichinan Line
459	Nakahanda	Hōhi Main Line	524	Uchinoda	Nichinan Line
460	Ōita-Daigaku-mae	Hōhi Main Line	525	Obi	Nichinan Line
461	Shikido	Hōhi Main Line	526	Nichinan	Nichinan Line
462	Takio	Hōhi Main Line	527	Aburatsu	Nichinan Line
463	Midorikawa	Misumi Line	528	Ōdōtsu	Nichinan Line
464	Sumiyoshi	Misumi Line	529	Nangō	Nichinan Line
465	Higo-Nagahama	Misumi Line	530	Taninokuchi	Nichinan Line
466	Ōda	Misumi Line	531	Yowara	Nichinan Line
467	Akase	Misumi Line	532	Hyūga-Ōtsuka	Nichinan Line
468	Ishiuchi Dam	Misumi Line	533	Hyūga-Kitakata	Nichinan Line
469	Hataura	Misumi Line	534	Kushima	Nichinan Line
470	Misumi	Misumi Line	535	Fukushima-Imamachi	Nichinan Line
471	Dan	Hisatsu Line	536	Fukushima-Takamatsu	Nichinan Line
472	Sakamoto	Hisatsu Line	537	Ōsumi-Natsui	Nichinan Line
473	Haki	Hisatsu Line	538	Shibushi	Nichinan Line
474	Kamase	Hisatsu Line	539	Miyazaki Airport	Miyazaki Kūkō Line
475	Setoishi	Hisatsu Line	540	Kōrimoto	Ibusuki Makurazaki Line
476	Kaiji	Hisatsu Line	541	Minami-Kagoshima	Ibusuki Makurazaki Line
477	Yoshio	Hisatsu Line	542	Usuki	Ibusuki Makurazaki Line
478	Shiroishi	Hisatsu Line	543	Taniyama	Ibusuki Makurazaki Line
479	Kyūsendō	Hisatsu Line	544	Jigenji	Ibusuki Makurazaki Line

Table 2. Node number and station name of the Kyushu railway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
545	Sakanoue	Ibusuki Makurazaki Line	577	Nishitetsu Fukuoka (Tenjin)	Nishitetsu-Tenjin Ōmuta Line
546	Goino	Ibusuki Makurazaki Line	578	Yakuin	Nishitetsu-Tenjin Ōmuta Line
547	Hirakawa	Ibusuki Makurazaki Line	579	Nishitetsu Hirao	Nishitetsu-Tenjin Ōmuta Line
548	Sesekushi	Ibusuki Makurazaki Line	580	Takamiya	Nishitetsu-Tenjin Ōmuta Line
549	Nakamyō	Ibusuki Makurazaki Line	581	Ōhashi	Nishitetsu-Tenjin Ōmuta Line
550	Kiire	Ibusuki Makurazaki Line	582	Ijiri	Nishitetsu-Tenjin Ōmuta Line
551	Maenohama	Ibusuki Makurazaki Line	583	Zasshonokuma	Nishitetsu-Tenjin Ōmuta Line
552	Nukumi	Ibusuki Makurazaki Line	584	Kasugabaru	Nishitetsu-Tenjin Ōmuta Line
553	Satsuma-Imaizumi	Ibusuki Makurazaki Line	585	Shirakibaru	Nishitetsu-Tenjin Ōmuta Line
554	Miyagahama	Ibusuki Makurazaki Line	586	Shimoōri	Nishitetsu-Tenjin Ōmuta Line
555	Nigatsuden	Ibusuki Makurazaki Line	587	Tofurōmae	Nishitetsu-Tenjin Ōmuta Line
556	Ibusuki	Ibusuki Makurazaki Line	588	Nishitetsu Futsukaichi	Nishitetsu-Tenjin Ōmuta Line
557	Yamakawa	Ibusuki Makurazaki Line	589	Murasaki	Nishitetsu-Tenjin Ōmuta Line
558	Ōyama	Ibusuki Makurazaki Line	590	Asakuragaidō	Nishitetsu-Tenjin Ōmuta Line
559	Nishi-Ōyama	Ibusuki Makurazaki Line	591	Sakuradai	Nishitetsu-Tenjin Ōmuta Line
560	Satsuma-Kawashiri	Ibusuki Makurazaki Line	592	Chikushi	Nishitetsu-Tenjin Ōmuta Line
561	Higashi-Kaimon	Ibusuki Makurazaki Line	593	Tsuko	Nishitetsu-Tenjin Ōmuta Line
562	Kaimon	Ibusuki Makurazaki Line	594	Mikunigaoka	Nishitetsu-Tenjin Ōmuta Line
563	Irino	Ibusuki Makurazaki Line	595	Mitsusawa	Nishitetsu-Tenjin Ōmuta Line
564	Ei	Ibusuki Makurazaki Line	596	Ōho	Nishitetsu-Tenjin Ōmuta Line
565	Nishi-Ei	Ibusuki Makurazaki Line	597	Nishitetsu Ogōri	Nishitetsu-Tenjin Ōmuta Line
566	Goryō	Ibusuki Makurazaki Line	598	Hatama	Nishitetsu-Tenjin Ōmuta Line
567	Ishikaki	Ibusuki Makurazaki Line	599	Ajisaka	Nishitetsu-Tenjin Ōmuta Line
568	Mizunarikawa	Ibusuki Makurazaki Line	600	Miyanojin	Nishitetsu-Tenjin Ōmuta Line
569	Ei-Ōkawa	Ibusuki Makurazaki Line	601	Kushiwara	Nishitetsu-Tenjin Ōmuta Line
570	Matsugaura	Ibusuki Makurazaki Line	602	Nishitetsu Kurume	Nishitetsu-Tenjin Ōmuta Line
571	Satsuma-Shioya	Ibusuki Makurazaki Line	603	Hanabatake	Nishitetsu-Tenjin Ōmuta Line
572	Shirasawa	Ibusuki Makurazaki Line	604	Shikenjōmae	Nishitetsu-Tenjin Ōmuta Line
573	Satsuma-Itashiki	Ibusuki Makurazaki Line	605	Tsubuku	Nishitetsu-Tenjin Ōmuta Line
574	Makurazaki	Ibusuki Makurazaki Line	606	Yasutake	Nishitetsu-Tenjin Ōmuta Line
575	Shin-Minamata	Shinkansen	607	Daizenji	Nishitetsu-Tenjin Ōmuta Line
576	Izumi	Shinkansen	608	Mizuma	Nishitetsu-Tenjin Ōmuta Line

Table 2. Node number and station name of the Kyushu railway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
609	Inuzuka	Nishitetsu-Tenjin Ōmuta Line	641	Nishitetsu Kashii	Nishitetsu-Kaizuka Line
610	Ōmizo	Nishitetsu-Tenjin Ōmuta Line	642	Kashii-Kaenmae	Nishitetsu-Kaizuka Line
611	Hatchōmuta	Nishitetsu-Tenjin Ōmuta Line	643	Tōnoharu	Nishitetsu-Kaizuka Line
612	Kamachi	Nishitetsu-Tenjin Ōmuta Line	644	Mitoma	Nishitetsu-Kaizuka Line
613	Yakabe	Nishitetsu-Tenjin Ōmuta Line	645	Nishitetsu Shingū	Nishitetsu-Kaizuka Line
614	Nishitetsu Yanagawa	Nishitetsu-Tenjin Ōmuta Line	646	Higashi-Hie	Fukuoka City Subway-Kūkō Line
615	Tokumasu	Nishitetsu-Tenjin Ōmuta Line	647	Fukuokakūkō (Airport)	Fukuoka City Subway-Kūkō Line
616	Shiotsuka	Nishitetsu-Tenjin Ōmuta Line	648	Gofukumachi	Fukuoka City Subway-Hakozaki Line
617	Nishitetsu Nakashima	Nishitetsu-Tenjin Ōmuta Line	649	Chiyo-Kenchōguchi	Fukuoka City Subway-Hakozaki Line
618	Enoura	Nishitetsu-Tenjin Ōmuta Line	650	Maidashi-Kyūdai-byōin-mae	Fukuoka City Subway-Hakozaki Line
619	Hiraki	Nishitetsu-Tenjin Ōmuta Line	651	Hakozaki-Miyamae	Fukuoka City Subway-Hakozaki Line
620	Nishitetsu Wataze	Nishitetsu-Tenjin Ōmuta Line	652	Hakozaki-Kyūdai-mae	Fukuoka City Subway-Hakozaki Line
621	Kuranaga	Nishitetsu-Tenjin Ōmuta Line	653	Tenjin-Minami	Fukuoka City Subway-Nanakuma Line
622	Higashi-Amagi	Nishitetsu-Tenjin Ōmuta Line	654	Watanabe-dōri	Fukuoka City Subway-Nanakuma Line
623	Nishitetsu Ginsui	Nishitetsu-Tenjin Ōmuta Line	655	Yakuin-ōdōri	Fukuoka City Subway-Nanakuma Line
624	Shin-Sakaemachi	Nishitetsu-Tenjin Ōmuta Line	656	Sakurazaka	Fukuoka City Subway-Nanakuma Line
625	Nishitetsu Gojō	Nishitetsu-Dazaifu Line	657	Ropponmatsu	Fukuoka City Subway-Nanakuma Line
626	Dazaifu	Nishitetsu-Dazaifu Line	658	Befu	Fukuoka City Subway-Nanakuma Line
627	Gorōmaru	Nishitetsu-Amagi Line	659	Chayama	Fukuoka City Subway-Nanakuma Line
628	Gakkōmae	Nishitetsu-Amagi Line	660	Kanayama	Fukuoka City Subway-Nanakuma Line
629	Koganchaya	Nishitetsu-Amagi Line	661	Nanakuma	Fukuoka City Subway-Nanakuma Line
630	Kitano	Nishitetsu-Amagi Line	662	Fukudaimae	Fukuoka City Subway-Nanakuma Line
631	Ōki	Nishitetsu-Amagi Line	663	Umebayashi	Fukuoka City Subway-Nanakuma Line
632	Kaneshima	Nishitetsu-Amagi Line	664	Noke	Fukuoka City Subway-Nanakuma Line
633	Ōzeki	Nishitetsu-Amagi Line	665	Kamo	Fukuoka City Subway-Nanakuma Line
634	Hongō	Nishitetsu-Amagi Line	666	Jirōmaru	Fukuoka City Subway-Nanakuma Line
635	Kamiura	Nishitetsu-Amagi Line	667	Hashimoto	Fukuoka City Subway-Nanakuma Line
636	Mada	Nishitetsu-Amagi Line	668	Shin-Shimonoseki	Shinkansen
637	Amagi	Nishitetsu-Amagi Line	669	Hakata-Minami	Shinkansen
638	Kaizuka	Nishitetsu-Kaizuka Line	670	Shin-Ōmuta	Shinkansen
639	Najima	Nishitetsu-Kaizuka Line	671	Shin-Tamana	Shinkansen
640	Kashii-Miyamae	Nishitetsu-Kaizuka Line			

Table 3. Link number and pair of nodes of the Kyushu railway network

No.	**From node**	**To node**	**Line**	**No.**	**From node**	**To node**	**Line**
1	1	2	Kagoshima Main Line	65	65	66	Kagoshima Main Line
2	2	3	Kagoshima Main Line	66	66	67	Kagoshima Main Line
3	3	4	Kagoshima Main Line	67	67	68	Kagoshima Main Line
4	4	5	Kagoshima Main Line	68	68	69	Kagoshima Main Line
5	5	6	Kagoshima Main Line	69	69	70	Kagoshima Main Line
6	6	7	Kagoshima Main Line	70	70	71	Kagoshima Main Line
7	7	8	Kagoshima Main Line	71	71	72	Kagoshima Main Line
8	8	9	Kagoshima Main Line	72	72	73	Kagoshima Main Line
9	9	10	Kagoshima Main Line	73	73	74	Kagoshima Main Line
10	10	11	Kagoshima Main Line	74	74	75	Kagoshima Main Line
11	11	12	Kagoshima Main Line	75	75	76	Kagoshima Main Line
12	12	13	Kagoshima Main Line	76	76	77	Kagoshima Main Line
13	13	14	Kagoshima Main Line	77	77	78	Kagoshima Main Line
14	14	15	Kagoshima Main Line	78	78	79	Kagoshima Main Line
15	15	16	Kagoshima Main Line	79	79	80	Kagoshima Main Line
16	16	17	Kagoshima Main Line	80	80	81	Kagoshima Main Line
17	17	18	Kagoshima Main Line	81	81	82	Kagoshima Main Line
18	18	19	Kagoshima Main Line	82	83	84	Kagoshima Main Line
19	19	20	Kagoshima Main Line	83	84	85	Kagoshima Main Line
20	20	21	Kagoshima Main Line	84	85	86	Kagoshima Main Line
21	21	22	Kagoshima Main Line	85	86	87	Kagoshima Main Line
22	22	23	Kagoshima Main Line	86	87	88	Kagoshima Main Line
23	23	24	Kagoshima Main Line	87	88	89	Kagoshima Main Line
24	24	25	Kagoshima Main Line	88	89	90	Kagoshima Main Line
25	25	26	Kagoshima Main Line	89	90	91	Kagoshima Main Line
26	26	27	Kagoshima Main Line	90	91	92	Kagoshima Main Line
27	27	28	Kagoshima Main Line	91	92	93	Kagoshima Main Line
28	28	29	Kagoshima Main Line	92	93	94	Kagoshima Main Line
29	29	30	Kagoshima Main Line	93	94	95	Kagoshima Main Line
30	30	31	Kagoshima Main Line	94	95	96	Kagoshima Main Line
31	31	32	Kagoshima Main Line	95	97	3	Sanyo Main Line
32	32	33	Kagoshima Main Line	96	5	98	Nippō Main Line
33	33	34	Kagoshima Main Line	97	98	99	Nippō Main Line
34	34	35	Kagoshima Main Line	98	99	100	Nippō Main Line
35	35	36	Kagoshima Main Line	99	100	101	Nippō Main Line
36	36	37	Kagoshima Main Line	100	101	102	Nippō Main Line
37	37	38	Kagoshima Main Line	101	102	103	Nippō Main Line
38	38	39	Kagoshima Main Line	102	103	104	Nippō Main Line
39	39	40	Kagoshima Main Line	103	104	105	Nippō Main Line
40	40	41	Kagoshima Main Line	104	105	106	Nippō Main Line
41	41	42	Kagoshima Main Line	105	106	107	Nippō Main Line
42	42	43	Kagoshima Main Line	106	107	108	Nippō Main Line
43	43	44	Kagoshima Main Line	107	108	109	Nippō Main Line
44	44	45	Kagoshima Main Line	108	109	110	Nippō Main Line
45	45	46	Kagoshima Main Line	109	110	111	Nippō Main Line
46	46	47	Kagoshima Main Line	110	111	112	Nippō Main Line
47	47	48	Kagoshima Main Line	111	112	113	Nippō Main Line
48	48	49	Kagoshima Main Line	112	113	114	Nippō Main Line
49	49	50	Kagoshima Main Line	113	114	115	Nippō Main Line
50	50	51	Kagoshima Main Line	114	115	116	Nippō Main Line
51	51	52	Kagoshima Main Line	115	116	117	Nippō Main Line
52	52	53	Kagoshima Main Line	116	117	118	Nippō Main Line
53	53	54	Kagoshima Main Line	117	118	119	Nippō Main Line
54	54	55	Kagoshima Main Line	118	119	120	Nippō Main Line
55	55	56	Kagoshima Main Line	119	120	121	Nippō Main Line
56	56	57	Kagoshima Main Line	120	121	122	Nippō Main Line
57	57	58	Kagoshima Main Line	121	122	123	Nippō Main Line
58	58	59	Kagoshima Main Line	122	123	124	Nippō Main Line
59	59	60	Kagoshima Main Line	123	124	125	Nippō Main Line
60	60	61	Kagoshima Main Line	124	125	126	Nippō Main Line
61	61	62	Kagoshima Main Line	125	126	127	Nippō Main Line
62	62	63	Kagoshima Main Line	126	127	128	Nippō Main Line
63	63	64	Kagoshima Main Line	127	128	129	Nippō Main Line
64	64	65	Kagoshima Main Line	128	129	130	Nippō Main Line

Table 3. Link number and pair of nodes of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**No.**	**From node**	**To node**	**Line**
129	130	131	Nippō Main Line	193	194	195	Nippō Main Line
130	131	132	Nippō Main Line	194	195	196	Nippō Main Line
131	132	133	Nippō Main Line	195	196	197	Nippō Main Line
132	133	134	Nippō Main Line	196	197	198	Nippō Main Line
133	134	135	Nippō Main Line	197	198	199	Nippō Main Line
134	135	136	Nippō Main Line	198	199	200	Nippō Main Line
135	136	137	Nippō Main Line	199	200	201	Nippō Main Line
136	137	138	Nippō Main Line	200	201	202	Nippō Main Line
137	138	139	Nippō Main Line	201	202	203	Nippō Main Line
138	139	140	Nippō Main Line	202	203	204	Nippō Main Line
139	140	141	Nippō Main Line	203	204	205	Nippō Main Line
140	141	142	Nippō Main Line	204	205	96	Nippō Main Line
141	142	143	Nippō Main Line	205	206	207	Chikuhō Main Line
142	143	144	Nippō Main Line	206	207	208	Chikuhō Main Line
143	144	145	Nippō Main Line	207	208	209	Chikuhō Main Line
144	145	146	Nippō Main Line	208	209	210	Chikuhō Main Line
145	146	147	Nippō Main Line	209	210	13	Chikuhō Main Line
146	147	148	Nippō Main Line	210	13	211	Chikuhō Main Line
147	148	149	Nippō Main Line	211	211	212	Chikuhō Main Line
148	149	150	Nippō Main Line	212	212	213	Chikuhō Main Line
149	150	151	Nippō Main Line	213	213	214	Chikuhō Main Line
150	151	152	Nippō Main Line	214	214	215	Chikuhō Main Line
151	152	153	Nippō Main Line	215	215	216	Chikuhō Main Line
152	153	154	Nippō Main Line	216	216	217	Chikuhō Main Line
153	154	155	Nippō Main Line	217	217	218	Chikuhō Main Line
154	155	156	Nippō Main Line	218	218	219	Chikuhō Main Line
155	156	157	Nippō Main Line	219	219	220	Chikuhō Main Line
156	157	158	Nippō Main Line	220	220	221	Chikuhō Main Line
157	158	159	Nippō Main Line	221	221	222	Chikuhō Main Line
158	159	160	Nippō Main Line	222	222	223	Chikuhō Main Line
159	160	161	Nippō Main Line	223	223	224	Chikuhō Main Line
160	161	162	Nippō Main Line	224	224	225	Chikuhō Main Line
161	162	163	Nippō Main Line	225	225	226	Chikuhō Main Line
162	163	164	Nippō Main Line	226	226	227	Chikuhō Main Line
163	164	165	Nippō Main Line	227	227	228	Chikuhō Main Line
164	165	166	Nippō Main Line	228	228	42	Chikuhō Main Line
165	166	167	Nippō Main Line	229	31	229	Sasaguri Line
166	167	168	Nippō Main Line	230	229	230	Sasaguri Line
167	168	169	Nippō Main Line	231	230	231	Sasaguri Line
168	169	170	Nippō Main Line	232	231	232	Sasaguri Line
169	170	171	Nippō Main Line	233	232	233	Sasaguri Line
170	171	172	Nippō Main Line	234	233	234	Sasaguri Line
171	172	173	Nippō Main Line	235	234	235	Sasaguri Line
172	173	174	Nippō Main Line	236	235	236	Sasaguri Line
173	174	175	Nippō Main Line	237	236	237	Sasaguri Line
174	175	176	Nippō Main Line	238	237	225	Sasaguri Line
175	176	177	Nippō Main Line	239	238	239	Kashii Line
176	177	178	Nippō Main Line	240	239	240	Kashii Line
177	178	179	Nippō Main Line	241	240	241	Kashii Line
178	179	180	Nippō Main Line	242	241	242	Kashii Line
179	180	181	Nippō Main Line	243	242	28	Kashii Line
180	181	182	Nippō Main Line	244	28	243	Kashii Line
181	182	183	Nippō Main Line	245	243	244	Kashii Line
182	183	184	Nippō Main Line	246	244	245	Kashii Line
183	184	185	Nippō Main Line	247	245	246	Kashii Line
184	185	186	Nippō Main Line	248	246	231	Kashii Line
185	186	187	Nippō Main Line	249	231	247	Kashii Line
186	187	188	Nippō Main Line	250	247	248	Kashii Line
187	188	189	Nippō Main Line	251	248	249	Kashii Line
188	189	190	Nippō Main Line	252	249	250	Kashii Line
189	190	191	Nippō Main Line	253	250	251	Kashii Line
190	191	192	Nippō Main Line	254	99	252	Hitahikosan Line
191	192	193	Nippō Main Line	255	252	253	Hitahikosan Line
192	193	194	Nippō Main Line	256	253	254	Hitahikosan Line

Table 3. Link number and pair of nodes of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**No.**	**From node**	**To node**	**Line**
257	254	255	Hitahikosan Line	321	317	318	Chikuhi Line
258	255	256	Hitahikosan Line	322	318	319	Chikuhi Line
259	256	257	Hitahikosan Line	323	319	320	Chikuhi Line
260	257	258	Hitahikosan Line	324	47	321	Nagasaki Main Line
261	258	259	Hitahikosan Line	325	321	322	Nagasaki Main Line
262	259	260	Hitahikosan Line	326	322	323	Nagasaki Main Line
263	260	261	Hitahikosan Line	327	323	324	Nagasaki Main Line
264	261	262	Hitahikosan Line	328	324	325	Nagasaki Main Line
265	262	263	Hitahikosan Line	329	325	326	Nagasaki Main Line
266	263	264	Hitahikosan Line	330	326	327	Nagasaki Main Line
267	264	265	Hitahikosan Line	331	327	328	Nagasaki Main Line
268	265	266	Hitahikosan Line	332	328	329	Nagasaki Main Line
269	266	267	Hitahikosan Line	333	329	330	Nagasaki Main Line
270	267	268	Hitahikosan Line	334	330	331	Nagasaki Main Line
271	268	269	Hitahikosan Line	335	331	332	Nagasaki Main Line
272	269	270	Hitahikosan Line	336	332	333	Nagasaki Main Line
273	270	271	Hitahikosan Line	337	333	334	Nagasaki Main Line
274	271	272	Hitahikosan Line	338	334	335	Nagasaki Main Line
275	272	273	Hitahikosan Line	339	335	336	Nagasaki Main Line
276	273	274	Hitahikosan Line	340	336	337	Nagasaki Main Line
277	222	275	Gotōji Line	341	337	338	Nagasaki Main Line
278	275	276	Gotōji Line	342	338	339	Nagasaki Main Line
279	276	277	Gotōji Line	343	339	340	Nagasaki Main Line
280	277	278	Gotōji Line	344	340	341	Nagasaki Main Line
281	278	261	Gotōji Line	345	341	342	Nagasaki Main Line
282	32	279	Chikuhi Line	346	342	343	Nagasaki Main Line
283	279	280	Chikuhi Line	347	343	344	Nagasaki Main Line
284	280	281	Chikuhi Line	348	344	345	Nagasaki Main Line
285	281	282	Chikuhi Line	349	345	346	Nagasaki Main Line
286	282	283	Chikuhi Line	350	346	347	Nagasaki Main Line
287	283	284	Chikuhi Line	351	347	348	Nagasaki Main Line
288	284	285	Chikuhi Line	352	348	349	Nagasaki Main Line
289	285	286	Chikuhi Line	353	349	350	Nagasaki Main Line
290	286	287	Chikuhi Line	354	350	351	Nagasaki Main Line
291	287	288	Chikuhi Line	355	351	352	Nagasaki Main Line
292	288	289	Chikuhi Line	356	352	353	Nagasaki Main Line
293	289	290	Chikuhi Line	357	353	354	Nagasaki Main Line
294	290	291	Chikuhi Line	358	349	355	Nagasaki Main Line (old)
295	291	292	Chikuhi Line	359	355	356	Nagasaki Main Line (old)
296	292	293	Chikuhi Line	360	356	357	Nagasaki Main Line (old)
297	293	294	Chikuhi Line	361	357	358	Nagasaki Main Line (old)
298	294	295	Chikuhi Line	362	358	359	Nagasaki Main Line (old)
299	295	296	Chikuhi Line	363	359	360	Nagasaki Main Line (old)
300	296	297	Chikuhi Line	364	360	361	Nagasaki Main Line (old)
301	297	298	Chikuhi Line	365	361	353	Nagasaki Main Line (old)
302	298	299	Chikuhi Line	366	330	362	Karatsu Line
303	299	300	Chikuhi Line	367	362	363	Karatsu Line
304	300	301	Chikuhi Line	368	363	364	Karatsu Line
305	301	302	Chikuhi Line	369	364	365	Karatsu Line
306	302	303	Chikuhi Line	370	365	366	Karatsu Line
307	303	304	Chikuhi Line	371	366	367	Karatsu Line
308	304	305	Chikuhi Line	372	367	368	Karatsu Line
309	305	306	Chikuhi Line	373	368	369	Karatsu Line
310	306	307	Chikuhi Line	374	369	310	Karatsu Line
311	307	308	Chikuhi Line	375	332	370	Sasebo Line
312	307	309	Chikuhi Line	376	370	371	Sasebo Line
313	309	310	Chikuhi Line	377	371	372	Sasebo Line
314	310	311	Chikuhi Line	378	372	373	Sasebo Line
315	311	312	Chikuhi Line	379	373	374	Sasebo Line
316	312	313	Chikuhi Line	380	374	375	Sasebo Line
317	313	314	Chikuhi Line	381	375	376	Sasebo Line
318	314	315	Chikuhi Line	382	376	377	Sasebo Line
319	315	316	Chikuhi Line	383	377	378	Sasebo Line
320	316	317	Chikuhi Line	384	378	379	Sasebo Line

Table 3. Link number and pair of nodes of the Kyushu railway network (Cont.)

No.	From node	To node	Line	No.	From node	To node	Line
385	379	380	Sasebo Line	449	440	441	Hōhi Main Line
386	380	381	Sasebo Line	450	441	442	Hōhi Main Line
387	381	382	Sasebo Line	451	442	443	Hōhi Main Line
388	379	383	Ōmura Line	452	443	444	Hōhi Main Line
389	383	384	Ōmura Line	453	444	445	Hōhi Main Line
390	384	385	Ōmura Line	454	445	446	Hōhi Main Line
391	385	386	Ōmura Line	455	446	447	Hōhi Main Line
392	386	387	Ōmura Line	456	447	448	Hōhi Main Line
393	387	388	Ōmura Line	457	448	449	Hōhi Main Line
394	388	389	Ōmura Line	458	449	450	Hōhi Main Line
395	389	390	Ōmura Line	459	450	451	Hōhi Main Line
396	390	391	Ōmura Line	460	451	452	Hōhi Main Line
397	391	392	Ōmura Line	461	452	453	Hōhi Main Line
398	392	393	Ōmura Line	462	453	454	Hōhi Main Line
399	393	347	Ōmura Line	463	454	455	Hōhi Main Line
400	49	394	Kyūdai Main Line	464	455	456	Hōhi Main Line
401	394	395	Kyūdai Main Line	465	456	457	Hōhi Main Line
402	395	396	Kyūdai Main Line	466	457	458	Hōhi Main Line
403	396	397	Kyūdai Main Line	467	458	459	Hōhi Main Line
404	397	398	Kyūdai Main Line	468	459	460	Hōhi Main Line
405	398	399	Kyūdai Main Line	469	460	461	Hōhi Main Line
406	399	400	Kyūdai Main Line	470	461	462	Hōhi Main Line
407	400	401	Kyūdai Main Line	471	462	135	Hōhi Main Line
408	401	402	Kyūdai Main Line	472	76	463	Misumi Line
409	402	403	Kyūdai Main Line	473	463	464	Misumi Line
410	403	274	Kyūdai Main Line	474	464	465	Misumi Line
411	274	404	Kyūdai Main Line	475	465	466	Misumi Line
412	404	405	Kyūdai Main Line	476	466	467	Misumi Line
413	405	406	Kyūdai Main Line	477	467	468	Misumi Line
414	406	407	Kyūdai Main Line	478	468	469	Misumi Line
415	407	408	Kyūdai Main Line	479	469	470	Misumi Line
416	408	409	Kyūdai Main Line	480	82	471	Hisatsu Line
417	409	410	Kyūdai Main Line	481	471	472	Hisatsu Line
418	410	411	Kyūdai Main Line	482	472	473	Hisatsu Line
419	411	412	Kyūdai Main Line	483	473	474	Hisatsu Line
420	412	413	Kyūdai Main Line	484	474	475	Hisatsu Line
421	413	414	Kyūdai Main Line	485	475	476	Hisatsu Line
422	414	415	Kyūdai Main Line	486	476	477	Hisatsu Line
423	415	416	Kyūdai Main Line	487	477	478	Hisatsu Line
424	416	417	Kyūdai Main Line	488	478	479	Hisatsu Line
425	417	418	Kyūdai Main Line	489	479	480	Hisatsu Line
426	418	419	Kyūdai Main Line	490	480	481	Hisatsu Line
427	419	420	Kyūdai Main Line	491	481	482	Hisatsu Line
428	420	421	Kyūdai Main Line	492	482	483	Hisatsu Line
429	421	422	Kyūdai Main Line	493	483	484	Hisatsu Line
430	422	423	Kyūdai Main Line	494	484	485	Hisatsu Line
431	423	424	Kyūdai Main Line	495	485	486	Hisatsu Line
432	424	425	Kyūdai Main Line	496	486	487	Hisatsu Line
433	425	426	Kyūdai Main Line	497	487	488	Hisatsu Line
434	426	427	Kyūdai Main Line	498	488	489	Hisatsu Line
435	427	135	Kyūdai Main Line	499	489	490	Hisatsu Line
436	72	428	Hōhi Main Line	500	490	491	Hisatsu Line
437	428	429	Hōhi Main Line	501	491	492	Hisatsu Line
438	429	430	Hōhi Main Line	502	492	493	Hisatsu Line
439	430	431	Hōhi Main Line	503	493	494	Hisatsu Line
440	431	432	Hōhi Main Line	504	494	495	Hisatsu Line
441	432	433	Hōhi Main Line	505	495	496	Hisatsu Line
442	433	434	Hōhi Main Line	506	496	199	Hisatsu Line
443	434	435	Hōhi Main Line	507	488	497	Kitto Line
444	435	436	Hōhi Main Line	508	497	498	Kitto Line
445	436	437	Hōhi Main Line	509	498	499	Kitto Line
446	437	438	Hōhi Main Line	510	499	500	Kitto Line
447	438	439	Hōhi Main Line	511	500	501	Kitto Line
448	439	440	Hōhi Main Line	512	501	502	Kitto Line

Table 3. Link number and pair of nodes of the Kyushu railway network (Cont.)

No.	From node	To node	Line	No.	From node	To node	Line
513	502	503	Kitto Line	577	565	566	Ibusuki Makurazaki Line
514	503	504	Kitto Line	578	566	567	Ibusuki Makurazaki Line
515	504	505	Kitto Line	579	567	568	Ibusuki Makurazaki Line
516	505	506	Kitto Line	580	568	569	Ibusuki Makurazaki Line
517	506	507	Kitto Line	581	569	570	Ibusuki Makurazaki Line
518	507	508	Kitto Line	582	570	571	Ibusuki Makurazaki Line
519	508	509	Kitto Line	583	571	572	Ibusuki Makurazaki Line
520	509	510	Kitto Line	584	572	573	Ibusuki Makurazaki Line
521	510	511	Kitto Line	585	573	574	Ibusuki Makurazaki Line
522	511	190	Kitto Line	586	577	578	Nishitetsu-Tenjin Ōmuta Line
523	181	512	Nichinan Line	587	578	579	Nishitetsu-Tenjin Ōmuta Line
524	512	513	Nichinan Line	588	579	580	Nishitetsu-Tenjin Ōmuta Line
525	513	514	Nichinan Line	589	580	581	Nishitetsu-Tenjin Ōmuta Line
526	514	515	Nichinan Line	590	581	582	Nishitetsu-Tenjin Ōmuta Line
527	515	516	Nichinan Line	591	582	583	Nishitetsu-Tenjin Ōmuta Line
528	516	517	Nichinan Line	592	583	584	Nishitetsu-Tenjin Ōmuta Line
529	517	518	Nichinan Line	593	584	585	Nishitetsu-Tenjin Ōmuta Line
530	518	519	Nichinan Line	594	585	586	Nishitetsu-Tenjin Ōmuta Line
531	519	520	Nichinan Line	595	586	587	Nishitetsu-Tenjin Ōmuta Line
532	520	521	Nichinan Line	596	587	588	Nishitetsu-Tenjin Ōmuta Line
533	521	522	Nichinan Line	597	588	589	Nishitetsu-Tenjin Ōmuta Line
534	522	523	Nichinan Line	598	589	590	Nishitetsu-Tenjin Ōmuta Line
535	523	524	Nichinan Line	599	590	591	Nishitetsu-Tenjin Ōmuta Line
536	524	525	Nichinan Line	600	591	592	Nishitetsu-Tenjin Ōmuta Line
537	525	526	Nichinan Line	601	592	593	Nishitetsu-Tenjin Ōmuta Line
538	526	527	Nichinan Line	602	593	594	Nishitetsu-Tenjin Ōmuta Line
539	527	528	Nichinan Line	603	594	595	Nishitetsu-Tenjin Ōmuta Line
540	528	529	Nichinan Line	604	595	596	Nishitetsu-Tenjin Ōmuta Line
541	529	530	Nichinan Line	605	596	597	Nishitetsu-Tenjin Ōmuta Line
542	530	531	Nichinan Line	606	597	598	Nishitetsu-Tenjin Ōmuta Line
543	531	532	Nichinan Line	607	598	599	Nishitetsu-Tenjin Ōmuta Line
544	532	533	Nichinan Line	608	599	600	Nishitetsu-Tenjin Ōmuta Line
545	533	534	Nichinan Line	609	600	601	Nishitetsu-Tenjin Ōmuta Line
546	534	535	Nichinan Line	610	601	602	Nishitetsu-Tenjin Ōmuta Line
547	535	536	Nichinan Line	611	602	603	Nishitetsu-Tenjin Ōmuta Line
548	536	537	Nichinan Line	612	603	604	Nishitetsu-Tenjin Ōmuta Line
549	537	538	Nichinan Line	613	604	605	Nishitetsu-Tenjin Ōmuta Line
550	512	539	Nichinan Line	614	605	606	Nishitetsu-Tenjin Ōmuta Line
551	95	540	Ibusuki Makurazaki Line	615	606	607	Nishitetsu-Tenjin Ōmuta Line
552	540	541	Ibusuki Makurazaki Line	616	607	608	Nishitetsu-Tenjin Ōmuta Line
553	541	542	Ibusuki Makurazaki Line	617	608	609	Nishitetsu-Tenjin Ōmuta Line
554	542	543	Ibusuki Makurazaki Line	618	609	610	Nishitetsu-Tenjin Ōmuta Line
555	543	544	Ibusuki Makurazaki Line	619	610	611	Nishitetsu-Tenjin Ōmuta Line
556	544	545	Ibusuki Makurazaki Line	620	611	612	Nishitetsu-Tenjin Ōmuta Line
557	545	546	Ibusuki Makurazaki Line	621	612	613	Nishitetsu-Tenjin Ōmuta Line
558	546	547	Ibusuki Makurazaki Line	622	613	614	Nishitetsu-Tenjin Ōmuta Line
559	547	548	Ibusuki Makurazaki Line	623	614	615	Nishitetsu-Tenjin Ōmuta Line
560	548	549	Ibusuki Makurazaki Line	624	615	616	Nishitetsu-Tenjin Ōmuta Line
561	549	550	Ibusuki Makurazaki Line	625	616	617	Nishitetsu-Tenjin Ōmuta Line
562	550	551	Ibusuki Makurazaki Line	626	617	618	Nishitetsu-Tenjin Ōmuta Line
563	551	552	Ibusuki Makurazaki Line	627	618	619	Nishitetsu-Tenjin Ōmuta Line
564	552	553	Ibusuki Makurazaki Line	628	619	620	Nishitetsu-Tenjin Ōmuta Line
565	553	554	Ibusuki Makurazaki Line	629	620	621	Nishitetsu-Tenjin Ōmuta Line
566	554	555	Ibusuki Makurazaki Line	630	621	622	Nishitetsu-Tenjin Ōmuta Line
567	555	556	Ibusuki Makurazaki Line	631	622	623	Nishitetsu-Tenjin Ōmuta Line
568	556	557	Ibusuki Makurazaki Line	632	623	624	Nishitetsu-Tenjin Ōmuta Line
569	557	558	Ibusuki Makurazaki Line	633	624	59	Nishitetsu-Tenjin Ōmuta Line
570	558	559	Ibusuki Makurazaki Line	634	589	625	Nishitetsu-Dazaifu Line
571	559	560	Ibusuki Makurazaki Line	635	626	626	Nishitetsu-Dazaifu Line
572	560	561	Ibusuki Makurazaki Line	636	600	627	Nishitetsu-Amagi Line
573	561	562	Ibusuki Makurazaki Line	637	627	628	Nishitetsu-Amagi Line
574	562	563	Ibusuki Makurazaki Line	638	628	629	Nishitetsu-Amagi Line
575	563	564	Ibusuki Makurazaki Line	639	629	630	Nishitetsu-Amagi Line
576	564	565	Ibusuki Makurazaki Line	640	630	631	Nishitetsu-Amagi Line

Table 3. Link number and pair of nodes of the Kyushu railway network (Cont.)

No.	From node	To node	Line	No.	From node	To node	Line
641	631	632	Nishitetsu-Amagi Line	667	655	656	Fukuoka City Subway-Nanakuma Line
642	632	633	Nishitetsu-Amagi Line	668	656	657	Fukuoka City Subway-Nanakuma Line
643	633	634	Nishitetsu-Amagi Line	669	657	658	Fukuoka City Subway-Nanakuma Line
644	634	635	Nishitetsu-Amagi Line	670	658	659	Fukuoka City Subway-Nanakuma Line
645	635	636	Nishitetsu-Amagi Line	671	659	660	Fukuoka City Subway-Nanakuma Line
646	636	637	Nishitetsu-Amagi Line	672	660	661	Fukuoka City Subway-Nanakuma Line
647	652	638	Nishitetsu-Kaizuka Line	673	661	662	Fukuoka City Subway-Nanakuma Line
648	638	639	Nishitetsu-Kaizuka Line	674	662	663	Fukuoka City Subway-Nanakuma Line
649	639	29	Nishitetsu-Kaizuka Line	675	663	664	Fukuoka City Subway-Nanakuma Line
650	29	640	Nishitetsu-Kaizuka Line	676	664	665	Fukuoka City Subway-Nanakuma Line
651	640	641	Nishitetsu-Kaizuka Line	677	665	666	Fukuoka City Subway-Nanakuma Line
652	641	642	Nishitetsu-Kaizuka Line	678	666	667	Fukuoka City Subway-Nanakuma Line
653	642	643	Nishitetsu-Kaizuka Line	679	668	4	Shinkansen
654	643	242	Nishitetsu-Kaizuka Line	680	4	32	Shinkansen
655	242	644	Nishitetsu-Kaizuka Line	681	32	669	Shinkansen
656	644	645	Nishitetsu-Kaizuka Line	682	669	321	Shinkansen
657	32	646	Fukuoka City Subway-Kūkō Line	683	321	49	Shinkansen
658	646	647	Fukuoka City Subway-Kūkō Line	684	49	53	Shinkansen
659	280	648	Fukuoka City Subway-Hakozaki Line	685	53	670	Shinkansen
660	648	649	Fukuoka City Subway-Hakozaki Line	686	670	671	Shinkansen
661	649	650	Fukuoka City Subway-Hakozaki Line	687	671	72	Shinkansen
662	650	651	Fukuoka City Subway-Hakozaki Line	688	72	81	Shinkansen
663	651	652	Fukuoka City Subway-Hakozaki Line	689	81	575	Shinkansen
664	653	654	Fukuoka City Subway-Nanakuma Line	690	575	576	Shinkansen
665	654	578	Fukuoka City Subway-Nanakuma Line	691	576	83	Shinkansen
666	578	655	Fukuoka City Subway-Nanakuma Line	692	83	95	Shinkansen

4.2 Case Study of Tokyo Subway Network

This case study aims to evaluate the critical node and vulnerability section of the urban railway network in Tokyo that serves passengers in the world's largest city with about 13.5 million population in 2015 (Statistic Bureau of Japan, 2022). The network composes of two main operators, the Tokyo Metro subway and the Toei subway (Tokyo Metro, 2021), which are denser and more similar to the grid network. The network has 201 nodes and 252 links, which are divided into 13 lines, as illustrated in Fig. 8, Tables 4 and 5.

Figure 8. Subway network in Tokyo (Tokyo Metro, 2021)

From Fig. 8, railway lines by operators are

1) Toei subway
    1.1) Asakusa Line
    1.2) Mita Line
    1.3) Shinjuku Line
    1.4) Oedo Line
2) Tokyo Metro subway
    2.1) Ginza Line
    2.2) Marunouchi Line
    2.3) Hibiya Line
    2.4) Tozai Line
    2.5) Chiyoda Line
    2.6) Yurakucho Line
    2.7) Hanzomon Line
    2.8) Namboku Line
    2.9) Fukutoshin Line

Table 4. Node number and station name of the Tokyo subway network

No.	Station Name	Line	No.	Station Name	Line
1	Shibuya	Tokyo Metro Ginza Line	33	Yotsuya-sanchome	Tokyo Metro Marunouchi Line
2	Omote-sando	Tokyo Metro Ginza Line	34	Yotsuya	Tokyo Metro Marunouchi Line
3	Gaiemmae	Tokyo Metro Ginza Line	35	Kasumigaseki	Tokyo Metro Marunouchi Line
4	Aoyama-itchome	Tokyo Metro Ginza Line	36	Tokyo	Tokyo Metro Marunouchi Line
5	Akasaka-mitsuke/Nagatacho	Tokyo Metro Ginza Line	37	Otemachi	Tokyo Metro Marunouchi Line
6	Tameike-sanno/Kokkai-gijidomae	Tokyo Metro Ginza Line	38	Awajicho/Shin-ochanomizu/Ogawamachi	Tokyo Metro Marunouchi Line
7	Toranomon/Toranomon-hills	Tokyo Metro Ginza Line	39	Ochanomizu	Tokyo Metro Marunouchi Line
8	Shimbashi	Tokyo Metro Ginza Line	40	Hongo-sanchome	Tokyo Metro Marunouchi Line
9	Ginza/Ginza-itchome	Tokyo Metro Ginza Line	41	Korakuen/Kasuga	Tokyo Metro Marunouchi Line
10	Kyobashi	Tokyo Metro Ginza Line	42	Myogadani	Tokyo Metro Marunouchi Line
11	Nihombashi	Tokyo Metro Ginza Line	43	Shin-otsuka	Tokyo Metro Marunouchi Line
12	Mitsukoshimae	Tokyo Metro Ginza Line	44	Ikebukuro	Tokyo Metro Marunouchi Line
13	Kanda	Tokyo Metro Ginza Line	45	Naka-meguro	Tokyo Metro Hibiya Line
14	Suehirocho	Tokyo Metro Ginza Line	46	Ebisu	Tokyo Metro Hibiya Line
15	Ueno-hirokoji/Ueno-Okachimachi/Naka-okachimachi	Tokyo Metro Ginza Line	47	Hiro-o	Tokyo Metro Hibiya Line
16	Ueno	Tokyo Metro Ginza Line	48	Roppongi	Tokyo Metro Hibiya Line
17	Inaricho	Tokyo Metro Ginza Line	49	Kamiyacho	Tokyo Metro Hibiya Line
18	Tawaramachi	Tokyo Metro Ginza Line	50	Hibiya/Yurakucho	Tokyo Metro Hibiya Line
19	Asakusa	Tokyo Metro Ginza Line	51	Higashi-ginza	Tokyo Metro Hibiya Line
20	Ogikubo	Tokyo Metro Marunouchi Line	52	Tsukiji/Shintomicho	Tokyo Metro Hibiya Line
21	Minami-asagaya	Tokyo Metro Marunouchi Line	53	Hatchobori	Tokyo Metro Hibiya Line
22	Shin-koenji	Tokyo Metro Marunouchi Line	54	Kayabacho	Tokyo Metro Hibiya Line
23	Higashi-koenji	Tokyo Metro Marunouchi Line	55	Ningyocho/Suitengumae	Tokyo Metro Hibiya Line
24	Shin-nakano	Tokyo Metro Marunouchi Line	56	Kodemmacho	Tokyo Metro Hibiya Line
25	Honancho	Tokyo Metro Marunouchi Line	57	Akihabara/Iwamotocho	Tokyo Metro Hibiya Line
26	Nakano-fujimicho	Tokyo Metro Marunouchi Line	58	Iriya	Tokyo Metro Hibiya Line
27	NakanoShimbashi	Tokyo Metro Marunouchi Line	59	Minowa	Tokyo Metro Hibiya Line
28	Nakano-sakaue	Tokyo Metro Marunouchi Line	60	Minami-senju	Tokyo Metro Hibiya Line
29	Nishi-shinjuku	Tokyo Metro Marunouchi Line	61	Kita-senju	Tokyo Metro Hibiya Line
30	Shinjuku/Shinjuku-nishiguchi	Tokyo Metro Marunouchi Line	62	Nakano	Tokyo Metro Tozai Line
31	Shinjuku-sanchome	Tokyo Metro Marunouchi Line	63	Ochiai	Tokyo Metro Tozai Line
32	Shinjuku-gyoemmae	Tokyo Metro Marunouchi Line	64	Takadanobaba	Tokyo Metro Tozai Line

Table 4. Node number and station name of the Tokyo subway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
65	Waseda	Tokyo Metro Tozai Line	97	Chikatetsu-akatsuka	Tokyo Metro Yurakucho Line
66	Kagurazaka	Tokyo Metro Tozai Line	98	Heiwadai	Tokyo Metro Yurakucho Line
67	Iidabashi	Tokyo Metro Tozai Line	99	Hikawadai	Tokyo Metro Yurakucho Line
68	Kudanshita	Tokyo Metro Tozai Line	100	Kotake-mukaihara	Tokyo Metro Yurakucho Line
69	Takebashi	Tokyo Metro Tozai Line	101	Senkawa	Tokyo Metro Yurakucho Line
70	Monzen-nakacho	Tokyo Metro Tozai Line	102	Kanamecho	Tokyo Metro Yurakucho Line
71	Kiba	Tokyo Metro Tozai Line	103	Higashi-ikebukuro	Tokyo Metro Yurakucho Line
72	Toyocho	Tokyo Metro Tozai Line	104	Gokokuji	Tokyo Metro Yurakucho Line
73	Minami-sunamachi	Tokyo Metro Tozai Line	105	Edogawabashi	Tokyo Metro Yurakucho Line
74	Nishi-kasai	Tokyo Metro Tozai Line	106	Ichigaya	Tokyo Metro Yurakucho Line
75	Kasai	Tokyo Metro Tozai Line	107	Kojimachi	Tokyo Metro Yurakucho Line
76	Urayasu	Tokyo Metro Tozai Line	108	Sakuradamon	Tokyo Metro Yurakucho Line
77	Minami-gyotoku	Tokyo Metro Tozai Line	109	Tsukishima	Tokyo Metro Yurakucho Line
78	Gyotoku	Tokyo Metro Tozai Line	110	Toyosu	Tokyo Metro Yurakucho Line
79	Myoden	Tokyo Metro Tozai Line	111	Tatsumi	Tokyo Metro Yurakucho Line
80	Baraki-nakayama	Tokyo Metro Tozai Line	112	Shin-kiba	Tokyo Metro Yurakucho Line
81	Nishi-funabashi	Tokyo Metro Tozai Line	113	Hanzomon	Tokyo Metro Hanzomon Line
82	Yoyogi-uehara	Tokyo Metro Chiyoda Line	114	Jimbocho	Tokyo Metro Hanzomon Line
83	Yoyogi-koen	Tokyo Metro Chiyoda Line	115	Kiyosumi-shirakawa	Tokyo Metro Hanzomon Line
84	Meiji-jingumae	Tokyo Metro Chiyoda Line	116	Sumiyoshi	Tokyo Metro Hanzomon Line
85	Nogizaka	Tokyo Metro Chiyoda Line	117	Kinshicho	Tokyo Metro Hanzomon Line
86	Akasaka	Tokyo Metro Chiyoda Line	118	Oshiage	Tokyo Metro Hanzomon Line
87	Nijubashimae	Tokyo Metro Chiyoda Line	119	Meguro	Tokyo Metro Namboku Line
88	Yushima	Tokyo Metro Chiyoda Line	120	Shirokanedai	Tokyo Metro Namboku Line
89	Nezu	Tokyo Metro Chiyoda Line	121	Shirokane-takanawa	Tokyo Metro Namboku Line
90	Sendagi	Tokyo Metro Chiyoda Line	122	Azabu-juban	Tokyo Metro Namboku Line
91	Nishi-nippori	Tokyo Metro Chiyoda Line	123	Roppongi-itchome	Tokyo Metro Namboku Line
92	Machiya	Tokyo Metro Chiyoda Line	124	Todaimae	Tokyo Metro Namboku Line
93	Ayase	Tokyo Metro Chiyoda Line	125	Hon-komagome	Tokyo Metro Namboku Line
94	Kita-ayase	Tokyo Metro Chiyoda Line	126	Komagome	Tokyo Metro Namboku Line
95	Wakoshi	Tokyo Metro Yurakucho Line	127	Nishigahara	Tokyo Metro Namboku Line
96	Chikatetsu-narimasu	Tokyo Metro Yurakucho Line	128	Oji	Tokyo Metro Namboku Line

Table 4. Node number and station name of the Tokyo subway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
129	Oji-kamiya	Tokyo Metro Namboku Line	166	Takashimadaira	Toei Mita Line
130	Shimo	Tokyo Metro Namboku Line	167	Shin-takashimadaira	Toei Mita Line
131	Akabane-iwabuchi	Tokyo Metro Namboku Line	168	Nishi-takashimadaira	Toei Mita Line
132	Zoshigaya	Tokyo Metro Fukutoshin Line	169	Akebonobashi	Toei Shinjuku Line
133	Nishi-waseda	Tokyo Metro Fukutoshin Line	170	Hamacho	Toei Shinjuku Line
134	Higashi-shinjuku	Tokyo Metro Fukutoshin Line	171	Morishita	Toei Shinjuku Line
135	Kita-sando	Tokyo Metro Fukutoshin Line	172	Kikukawa	Toei Shinjuku Line
136	Nishi-magome	Toei Asakusa Line	173	Nishi-ojima	Toei Shinjuku Line
137	Magome	Toei Asakusa Line	174	Ojima	Toei Shinjuku Line
138	Nakanobu	Toei Asakusa Line	175	Higashi-ojima	Toei Shinjuku Line
139	Togoshi	Toei Asakusa Line	176	Funabori	Toei Shinjuku Line
140	Gotanda	Toei Asakusa Line	177	Ichinoe	Toei Shinjuku Line
141	Takanawadai	Toei Asakusa Line	178	Mizue	Toei Shinjuku Line
142	Sengakuji	Toei Asakusa Line	179	Shinozaki	Toei Shinjuku Line
143	Mita	Toei Asakusa Line	180	Motoyawata	Toei Shinjuku Line
144	Daimon	Toei Asakusa Line	181	Tochomae	Toei Oedo Line
145	Takaracho	Toei Asakusa Line	182	Wakamatsu-kawada	Toei Oedo Line
146	Higashi-nihombashi/Bakuro-yokoyama	Toei Asakusa Line	183	Ushigome-yanagicho	Toei Oedo Line
147	Asakusabashi	Toei Asakusa Line	184	Ushigome-kagurazaka	Toei Oedo Line
148	Kuramae	Toei Asakusa Line	185	Shin-okachimachi	Toei Oedo Line
149	Honjo-azumabashi	Toei Asakusa Line	186	Ryōgoku	Toei Oedo Line
150	Shibakoen	Toei Mita Line	187	Kachidoki	Toei Oedo Line
151	Onarimon	Toei Mita Line	188	Tsukijishijō	Toei Oedo Line
152	Uchisaiwaicho	Toei Mita Line	189	Shiodome	Toei Oedo Line
153	Suidobashi	Toei Mita Line	190	Akabanebashi	Toei Oedo Line
154	Hakusan	Toei Mita Line	191	Kokuritsu-Kyōgijō	Toei Oedo Line
155	Sengoku	Toei Mita Line	192	Yoyogi	Toei Oedo Line
156	Sugamo	Toei Mita Line	193	Nishi-shinjuku-gochome	Toei Oedo Line
157	Nishi-sugamo	Toei Mita Line	194	Higashi-Nakano	Toei Oedo Line
158	Shin-itabashi	Toei Mita Line	195	Nakai	Toei Oedo Line
159	Itabashikuyakushomae	Toei Mita Line	196	Ochiai-minami-nagasaki	Toei Oedo Line
160	Itabashihoncho	Toei Mita Line	197	Shin-egota	Toei Oedo Line
161	Motohasunuma	Toei Mita Line	198	Nerima	Toei Oedo Line
162	Shimura-sakaue	Toei Mita Line	199	Toshimaen	Toei Oedo Line
163	Shimura-sanchome	Toei Mita Line	200	Nerima-kasugachō	Toei Oedo Line
164	Hasune	Toei Mita Line	201	Hikarigaoka	Toei Oedo Line
165	Nishidai	Toei Mita Line			

Table 5. Link number and pair of nodes of the Tokyo subway network

No.	From node	To node	Line	No.	From node	To node	Line
1	1	2	Tokyo Metro Ginza Line	13	13	14	Tokyo Metro Ginza Line
2	2	3	Tokyo Metro Ginza Line	14	14	15	Tokyo Metro Ginza Line
3	3	4	Tokyo Metro Ginza Line	15	15	16	Tokyo Metro Ginza Line
4	4	5	Tokyo Metro Ginza Line	16	16	17	Tokyo Metro Ginza Line
5	5	6	Tokyo Metro Ginza Line	17	17	18	Tokyo Metro Ginza Line
6	6	7	Tokyo Metro Ginza Line	18	18	19	Tokyo Metro Ginza Line
7	7	8	Tokyo Metro Ginza Line	19	20	21	Tokyo Metro Marunouchi Line
8	8	9	Tokyo Metro Ginza Line	20	21	22	Tokyo Metro Marunouchi Line
9	9	10	Tokyo Metro Ginza Line	21	22	23	Tokyo Metro Marunouchi Line
10	10	11	Tokyo Metro Ginza Line	22	23	24	Tokyo Metro Marunouchi Line
11	11	12	Tokyo Metro Ginza Line	23	24	28	Tokyo Metro Marunouchi Line
12	12	13	Tokyo Metro Ginza Line	24	25	26	Tokyo Metro Marunouchi Line

Table 5. Link number and pair of nodes of the Tokyo subway network (Cont.)

No.	From node	To node	Line	No.	From node	To node	Line
25	26	27	Tokyo Metro Marunouchi Line	89	84	2	Tokyo Metro Chiyoda Line
26	27	28	Tokyo Metro Marunouchi Line	90	2	85	Tokyo Metro Chiyoda Line
27	28	29	Tokyo Metro Marunouchi Line	91	85	86	Tokyo Metro Chiyoda Line
28	29	30	Tokyo Metro Marunouchi Line	92	86	6	Tokyo Metro Chiyoda Line
29	30	31	Tokyo Metro Marunouchi Line	93	50	87	Tokyo Metro Chiyoda Line
30	31	32	Tokyo Metro Marunouchi Line	94	87	37	Tokyo Metro Chiyoda Line
31	32	33	Tokyo Metro Marunouchi Line	95	38	88	Tokyo Metro Chiyoda Line
32	33	34	Tokyo Metro Marunouchi Line	96	88	89	Tokyo Metro Chiyoda Line
33	34	5	Tokyo Metro Marunouchi Line	97	89	90	Tokyo Metro Chiyoda Line
34	6	35	Tokyo Metro Marunouchi Line	98	90	91	Tokyo Metro Chiyoda Line
35	35	9	Tokyo Metro Marunouchi Line	99	91	92	Tokyo Metro Chiyoda Line
36	9	36	Tokyo Metro Marunouchi Line	100	92	61	Tokyo Metro Chiyoda Line
37	36	37	Tokyo Metro Marunouchi Line	101	61	93	Tokyo Metro Chiyoda Line
38	37	38	Tokyo Metro Marunouchi Line	102	93	94	Tokyo Metro Chiyoda Line
39	38	39	Tokyo Metro Marunouchi Line	103	95	96	Tokyo Metro Yurakucho Line
40	39	40	Tokyo Metro Marunouchi Line	104	96	97	Tokyo Metro Yurakucho Line
41	40	41	Tokyo Metro Marunouchi Line	105	97	98	Tokyo Metro Yurakucho Line
42	41	42	Tokyo Metro Marunouchi Line	106	98	99	Tokyo Metro Yurakucho Line
43	42	43	Tokyo Metro Marunouchi Line	107	99	100	Tokyo Metro Yurakucho Line
44	43	44	Tokyo Metro Marunouchi Line	108	100	101	Tokyo Metro Yurakucho Line
45	45	46	Tokyo Metro Hibiya Line	109	101	102	Tokyo Metro Yurakucho Line
46	46	47	Tokyo Metro Hibiya Line	110	102	44	Tokyo Metro Yurakucho Line
47	47	48	Tokyo Metro Hibiya Line	111	44	103	Tokyo Metro Yurakucho Line
48	48	49	Tokyo Metro Hibiya Line	112	103	104	Tokyo Metro Yurakucho Line
49	49	7	Tokyo Metro Hibiya Line	113	104	105	Tokyo Metro Yurakucho Line
50	7	35	Tokyo Metro Hibiya Line	114	105	67	Tokyo Metro Yurakucho Line
51	35	50	Tokyo Metro Hibiya Line	115	67	106	Tokyo Metro Yurakucho Line
52	50	9	Tokyo Metro Hibiya Line	116	106	107	Tokyo Metro Yurakucho Line
53	9	51	Tokyo Metro Hibiya Line	117	107	5	Tokyo Metro Yurakucho Line
54	51	52	Tokyo Metro Hibiya Line	118	5	108	Tokyo Metro Yurakucho Line
55	52	53	Tokyo Metro Hibiya Line	119	108	9	Tokyo Metro Yurakucho Line
56	53	54	Tokyo Metro Hibiya Line	120	9	52	Tokyo Metro Yurakucho Line
57	54	55	Tokyo Metro Hibiya Line	121	52	109	Tokyo Metro Yurakucho Line
58	55	56	Tokyo Metro Hibiya Line	122	109	110	Tokyo Metro Yurakucho Line
59	56	57	Tokyo Metro Hibiya Line	123	110	111	Tokyo Metro Yurakucho Line
60	57	15	Tokyo Metro Hibiya Line	124	111	112	Tokyo Metro Yurakucho Line
61	16	58	Tokyo Metro Hibiya Line	125	2	4	Tokyo Metro Hanzomon Line
62	58	59	Tokyo Metro Hibiya Line	126	5	113	Tokyo Metro Hanzomon Line
63	59	60	Tokyo Metro Hibiya Line	127	113	68	Tokyo Metro Hanzomon Line
64	60	61	Tokyo Metro Hibiya Line	128	68	114	Tokyo Metro Hanzomon Line
65	62	63	Tokyo Metro Tozai Line	129	114	37	Tokyo Metro Hanzomon Line
66	63	64	Tokyo Metro Tozai Line	130	37	12	Tokyo Metro Hanzomon Line
67	64	65	Tokyo Metro Tozai Line	131	12	55	Tokyo Metro Hanzomon Line
68	65	66	Tokyo Metro Tozai Line	132	55	115	Tokyo Metro Hanzomon Line
69	66	67	Tokyo Metro Tozai Line	133	115	116	Tokyo Metro Hanzomon Line
70	67	68	Tokyo Metro Tozai Line	134	116	117	Tokyo Metro Hanzomon Line
71	68	69	Tokyo Metro Tozai Line	135	117	118	Tokyo Metro Hanzomon Line
72	69	37	Tokyo Metro Tozai Line	136	119	120	Tokyo Metro Namboku Line
73	37	11	Tokyo Metro Tozai Line	137	120	121	Tokyo Metro Namboku Line
74	11	54	Tokyo Metro Tozai Line	138	121	122	Tokyo Metro Namboku Line
75	54	70	Tokyo Metro Tozai Line	139	122	123	Tokyo Metro Namboku Line
76	70	71	Tokyo Metro Tozai Line	140	123	6	Tokyo Metro Namboku Line
77	71	72	Tokyo Metro Tozai Line	141	34	106	Tokyo Metro Namboku Line
78	72	73	Tokyo Metro Tozai Line	142	67	41	Tokyo Metro Namboku Line
79	73	74	Tokyo Metro Tozai Line	143	41	124	Tokyo Metro Namboku Line
80	74	75	Tokyo Metro Tozai Line	144	124	125	Tokyo Metro Namboku Line
81	75	76	Tokyo Metro Tozai Line	145	125	126	Tokyo Metro Namboku Line
82	76	77	Tokyo Metro Tozai Line	146	126	127	Tokyo Metro Namboku Line
83	77	78	Tokyo Metro Tozai Line	147	127	128	Tokyo Metro Namboku Line
84	78	79	Tokyo Metro Tozai Line	148	128	129	Tokyo Metro Namboku Line
85	79	80	Tokyo Metro Tozai Line	149	129	130	Tokyo Metro Namboku Line
86	80	81	Tokyo Metro Tozai Line	150	130	131	Tokyo Metro Namboku Line
87	82	83	Tokyo Metro Chiyoda Line	151	44	132	Tokyo Metro Fukutoshin Line
88	83	84	Tokyo Metro Chiyoda Line	152	132	133	Tokyo Metro Fukutoshin Line

Table 5. Link number and pair of nodes of the Tokyo subway network (Cont.)

No.	From node	To node	Line	No.	From node	To node	Line
153	133	134	Tokyo Metro Fukutoshin Line	203	114	38	Toei Shinjuku Line
154	134	31	Tokyo Metro Fukutoshin Line	204	38	57	Toei Shinjuku Line
155	31	135	Tokyo Metro Fukutoshin Line	205	57	146	Toei Shinjuku Line
156	135	84	Tokyo Metro Fukutoshin Line	206	146	170	Toei Shinjuku Line
157	84	1	Tokyo Metro Fukutoshin Line	207	170	171	Toei Shinjuku Line
158	136	137	Toei Asakusa Line	208	171	172	Toei Shinjuku Line
159	137	138	Toei Asakusa Line	209	172	116	Toei Shinjuku Line
160	138	139	Toei Asakusa Line	210	116	173	Toei Shinjuku Line
161	139	140	Toei Asakusa Line	211	173	174	Toei Shinjuku Line
162	140	141	Toei Asakusa Line	212	174	175	Toei Shinjuku Line
163	141	142	Toei Asakusa Line	213	175	176	Toei Shinjuku Line
164	142	143	Toei Asakusa Line	214	176	177	Toei Shinjuku Line
165	143	144	Toei Asakusa Line	215	177	178	Toei Shinjuku Line
166	144	8	Toei Asakusa Line	216	178	179	Toei Shinjuku Line
167	8	51	Toei Asakusa Line	217	179	180	Toei Shinjuku Line
168	51	145	Toei Asakusa Line	218	181	30	Toei Oedo Line
169	145	11	Toei Asakusa Line	219	30	134	Toei Oedo Line
170	11	55	Toei Asakusa Line	220	134	182	Toei Oedo Line
171	55	146	Toei Asakusa Line	221	182	183	Toei Oedo Line
172	146	147	Toei Asakusa Line	222	183	184	Toei Oedo Line
173	147	148	Toei Asakusa Line	223	184	67	Toei Oedo Line
174	148	19	Toei Asakusa Line	224	40	15	Toei Oedo Line
175	19	149	Toei Asakusa Line	225	15	185	Toei Oedo Line
176	149	118	Toei Asakusa Line	226	185	148	Toei Oedo Line
177	121	143	Toei Mita Line	227	148	186	Toei Oedo Line
178	143	150	Toei Mita Line	228	186	171	Toei Oedo Line
179	150	151	Toei Mita Line	229	171	115	Toei Oedo Line
180	151	152	Toei Mita Line	230	115	70	Toei Oedo Line
181	152	50	Toei Mita Line	231	70	109	Toei Oedo Line
182	50	37	Toei Mita Line	232	109	187	Toei Oedo Line
183	114	153	Toei Mita Line	233	187	188	Toei Oedo Line
184	153	41	Toei Mita Line	234	188	189	Toei Oedo Line
185	41	154	Toei Mita Line	235	189	144	Toei Oedo Line
186	154	155	Toei Mita Line	236	144	190	Toei Oedo Line
187	155	156	Toei Mita Line	237	190	122	Toei Oedo Line
188	156	157	Toei Mita Line	238	122	48	Toei Oedo Line
189	157	158	Toei Mita Line	239	48	4	Toei Oedo Line
190	158	159	Toei Mita Line	240	4	191	Toei Oedo Line
191	159	160	Toei Mita Line	241	191	192	Toei Oedo Line
192	160	161	Toei Mita Line	242	192	30	Toei Oedo Line
193	161	162	Toei Mita Line	243	181	193	Toei Oedo Line
194	162	163	Toei Mita Line	244	193	28	Toei Oedo Line
195	163	164	Toei Mita Line	245	28	194	Toei Oedo Line
196	164	165	Toei Mita Line	246	194	195	Toei Oedo Line
197	165	166	Toei Mita Line	247	195	196	Toei Oedo Line
198	166	167	Toei Mita Line	248	196	197	Toei Oedo Line
199	167	168	Toei Mita Line	249	197	198	Toei Oedo Line
200	31	169	Toei Shinjuku Line	250	198	199	Toei Oedo Line
201	169	106	Toei Shinjuku Line	251	199	200	Toei Oedo Line
202	106	68	Toei Shinjuku Line	252	200	201	Toei Oedo Line

4.3 Case Study of Osaka Subway Network

The Osaka subway network is similar to the Tokyo subway that it is dense, and the central area is similar to the grid network. This network, operated by the Osaka metro, served about 2.5 million daily ridership in 2019 (Osaka Metro, 2022). The network has 106 nodes and 121 links, which are divided into 8 lines plus 1 automated guideway transit line (Osaka Metro, 2020), as illustrated in Fig. 9, Tables 6 and 7.

From Fig. 9, the Osaka subway lines are

1) Midosuji Line
2) Tanimachi Line
3) Yotsubashi Line
4) Chūō Line
5) Sennichimae Line
6) Sakaisuji Line
7) Nagahori Tsurumi-ryokuchi Line
8) Imazatosuji Line
9) Nankō Port Town Line (Automated guideway transit)

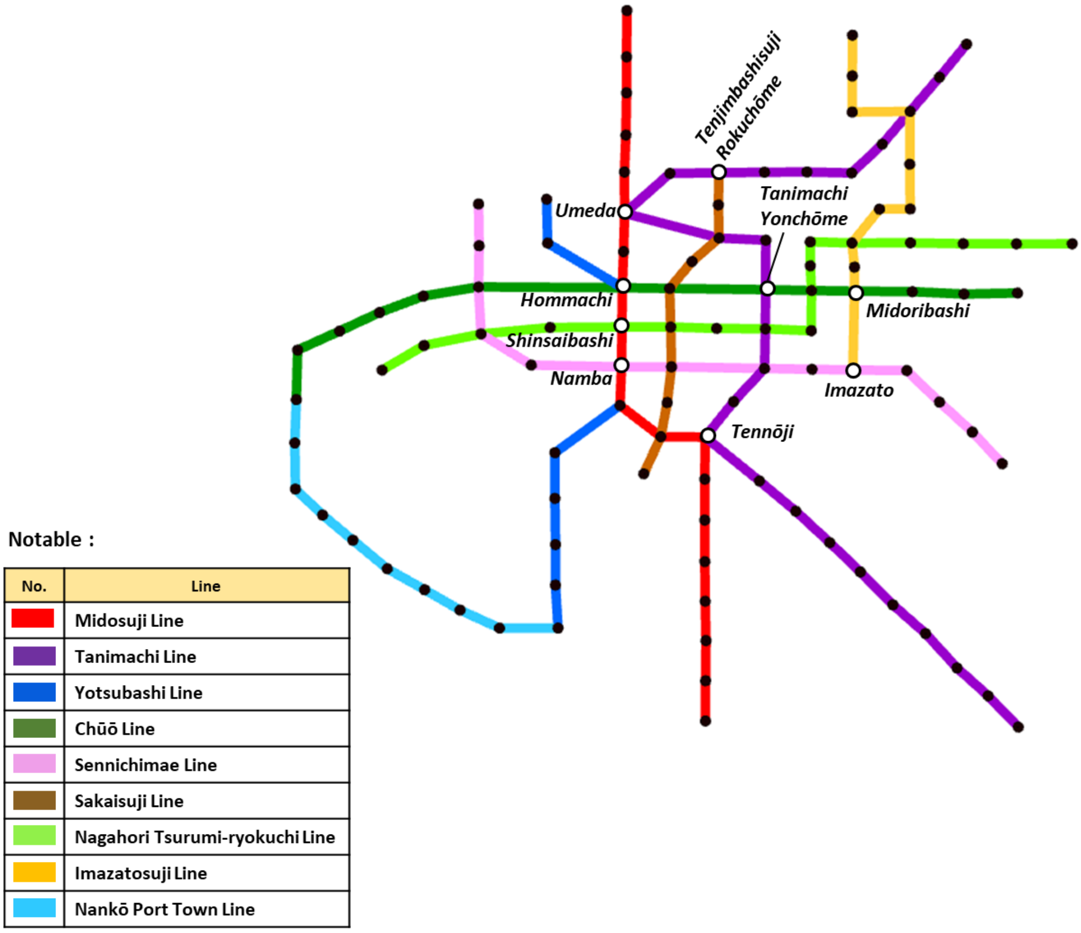


Figure 9. Subway network in Osaka (Osaka Metro, 2020)

Table 6. Node number and station name of the Osaka subway network

No.	Station Name	Line	No.	Station Name	Line
1	Esaka	Midosuji Line	6	Umeda	Midosuji Line
2	Higashi-Mikuni	Midosuji Line	7	Yodoyabashi (Osaka City Hall)	Midosuji Line
3	Shin-Ōsaka	Midosuji Line	8	Hommachi (Semba-nishi)	Midosuji Line
4	Nishinakajima-Minamigata	Midosuji Line	9	Shinsaibashi	Midosuji Line
5	Nakatsu	Midosuji Line	10	Namba	Midosuji Line

Table 6. Node number and station name of the Osaka subway network (Cont.)

No.	Station Name	Line	No.	Station Name	Line
11	Daikokuchō	Midosuji Line	59	Morinomiya	Chūō Line
12	Dōbutsuen-mae (Shinsekai)	Midosuji Line	60	Midoribashi	Chūō Line
13	Tennōji	Midosuji Line	61	Fukaebashi	Chūō Line
14	Shōwachō	Midosuji Line	62	Takaida	Chūō Line
15	Nishitanabe	Midosuji Line	63	Nagata	Chūō Line
16	Nagai	Midosuji Line	64	Nodahanshin	Sennichimae Line
17	Abiko	Midosuji Line	65	Tamagawa	Sennichimae Line
18	Kitahanada	Midosuji Line	66	Nishi-Nagahori	Sennichimae Line
19	Shinkanaoka	Midosuji Line	67	Sakuragawa	Sennichimae Line
20	Nakamozu	Midosuji Line	68	Nippombashi	Sennichimae Line
21	Dainichi	Tanimachi Line	69	Tsuruhashi	Sennichimae Line
22	Moriguchi	Tanimachi Line	70	Imazato	Sennichimae Line
23	Taishibashi-Imaichi	Tanimachi Line	71	Shin-Fukae	Sennichimae Line
24	Sembayashi-Omiya	Tanimachi Line	72	Shōji	Sennichimae Line
25	Sekime-Takadono	Tanimachi Line	73	Kita-Tatsumi	Sennichimae Line
26	Noe-Uchindai	Tanimachi Line	74	Minami-Tatsumi	Sennichimae Line
27	Miyakojima	Tanimachi Line	75	Ōgimachi	Sakaisuji Line
28	Tenjimbashisuji Rokuchōme	Tanimachi Line	76	Kitahama	Sakaisuji Line
29	Nakazakichō	Tanimachi Line	77	Nagahoribashi	Sakaisuji Line
30	Minami-morimachi	Tanimachi Line	78	Ebisuchō (Nippombashi-suji)	Sakaisuji Line
31	Temmabashi	Tanimachi Line	79	Tengachaya	Sakaisuji Line
32	Tanimachi Yonchōme	Tanimachi Line	80	Taishō	Nagahori Tsurumi-ryokuchi Line
33	Tanimachi Rokuchōme	Tanimachi Line	81	Dome-mae Chiyozaki (Kyocera Dome Osaka)	Nagahori Tsurumi-ryokuchi Line
34	Tanimachi Kyūchōme	Tanimachi Line	82	Nishiōhashi	Nagahori Tsurumi-ryokuchi Line
35	Shitennōji-mae Yūhigaoka	Tanimachi Line	83	Matsuyamachi	Nagahori Tsurumi-ryokuchi Line
36	Abeno	Tanimachi Line	84	Tamatsukuri	Nagahori Tsurumi-ryokuchi Line
37	Fuminosato	Tanimachi Line	85	Osaka Business Park (Osaka-jo Hall)	Nagahori Tsurumi-ryokuchi Line
38	Tanabe	Tanimachi Line	86	Kyōbashi	Nagahori Tsurumi-ryokuchi Line
39	Komagawa-Nakano	Tanimachi Line	87	Gamō-yonchōme	Nagahori Tsurumi-ryokuchi Line
40	Hirano	Tanimachi Line	88	Imafuku-Tsurumi	Nagahori Tsurumi-ryokuchi Line
41	Kire-Uriwari	Tanimachi Line	89	Yokozutsumi	Nagahori Tsurumi-ryokuchi Line
42	Deto	Tanimachi Line	90	Tsurumi-ryokuchi	Nagahori Tsurumi-ryokuchi Line
43	Nagahara	Tanimachi Line	91	Kadoma-minami	Nagahori Tsurumi-ryokuchi Line
44	Yaominami	Tanimachi Line	92	Itakano	Imazatosuji Line
45	Nishi-Umeda	Yotsubashi Line	93	Zuikō Yonchōme	Imazatosuji Line
46	Higobashi	Yotsubashi Line	94	Daidō-Toyosato	Imazatosuji Line
47	Hanazonochō	Yotsubashi Line	95	Shimizu	Imazatosuji Line
48	Kishinosato	Yotsubashi Line	96	Shimmori-Furuichi	Imazatosuji Line
49	Tamade	Yotsubashi Line	97	Sekime-Seiiku	Imazatosuji Line
50	Kitakagaya	Yotsubashi Line	98	Shigino	Imazatosuji Line
51	Suminoekōen	Yotsubashi Line	99	Trade Center-mae	Nankō Port Town Line
52	Cosmosquare	Chūō Line	100	Nakafuto	Nankō Port Town Line
53	Osakako (Tempozan)	Chūō Line	101	Port Town-nishi	Nankō Port Town Line
54	Asashiobashi	Chūō Line	102	Port Town-higashi	Nankō Port Town Line
55	Bentencho	Chūō Line	103	Ferry Terminal	Nankō Port Town Line
56	Kujo	Chūō Line	104	Nankō-higashi	Nankō Port Town Line
57	Awaza	Chūō Line	105	Nankōguchi	Nankō Port Town Line
58	Sakaisuji-Hommachi (Semba-higashi)	Chūō Line	106	Hirabayashi	Nankō Port Town Line

Table 7. Link number and pair of nodes of the Osaka subway network

No.	From node	To node	Line	No.	From node	To node	Line
1	1	2	Midosuji Line	51	50	51	Yotsubashi Line
2	2	3	Midosuji Line	52	52	53	Chūō Line
3	3	4	Midosuji Line	53	53	54	Chūō Line
4	4	5	Midosuji Line	54	54	55	Chūō Line
5	5	6	Midosuji Line	55	55	56	Chūō Line
6	6	7	Midosuji Line	56	56	57	Chūō Line
7	7	8	Midosuji Line	57	57	8	Chūō Line
8	8	9	Midosuji Line	58	5	58	Chūō Line
9	9	10	Midosuji Line	59	58	32	Chūō Line
10	10	11	Midosuji Line	60	32	59	Chūō Line
11	11	12	Midosuji Line	61	59	60	Chūō Line
12	12	13	Midosuji Line	62	60	61	Chūō Line
13	13	14	Midosuji Line	63	61	62	Chūō Line
14	14	15	Midosuji Line	64	62	63	Chūō Line
15	15	16	Midosuji Line	65	64	65	Sennichimae Line
16	16	17	Midosuji Line	66	65	57	Sennichimae Line
17	17	18	Midosuji Line	67	57	66	Sennichimae Line
18	18	19	Midosuji Line	68	66	67	Sennichimae Line
19	19	20	Midosuji Line	69	67	10	Sennichimae Line
20	21	22	Tanimachi Line	70	10	68	Sennichimae Line
21	22	23	Tanimachi Line	71	68	34	Sennichimae Line
22	23	24	Tanimachi Line	72	34	69	Sennichimae Line
23	24	25	Tanimachi Line	73	69	70	Sennichimae Line
24	25	26	Tanimachi Line	74	70	71	Sennichimae Line
25	26	27	Tanimachi Line	75	71	72	Sennichimae Line
26	27	28	Tanimachi Line	76	72	73	Sennichimae Line
27	28	29	Tanimachi Line	77	73	74	Sennichimae Line
28	29	6	Tanimachi Line	78	28	75	Sakaisuji Line
29	6	30	Tanimachi Line	79	75	30	Sakaisuji Line
30	30	31	Tanimachi Line	80	30	76	Sakaisuji Line
31	31	32	Tanimachi Line	81	76	58	Sakaisuji Line
32	32	33	Tanimachi Line	82	58	77	Sakaisuji Line
33	33	34	Tanimachi Line	83	77	68	Sakaisuji Line
34	34	35	Tanimachi Line	84	68	78	Sakaisuji Line
35	35	13	Tanimachi Line	85	78	12	Sakaisuji Line
36	13	36	Tanimachi Line	86	12	79	Sakaisuji Line
37	36	37	Tanimachi Line	87	80	81	Nagahori Tsurumi-ryokuchi Line
38	37	38	Tanimachi Line	88	81	66	Nagahori Tsurumi-ryokuchi Line
39	38	39	Tanimachi Line	89	66	82	Nagahori Tsurumi-ryokuchi Line
40	39	40	Tanimachi Line	90	82	9	Nagahori Tsurumi-ryokuchi Line
41	40	41	Tanimachi Line	91	9	77	Nagahori Tsurumi-ryokuchi Line
42	41	42	Tanimachi Line	92	77	83	Nagahori Tsurumi-ryokuchi Line
43	42	43	Tanimachi Line	93	83	33	Nagahori Tsurumi-ryokuchi Line
44	43	44	Tanimachi Line	94	33	84	Nagahori Tsurumi-ryokuchi Line
45	45	46	Yotsubashi Line	95	84	59	Nagahori Tsurumi-ryokuchi Line
46	46	8	Yotsubashi Line	96	59	85	Nagahori Tsurumi-ryokuchi Line
47	11	47	Yotsubashi Line	97	85	86	Nagahori Tsurumi-ryokuchi Line
48	47	48	Yotsubashi Line	98	86	87	Nagahori Tsurumi-ryokuchi Line
49	48	49	Yotsubashi Line	99	87	88	Nagahori Tsurumi-ryokuchi Line
50	49	50	Yotsubashi Line	100	88	89	Nagahori Tsurumi-ryokuchi Line

Table 7. Link number and pair of nodes of the Osaka subway network (Cont.)

No.	**From node**	**To node**	**Line**	**No.**	**From node**	**To node**	**Line**
101	89	90	Nagahori Tsurumi-ryokuchi Line	112	60	70	Imazatosuji Line
102	90	91	Nagahori Tsurumi-ryokuchi Line	113	52	99	Nankō Port Town Line
103	92	93	Imazatosuji Line	114	99	100	Nankō Port Town Line
104	93	94	Imazatosuji Line	115	100	101	Nankō Port Town Line
105	94	23	Imazatosuji Line	116	101	102	Nankō Port Town Line
106	23	95	Imazatosuji Line	117	102	103	Nankō Port Town Line
107	95	96	Imazatosuji Line	118	103	104	Nankō Port Town Line
108	96	97	Imazatosuji Line	119	104	105	Nankō Port Town Line
109	97	87	Imazatosuji Line	120	105	106	Nankō Port Town Line
110	87	98	Imazatosuji Line	121	106	51	Nankō Port Town Line
111	98	60	Imazatosuji Line				

References

[1] Bastian M., Heymann S., & Jacomy M. (2009). Gephi: an open source software for exploring and manipulating networks. *International AAAI Conference on Weblogs and Social Media.*

[2] Fukuoka City Subway. (2020). Information of the Transportation Bureau. (https://subway.city.fukuoka.lg.jp/subway/about/pdf/subway_05.pdf; Accessed November 17, 2020) (In Japanese)

[3] Google Map. (2020) Google Map. (https://www.google.co.th/maps/; Accessed July 17, 2020)

[4] JR Kyushu Railway Company. (2020). Face Sheet 2020. (https://www.jrkyushu.co.jp/ company/ir_eng/library/fact_sheet/; Accessed November 17, 2020)

[5] Kiyasu, K. (2017) Efforts for recovery of roads from the 2016 Kumamoto earthquake. (http://www.nilim.go.jp/english/hottopics/pdf/im1.pdf; Accessed November 18, 2020)

[6] National land information division, national spatial planning and regional policy bureau, MLIT of Japan (2020). railroad data. (https://nlftp.mlit.go.jp/ksj/gml/datalist /KsjTmplt-N02-v2_3.html; Accessed December 10, 2020) (In Japanese)

[7] Nishitetsu. (2020). Railway business. (https://www.nishitetsu.co.jp/group/ enterprise_1.html; Accessed November 17, 2020) (In Japanese)

[8] Osaka Metro (2020). List of routes. (https://subway.osakametro.co.jp/en/station_guide/; Accessed February 16, 2020)

[9] Osaka Metro (2022). Company Profile. (https://www.osakametro.co.jp/company/company_profile/kaisya_gaiyou.php; Accessed June 17, 2020)

[10] Statistics Bureau of Japan (2022). Statistical Handbook of Japan 2021. (https://www.stat.go.jp/english/data/handbook/c0117.html; Accessed January 16, 2022)

[11] The Asahi Shimbun. (2020). Rail networks in Kyushu severed by torrential rain disaster. (https://www.asahi.com/ajw/articles/13529875; Accessed February 2, 2022)

[12] Tokyo Metro (2021). Subway Map. (https://www.tokyometro.jp/en/subwaymap/; Accessed December 1, 2020)

Chapter 5

Network Centrality Analysis for the Railway Network

5.1 Overview of Centrality Analysis

This research uses multiple centrality analyses to identify the most critical node, which represents the most influence station within the network. Any node with a higher centrality means that the node has more influence.

The centralities comprise of degree centrality, closeness centrality, betweenness centrality, eigenvector centrality, and information centrality. All five methods will be compared with their results and performance, then identifying the critical nodes that consider important stations in multiple views. In addition, each centrality type result will be divided into five levels to help classify the important level of nodes in the network.

5.2 Square Grid Network Testing for All Five Centralities

The testing was conducted to compare the centrality's performance, which was divided into two main parts. First is the computing time testing, and another is grid network testing to compare node centrality under the same network.

5.2.1 The calculation time testing of centrality analyses

The computing time testing was conducted with six different sizes of square grid network, 25, 100, 225, 400, 625, and 900 nodes matrix, in which each condition was testing by 30 rounds. The purpose is to compare the calculation time of all five centrality methods. All testing computed by MATLAB program on a computer with Intel(R) Core(TM) i5-4570 CPU @ 3.20GHz, and 16.0 GB of RAM due to more convenience for linking data with spreadsheet software, especially Microsoft Excel, and ease of the testing and checking algorithm.

Table 8. The average processing time of the centrality analyses on various sizes of square grid networks

	Average calculating time (seconds)				
Grid Network Size	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
5 × 5 (25 nodes)	1.9618	2.0760	2.0831	1.9924	0.0240
10 × 10 (100 nodes)	1.9656	2.0761	2.1086	1.9966	0.2247
15 × 15 (225 nodes)	2.0006	2.1201	2.1706	2.0533	2.5422
20 × 20 (400 nodes)	2.1175	2.2393	2.3212	2.2060	28.9578
25 × 25 (625 nodes)	2.4115	2.5773	2.6405	2.5136	174.6976
30 × 30 (900 nodes)	3.0439	3.1688	3.2916	3.1565	715.8645

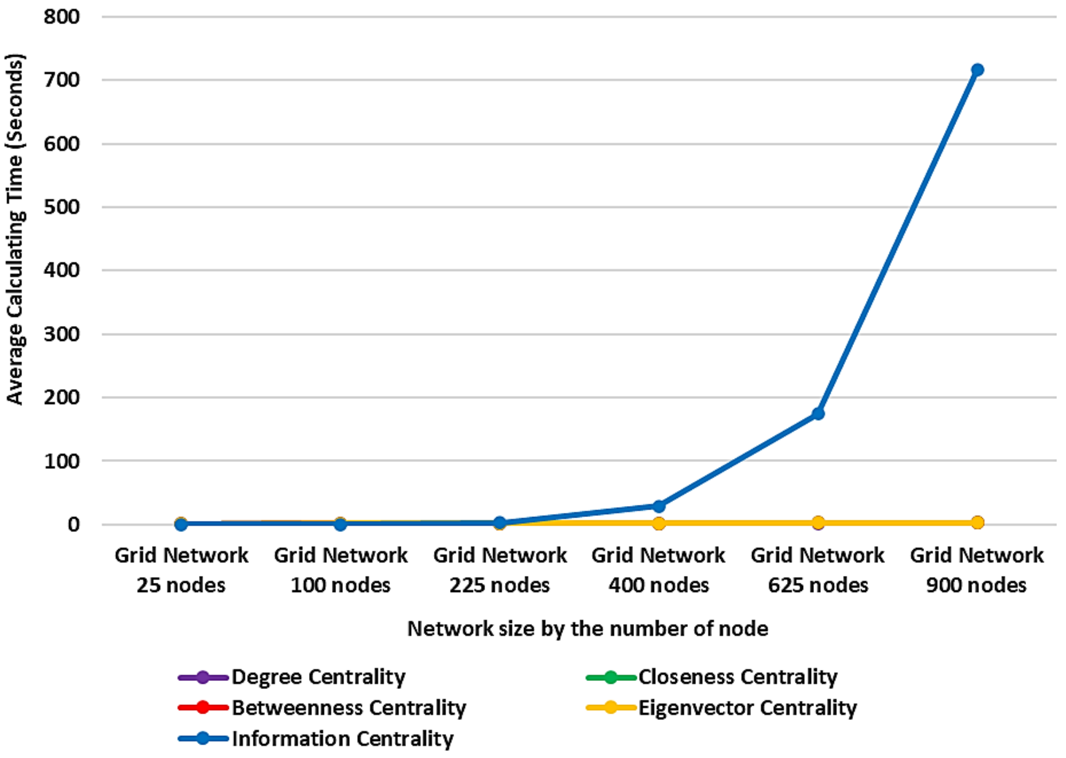


Figure 10. Calculation time results of centralities on the six different sizes of square grid networks

From Table 8 and Fig. 10, the degree, eigenvector, closeness, and betweenness centralities were not significantly different, even though the degree centrality used the least average calculating time from all six different sizes of square grid networks. However, the information centrality spent the longest computing time on the large square grid network, especially with the 900 nodes network that consumed about 716 seconds on average (nearly 12 minutes). The main reason is that information centrality needs two stages to measure the critical node, which according to Eq. 3.13 and 3.14. The first stage is calculating the global efficiency of the entire network and the next stage is removing every link that connected the measured node and calculating the global efficiency of the remaining network.

5.2.2 Comparison of the centrality of square grid networks

The comparison of all five centrality analyses on the square grid network was conducted on the 625 nodes-network due to its similar size to the Kyushu railway network. The result of each case was divided into five levels and independent from each other.

From Fig. 11, the characteristic of the example network has the same size in each dimension, so the centrality level of each node is symmetric in every case. The degree centrality measured the only number of links that connected each node, so it cannot point to the most critical node in this case due to the same degree of inner nodes. The eigenvector centrality result showed the very high centrality nodes in the more specific area at the center of the network as well as closeness, betweenness, and information centralities. However, the closeness centrality showed its weak point: the range of centrality levels, especially at the moderate level, is narrow, and the very high-level centrality nodes are assembled as a large cluster group. This condition made this method is difficult to classify each level for priority management. The betweenness centrality illustrated the very high centrality nodes that are limited in the central area as well as information centrality but identified the critical nodes in a more specific area.

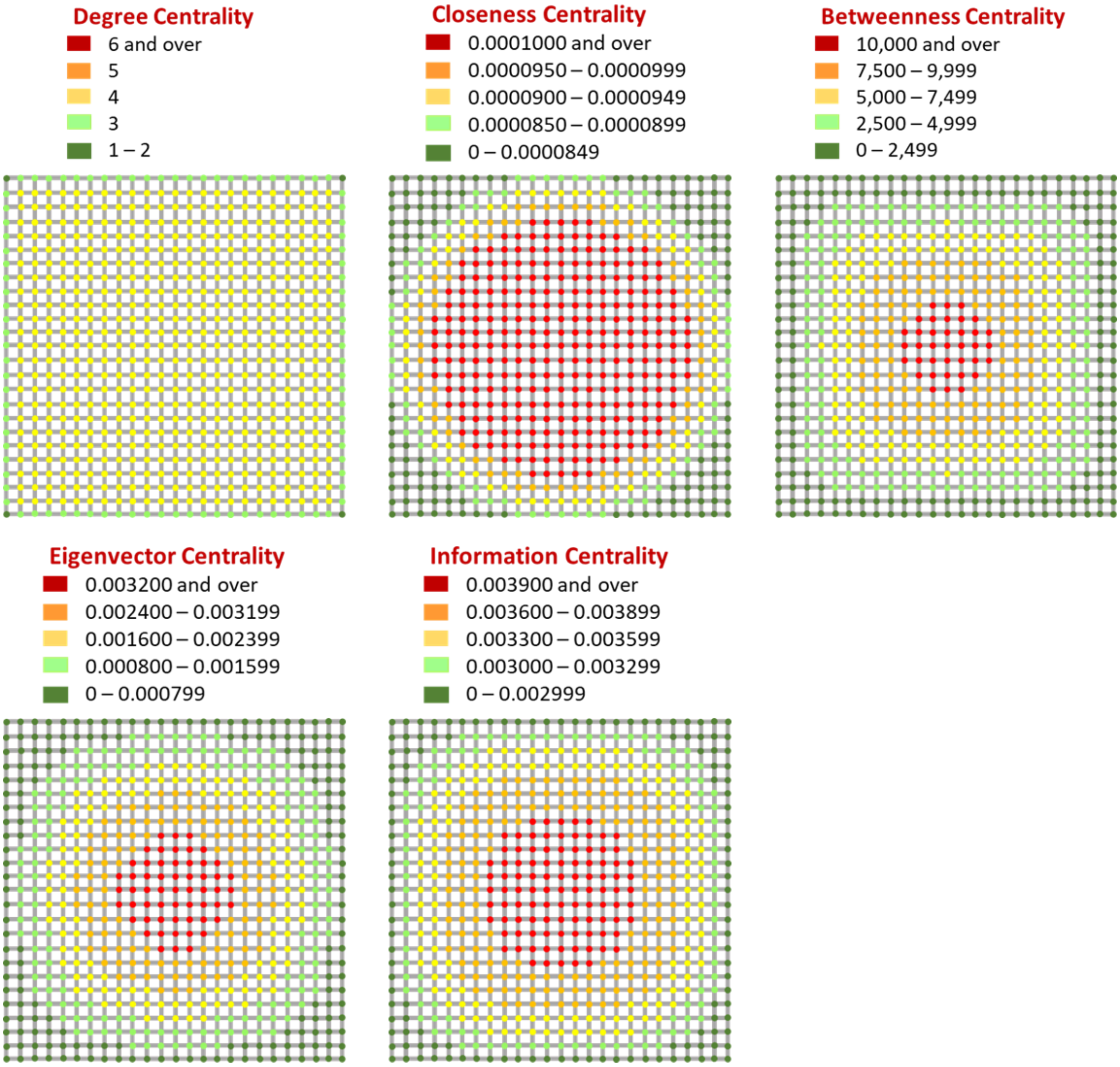


Figure 11. Comparison of the critical nodes from all five centralities on the example of a square grid network

5.3 Centralities Testing Results with a Case Study of the Kyushu Railway Network

All five node-based centrality analyses were used to analyze all 671 nodes of the Kyushu railway network case study for the node influence evaluation. The result is illustrated in the following table.

Table 9. All five node-based centralities of the Kyushu railway network

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
1	Mojikō	1	0.0360	0.0000	0.0027	0.0039
2	Komorie	2	0.0373	669.0000	0.0078	0.0081
3	Moji	3	0.0388	2,003.0000	0.0200	0.0168
4	Kokura	4	0.0403	39,098.9167	0.0435	0.0683
5	Nishi-Kokura	3	0.0393	37,761.7500	0.0207	0.0514
6	Kyūshūkōdaimae	2	0.0379	13,418.3333	0.0082	0.0190
7	Tobata	2	0.0366	12,808.3333	0.0033	0.0169
8	Edamitsu	2	0.0355	12,201.3333	0.0013	0.0152
9	Space World	2	0.0344	11,598.3333	0.0005	0.0138
10	Yahata	2	0.0333	10,999.3333	0.0002	0.0126
11	Kurosaki	2	0.0324	10,405.3333	0.0001	0.0118
12	Jinnoharu	2	0.0315	9,825.8333	0.0000	0.0112
13	Orio	4	0.0307	9,818.8333	0.0000	0.0227
14	Mizumaki	2	0.0298	2,290.1667	0.0000	0.0059
15	Ongagawa	2	0.0290	1,677.1667	0.0000	0.0051
16	Ebitsu	2	0.0283	1,077.5000	0.0000	0.0046
17	Kyōikudaimae	2	0.0276	492.8333	0.0000	0.0042
18	Akama	2	0.0275	410.8333	0.0000	0.0041
19	Tōgō	2	0.0280	846.1667	0.0000	0.0041
20	Higashi-Fukuma	2	0.0286	1,365.5000	0.0000	0.0044
21	Fukuma	2	0.0294	1,951.5000	0.0000	0.0048
22	Chidori	2	0.0302	2,550.1667	0.0000	0.0054
23	Koga	2	0.0310	3,158.3333	0.0001	0.0062
24	Shishibu	2	0.0319	3,770.3333	0.0002	0.0072
25	Shingū-Chūō	2	0.0329	4,384.3333	0.0005	0.0085
26	Fukkōdaimae	2	0.0339	5,000.3333	0.0011	0.0100
27	Kyūsandaimae	2	0.0350	5,618.3333	0.0029	0.0119
28	Kashii	4	0.0362	12,660.8333	0.0072	0.0194
29	Chihaya	4	0.0374	16,397.6667	0.0108	0.0208
30	Hakozaki	2	0.0387	16,789.1667	0.0156	0.0189
31	Yoshizuka	3	0.0401	26,694.5000	0.0345	0.0365
32	Hakata	**6**	0.0415	79,248.7500	**0.0709**	0.1282
33	Takeshita	2	0.0399	3,693.8333	0.0282	0.0112
34	Sasabaru	2	0.0384	3,055.8333	0.0112	0.0089
35	Minami-Fukuoka	2	0.0371	2,432.8333	0.0045	0.0072
36	Kasuga	2	0.0359	1,818.8333	0.0018	0.0059
37	Ōnojō	2	0.0348	1,217.8333	0.0007	0.0050
38	Mizuki	2	0.0337	628.1667	0.0003	0.0044
39	Tofurōminami	2	0.0328	348.0000	0.0001	0.0042
40	Futsukaichi	2	0.0334	550.5000	0.0001	0.0043
41	Tenpaizan	2	0.0340	1,004.8333	0.0002	0.0047
42	Haruda	3	0.0352	9,131.3333	0.0004	0.0091
43	Keyakidai	2	0.0363	9,448.1667	0.0009	0.0083
44	Kiyama	2	0.0375	10,026.1667	0.0022	0.0091
45	Yayoigaoka	2	0.0387	10,612.1667	0.0055	0.0102
46	Tashiro	2	0.0401	11,206.1667	0.0139	0.0119
47	Tosu	3	0.0416	11,933.8333	0.0349	0.0141
48	Hizen-Asahi	2	0.0419	4,453.5833	0.0326	0.0050
49	Kurume	5	**0.0436**	11,6024.9167	0.0599	**0.2133**

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
50	Araki	2	0.0418	349.5000	0.0257	0.0052
51	Nishimuta	2	0.0409	1.0000	0.0150	0.0046
52	Hainuzuka	2	0.0417	318.5000	0.0179	0.0050
53	Chikugo-Funagoya	4	0.0435	**120,336.4167**	0.0371	0.1994
54	Setaka	2	0.0421	48,740.0000	0.0148	0.0273
55	Minami-Setaka	2	0.0408	48,243.0000	0.0059	0.0252
56	Wataze	2	0.0396	47,746.0000	0.0023	0.0235
57	Yoshino	2	0.0385	47,249.0000	0.0009	0.0222
58	Ginsui	2	0.0374	46,752.0000	0.0004	0.0212
59	Ōmuta	3	0.0364	47,053.0000	0.0002	0.1018
60	Arao	2	0.0352	2,058.0000	0.0001	0.0056
61	Minami-Arao	2	0.0340	1,519.8333	0.0000	0.0049
62	Nagasu	2	0.0337	1,216.6667	0.0000	0.0044
63	Ōnoshimo	2	0.0334	1,051.1667	0.0000	0.0042
64	Tamana	2	0.0332	1,062.3333	0.0000	0.0042
65	Higo-Ikura	2	0.0340	1,398.0000	0.0000	0.0044
66	Konoha	2	0.0349	1,895.0000	0.0000	0.0049
67	Tabaruzaka	2	0.0358	2,392.0000	0.0001	0.0056
68	Ueki	2	0.0368	2,889.0000	0.0002	0.0067
69	Nishisato	2	0.0378	3,386.0000	0.0004	0.0080
70	Sōjōdaigakumae	2	0.0389	3,921.0000	0.0011	0.0096
71	Kami-Kumamoto	2	0.0404	4,532.0000	0.0027	0.0117
72	Kumamoto	5	0.0421	101,117.4167	0.0067	0.1977
73	Nishi-Kumamoto	2	0.0405	6,674.0000	0.0027	0.0081
74	Kawashiri	2	0.0389	6,029.0000	0.0011	0.0067
75	Tomiai	2	0.0375	5,384.0000	0.0005	0.0059
76	Uto	3	0.0362	5,544.5000	0.0003	0.0257
77	Matsubase	2	0.0356	805.5000	0.0002	0.0040
78	Ogawa	2	0.0369	1,210.0000	0.0003	0.0042
79	Arisa	2	0.0382	1,855.0000	0.0006	0.0049
80	Senchō	2	0.0397	2,500.0000	0.0016	0.0061
81	Shin-Yatsushiro	4	0.0413	82,016.0833	0.0039	0.1535
82	Yatsushiro	2	0.0398	14,489.0000	0.0016	0.0224
83	Sendai	3	0.0386	66,249.5833	0.0003	0.0712
84	Kumanojō	2	0.0372	3,160.5000	0.0001	0.0071
85	Kobanchaya	2	0.0359	2,502.5000	0.0001	0.0057
86	Kushikino	2	0.0347	1,844.5000	0.0000	0.0047
87	Kamimuragakuenmae	2	0.0336	1,186.5000	0.0000	0.0040
88	Ichiki	2	0.0325	528.5000	0.0000	0.0035
89	Yunomoto	2	0.0317	15.0000	0.0000	0.0033
90	Higashi-Ichiki	2	0.0319	159.5000	0.0000	0.0034
91	Ijūin	2	0.0329	817.5000	0.0000	0.0037
92	Satsuma-Matsumoto	2	0.0340	1,475.5000	0.0000	0.0043
93	Kami-Ijūin	2	0.0352	2,133.5000	0.0000	0.0052
94	Hiroki	2	0.0365	2,791.5000	0.0001	0.0065
95	Kagoshima-Chūō	4	0.0378	67,534.0833	0.0002	0.0980
96	Kagoshima	2	0.0369	46,151.0833	0.0001	0.0340
97	Shimonoseki	1	0.0373	0.0000	0.0069	0.0043
98	Minami-Kokura	2	0.0382	25,128.2500	0.0085	0.0350
99	Jōno	3	0.0372	25,719.5833	0.0040	0.0428
100	Abeyamakōen	2	0.0361	17,898.3333	0.0016	0.0314
101	Shimosone	2	0.0351	17,373.3333	0.0006	0.0290
102	Kusami	2	0.0342	16,852.3333	0.0003	0.0268
103	Kanda	2	0.0333	16,335.3333	0.0001	0.0249
104	Obase Nishikōdai-mae	2	0.0325	15,822.3333	0.0000	0.0231
105	Yukuhashi	2	0.0317	15,313.3333	0.0000	0.0214
106	Minami-Yukuhashi	2	0.0309	14,808.3333	0.0000	0.0199

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
107	Shindenbaru	2	0.0302	14,307.3333	0.0000	0.0185
108	Tsuiki	2	0.0296	13,810.3333	0.0000	0.0173
109	Shiida	2	0.0290	13,317.3333	0.0000	0.0161
110	Buzen-Shōe	2	0.0284	12,828.3333	0.0000	0.0150
111	Unoshima	2	0.0278	12,343.3333	0.0000	0.0140
112	Mikekado	2	0.0272	11,862.3333	0.0000	0.0131
113	Yoshitomi	2	0.0267	11,385.3333	0.0000	0.0123
114	Nakatsu	2	0.0262	10,912.3333	0.0000	0.0115
115	Higashi-Nakatsu	2	0.0258	10,443.3333	0.0000	0.0108
116	Imazu	2	0.0253	9,978.3333	0.0000	0.0102
117	Amatsu	2	0.0249	9,517.3333	0.0000	0.0096
118	Buzen-Zenkōji	2	0.0245	9,060.3333	0.0000	0.0091
119	Yanagigaura	2	0.0241	8,621.3333	0.0000	0.0087
120	Buzen-Nagasu	2	0.0237	8,214.3333	0.0000	0.0083
121	Usa	2	0.0234	7,827.8333	0.0000	0.0080
122	Nishi-Yashiki	2	0.0231	7,448.8333	0.0000	0.0078
123	Tateishi	2	0.0228	7,075.8333	0.0000	0.0076
124	Naka-Yamaga	2	0.0225	6,708.8333	0.0000	0.0074
125	Kitsuki	2	0.0223	6,347.8333	0.0000	0.0073
126	Ōga	2	0.0220	5,992.8333	0.0000	0.0073
127	Hiji	2	0.0217	5,643.8333	0.0000	0.0074
128	Yōkoku	2	0.0215	5,304.1667	0.0000	0.0075
129	Bungo-Toyooka	2	0.0213	4,978.0000	0.0000	0.0076
130	Kamegawa	2	0.0212	4,786.0000	0.0000	0.0079
131	Beppu-Daigaku	2	0.0211	4,728.5000	0.0000	0.0082
132	Beppu	2	0.0211	4,777.0000	0.0000	0.0086
133	Higashi-Beppu	2	0.0213	5,026.5000	0.0000	0.0091
134	Nishi-Ōita	2	0.0216	5,383.5000	0.0000	0.0098
135	Ōita	4	0.0219	19,348.5833	0.0000	0.0238
136	Maki	2	0.0215	14,456.0833	0.0000	0.0158
137	Takajō	2	0.0212	13,973.0833	0.0000	0.0149
138	Tsurusaki	2	0.0209	13,492.0833	0.0000	0.0140
139	Ōzai	2	0.0206	13,013.0833	0.0000	0.0133
140	Sakanoichi	2	0.0203	12,536.0833	0.0000	0.0126
141	Kōzaki	2	0.0200	12,061.0833	0.0000	0.0119
142	Sashiu	2	0.0197	11,588.0833	0.0000	0.0114
143	Shitanoe	2	0.0194	11,125.0833	0.0000	0.0109
144	Kumasaki	2	0.0192	10,672.0833	0.0000	0.0104
145	Kami-Usuki	2	0.0190	10,221.0833	0.0000	0.0099
146	Usuki	2	0.0187	9,772.0833	0.0000	0.0095
147	Tsukumi	2	0.0185	9,325.0833	0.0000	0.0092
148	Hishiro	2	0.0183	8,880.0833	0.0000	0.0089
149	Azamui	2	0.0180	8,437.0833	0.0000	0.0086
150	Kariu	2	0.0178	7,996.0833	0.0000	0.0083
151	Kaizaki	2	0.0176	7,597.5833	0.0000	0.0081
152	Saiki	2	0.0175	7,247.0833	0.0000	0.0079
153	Kamioka	2	0.0173	6,904.0833	0.0000	0.0078
154	Naomi	2	0.0172	6,563.0833	0.0000	0.0077
155	Naokawa	2	0.0170	6,234.0833	0.0000	0.0076
156	Shigeoka	2	0.0169	5,940.0833	0.0000	0.0075
157	Sōtarō	2	0.0168	5,756.5833	0.0000	0.0075
158	Ichitana	2	0.0169	5,852.4167	0.0000	0.0075
159	Kitagawa	2	0.0170	6,158.4167	0.0000	0.0075
160	Hyūga-Nagai	2	0.0171	6,496.2500	0.0000	0.0075
161	Kita-Nobeoka	2	0.0173	6,858.7500	0.0000	0.0076
162	Nobeoka	2	0.0175	7,239.7500	0.0000	0.0077
163	Minami-Nobeoka	2	0.0177	7,633.0833	0.0000	0.0079

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	**Station**	**Degree Centrality**	**Closeness Centrality**	**Betweenness Centrality**	**Eigenvector Centrality**	**Information Centrality**
164	Asahigaoka	2	0.0178	8,034.4167	0.0000	0.0081
165	Totoro	2	0.0180	8,443.8333	0.0000	0.0083
166	Kadogawa	2	0.0182	8,860.5833	0.0000	0.0085
167	Hyūgashi	2	0.0185	9,283.5833	0.0000	0.0088
168	Zaikōji	2	0.0187	9,712.5833	0.0000	0.0091
169	Minami-Hyūga	2	0.0189	10,156.0833	0.0000	0.0094
170	Mimitsu	2	0.0192	10,613.0833	0.0000	0.0098
171	Higashi-Tsuno	2	0.0194	11,074.0833	0.0000	0.0102
172	Tsuno	2	0.0197	11,539.0833	0.0000	0.0106
173	Kawaminami	2	0.0200	12,008.0833	0.0000	0.0111
174	Takanabe	2	0.0202	12,481.0833	0.0000	0.0116
175	Hyūga-Shintomi	2	0.0205	12,958.0833	0.0000	0.0122
176	Sadowara	2	0.0208	13,439.0833	0.0000	0.0128
177	Hyūga-Sumiyoshi	2	0.0212	13,924.0833	0.0000	0.0135
178	Hasugaike	2	0.0215	14,413.0833	0.0000	0.0142
179	Miyazaki-Jingū	2	0.0218	14,906.0833	0.0000	0.0150
180	Miyazaki	2	0.0222	15,403.0833	0.0000	0.0159
181	Minami-Miyazaki	3	0.0226	31,542.0833	0.0000	0.0505
182	Kanō	2	0.0229	29,653.0833	0.0000	0.0289
183	Kiyotake	2	0.0233	30,106.0833	0.0000	0.0293
184	Hyūga-Kutsukake	2	0.0237	30,563.0833	0.0000	0.0298
185	Tano	2	0.0241	31,024.0833	0.0000	0.0304
186	Aoidake	2	0.0245	31,489.0833	0.0000	0.0311
187	Yamanokuchi	2	0.0249	31,958.0833	0.0000	0.0318
188	Mochibaru	2	0.0253	32,431.0833	0.0000	0.0327
189	Mimata	2	0.0258	32,908.0833	0.0000	0.0336
190	Miyakonojō	3	0.0263	37,013.0833	0.0000	0.0368
191	Nishi-Miyakonojō	2	0.0267	35,248.4167	0.0000	0.0188
192	Isoichi	2	0.0272	35,707.5833	0.0000	0.0191
193	Takarabe	2	0.0278	36,181.5833	0.0000	0.0194
194	Kitamata	2	0.0283	36,657.5833	0.0000	0.0199
195	Ōsumi-Ōkawara	2	0.0289	37,135.5833	0.0000	0.0205
196	Kita-Naganoda	2	0.0295	37,615.5833	0.0000	0.0212
197	Kirishima-Jingū	2	0.0302	38,097.5833	0.0000	0.0221
198	Kokubu	2	0.0308	38,581.5833	0.0000	0.0232
199	Hayato	3	0.0316	43,840.4167	0.0000	0.0295
200	Kajiki	2	0.0322	43,505.0833	0.0000	0.0274
201	Kinkō	2	0.0329	43,941.0833	0.0000	0.0280
202	Chōsa	2	0.0336	44,379.0833	0.0000	0.0288
203	Aira	2	0.0344	44,819.0833	0.0000	0.0297
204	Shigetomi	2	0.0352	45,261.0833	0.0000	0.0309
205	Ryūgamizu	2	0.0360	45,705.0833	0.0000	0.0323
206	Wakamatsu	1	0.0266	0.0000	0.0000	0.0026
207	Fujinoki	2	0.0274	669.0000	0.0000	0.0053
208	Okudōkai	2	0.0281	1,336.0000	0.0000	0.0079
209	Futajima	2	0.0289	2,001.0000	0.0000	0.0107
210	Honjō	2	0.0298	2,664.0000	0.0000	0.0136
211	Higashi-Mizumaki	2	0.0299	3,535.6667	0.0000	0.0071
212	Nakama	2	0.0291	2,949.6667	0.0000	0.0063
213	Chikuzen-Habu	2	0.0284	2,383.1667	0.0000	0.0056
214	Kurate	2	0.0278	1,852.1667	0.0000	0.0051
215	Chikuzen-Ueki	2	0.0273	1,378.1667	0.0000	0.0048
216	Shinnyū	2	0.0268	950.1667	0.0000	0.0047
217	Nōgata	2	0.0268	961.3333	0.0000	0.0047
218	Katsuno	2	0.0273	1,428.1667	0.0000	0.0049
219	Kotake	2	0.0279	1,971.8333	0.0000	0.0053
220	Namazuta	2	0.0286	2,571.3333	0.0000	0.0058

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
221	Urata	2	0.0294	3,176.3333	0.0000	0.0066
222	Shin Iizuka	3	0.0302	6,269.0000	0.0000	0.0099
223	Iizuka	2	0.0307	5,588.8333	0.0000	0.0084
224	Tentō	2	0.0313	5,999.1667	0.0000	0.0090
225	Keisen	3	0.0320	7,033.8333	0.0000	0.0105
226	Kami Honami	2	0.0325	6,204.6667	0.0000	0.0069
227	Chikuzen Uchino	2	0.0333	6,699.0000	0.0001	0.0070
228	Chikuzen Yamae	2	0.0342	7,226.8333	0.0002	0.0075
229	Yusu	2	0.0387	8,956.1667	0.0139	0.0124
230	Harumachi	2	0.0374	8,358.6667	0.0060	0.0112
231	Chojabaru	4	0.0363	8,559.8333	0.0036	0.0263
232	Kadomatsu	2	0.0351	3,630.8333	0.0014	0.0084
233	Sasaguri	2	0.0341	3,044.8333	0.0006	0.0071
234	Chikuzen-Yamate	2	0.0331	2,466.8333	0.0002	0.0063
235	Kido Nanzōin-mae	2	0.0322	1,900.6667	0.0001	0.0058
236	Kurōbaru	2	0.0314	1,376.3333	0.0000	0.0056
237	Chikuzen-Daibu	2	0.0314	1,336.6667	0.0000	0.0057
238	Saitozaki	1	0.0307	0.0000	0.0001	0.0031
239	Umi-no-Nakamichi	2	0.0317	669.0000	0.0003	0.0064
240	Gannosu	2	0.0327	1,336.0000	0.0007	0.0098
241	Nata	2	0.0338	2,001.0000	0.0017	0.0134
242	Wajiro	4	0.0350	4,682.8333	0.0043	0.0246
243	Kashii-Jingū	2	0.0351	1,142.5000	0.0029	0.0053
244	Maimatsubara	2	0.0340	551.8333	0.0014	0.0047
245	Doi	2	0.0340	560.1667	0.0010	0.0046
246	Iga	2	0.0351	1,162.1667	0.0016	0.0051
247	Sakado	2	0.0350	2,664.0000	0.0014	0.0166
248	Sue	2	0.0338	2,001.0000	0.0006	0.0130
249	Sue-Chūō	2	0.0327	1,336.0000	0.0002	0.0095
250	Shinbaru	2	0.0317	669.0000	0.0001	0.0063
251	Umi	1	0.0307	0.0000	0.0000	0.0031
252	Ishida	2	0.0359	7,751.5833	0.0016	0.0127
253	Shii-Kōen	2	0.0348	7,146.0833	0.0006	0.0110
254	Shii	2	0.0338	6,567.5833	0.0003	0.0096
255	Ishiharamachi	2	0.0329	6,017.5833	0.0001	0.0085
256	Yobuno	2	0.0320	5,475.9167	0.0000	0.0076
257	Saidōsho	2	0.0312	4,959.0833	0.0000	0.0069
258	Kawara	2	0.0306	4,484.9167	0.0000	0.0064
259	Ipponmatsu	2	0.0300	4,039.9167	0.0000	0.0061
260	Tagawa-Ita	2	0.0294	3,627.9167	0.0000	0.0061
261	Tagawa-Gotōji	3	0.0289	4,916.6667	0.0000	0.0090
262	Ikejiri	2	0.0283	3,621.2500	0.0000	0.0069
263	Buzen-Kawasaki	2	0.0278	3,134.2500	0.0000	0.0062
264	Nishi-Soeda	2	0.0272	2,653.2500	0.0000	0.0058
265	Soeda	2	0.0270	2,476.7500	0.0000	0.0055
266	Kanyūsha-Hikosan	2	0.0273	2,693.2500	0.0000	0.0054
267	Buzen-Masuda	2	0.0276	3,004.2500	0.0000	0.0054
268	Hikosan	2	0.0281	3,381.2500	0.0000	0.0055
269	Chikuzen-Iwaya	2	0.0286	3,835.7500	0.0000	0.0058
270	Daigyōji	2	0.0292	4,319.7500	0.0000	0.0062
271	Hōshuyama	2	0.0299	4,831.2500	0.0000	0.0067
272	Ōtsuru	2	0.0306	5,384.2500	0.0000	0.0075
273	Imayama	2	0.0315	5,969.2500	0.0000	0.0083
274	Yoake	3	0.0324	25,707.9167	0.0000	0.0251
275	Kami-Mio	2	0.0296	3,113.8333	0.0000	0.0067
276	Shimo-Kamoo	2	0.0291	2,707.5000	0.0000	0.0063
277	Chikuzen-Shōnai	2	0.0287	2,345.6667	0.0000	0.0061

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	**Station**	**Degree Centrality**	**Closeness Centrality**	**Betweenness Centrality**	**Eigenvector Centrality**	**Information Centrality**
278	Funao	2	0.0283	2,189.8333	0.0000	0.0062
279	Gion	2	0.0400	15,827.6667	0.0291	0.0207
280	Nakasu-Kawabata	3	0.0386	15,366.6667	0.0138	0.0298
281	Tenjin	2	0.0372	12,305.1667	0.0055	0.0242
282	Akasaka	2	0.0360	11,686.1667	0.0022	0.0217
283	Ōhorikōen	2	0.0348	11,068.1667	0.0009	0.0194
284	Tōjinmachi	2	0.0338	10,460.1667	0.0003	0.0174
285	Nishijin	2	0.0328	9,862.1667	0.0001	0.0157
286	Fujisaki	2	0.0319	9,264.1667	0.0001	0.0141
287	Muromi	2	0.0310	8,666.1667	0.0000	0.0127
288	Meinohama	2	0.0301	8,068.1667	0.0000	0.0114
289	Shimoyamato	2	0.0294	7,470.1667	0.0000	0.0102
290	Imajuku	2	0.0286	6,872.1667	0.0000	0.0092
291	Kyūdai-Gakkentoshi	2	0.0279	6,274.1667	0.0000	0.0083
292	Susenji	2	0.0272	5,676.1667	0.0000	0.0075
293	Hatae	2	0.0266	5,133.1667	0.0000	0.0068
294	Itoshima-Kokomae	2	0.0261	4,645.1667	0.0000	0.0062
295	Chikuzen-Maebaru	2	0.0256	4,157.1667	0.0000	0.0057
296	Misakigaoka	2	0.0252	3,669.1667	0.0000	0.0053
297	Kafuri	2	0.0247	3,181.1667	0.0000	0.0049
298	Ikisan	2	0.0243	2,693.1667	0.0000	0.0047
299	Chikuzen-Fukae	2	0.0239	2,205.1667	0.0000	0.0045
300	Dainyū	2	0.0234	1,717.1667	0.0000	0.0045
301	Fukuyoshi	2	0.0230	1,229.1667	0.0000	0.0045
302	Shikaka	2	0.0230	1,145.0000	0.0000	0.0046
303	Hamasaki	2	0.0232	1,474.8333	0.0000	0.0048
304	Nijinomatsubara	2	0.0236	1,949.3333	0.0000	0.0051
305	Higashi-Karatsu	2	0.0242	2,559.3333	0.0000	0.0055
306	Watada	2	0.0247	3,174.8333	0.0000	0.0060
307	Karatsu	3	0.0253	4,438.8333	0.0000	0.0088
308	Nishi-Karatsu	1	0.0247	0.0000	0.0000	0.0025
309	Onizuka	2	0.0259	5,031.8333	0.0000	0.0078
310	Yamamoto	3	0.0265	11,989.8333	0.0000	0.0269
311	Hizen-Kubo	2	0.0259	5,949.0000	0.0000	0.0202
312	Nishi-Ōchi	2	0.0252	5,296.0000	0.0000	0.0180
313	Sari	2	0.0246	4,641.0000	0.0000	0.0158
314	Komanaki	2	0.0240	3,984.0000	0.0000	0.0138
315	Ōkawano	2	0.0235	3,325.0000	0.0000	0.0118
316	Hizen-Nagano	2	0.0229	2,664.0000	0.0000	0.0098
317	Momonokawa	2	0.0224	2,001.0000	0.0000	0.0078
318	Kanaishihara	2	0.0219	1,336.0000	0.0000	0.0059
319	Kami-Imari	2	0.0215	669.0000	0.0000	0.0040
320	Imari	1	0.0210	0.0000	0.0000	0.0020
321	Shin-Tosu	4	0.0432	113,426.3333	0.0550	0.1452
322	Hizen-Fumoto	2	0.0419	50,880.8333	0.0219	0.0709
323	Nakabaru	2	0.0407	50,392.8333	0.0087	0.0680
324	Yoshinogari-Kōen	2	0.0395	49,904.8333	0.0035	0.0656
325	Kanzaki	2	0.0384	49,416.8333	0.0014	0.0636
326	Igaya	2	0.0374	48,928.8333	0.0006	0.0618
327	Saga	2	0.0364	48,440.8333	0.0002	0.0603
328	Nabeshima	2	0.0354	47,952.8333	0.0001	0.0590
329	Balloon Saga (seasonal)	2	0.0345	47,464.8333	0.0000	0.0580
330	Kubota	3	0.0337	48,956.8333	0.0000	0.1114
331	Ushizu	2	0.0328	33,264.0000	0.0000	0.0953
332	Hizen-Yamaguchi	3	0.0319	32,920.0000	0.0000	0.0941
333	Hizen-Shiroishi	2	0.0310	19,357.0000	0.0000	0.0175
334	Hizen-Ryūō	2	0.0302	18,751.0000	0.0000	0.0160

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
335	Hizen-Kashima	2	0.0294	18,145.0000	0.0000	0.0147
336	Hizen-Hama	2	0.0286	17,539.0000	0.0000	0.0135
337	Hizen-Nanaura	2	0.0279	16,933.0000	0.0000	0.0124
338	Hizen-Iida	2	0.0272	16,327.0000	0.0000	0.0115
339	Tara	2	0.0265	15,721.0000	0.0000	0.0106
340	Hizen-Ōura	2	0.0259	15,115.0000	0.0000	0.0099
341	Konagai	2	0.0253	14,512.0000	0.0000	0.0093
342	Nagasato	2	0.0248	13,912.0000	0.0000	0.0087
343	Yue	2	0.0242	13,312.0000	0.0000	0.0083
344	Oe	2	0.0237	12,712.0000	0.0000	0.0080
345	Hizen-Nagata	2	0.0232	12,112.0000	0.0000	0.0078
346	Higashi-Isahaya	2	0.0227	11,512.0000	0.0000	0.0077
347	Isahaya	3	0.0223	11,206.0000	0.0000	0.0273
348	Nishi-Isahaya	2	0.0218	8,541.0000	0.0000	0.0244
349	Kikitsu	3	0.0214	7,910.0000	0.0000	0.0229
350	Ichinuno	2	0.0209	3,634.0000	0.0000	0.0037
351	Hizen-Koga	2	0.0205	2,977.0000	0.0000	0.0032
352	Utsutsugawa	2	0.0201	2,320.0000	0.0000	0.0029
353	Urakami	3	0.0197	1,668.5000	0.0000	0.0044
354	Nagasaki	1	0.0193	0.0000	0.0000	0.0020
355	Higashisono	2	0.0209	2,974.0000	0.0000	0.0035
356	Ōkusa	2	0.0205	2,315.5000	0.0000	0.0030
357	Honkawachi	2	0.0201	1,658.0000	0.0000	0.0026
358	Nagayo	2	0.0197	1,001.0000	0.0000	0.0024
359	Kōda	2	0.0193	344.0000	0.0000	0.0023
360	Michinoo	2	0.0190	16.0000	0.0000	0.0022
361	Nishi-Urakami	2	0.0193	346.0000	0.0000	0.0023
362	Ogi	2	0.0327	16,513.8333	0.0000	0.0208
363	Higashi-Taku	2	0.0318	15,915.8333	0.0000	0.0193
364	Naka-Taku	2	0.0309	15,317.8333	0.0000	0.0180
365	Taku	2	0.0301	14,719.8333	0.0000	0.0168
366	Kyūragi	2	0.0293	14,121.8333	0.0000	0.0158
367	Iwaya	2	0.0285	13,523.8333	0.0000	0.0149
368	Ōchi	2	0.0278	12,925.8333	0.0000	0.0142
369	Honmutabe	2	0.0272	12,327.8333	0.0000	0.0136
370	Ōmachi	2	0.0310	12,548.0000	0.0000	0.0166
371	Kitagata	2	0.0301	11,920.0000	0.0000	0.0150
372	Takahashi	2	0.0293	11,292.0000	0.0000	0.0137
373	Takeo-Onsen	2	0.0285	10,678.0000	0.0000	0.0125
374	Nagao	2	0.0278	10,078.0000	0.0000	0.0114
375	Mimasaka	2	0.0271	9,478.0000	0.0000	0.0104
376	Kami-Arita	2	0.0265	8,878.0000	0.0000	0.0096
377	Arita	2	0.0259	8,278.0000	0.0000	0.0089
378	Mikawachi	2	0.0253	7,678.0000	0.0000	0.0083
379	Haiki	3	0.0247	7,174.0000	0.0000	0.0123
380	Daitō	2	0.0241	1,336.0000	0.0000	0.0068
381	Hiu	2	0.0236	669.0000	0.0000	0.0045
382	Sasebo	1	0.0230	0.0000	0.0000	0.0022
383	Huis Ten Bosch	2	0.0242	4,663.0000	0.0000	0.0057
384	Haenosaki	2	0.0237	4,057.0000	0.0000	0.0051
385	Ogushigō	2	0.0232	3,451.0000	0.0000	0.0047
386	Kawatana	2	0.0227	2,845.0000	0.0000	0.0044
387	Sonogi	2	0.0222	2,239.0000	0.0000	0.0042
388	Chiwata	2	0.0218	1,633.0000	0.0000	0.0040
389	Matsubara	2	0.0214	1,027.0000	0.0000	0.0040
390	Takematsu	2	0.0210	421.0000	0.0000	0.0040
391	Suwa	2	0.0210	432.0000	0.0000	0.0042

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
392	Ōmura	2	0.0214	1,060.0000	0.0000	0.0044
393	Iwamatsu	2	0.0218	1,688.0000	0.0000	0.0048
394	Kurume-Kōkōmae	2	0.0421	28,205.6667	0.0238	0.0323
395	Minami-Kurume	2	0.0408	27,687.6667	0.0095	0.0294
396	Kurume-Daigakumae	2	0.0396	27,175.6667	0.0038	0.0271
397	Mii	2	0.0384	26,670.6667	0.0015	0.0250
398	Zendōji	2	0.0374	26,173.6667	0.0006	0.0233
399	Chikugo-Kusano	2	0.0364	25,684.6667	0.0002	0.0218
400	Tanushimaru	2	0.0354	25,203.6667	0.0001	0.0205
401	Chikugo-Yoshii	2	0.0346	24,736.1667	0.0000	0.0195
402	Ukiha	2	0.0338	24,283.1667	0.0000	0.0186
403	Chikugo-Ōishi	2	0.0331	23,840.1667	0.0000	0.0179
404	Teruoka	2	0.0316	20,947.9167	0.0000	0.0199
405	Hita	2	0.0309	20,452.9167	0.0000	0.0185
406	Bungo-Miyoshi	2	0.0302	19,961.9167	0.0000	0.0172
407	Bungo-Nakagawa	2	0.0296	19,474.9167	0.0000	0.0161
408	Amagase	2	0.0289	18,991.9167	0.0000	0.0150
409	Sugikawachi	2	0.0284	18,512.9167	0.0000	0.0141
410	Kita-Yamada	2	0.0278	18,037.9167	0.0000	0.0132
411	Bungo-Mori	2	0.0273	17,566.9167	0.0000	0.0125
412	Era	2	0.0268	17,099.9167	0.0000	0.0118
413	Hikiji	2	0.0263	16,636.9167	0.0000	0.0112
414	Bungo-Nakamura	2	0.0258	16,177.9167	0.0000	0.0106
415	Noya	2	0.0254	15,722.9167	0.0000	0.0101
416	Yufuin	2	0.0250	15,299.9167	0.0000	0.0097
417	Minami-Yufu	2	0.0246	14,908.9167	0.0000	0.0094
418	Yunohira	2	0.0243	14,521.9167	0.0000	0.0091
419	Shōnai	2	0.0239	14,138.9167	0.0000	0.0089
420	Tenjinyama	2	0.0236	13,759.9167	0.0000	0.0088
421	Onoya	2	0.0233	13,384.9167	0.0000	0.0087
422	Onigase	2	0.0230	13,013.9167	0.0000	0.0087
423	Mukainoharu	2	0.0227	12,646.9167	0.0000	0.0088
424	Bungo-Kokubu	2	0.0224	12,283.9167	0.0000	0.0090
425	Kaku	2	0.0222	11,934.4167	0.0000	0.0092
426	Minami-Ōita	2	0.0219	11,662.2500	0.0000	0.0096
427	Furugō	2	0.0219	11,541.0833	0.0000	0.0101
428	Heisei	2	0.0407	20,585.6667	0.0027	0.0371
429	Minami-Kumamoto	2	0.0394	20,046.6667	0.0011	0.0340
430	Shin-Suizenji	2	0.0382	19,511.6667	0.0004	0.0314
431	Suizenji	2	0.0371	18,980.6667	0.0002	0.0290
432	Tōkai-Gakuen-mae	2	0.0360	18,453.6667	0.0001	0.0269
433	Tatsutaguchi	2	0.0351	17,930.6667	0.0000	0.0250
434	Musashizuka	2	0.0341	17,411.6667	0.0000	0.0232
435	Hikari no Mori	2	0.0333	16,896.6667	0.0000	0.0215
436	Sanrigi	2	0.0324	16,385.6667	0.0000	0.0200
437	Haramizu	2	0.0317	15,878.6667	0.0000	0.0187
438	Higo-Ōzu	2	0.0309	15,375.6667	0.0000	0.0174
439	Seta	2	0.0302	14,876.6667	0.0000	0.0162
440	Tateno	2	0.0296	14,381.6667	0.0000	0.0151
441	Akamizu	2	0.0289	13,890.6667	0.0000	0.0141
442	Ichinokawa	2	0.0283	13,403.6667	0.0000	0.0132
443	Uchinomaki	2	0.0278	12,920.6667	0.0000	0.0124
444	Aso	2	0.0272	12,441.6667	0.0000	0.0116
445	Ikoi-no-Mura	2	0.0267	11,966.6667	0.0000	0.0110
446	Miyaji	2	0.0262	11,495.6667	0.0000	0.0103
447	Namino	2	0.0258	11,028.6667	0.0000	0.0098
448	Takimizu	2	0.0253	10,565.6667	0.0000	0.0093

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	**Station**	**Degree Centrality**	**Closeness Centrality**	**Betweenness Centrality**	**Eigenvector Centrality**	**Information Centrality**
449	Bungo-Ogi	2	0.0249	10,106.6667	0.0000	0.0089
450	Tamarai	2	0.0245	9,656.1667	0.0000	0.0085
451	Bungo-Taketa	2	0.0241	9,219.6667	0.0000	0.0082
452	Asaji	2	0.0237	8,793.6667	0.0000	0.0080
453	Ogata	2	0.0234	8,401.6667	0.0000	0.0078
454	Bungo-Kiyokawa	2	0.0231	8,044.6667	0.0000	0.0077
455	Miemachi	2	0.0228	7,695.6667	0.0000	0.0077
456	Sugao	2	0.0226	7,354.6667	0.0000	0.0077
457	Inukai	2	0.0223	7,021.6667	0.0000	0.0078
458	Takenaka	2	0.0221	6,707.1667	0.0000	0.0080
459	Nakahanda	2	0.0219	6,411.1667	0.0000	0.0082
460	Ōita-Daigaku-mae	2	0.0217	6,144.1667	0.0000	0.0085
461	Shikido	2	0.0215	6,005.5000	0.0000	0.0090
462	Takio	2	0.0216	6,114.6667	0.0000	0.0096
463	Midorikawa	2	0.0350	4,641.0000	0.0001	0.0222
464	Sumiyoshi	2	0.0338	3,984.0000	0.0000	0.0190
465	Higo-Nagahama	2	0.0327	3,325.0000	0.0000	0.0161
466	Ōda	2	0.0317	2,664.0000	0.0000	0.0132
467	Akase	2	0.0307	2,001.0000	0.0000	0.0105
468	Ishiuchi Dam	2	0.0298	1,336.0000	0.0000	0.0078
469	Hataura	2	0.0290	669.0000	0.0000	0.0052
470	Misumi	1	0.0282	0.0000	0.0000	0.0026
471	Dan	2	0.0384	13,878.0000	0.0006	0.0198
472	Sakamoto	2	0.0371	13,269.0000	0.0002	0.0176
473	Haki	2	0.0359	12,662.0000	0.0001	0.0157
474	Kamase	2	0.0348	12,057.0000	0.0000	0.0140
475	Setoishi	2	0.0337	11,454.0000	0.0000	0.0125
476	Kaiji	2	0.0327	10,884.5000	0.0000	0.0112
477	Yoshio	2	0.0320	10,384.6667	0.0000	0.0100
478	Shiroishi	2	0.0313	9,926.8333	0.0000	0.0090
479	Kyūsendō	2	0.0306	9,474.8333	0.0000	0.0081
480	Isshōchi	2	0.0300	9,024.8333	0.0000	0.0074
481	Naraguchi	2	0.0294	8,576.8333	0.0000	0.0068
482	Watari	2	0.0289	8,130.8333	0.0000	0.0063
483	Nishi Hitoyoshi	2	0.0283	7,686.8333	0.0000	0.0060
484	Hitoyoshi	2	0.0278	7,244.8333	0.0000	0.0057
485	Okoba	2	0.0273	6,824.8333	0.0000	0.0056
486	Yatake	2	0.0269	6,450.3333	0.0000	0.0056
487	Masaki	2	0.0266	6,103.8333	0.0000	0.0058
488	Yoshimatsu	3	0.0262	6,351.1667	0.0000	0.0085
489	Kurino	2	0.0257	958.6667	0.0000	0.0047
490	Ōsumi-Yokogawa	2	0.0263	1,279.1667	0.0000	0.0046
491	Uemura	2	0.0269	1,882.1667	0.0000	0.0047
492	Kirishima Onsen	2	0.0276	2,487.1667	0.0000	0.0050
493	Kareigawa	2	0.0283	3,094.1667	0.0000	0.0054
494	Naka-fukura	2	0.0290	3,703.1667	0.0000	0.0059
495	Hyōkiyama	2	0.0298	4,314.1667	0.0000	0.0066
496	Hinatayama	2	0.0307	4,927.1667	0.0000	0.0075
497	Tsurumaru	2	0.0257	5,253.3333	0.0000	0.0061
498	Kyōmachi Onsen	2	0.0252	4,769.3333	0.0000	0.0055
499	Ebino	2	0.0248	4,287.3333	0.0000	0.0049
500	Ebino Uwae	2	0.0244	3,807.3333	0.0000	0.0045
501	Ebino Iino	2	0.0239	3,329.3333	0.0000	0.0043
502	Nishi Kobayashi	2	0.0235	2,853.3333	0.0000	0.0040
503	Kobayashi	2	0.0232	2,379.3333	0.0000	0.0039
504	Hirowara	2	0.0228	1,943.1667	0.0000	0.0039
505	Takaharu	2	0.0225	1,809.6667	0.0000	0.0040

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
506	Hyūga Maeda	2	0.0230	2,176.8333	0.0000	0.0042
507	Takasaki Shinden	2	0.0235	2,779.8333	0.0000	0.0044
508	Higashi Takasaki	2	0.0240	3,384.8333	0.0000	0.0048
509	Mangatsuka	2	0.0245	3,991.8333	0.0000	0.0052
510	Tanigashira	2	0.0251	4,600.8333	0.0000	0.0058
511	Hyūga Shōnai	2	0.0257	5,211.8333	0.0000	0.0065
512	Tayoshi	3	0.0221	17,387.0000	0.0000	0.0386
513	Minamikata	2	0.0217	16,125.0000	0.0000	0.0350
514	Kibana	2	0.0213	15,504.0000	0.0000	0.0333
515	Undōkōen	2	0.0208	14,881.0000	0.0000	0.0317
516	Sosanji	2	0.0204	14,256.0000	0.0000	0.0302
517	Kodomonokuni	2	0.0201	13,629.0000	0.0000	0.0287
518	Aoshima	2	0.0197	13,000.0000	0.0000	0.0273
519	Oryūzako	2	0.0193	12,369.0000	0.0000	0.0258
520	Uchiumi	2	0.0190	11,736.0000	0.0000	0.0245
521	Kouchiumi	2	0.0186	11,101.0000	0.0000	0.0231
522	Ibii	2	0.0183	10,464.0000	0.0000	0.0218
523	Kitagō	2	0.0180	9,825.0000	0.0000	0.0204
524	Uchinoda	2	0.0177	9,184.0000	0.0000	0.0191
525	Obi	2	0.0174	8,541.0000	0.0000	0.0178
526	Nichinan	2	0.0171	7,896.0000	0.0000	0.0166
527	Aburatsu	2	0.0168	7,249.0000	0.0000	0.0153
528	Ōdōtsu	2	0.0166	6,600.0000	0.0000	0.0140
529	Nangō	2	0.0163	5,949.0000	0.0000	0.0128
530	Taninokuchi	2	0.0161	5,296.0000	0.0000	0.0116
531	Yowara	2	0.0158	4,641.0000	0.0000	0.0103
532	Hyūga-Ōtsuka	2	0.0156	3,984.0000	0.0000	0.0091
533	Hyūga-Kitakata	2	0.0153	3,325.0000	0.0000	0.0079
534	Kushima	2	0.0151	2,664.0000	0.0000	0.0066
535	Fukushima-Imamachi	2	0.0149	2,001.0000	0.0000	0.0054
536	Fukushima-Takamatsu	2	0.0147	1,336.0000	0.0000	0.0041
537	Ōsumi-Natsui	2	0.0145	669.0000	0.0000	0.0028
538	Shibushi	1	0.0142	0.0000	0.0000	0.0014
539	Miyazaki Airport	1	0.0216	0.0000	0.0000	0.0023
540	Kōrimoto	2	0.0366	21,624.0000	0.0001	0.0642
541	Minami-Kagoshima	2	0.0354	21,021.0000	0.0000	0.0611
542	Usuki	2	0.0343	20,416.0000	0.0000	0.0582
543	Taniyama	2	0.0333	19,809.0000	0.0000	0.0555
544	Jigenji	2	0.0323	19,200.0000	0.0000	0.0529
545	Sakanoue	2	0.0314	18,589.0000	0.0000	0.0505
546	Goino	2	0.0305	17,976.0000	0.0000	0.0481
547	Hirakawa	2	0.0297	17,361.0000	0.0000	0.0459
548	Sesekushi	2	0.0289	16,744.0000	0.0000	0.0437
549	Nakamyō	2	0.0281	16,125.0000	0.0000	0.0417
550	Kiire	2	0.0274	15,504.0000	0.0000	0.0397
551	Maenohama	2	0.0267	14,881.0000	0.0000	0.0377
552	Nukumi	2	0.0261	14,256.0000	0.0000	0.0358
553	Satsuma-Imaizumi	2	0.0255	13,629.0000	0.0000	0.0340
554	Miyagahama	2	0.0249	13,000.0000	0.0000	0.0322
555	Nigatsuden	2	0.0243	12,369.0000	0.0000	0.0304
556	Ibusuki	2	0.0238	11,736.0000	0.0000	0.0287
557	Yamakawa	2	0.0232	11,101.0000	0.0000	0.0271
558	Ōyama	2	0.0227	10,464.0000	0.0000	0.0254
559	Nishi-Ōyama	2	0.0223	9,825.0000	0.0000	0.0238
560	Satsuma-Kawashiri	2	0.0218	9,184.0000	0.0000	0.0222
561	Higashi-Kaimon	2	0.0213	8,541.0000	0.0000	0.0207
562	Kaimon	2	0.0209	7,896.0000	0.0000	0.0192

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
563	Irino	2	0.0205	7,249.0000	0.0000	0.0177
564	Ei	2	0.0201	6,600.0000	0.0000	0.0162
565	Nishi-Ei	2	0.0197	5,949.0000	0.0000	0.0147
566	Goryō	2	0.0193	5,296.0000	0.0000	0.0132
567	Ishikaki	2	0.0190	4,641.0000	0.0000	0.0118
568	Mizunarikawa	2	0.0186	3,984.0000	0.0000	0.0104
569	Ei-Ōkawa	2	0.0183	3,325.0000	0.0000	0.0089
570	Matsugaura	2	0.0180	2,664.0000	0.0000	0.0075
571	Satsuma-Shioya	2	0.0177	2,001.0000	0.0000	0.0061
572	Shirasawa	2	0.0174	1,336.0000	0.0000	0.0046
573	Satsuma-Itashiki	2	0.0171	669.0000	0.0000	0.0031
574	Makurazaki	1	0.0168	0.0000	0.0000	0.0016
575	Shin-Minamata	2	0.0403	66,242.9167	0.0016	0.0726
576	Izumi	2	0.0395	65,877.0833	0.0007	0.0709
577	Nishitetsu Fukuoka (Tenjin)	1	0.0145	0.0000	0.0000	0.0017
578	Yakuin	4	0.0147	10,505.0000	0.0000	0.0201
579	Nishitetsu Hirao	2	0.0149	11,101.0000	0.0000	0.0202
580	Takamiya	2	0.0152	11,736.0000	0.0000	0.0211
581	Ōhashi	2	0.0154	12,369.0000	0.0000	0.0221
582	Ijiri	2	0.0156	13,000.0000	0.0000	0.0231
583	Zasshonokuma	2	0.0158	13,629.0000	0.0000	0.0242
584	Kasugabaru	2	0.0161	14,256.0000	0.0000	0.0252
585	Shirakibaru	2	0.0163	14,881.0000	0.0000	0.0263
586	Shimoōri	2	0.0166	15,504.0000	0.0000	0.0274
587	Tofurōmae	2	0.0168	16,125.0000	0.0000	0.0286
588	Nishitetsu Futsukaichi	2	0.0171	16,744.0000	0.0000	0.0297
589	Murasaki	3	0.0174	18,643.0000	0.0000	0.0338
590	Asakuragaidō	2	0.0176	19,200.0000	0.0000	0.0341
591	Sakuradai	2	0.0179	19,809.0000	0.0000	0.0352
592	Chikushi	2	0.0182	20,416.0000	0.0000	0.0363
593	Tsuko	2	0.0185	21,021.0000	0.0000	0.0375
594	Mikunigaoka	2	0.0189	21,624.0000	0.0000	0.0386
595	Mitsusawa	2	0.0192	22,225.0000	0.0000	0.0399
596	Ōho	2	0.0195	22,824.0000	0.0000	0.0411
597	Nishitetsu Ogōri	2	0.0199	23,421.0000	0.0000	0.0424
598	Hatama	2	0.0202	24,016.0000	0.0000	0.0437
599	Ajisaka	2	0.0206	24,609.0000	0.0000	0.0451
600	Miyanojin	3	0.0210	32,009.0000	0.0000	0.0610
601	Kushiwara	2	0.0213	32,136.0000	0.0000	0.0594
602	Nishitetsu Kurume	2	0.0217	32,701.0000	0.0000	0.0604
603	Hanabatake	2	0.0221	33,264.0000	0.0000	0.0615
604	Shikenjōmae	2	0.0226	33,825.0000	0.0000	0.0627
605	Tsubuku	2	0.0230	34,384.0000	0.0000	0.0639
606	Yasutake	2	0.0234	34,941.0000	0.0000	0.0652
607	Daizenji	2	0.0239	35,496.0000	0.0000	0.0664
608	Mizuma	2	0.0244	36,049.0000	0.0000	0.0678
609	Inuzuka	2	0.0249	36,600.0000	0.0000	0.0692
610	Ōmizo	2	0.0254	37,149.0000	0.0000	0.0706
611	Hatchōmuta	2	0.0259	37,696.0000	0.0000	0.0721
612	Kamachi	2	0.0265	38,241.0000	0.0000	0.0736
613	Yakabe	2	0.0271	38,784.0000	0.0000	0.0751
614	Nishitetsu Yanagawa	2	0.0277	39,325.0000	0.0000	0.0768
615	Tokumasu	2	0.0283	39,864.0000	0.0000	0.0784
616	Shiotsuka	2	0.0290	40,401.0000	0.0000	0.0802
617	Nishitetsu Nakashima	2	0.0296	40,936.0000	0.0000	0.0820
618	Enoura	2	0.0304	41,469.0000	0.0000	0.0839
619	Hiraki	2	0.0311	42,000.0000	0.0000	0.0858

Table 9. All five node-based centralities of the Kyushu railway network (Cont.)

Node	**Station**	**Degree Centrality**	**Closeness Centrality**	**Betweenness Centrality**	**Eigenvector Centrality**	**Information Centrality**
620	Nishitetsu Wataze	2	0.0319	42,529.0000	0.0000	0.0879
621	Kuranaga	2	0.0327	43,056.0000	0.0000	0.0901
622	Higashi-Amagi	2	0.0336	43,581.0000	0.0000	0.0923
623	Nishitetsu Ginsui	2	0.0345	44,104.0000	0.0000	0.0947
624	Shin-Sakaemachi	2	0.0354	44,625.0000	0.0001	0.0973
625	Nishitetsu Gojō	2	0.0171	669.0000	0.0000	0.0036
626	Dazaifu	1	0.0168	0.0000	0.0000	0.0018
627	Gorōmaru	2	0.0205	6,600.0000	0.0000	0.0175
628	Gakkōmae	2	0.0201	5,949.0000	0.0000	0.0158
629	Koganchaya	2	0.0198	5,296.0000	0.0000	0.0142
630	Kitano	2	0.0194	4,641.0000	0.0000	0.0126
631	Ōki	2	0.0190	3,984.0000	0.0000	0.0110
632	Kaneshima	2	0.0187	3,325.0000	0.0000	0.0094
633	Ōzeki	2	0.0183	2,664.0000	0.0000	0.0079
634	Hongō	2	0.0180	2,001.0000	0.0000	0.0064
635	Kamiura	2	0.0177	1,336.0000	0.0000	0.0048
636	Mada	2	0.0174	669.0000	0.0000	0.0033
637	Amagi	1	0.0171	0.0000	0.0000	0.0016
638	Kaizuka	2	0.0349	754.0000	0.0018	0.0049
639	Najima	2	0.0361	1,388.5000	0.0043	0.0060
640	Kashii-Miyamae	2	0.0361	1,289.1667	0.0044	0.0052
641	Nishitetsu Kashii	2	0.0348	631.1667	0.0020	0.0042
642	Kashii-Kaenmae	2	0.0337	9.0000	0.0014	0.0039
643	Tōnoharu	2	0.0338	44.8333	0.0019	0.0041
644	Mitoma	2	0.0338	669.0000	0.0017	0.0073
645	Nishitetsu Shingū	1	0.0327	0.0000	0.0006	0.0035
646	Higashi-Hie	2	0.0398	669.0000	0.0276	0.0094
647	Fukuokakūkō (Airport)	1	0.0383	0.0000	0.0095	0.0044
648	Gofukumachi	2	0.0372	1,931.5000	0.0055	0.0061
649	Chiyo-Kenchōguchi	2	0.0359	1,298.5000	0.0022	0.0050
650	Maidashi-Kyūdai-byōin-mae	2	0.0348	692.5000	0.0010	0.0043
651	Hakozaki-Miyamae	2	0.0338	115.0000	0.0006	0.0041
652	Hakozaki-Kyūdai-mae	2	0.0338	144.0000	0.0008	0.0042
653	Tenjin-Minami	1	0.0143	0.0000	0.0000	0.0016
654	Watanabe-dōri	2	0.0145	669.0000	0.0000	0.0032
655	Yakuin-ōdōri	2	0.0145	7,896.0000	0.0000	0.0150
656	Sakurazaka	2	0.0143	7,249.0000	0.0000	0.0138
657	Ropponmatsu	2	0.0141	6,600.0000	0.0000	0.0127
658	Befu	2	0.0139	5,949.0000	0.0000	0.0115
659	Chayama	2	0.0138	5,296.0000	0.0000	0.0104
660	Kanayama	2	0.0136	4,641.0000	0.0000	0.0093
661	Nanakuma	2	0.0134	3,984.0000	0.0000	0.0082
662	Fukudaimae	2	0.0132	3,325.0000	0.0000	0.0071
663	Umebayashi	2	0.0130	2,664.0000	0.0000	0.0060
664	Noke	2	0.0129	2,001.0000	0.0000	0.0049
665	Kamo	2	0.0127	1,336.0000	0.0000	0.0037
666	Jirōmaru	2	0.0126	669.0000	0.0000	0.0026
667	Hashimoto	1	0.0124	0.0000	0.0000	0.0013
668	Shin-Shimonoseki	1	0.0388	0.0000	0.0150	0.0047
669	Hakata-Minami	2	0.0423	70,701.9167	0.0433	0.0828
670	Shin-Ōmuta	2	0.0429	89,445.7500	0.0154	0.0883
671	Shin-Tamana	2	0.0425	89,280.2500	0.0076	0.0877

From Table 9, after we reorganized the data, we can conclude the top 10 ranks of each centrality case in Table 10.

Table 10. The top 10 ranks the highest centrality of the Kyushu railway network

Rank	Degree Centrality		Closeness Centrality		Betweenness Centrality	
	Station	C_D	**Station**	C_C	**Station**	C_B
1	Hakata	6	Kurume	0.0436	Chikugo-Funagoya	120,336.4167
2	Kurume	5	Chikugo-Funagoya	0.0435	Kurume	116,024.9167
3	Kumamoto	5	Shin-Tosu	0.0432	Shin-Tosu	113,426.3333
4	Kokura	4	Shin-Ōmuta	0.0429	Kumamoto	101,117.4167
5	Ōita	4	Shin-Tamana	0.0425	Shin-Ōmuta	89,445.7500
6	Kagoshima-Chūō	4	Hakata-Minami	0.0423	Shin-Tamana	89,280.2500
7	Kashii	4	Kurume-Kōkōmae	0.0421	Shin-Yatsushiro	82,016.0833
8	Shin-Tosu	4	Setaka	0.0421	Hakata	79,248.7500
9	Chikugo-Funagoya	4	Kumamoto	0.0421	Hakata-Minami	70,701.9167
10	Shin-Yatsushiro	4	Hizen-Fumoto	0.0419	Kagoshima-Chūō	67,534.0833

Rank	Eigenvector Centrality		Information Centrality	
	Station	C_E	**Station**	C_I
1	Hakata	0.0709	Kurume	0.2133
2	Kurume	0.0599	Chikugo-Funagoya	0.1994
3	Shin-Tosu	0.0550	Kumamoto	0.1977
4	Kokura	0.0435	Shin-Yatsushiro	0.1535
5	Hakata-Minami	0.0433	Shin-Tosu	0.1452
6	Chikugo-Funagoya	0.0371	Hakata	0.1282
7	Tosu	0.0349	Kubota	0.1114
8	Yoshizuka	0.0345	Ōmuta	0.1018
9	Hizen-Asahi	0.0326	Kagoshima-Chūō	0.0980
10	Gion	0.0291	Shin-Sakaemachi	0.0973

5.4 Centralities Testing Results with a Case Study of the Tokyo Subway Network

In the case of the Tokyo subway network, which is denser in the urban environment, all five node-based centrality analyses were used to analyze all 201 nodes and are illustrated in Table 11.

Table 11. All five node-based centralities of the Tokyo subway network

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
1	Shibuya	2	0.1117	0.0000	0.0009	0.0104
2	Omote-sando	5	0.1223	596.0346	0.0027	0.0193
3	Gaiemmae	2	0.1205	0.0000	0.0020	0.0113
4	Aoyama-itchome	5	0.1360	2,743.9835	0.0057	0.0323
5	Akasaka-mitsuke/Nagatacho	6	0.1436	2,678.5849	0.0141	0.0331
6	Tameike-sanno/Kokkai-gijidomae	5	0.1370	931.9499	0.0179	0.0203
7	Toranomon/Toranomon hills	4	0.1300	449.9912	0.0194	0.0151
8	Shimbashi	4	0.1300	1,467.7221	0.0262	0.0183
9	Ginza/Ginza-itchome	**8**	0.1437	2,854.4751	0.0532	0.0270
10	Kyobashi	2	0.1381	377.4774	0.0242	0.0137
11	Nihombashi	6	0.1458	2,664.1166	0.0482	0.0220
12	Mitsukoshimae	4	0.1414	807.9239	0.0368	0.0171
13	Kanda	2	0.1293	96.6959	0.0097	0.0127
14	Suehirocho	2	0.1279	73.7429	0.0038	0.0125
15	Ueno-hirokoji/Ueno-Okachimachi/Naka-okachimachi	5	0.1381	2,656.5169	0.0064	0.0361

Table 11. All five node-based centralities of the Tokyo subway network

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
16	Ueno	3	0.1232	1,234.9143	0.0018	0.0238
17	Inaricho	2	0.1113	175.1429	0.0005	0.0111
18	Tawaramachi	2	0.1047	53.9429	0.0003	0.0101
19	Asakusa	3	0.1078	305.6254	0.0006	0.0133
20	Ogikubo	1	0.0689	0.0000	0.0000	0.0058
21	Minami-asagaya	2	0.0740	199.0000	0.0000	0.0120
22	Shin-koenji	2	0.0797	396.0000	0.0000	0.0184
23	Higashi-koenji	2	0.0864	591.0000	0.0000	0.0252
24	Shin-nakano	2	0.0942	784.0000	0.0000	0.0327
25	Honancho	1	0.0794	0.0000	0.0000	0.0068
26	Nakano-fujimicho	2	0.0861	199.0000	0.0000	0.0142
27	NakanoShimbashi	2	0.0941	396.0000	0.0000	0.0222
28	Nakano-sakaue	5	0.1035	3,024.0000	0.0001	0.1014
29	Nishi-shinjuku	2	0.1130	3,077.0000	0.0002	0.0186
30	Shinjuku/Shinjuku-nishiguchi	5	0.1245	3,943.1756	0.0006	0.1258
31	Shinjuku-sanchome	5	0.1305	3,062.5218	0.0012	0.0271
32	Shinjuku-gyoemmae	2	0.1214	73.1778	0.0007	0.0118
33	Yotsuya-sanchome	2	0.1298	139.0242	0.0016	0.0123
34	Yotsuya	3	0.1417	686.8126	0.0061	0.0166
35	Kasumigaseki	4	0.1386	662.4876	0.0319	0.0173
36	Tokyo	2	0.1420	517.7969	0.0271	0.0136
37	Otemachi	**8**	0.1541	4,959.2884	**0.0605**	0.0420
38	Awajicho/Shin-ochanomizu/Ogawamachi	5	0.1484	1,946.5785	0.0279	0.0374
39	Ochanomizu	2	0.1384	145.4008	0.0078	0.0131
40	Hongo-sanchome	3	0.1419	2,171.5107	0.0050	0.0250
41	Korakuen/Kasuga	6	0.1484	**6,437.0020**	0.0067	**0.1600**
42	Myogadani	2	0.1317	1,448.5540	0.0017	0.0187
43	Shin-otsuka	2	0.1223	1,319.9468	0.0005	0.0157
44	Ikebukuro	4	0.1159	1,817.7326	0.0002	0.0591
45	Naka-meguro	1	0.0911	0.0000	0.0001	0.0077
46	Ebisu	2	0.1001	199.0000	0.0002	0.0164
47	Hiro-o	2	0.1110	396.0000	0.0009	0.0259
48	Roppongi	4	0.1244	1,543.7068	0.0036	0.0421
49	Kamiyacho	2	0.1220	244.2183	0.0055	0.0122
50	Hibiya/Yurakucho	5	0.1440	1,665.0517	0.0432	0.0245
51	Higashi-ginza	4	0.1319	395.7659	0.0293	0.0142
52	Tsukiji/Shintomicho	4	0.1326	1,542.9910	0.0250	0.0193
53	Hatchobori	2	0.1271	31.4802	0.0122	0.0122
54	Kayabacho	4	0.1336	1,446.5175	0.0260	0.0168
55	Ningyocho/Suitengumae	6	0.1351	2,348.1276	0.0360	0.0225
56	Kodemmacho	2	0.1315	292.5722	0.0120	0.0124
57	Akihabara/Iwamotocho	4	0.1408	1,450.4867	0.0144	0.0202
58	Iriya	2	0.1107	748.7143	0.0004	0.0158
59	Minowa	2	0.1006	564.7143	0.0001	0.0124
60	Minami-senju	2	0.0921	380.7143	0.0000	0.0105
61	Kita-senju	3	0.0868	436.0095	0.0000	0.0209
62	Nakano	1	0.0866	0.0000	0.0000	0.0070
63	Ochiai	2	0.0947	199.0000	0.0000	0.0147
64	Takadanobaba	2	0.1044	396.0000	0.0002	0.0229
65	Waseda	2	0.1161	591.0000	0.0006	0.0319
66	Kagurazaka	2	0.1307	784.0000	0.0025	0.0425
67	Iidabashi	6	0.1492	4,097.5142	0.0100	0.0780
68	Kudanshita	5	0.1533	2,883.4149	0.0175	0.0264
69	Takebashi	2	0.1486	810.9660	0.0186	0.0138
70	Monzen-nakacho	4	0.1267	2,776.7057	0.0128	0.0781

Table 11. All five node-based centralities of the Tokyo subway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
71	Kiba	2	0.1138	1,900.0000	0.0032	0.0639
72	Toyocho	2	0.1032	1,719.0000	0.0008	0.0552
73	Minami-sunamachi	2	0.0944	1,536.0000	0.0002	0.0477
74	Nishi-kasai	2	0.0868	1,351.0000	0.0001	0.0410
75	Kasai	2	0.0803	1,164.0000	0.0000	0.0349
76	Urayasu	2	0.0747	975.0000	0.0000	0.0293
77	Minami-gyotoku	2	0.0697	784.0000	0.0000	0.0241
78	Gyotoku	2	0.0654	591.0000	0.0000	0.0191
79	Myoden	2	0.0615	396.0000	0.0000	0.0143
80	Baraki-nakayama	2	0.0580	199.0000	0.0000	0.0095
81	Nishi-funabashi	1	0.0548	0.0000	0.0000	0.0047
82	Yoyogi-uehara	1	0.0956	0.0000	0.0001	0.0081
83	Yoyogi-koen	2	0.1056	199.0000	0.0003	0.0173
84	Meiji-jingumae	4	0.1178	588.0425	0.0011	0.0294
85	Nogizaka	2	0.1153	50.2231	0.0018	0.0111
86	Akasaka	2	0.1221	137.9492	0.0047	0.0122
87	Nijubashimae	2	0.1384	0.0000	0.0247	0.0130
88	Yushima	2	0.1307	963.2857	0.0071	0.0229
89	Nezu	2	0.1167	779.2857	0.0018	0.0169
90	Sendagi	2	0.1055	595.2857	0.0005	0.0129
91	Nishi-nippori	2	0.0963	414.0952	0.0001	0.0105
92	Machiya	2	0.0890	255.1048	0.0000	0.0093
93	Ayase	2	0.0800	199.0000	0.0000	0.0130
94	Kita-ayase	1	0.0741	0.0000	0.0000	0.0063
95	Wakoshi	1	0.0615	0.0000	0.0000	0.0051
96	Chikatetsu-narimasu	2	0.0655	199.0000	0.0000	0.0105
97	Chikatetsu-akatsuka	2	0.0699	396.0000	0.0000	0.0158
98	Heiwadai	2	0.0750	591.0000	0.0000	0.0213
99	Hikawadai	2	0.0809	784.0000	0.0000	0.0271
100	Kotake-mukaihara	2	0.0876	975.0000	0.0000	0.0333
101	Senkawa	2	0.0954	1,164.0000	0.0000	0.0401
102	Kanamecho	2	0.1047	1,351.0000	0.0000	0.0477
103	Higashi-ikebukuro	2	0.1110	124.2143	0.0002	0.0106
104	Gokokuji	2	0.1187	228.9620	0.0007	0.0115
105	Edogawabashi	2	0.1321	389.9620	0.0025	0.0151
106	Ichigaya	5	0.1504	3,896.9948	0.0100	0.0291
107	Kojimachi	2	0.1414	405.6217	0.0057	0.0132
108	Sakuradamon	2	0.1398	1,150.8313	0.0160	0.0170
109	Tsukishima	4	0.1240	1,866.7178	0.0103	0.0428
110	Toyosu	2	0.1107	396.0000	0.0026	0.0258
111	Tatsumi	2	0.0999	199.0000	0.0007	0.0163
112	Shin-kiba	1	0.0909	0.0000	0.0002	0.0077
113	Hanzomon	2	0.1426	585.7554	0.0075	0.0144
114	Jimbocho	4	**0.1561**	3,610.2537	0.0272	0.0233
115	Kiyosumi-shirakawa	4	0.1246	2,497.6274	0.0140	0.0285
116	Sumiyoshi	4	0.1126	1,922.6796	0.0044	0.0610
117	Kinshicho	2	0.1023	204.7413	0.0011	0.0114
118	Oshiage	2	0.0984	105.3318	0.0003	0.0098
119	Meguro	1	0.0908	0.0000	0.0001	0.0079
120	Shirokanedai	2	0.0998	199.0000	0.0004	0.0168
121	Shirokane-takanawa	3	0.1106	887.4140	0.0015	0.0283
122	Azabu-juban	4	0.1184	1,056.8375	0.0030	0.0196
123	Roppongi-itchome	2	0.1235	247.0388	0.0050	0.0124
124	Todaimae	2	0.1306	1,351.0000	0.0017	0.0575
125	Hon-komagome	2	0.1165	1,164.0000	0.0004	0.0474
126	Komagome	2	0.1050	975.0000	0.0001	0.0388
127	Nishigahara	2	0.0954	784.0000	0.0000	0.0312

Table 11. All five node-based centralities of the Tokyo subway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
128	Oji	2	0.0874	591.0000	0.0000	0.0242
129	Oji-kamiya	2	0.0806	396.0000	0.0000	0.0178
130	Shimo	2	0.0747	199.0000	0.0000	0.0117
131	Akabane-iwabuchi	1	0.0696	0.0000	0.0000	0.0057
132	Zoshigaya	2	0.1115	610.5390	0.0001	0.0149
133	Nishi-waseda	2	0.1126	626.9515	0.0001	0.0158
134	Higashi-shinjuku	4	0.1212	877.1514	0.0005	0.0218
135	Kita-sando	2	0.1211	440.0925	0.0005	0.0144
136	Nishi-magome	1	0.0638	0.0000	0.0000	0.0054
137	Magome	2	0.0681	199.0000	0.0000	0.0111
138	Nakanobu	2	0.0730	396.0000	0.0000	0.0168
139	Togoshi	2	0.0785	591.0000	0.0000	0.0227
140	Gotanda	2	0.0849	784.0000	0.0000	0.0291
141	Takanawadai	2	0.0924	975.0000	0.0002	0.0362
142	Sengakuji	2	0.1012	1,164.0000	0.0007	0.0441
143	Mita	4	0.1117	1,672.5500	0.0028	0.0560
144	Daimon	4	0.1204	1,462.2692	0.0081	0.0204
145	Takaracho	2	0.1338	187.7738	0.0185	0.0130
146	Higashi-nihombashi/Bakuro-yokoyama	4	0.1328	831.9729	0.0140	0.0169
147	Asakusabashi	2	0.1184	143.0648	0.0038	0.0117
148	Kuramae	4	0.1188	642.8400	0.0019	0.0153
149	Honjo-azumabashi	2	0.0996	124.5905	0.0002	0.0101
150	Shibakoen	2	0.1107	236.2310	0.0014	0.0111
151	Onarimon	2	0.1162	326.1643	0.0030	0.0117
152	Uchisaiwaicho	2	0.1281	481.3810	0.0110	0.0150
153	Suidobashi	2	0.1449	1,989.0199	0.0081	0.0174
154	Hakusan	2	0.1318	2,604.0000	0.0017	0.0820
155	Sengoku	2	0.1184	2,431.0000	0.0004	0.0723
156	Sugamo	2	0.1073	2,256.0000	0.0001	0.0642
157	Nishi-sugamo	2	0.0980	2,079.0000	0.0000	0.0571
158	Shin-itabashi	2	0.0902	1,900.0000	0.0000	0.0507
159	Itabashikuyakushomae	2	0.0834	1,719.0000	0.0000	0.0449
160	Itabashihoncho	2	0.0775	1,536.0000	0.0000	0.0395
161	Motohasunuma	2	0.0724	1,351.0000	0.0000	0.0345
162	Shimura-sakaue	2	0.0678	1,164.0000	0.0000	0.0298
163	Shimura-sanchome	2	0.0637	975.0000	0.0000	0.0253
164	Hasune	2	0.0601	784.0000	0.0000	0.0210
165	Nishidai	2	0.0568	591.0000	0.0000	0.0168
166	Takashimadaira	2	0.0538	396.0000	0.0000	0.0127
167	Shin-takashimadaira	2	0.0511	199.0000	0.0000	0.0085
168	Nishi-takashimadaira	1	0.0487	0.0000	0.0000	0.0042
169	Akebonobashi	2	0.1380	2,768.0241	0.0027	0.0209
170	Hamacho	2	0.1199	84.4025	0.0046	0.0115
171	Morishita	4	0.1155	355.7784	0.0054	0.0141
172	Kikukawa	2	0.1049	25.0000	0.0023	0.0100
173	Nishi-ojima	2	0.1020	1,351.0000	0.0011	0.0484
174	Ojima	2	0.0932	1,164.0000	0.0003	0.0405
175	Higashi-ojima	2	0.0857	975.0000	0.0001	0.0335
176	Funabori	2	0.0792	784.0000	0.0000	0.0272
177	Ichinoe	2	0.0736	591.0000	0.0000	0.0214
178	Mizue	2	0.0687	396.0000	0.0000	0.0159
179	Shinozaki	2	0.0644	199.0000	0.0000	0.0105
180	Motoyawata	1	0.0605	0.0000	0.0000	0.0051
181	Tochomae	2	0.1120	181.0000	0.0002	0.0114
182	Wakamatsu-kawada	2	0.1185	103.5449	0.0003	0.0118
183	Ushigome-yanagicho	2	0.1229	174.1608	0.0007	0.0121

Table 11. All five node-based centralities of the Tokyo subway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
184	Ushigome-kagurazaka	2	0.1327	294.9493	0.0025	0.0152
185	Shin-okachimachi	2	0.1251	555.0085	0.0020	0.0139
186	Ryōgoku	2	0.1134	166.0313	0.0018	0.0115
187	Kachidoki	2	0.1128	258.3858	0.0027	0.0122
188	Tsukijishijō	2	0.1062	181.1530	0.0012	0.0107
189	Shiodome	2	0.1120	239.0868	0.0022	0.0118
190	Akabanebashi	2	0.1150	159.1622	0.0026	0.0119
191	Kokuritsu-Kyōgijō	2	0.1257	1,375.9542	0.0015	0.0176
192	Yoyogi	2	0.1206	1,319.9899	0.0005	0.0167
193	Nishi-shinjuku-gochome	2	0.1027	17.0000	0.0001	0.0099
194	Higashi-Nakano	2	0.0945	1,351.0000	0.0000	0.0453
195	Nakai	2	0.0869	1,164.0000	0.0000	0.0381
196	Ochiai-minami-nagasaki	2	0.0803	975.0000	0.0000	0.0317
197	Shin-egota	2	0.0746	784.0000	0.0000	0.0258
198	Nerima	2	0.0696	591.0000	0.0000	0.0204
199	Toshimaen	2	0.0652	396.0000	0.0000	0.0152
200	Nerima-kasugachō	2	0.0613	199.0000	0.0000	0.0101
201	Hikarigaoka	1	0.0578	0.0000	0.0000	0.0049

From Table 11, after we reorganized the data, we can conclude the top 10 ranks of each centrality case for the Tokyo subway network in Table 12.

Table 12. The top 10 ranks the highest centrality of the Tokyo subway network

Rank	**Degree Centrality**		**Closeness Centrality**		**Betweenness Centrality**	
	Station	C_D	**Station**	C_C	**Station**	C_B
1	Ginza/Ginza-itchome	8	Jimbocho	0.1561	Korakuen/Kasuga	6,437.0020
2	Otemachi	8	Otemachi	0.1541	Otemachi	4,959.2884
3	Akasaka-mitsuke/Nagatacho	6	Kudanshita	0.1533	Iidabashi	4,097.5142
4	Nihombashi	6	Ichigaya	0.1504	Shinjuku/Shinjuku-nishiguchi	3,943.1756
5	Korakuen/Kasuga	6	Iidabashi	0.1492	Ichigaya	3,896.9948
6	Ningyocho/Suitengumae	6	Takebashi	0.1486	Jimbocho	3,610.2537
7	Iidabashi	6	Awajicho/Shin-ochanomizu/Ogawamachi	0.1484	Nishi-shinjuku	3,077.0000
8	Shinjuku/Shinjuku-nishiguchi	5	Korakuen/Kasuga	0.1484	Shinjuku-sanchome	3,062.5218
9	Kudanshita	5	Nihombashi	0.1458	Nakano-sakaue	3,024.0000
10	Awajicho/Shin-ochanomizu/Ogawamachi	5	Suidobashi	0.1449	Kudanshita	2,883.4149

Rank	**Eigenvector Centrality**		**Information Centrality**	
	Station	C_E	**Station**	C_I
1	Otemachi	0.0605	Korakuen/Kasuga	0.1600
2	Ginza/Ginza-itchome	0.0532	Shinjuku/Shinjuku-nishiguchi	0.1258
3	Nihombashi	0.0482	Nakano-sakaue	0.1014
4	Hibiya/Yurakucho	0.0432	Hakusan	0.0820
5	Mitsukoshimae	0.0368	Monzen-nakacho	0.0781
6	Ningyocho/Suiten gumae	0.0360	Iidabashi	0.0780
7	Kasumigaseki	0.0319	Sengoku	0.0723
8	Higashi-ginza	0.0293	Sugamo	0.0642
9	Awajicho/Shin-ochanomizu/Ogawamachi	0.0279	Kiba	0.0639
10	Jimbocho	0.0272	Sumiyoshi	0.0610

5.5 Centralities Testing Results with a Case Study of the Osaka Subway Network

Similar to the Tokyo subway network, the Osaka subway network is dense and located in an urban environment. All five node-based centrality analyses were used to analyze all 106 nodes and are illustrated in Table 13.

Table 13. All five node-based centralities of the Osaka subway network

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
1	Esaka	1	0.0864	0.0000	0.0001	0.0123
2	Higashi-Mikuni	2	0.0944	104.0000	0.0002	0.0255
3	Shin-Ōsaka	2	0.1038	206.0000	0.0007	0.0386
4	Nishinakajima-Minamigata	2	0.1151	306.0000	0.0019	0.0528
5	Nakatsu	2	0.1288	404.0000	0.0057	0.0687
6	Umeda	4	0.1458	951.1667	0.0166	0.0954
7	Yodoyabashi (Osaka City Hall)	2	0.1530	670.6667	0.0221	0.0346
8	Hommachi (Semba-nishi)	**5**	0.1680	**1,566.5833**	0.0559	0.0972
9	Shinsaibashi	4	0.1636	704.8333	0.0560	0.0357
10	Namba	4	0.1623	1,215.5500	0.0451	0.0500
11	Daikokuchō	3	0.1523	1,280.0333	0.0216	0.0755
12	Dōbutsuen-mae (Shinsekai)	4	0.1472	792.7000	0.0184	0.0541
13	Tennōji	4	0.1480	1,532.4000	0.0127	**0.1881**
14	Shōwachō	2	0.1312	594.0000	0.0043	0.0852
15	Nishitanabe	2	0.1175	500.0000	0.0015	0.0700
16	Nagai	2	0.1062	404.0000	0.0005	0.0568
17	Abiko	2	0.0967	306.0000	0.0002	0.0448
18	Kitahanada	2	0.0886	206.0000	0.0001	0.0334
19	Shinkanaoka	2	0.0817	104.0000	0.0000	0.0224
20	Nakamozu	1	0.0756	0.0000	0.0000	0.0109
21	Dainichi	1	0.0820	0.0000	0.0000	0.0127
22	Moriguchi	2	0.0892	104.0000	0.0000	0.0268
23	Taishibashi-Imaichi	4	0.0975	594.6667	0.0001	0.0778
24	Sembayashi-Omiya	2	0.0978	316.1667	0.0001	0.0262
25	Sekime-Takadono	2	0.1023	355.1667	0.0002	0.0258
26	Noe-Uchindai	2	0.1086	414.6667	0.0006	0.0278
27	Miyakojima	2	0.1170	484.6667	0.0016	0.0320
28	Tenjimbashisuji Rokuchōme	3	0.1276	560.6667	0.0048	0.0394
29	Nakazakichō	2	0.1318	411.0000	0.0065	0.0250
30	Minami-morimachi	4	0.1483	559.8333	0.0201	0.0378
31	Temmabashi	2	0.1532	474.4167	0.0189	0.0302
32	Tanimachi Yonchōme	4	**0.1699**	1,178.4167	0.0419	0.0591
33	Tanimachi Rokuchōme	4	0.1643	475.2667	0.0383	0.0350
34	Tanimachi Kyūchōme	4	0.1656	1,354.2833	0.0349	0.0594
35	Shitennōji-mae Yūhigaoka	2	0.1525	827.9000	0.0145	0.0360
36	Abeno	2	0.1318	776.0000	0.0043	0.0987
37	Fuminosato	2	0.1186	686.0000	0.0015	0.0841
38	Tanabe	2	0.1075	594.0000	0.0005	0.0715
39	Komagawa-Nakano	2	0.0981	500.0000	0.0002	0.0601
40	Hirano	2	0.0901	404.0000	0.0001	0.0496
41	Kire-Uriwari	2	0.0832	306.0000	0.0000	0.0396
42	Deto	2	0.0771	206.0000	0.0000	0.0300
43	Nagahara	2	0.0718	104.0000	0.0000	0.0203
44	Yaominami	1	0.0671	0.0000	0.0000	0.0099
45	Nishi-Umeda	1	0.1266	0.0000	0.0057	0.0181
46	Higobashi	2	0.1446	104.0000	0.0188	0.0393

Table 13. All five node-based centralities of the Osaka subway network (Cont.)

Node	Station	Degree Centrality	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality	Information Centrality
47	Hanazonochō	2	0.1361	719.5333	0.0074	0.0553
48	Kishinosato	2	0.1230	636.5333	0.0025	0.0451
49	Tamade	2	0.1122	553.5333	0.0009	0.0372
50	Kitakagaya	2	0.1031	470.5333	0.0003	0.0311
51	Suminoekōen	2	0.0954	387.5333	0.0001	0.0264
52	Cosmosquare	2	0.0950	386.4667	0.0001	0.0266
53	Osakako (Tempozan)	2	0.1026	469.4667	0.0004	0.0314
54	Asashiobashi	2	0.1116	552.4667	0.0013	0.0377
55	Bentencho	2	0.1223	635.4667	0.0037	0.0458
56	Kujo	2	0.1352	718.4667	0.0109	0.0565
57	Awaza	4	0.1512	1,132.0333	0.0319	0.1053
58	Sakaisuji-Hommachi (Semba-higashi)	4	0.1669	780.0500	0.0542	0.0468
59	Morinomiya	4	0.1577	885.4167	0.0257	0.0556
60	Midoribashi	4	0.1534	1,292.5000	0.0137	0.0881
61	Fukaebashi	2	0.1340	206.0000	0.0047	0.0496
62	Takaida	2	0.1187	104.0000	0.0016	0.0316
63	Nagata	1	0.1063	0.0000	0.0005	0.0150
64	Nodahanshin	1	0.1169	0.0000	0.0033	0.0168
65	Tamagawa	2	0.1320	104.0000	0.0107	0.0362
66	Nishi-Nagahori	4	0.1468	382.0833	0.0269	0.0584
67	Sakuragawa	2	0.1458	257.2667	0.0220	0.0254
68	Nippombashi	4	0.1628	601.7167	0.0481	0.0374
69	Tsuruhashi	2	0.1534	844.1667	0.0135	0.0366
70	Imazato	3	0.1499	1,054.1667	0.0093	0.0855
71	Shin-Fukae	2	0.1317	306.0000	0.0032	0.0589
72	Shōji	2	0.1171	206.0000	0.0011	0.0426
73	Kita-Tatsumi	2	0.1053	104.0000	0.0004	0.0278
74	Minami-Tatsumi	1	0.0954	0.0000	0.0001	0.0133
75	Ōgimachi	2	0.1337	214.3333	0.0076	0.0239
76	Kitahama	2	0.1519	93.5833	0.0227	0.0249
77	Nagahoribashi	4	0.1626	242.7000	**0.0573**	0.0305
78	Ebisuchō (Nippombashi-suji)	2	0.1501	81.6000	0.0203	0.0240
79	Tengachaya	1	0.1286	0.0000	0.0056	0.0190
80	Taishō	1	0.1142	0.0000	0.0028	0.0163
81	Dome-mae Chiyozaki (Kyocera Dome Osaka)	2	0.1286	104.0000	0.0091	0.0350
82	Nishiōhashi	2	0.1446	10.7500	0.0253	0.0234
83	Matsuyamachi	2	0.1508	7.7500	0.0292	0.0238
84	Tamatsukuri	2	0.1516	61.4000	0.0195	0.0231
85	Osaka Business Park (Osaka-jo Hall)	2	0.1427	254.8333	0.0090	0.0244
86	Kyōbashi	2	0.1305	190.3333	0.0038	0.0215
87	Gamō-yonchōme	4	0.1282	981.8333	0.0035	0.0905
88	Imafuku-Tsurumi	2	0.1146	306.0000	0.0012	0.0540
89	Yokozutsumi	2	0.1034	206.0000	0.0004	0.0393
90	Tsurumi-ryokuchi	2	0.0941	104.0000	0.0001	0.0258
91	Kadoma-minami	1	0.0861	0.0000	0.0000	0.0124
92	Itakano	1	0.0761	0.0000	0.0000	0.0117
93	Zuikō Yonchōme	2	0.0822	104.0000	0.0000	0.0242
94	Daidō-Toyosato	2	0.0893	206.0000	0.0000	0.0369
95	Shimizu	2	0.1012	401.3333	0.0002	0.0310
96	Shimmori-Furuichi	2	0.1078	469.0000	0.0004	0.0327
97	Sekime-Seiiku	2	0.1170	544.6667	0.0012	0.0372
98	Shigino	2	0.1395	792.5000	0.0052	0.0309
99	Trade Center-mae	2	0.0884	303.4667	0.0000	0.0230
100	Nakafuto	2	0.0827	220.4667	0.0000	0.0203

Table 13. All five node-based centralities of the Osaka subway network (Cont.)

Node	**Station**	**Degree Centrality**	**Closeness Centrality**	**Betweenness Centrality**	**Eigenvector Centrality**	**Information Centrality**
101	Port Town-nishi	2	0.0788	156.8333	0.0000	0.0186
102	Port Town-higashi	2	0.0766	119.7000	0.0000	0.0178
103	Ferry Terminal	2	0.0755	102.8667	0.0000	0.0178
104	Nankō-higashi	2	0.0781	140.5333	0.0000	0.0186
105	Nankōguchi	2	0.0830	221.5333	0.0000	0.0202
106	Hirabayashi	2	0.0888	304.5333	0.0000	0.0228

From Table 13, after we reorganized the data, we can conclude the top 10 ranks of each centrality case for the Osaka subway network in Table 14.

Table 14. The top 10 ranks the highest centrality of the Osaka subway network

Rank	**Degree Centrality**		**Closeness Centrality**		**Betweenness Centrality**	
	Station	C_D	**Station**	C_C	**Station**	C_B
1	Hommachi (Semba-nishi)	5	Tanimachi Yonchōme	0.1699	Hommachi (Semba-nishi)	1,566.5833
2	Umeda	4	Hommachi (Semba-nishi)	0.1680	Tennōji	1,532.4000
3	Shinsaibashi	4	Sakaisuji-Hommachi (Semba-higashi)	0.1669	Tanimachi Kyūchōme	1,354.2833
4	Namba	4	Tanimachi Kyūchōme	0.1656	Midoribashi	1,292.5000
5	Tanimachi Yonchōme	4	Tanimachi Rokuchōme	0.1643	Daikokuchō	1,280.0333
6	Tanimachi Rokuchōme	4	Shinsaibashi	0.1636	Namba	1,215.5500
7	Tanimachi Kyūchōme	4	Nippombashi	0.1628	Tanimachi Yonchōme	1,178.4167
8	Sakaisuji-Hommachi (Semba-higashi)	4	Nagahoribashi	0.1626	Awaza	1,132.0333
9	Nagahoribashi	4	Namba	0.1623	Imazato	1,054.1667
10	Nippombashi	4	Morinomiya	0.1577	Gamō-yonchōme	981.8333

Rank	**Eigenvector Centrality**		**Information Centrality**	
	Station	C_E	**Station**	C_I
1	Nagahoribashi	0.0573	Tennōji	0.1881
2	Shinsaibashi	0.0560	Awaza	0.1053
3	Hommachi (Semba-nishi)	0.0559	Abeno	0.0987
4	Sakaisuji-Hommachi (Semba-higashi)	0.0542	Hommachi (Semba-nishi)	0.0972
5	Nippombashi	0.0481	Umeda	0.0954
6	Namba	0.0451	Gamō-yonchōme	0.0905
7	Tanimachi Yonchōme	0.0419	Midoribashi	0.0881
8	Tanimachi Rokuchōme	0.0383	Imazato	0.0855
9	Tanimachi Kyūchōme	0.0349	Shōwachō	0.0852
10	Awaza	0.0319	Fuminosato	0.0841

5.6 Comparison of Centrality Results from All Case Study Networks

To analyze each type of centrality, we divided each case into five levels, making it easy to analyze and manage priority, especially when considering the most critical or important node that strongly influences the network. The results from all three case studies are illustrated in Figs. 12 – 26 by following.

5.6.1 Degree centrality result

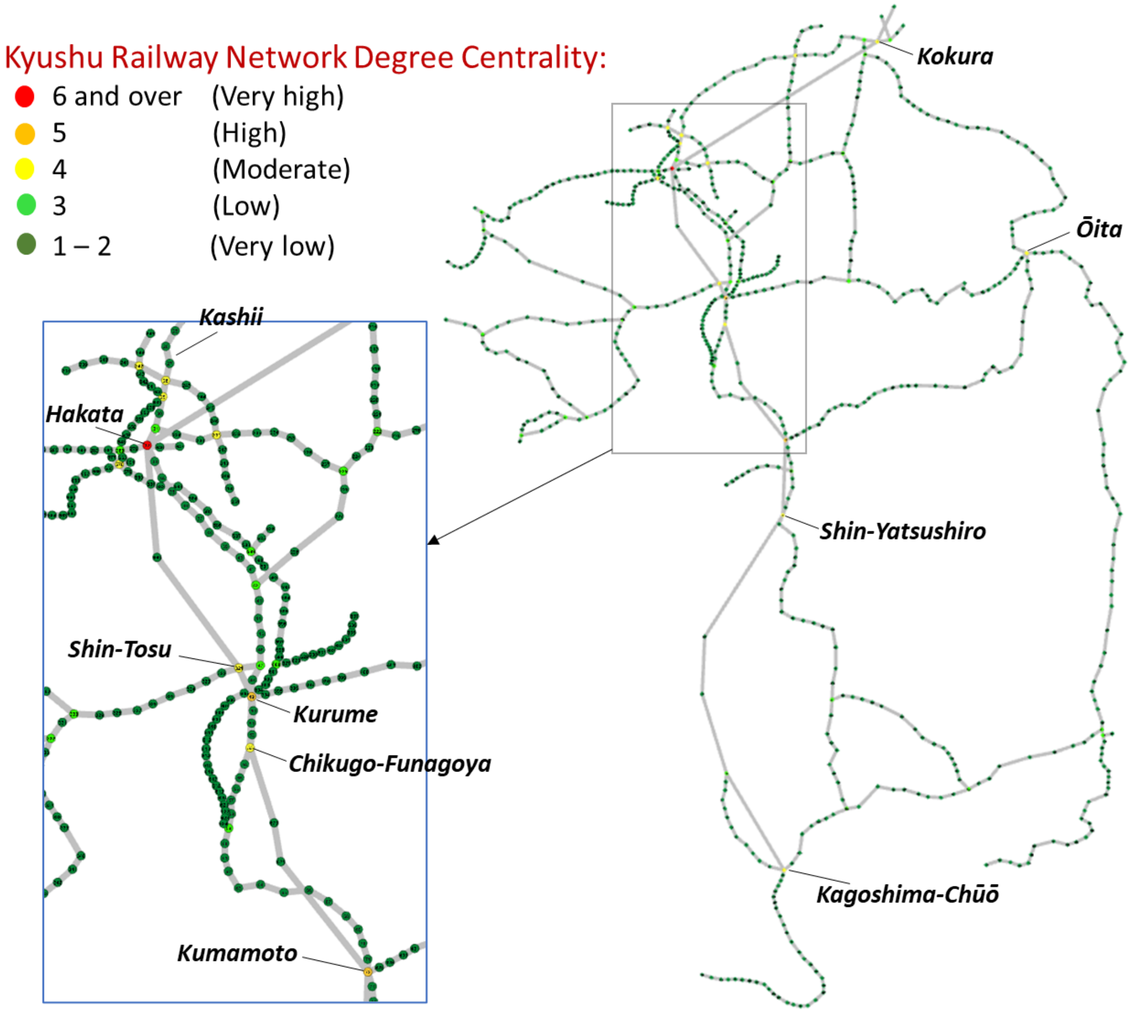


Figure 12. Degree centrality result of the Kyushu railway network

From Fig. 12, the degree centrality analysis of the Kyushu network showed that the Hakata station had the highest centrality, with a degree centrality of 6. This node connected six railway sections, including the Shinkansen line and subway line. The Kumamoto and Kurume stations shared the second rank with a degree centrality of 5.

Similarly, the Tokyo subway network in Fig. 13 showed the most critical nodes located at Ginza/Ginza-itchome and Otemachi stations, which had a degree centrality of 8. While in the case of the Osaka subway network (Fig. 14), the most critical node was Hommachi (Semba-nishi), which had a degree centrality of 5 and had fewer connections than the most critical nodes in the Tokyo subway case

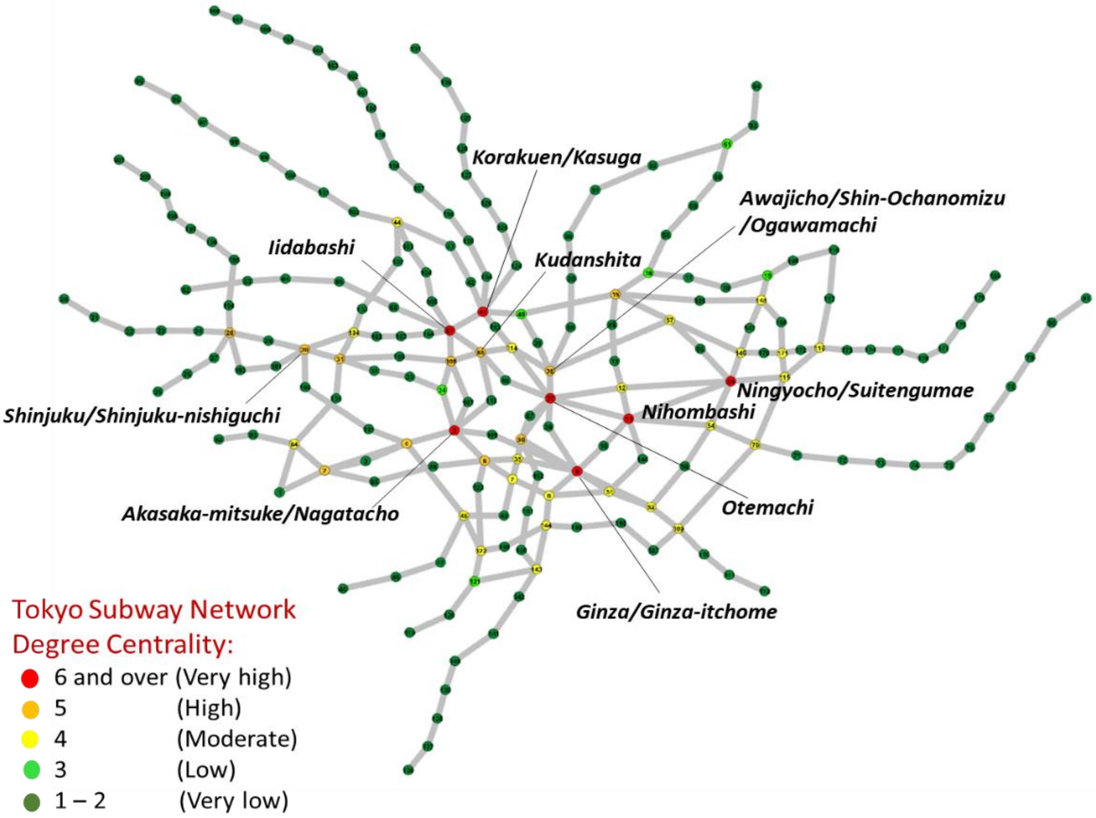


Figure 13. Degree centrality result of the Tokyo subway network

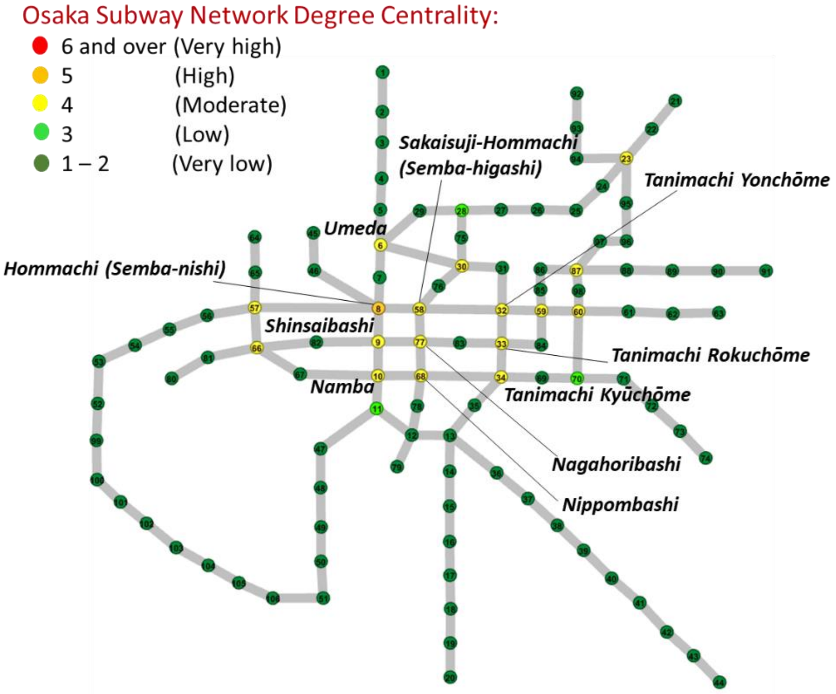


Figure 14. Degree centrality result of the Osaka subway network

However, the problem with this method is it is very difficult to manage priority because it shows only the number of lines that connect the node but cannot tell the influence of other factors, especially the flow and the average shortest distance. In the example, Ginza/Ginza-itchome and Otemachi stations, which both had the same degree centrality of 8, could not show what station is more important to protect or manage as the first priority.

5.6.2 Closeness centrality result

In the case of closeness centrality of the Kyushu railway network (Fig. 15), this network's very high centrality stations were located in the Fukuoka and Kumamoto corridor, especially along the Shinkansen line. The area considered very critically was the stations surrounding Kurume, Fukuoka, and Kumamoto city. The most centrality station was located at the Kurume station with a centrality value of 0.0436, followed by the Chikugo-Funagoya and Shin-Tosu stations, which had a centrality of about 0.0435 and 0.0432, respectively.

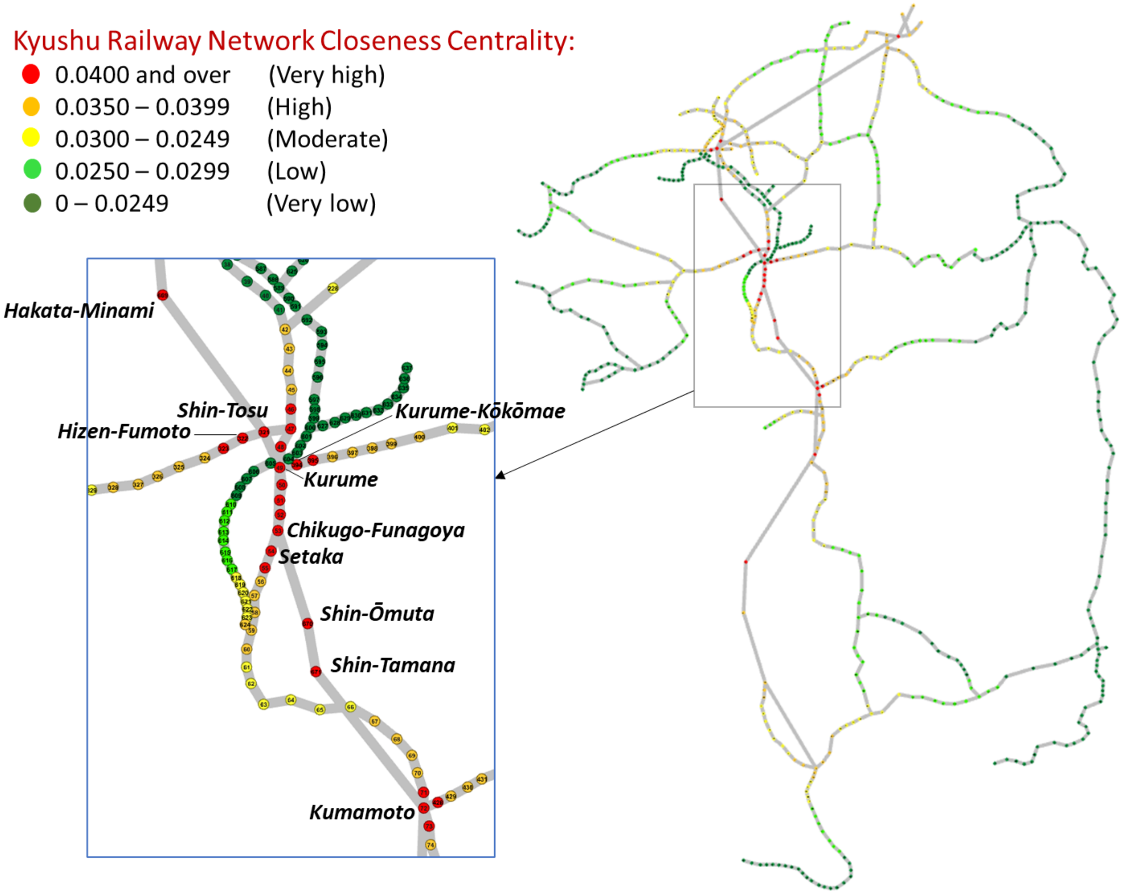


Figure 15. Closeness centrality result of the Kyushu railway network

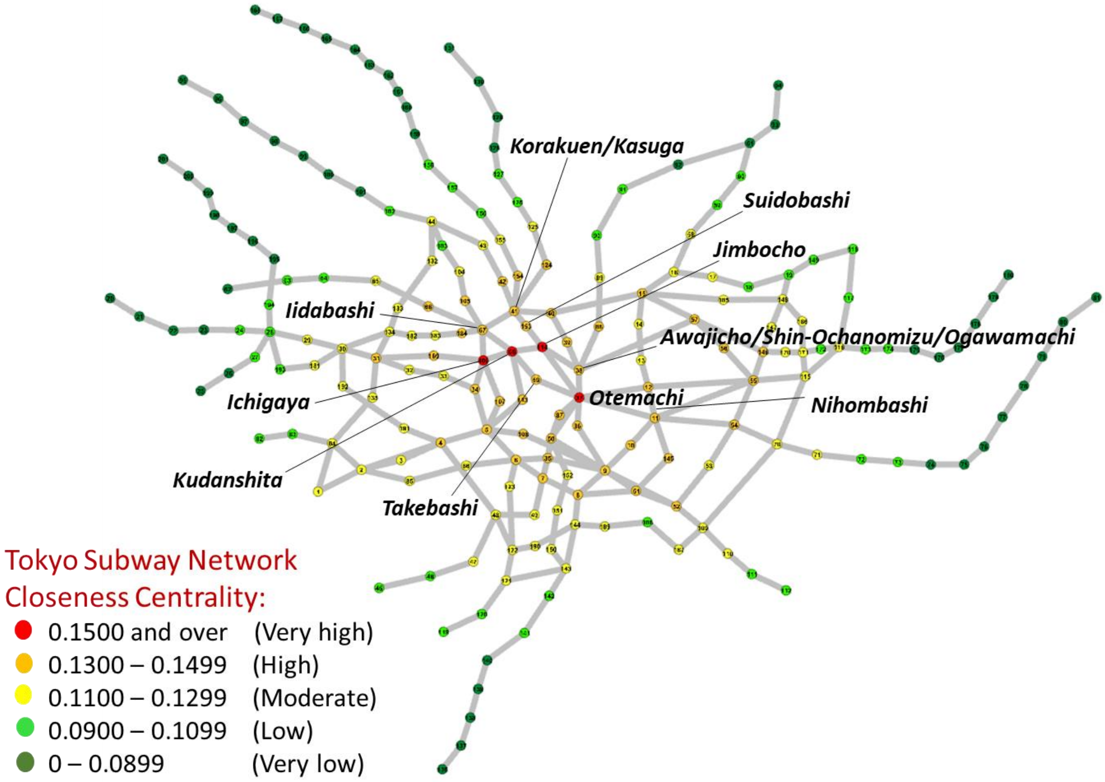


Figure 16. Closeness centrality result of the Tokyo subway network

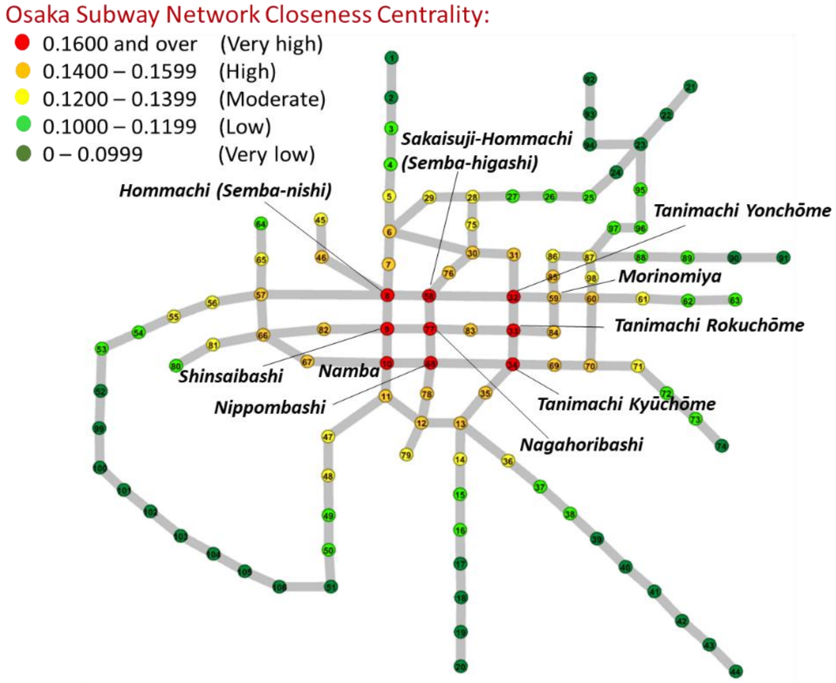


Figure 17. Closeness centrality result of the Osaka subway network

In the case of the Tokyo subway network from Fig. 16, Jimbocho, Otemachi, Kudanshita, and Ichigaya stations were very critical nodes for closeness centrality, located in the central area of the network. All four station had a centrality value 0.1561, 0.1541, 0.1533, and 0.1504, respectively. But in the Osaka subway network from Fig. 17, the very critical and high-influence nodes were located in the central area as a larger cluster group, similar to the Kyushu railway network case. The most important station in the Osaka subway network was Tanimachi Yonchōme station, which had 0.1699 of closeness centrality.

However, according to Rodrigues's (2019) explanation, the closeness centrality shows the range of centrality value, which is narrow and hard to divide into five levels for analyzing the level of centrality. Moreover, its result does not show the specific critical or influence nodes clearly if operators do not consider more detail on the centrality value of each node.

5.6.3 Eigenvector centrality result

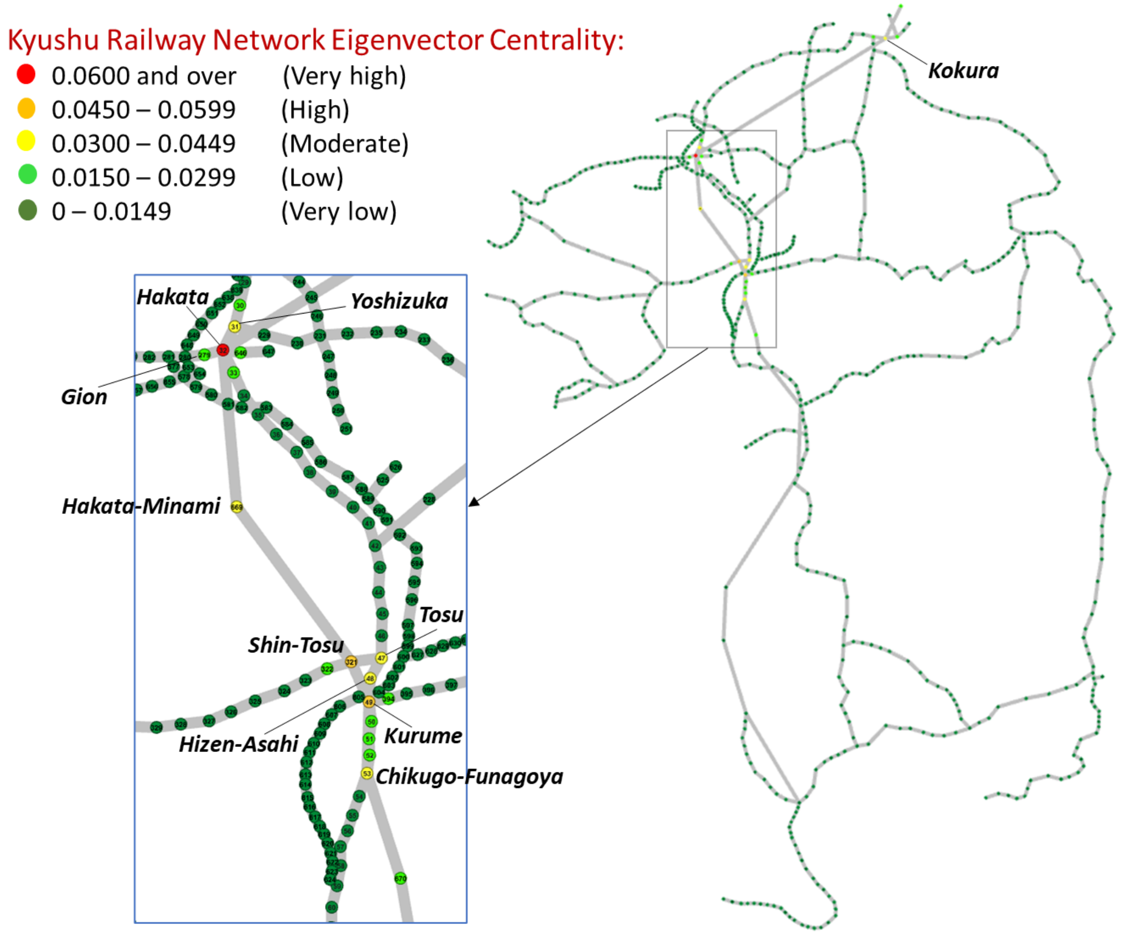


Figure 18. Eigenvector centrality result of the Kyushu railway network

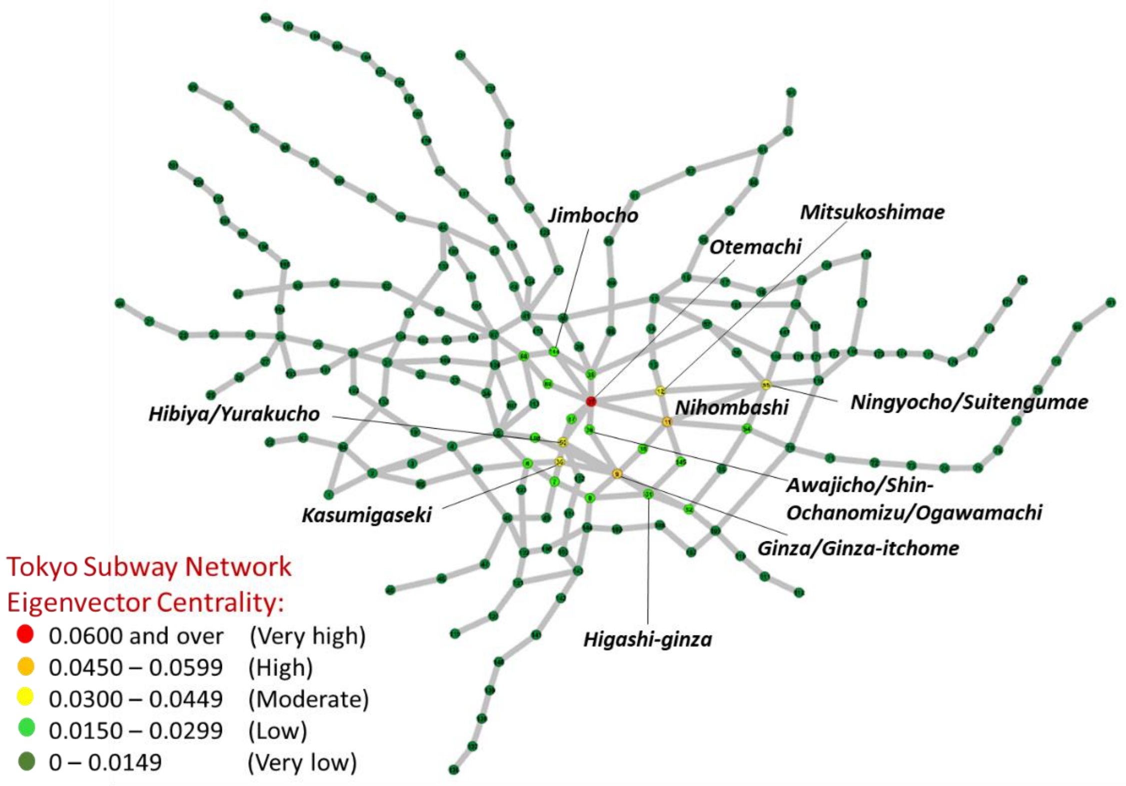


Figure 19. Eigenvector centrality result of the Tokyo subway network

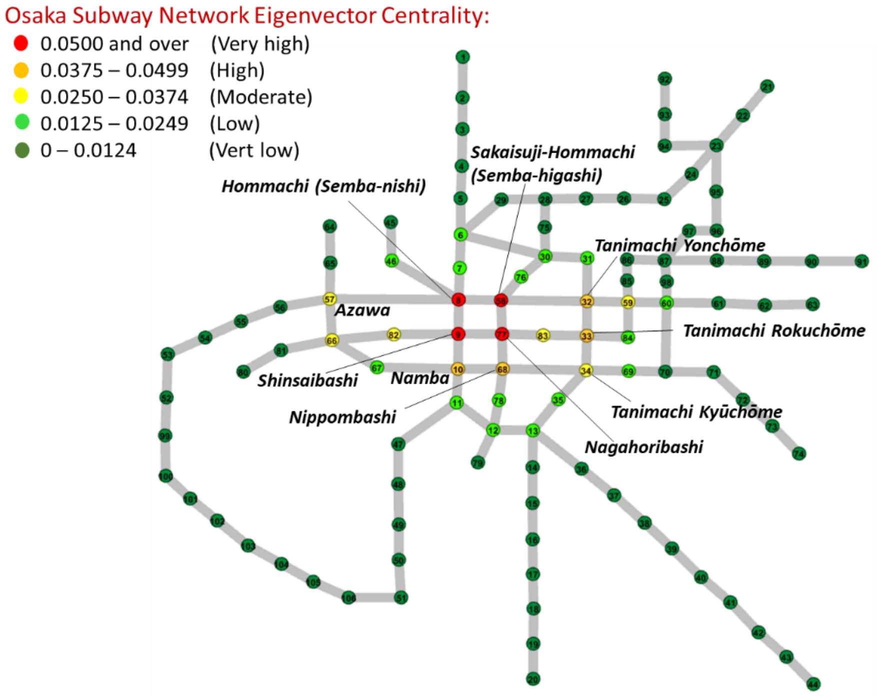


Figure 20. Eigenvector centrality result of the Osaka subway network

In the Kyushu network, as shown in Fig. 18, the most influential or critical station in the eigenvector centrality field was the Hakata station. The station had a centrality value of 0.0709 and connected six railway sections, including the subway line. This was followed by the Kurume and Shin-Tosu stations, which also connected several mainlines and had a centrality volume of 0.0599 and 0.0550, respectively.

In the Tokyo subway network, as shown in Fig. 19, The Otemachi station was the most influential station in this network, with a centrality value of 0.0605. This station is located in the central and was the most critical station in the degree centrality case. In the case of the Osaka subway network (Fig. 20), the very critical and important nodes were, Nagahoribashi, Shinsaibashi, Hommachi (Semba-nishi), and Sakaisuji-Hommachi (Semba-higashi) stations. Each station had an eigenvector centrality of 0.0573, 0.0560, 0.559, and 0.0542, respectively. In this case, all four very critical nodes were located very close to each other in the central area and had 4 – 5 connected links, which show they had a probability of influencing each other.

This method shows the most critical node more clearly than closeness centrality and illustrates the influence of surround or neighbor nodes too. However, this criterion does not mainly focus on the flow like closeness, betweenness, and information centralities.

5.6.4 Betweenness centrality result

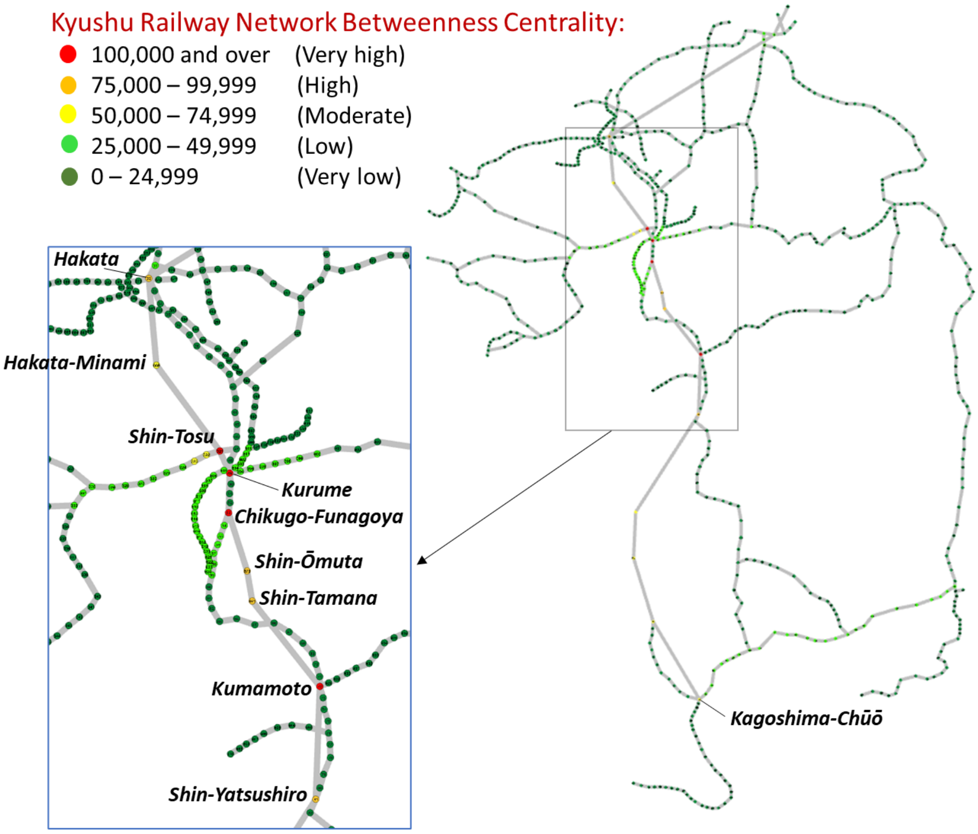


Figure 21. Betweenness centrality result of the Kyushu railway network

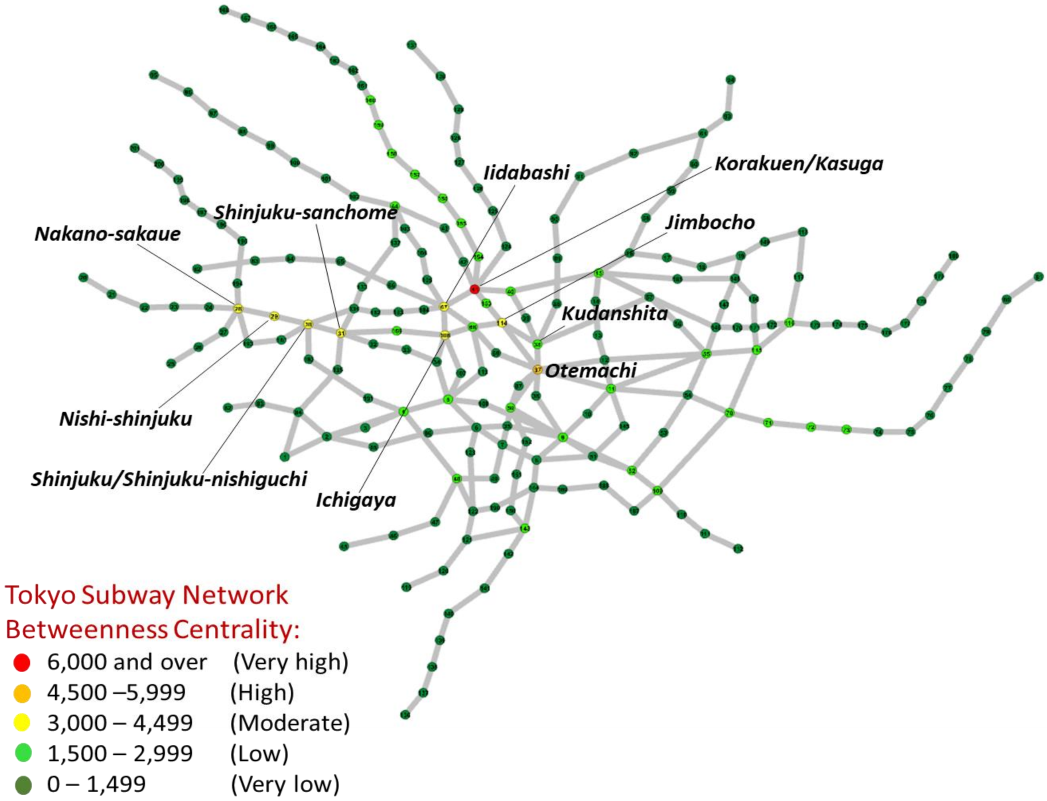


Figure 22. Betweenness centrality result of the Tokyo subway network

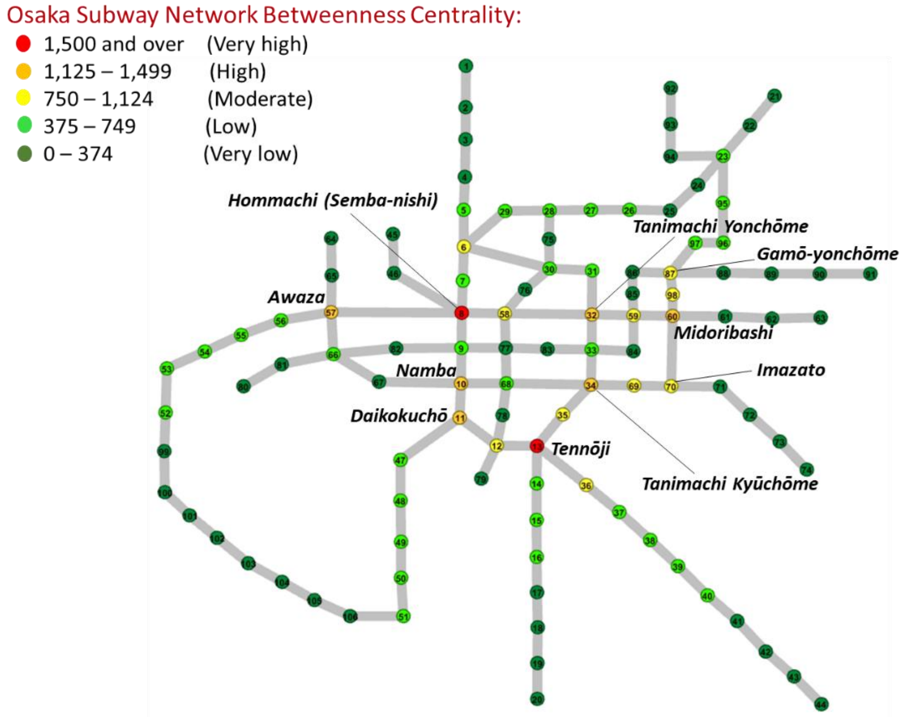


Figure 23. Betweenness centrality result of the Osaka subway network

In Fig. 21, the high or very high value of node-based betweenness centrality stations of the Kyushu railway network was mainly located in the Fukuoka and Kumamoto corridor, similar to the closeness centrality. The top four most critical stations, Chikugo-Funagoya, Kurume, Shin-Tosu, and Kumamoto, had a value of about 120,336, 116,025, 113,426, and 101,117, respectively. If we consider the Tokyo subway case in Fig. 22, the most critical station was the Korakuen/Kasuga station which had a betweenness centrality value of about 6,437. While the Osaka subway network in Fig. 23 showed the critical nodes at the Hommachi (Semba-nishi) and Tennōji stations that had betweenness centrality of about 1,567 and 1,532, respectively.

From all case study results, the betweenness centrality shows the critical or very important nodes more clearly than the closeness centrality analysis and makes it operators easier to classify the level of priority for strategic planning. All these nodes can be the very influence or critical nodes if we apply them to analyze traffic flow or passenger flow.

5.6.5 Information centrality result

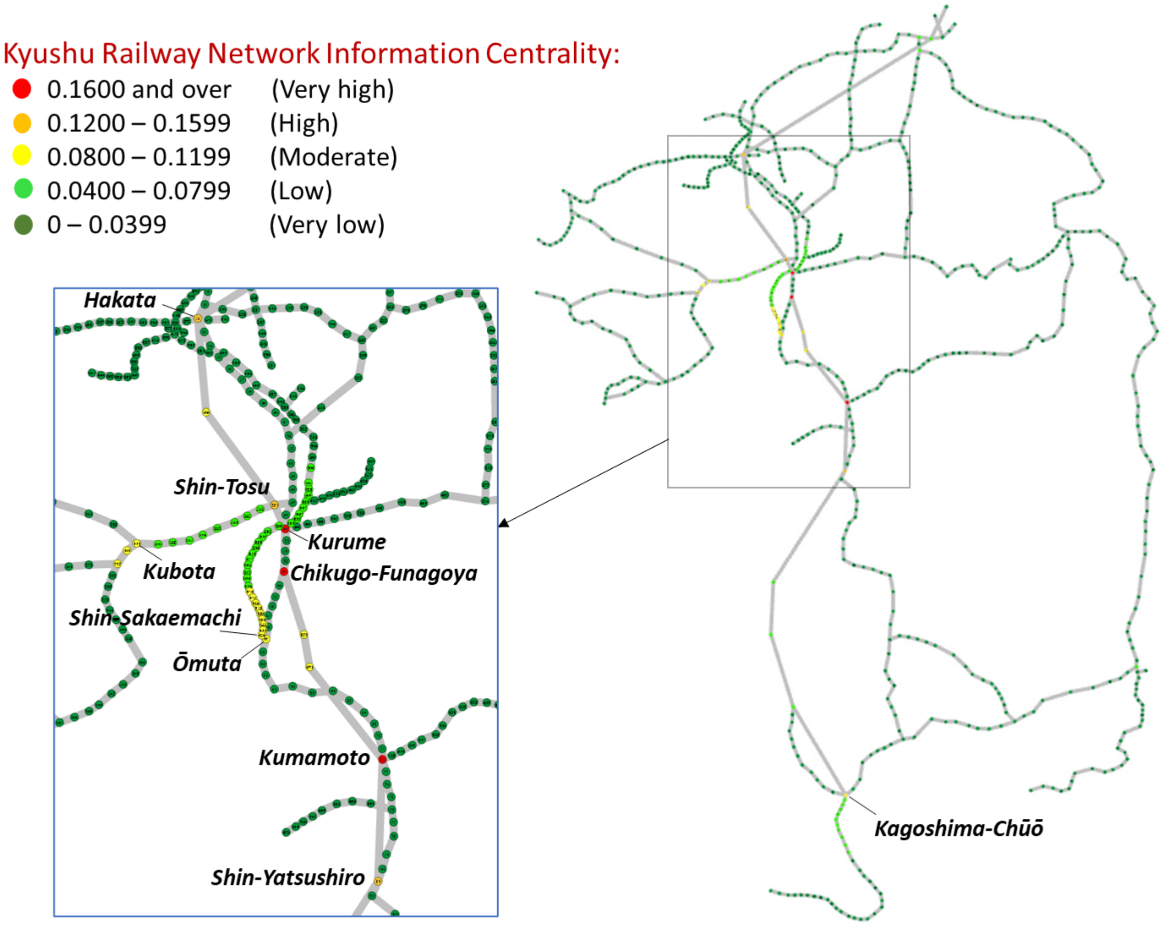


Figure 24. Information centrality result of the Kyushu railway network

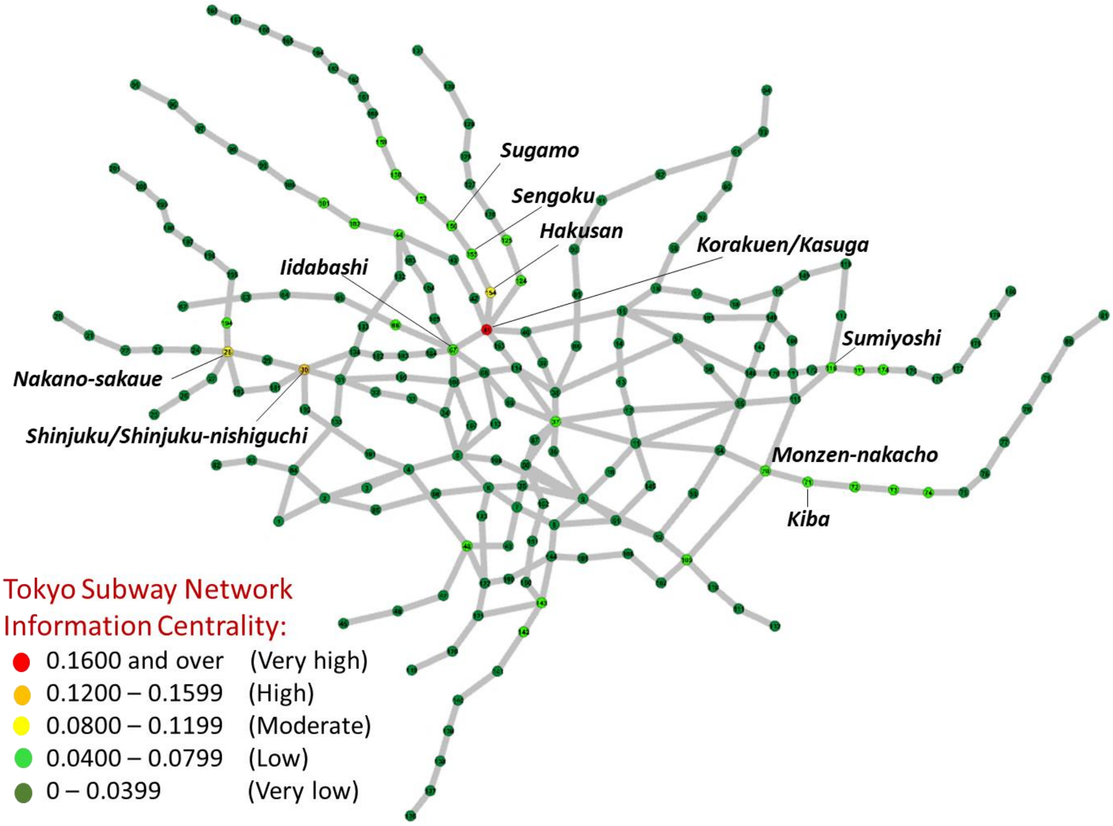


Figure 25. Information centrality result of the Tokyo subway network

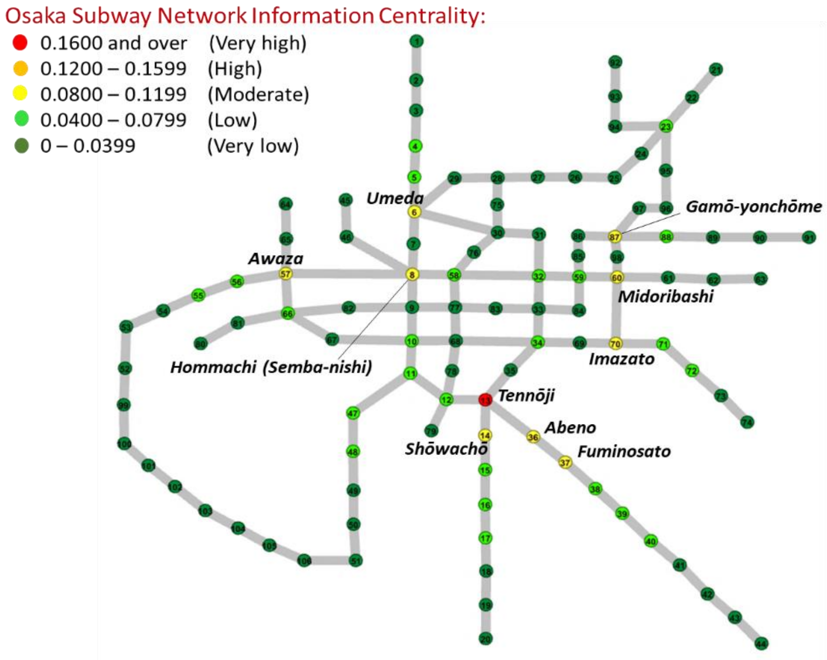


Figure 26. Information centrality result of the Osaka subway network

From Fig. 24, The very high influence station within the Kyushu railway network were Kurume, Chikugo-Funagoya, and Kumamoto stations, with an information centrality of 0.2133, 0.1994, and 0.1977, respectively. These stations are connected to several railway lines in the northern part of the network, which is dense. In addition, the very high or high centrality stations are connected or interchanged with the Shinkansen line. If these stations are disrupted, it has an extensive effect on the passenger who travels between Fukuoka and Kumamoto prefecture.

The result of the Tokyo subway network in Fig. 25 showed that Korakuen/Kasuga station had the highest value of information centrality that similar to the betweenness centrality case. The value of this station was 0.1600. In addition, the Shinjuku/Shinjuku-nishiguchi held the second most important node with 0.1258 of centrality.

In the Osaka subway network, the Tennōji station was the highest value of information centrality, as shown in Fig. 26, with a value of 0.1881. However, the other junction nodes had moderate or low values of this centrality type.

From all three-network analyses, some of the very high information centrality nodes were similar and corresponded to the betweenness centrality, which also shows a significantly high value. For example, Kurume and Chikugo-Funagoya stations from the Kyushu railway network or the Korakuen/Kasuka station of the Tokyo subway network.

5.7 Correlation Analysis

The result was analyzed correlation between pair of methods to analyze the relationship between all node-based centrality. The level of correlation can be measured by correlation coefficient $r = \mathrm{cov}_{xy}/s_x s_y$, where cov_{xy} is the covariance of variable x and y while s_x and s_y are the standard deviations of variable x and y (Howell, 2010). If the correlation coefficient is high, it means the pair of both centrality methods have a high corresponding. The correlation between the centrality criteria of all networks shows in Tables 15 - 17.

Table 15. Centrality correlation result of the Kyushu railway network

Kyushu railway network					
	Degree centrality	**Closeness centrality**	**Betweenness centrality**	**Eigenvector centrality**	**Information centrality**
Degree centrality	-	0.2200	0.4374	0.4522	0.4999
Closeness centrality	0.2200	-	0.3450	0.4199	0.2943
Betweenness centrality	0.4374	0.3450	-	0.4096	**0.8837**
Eigenvector centrality	0.4522	0.4199	0.4096	-	0.4079
Information centrality	0.4999	0.2943	**0.8837**	0.4079	-

Table 16. Centrality correlation result of the Tokyo subway network

Kyushu railway network					
	Degree centrality	**Closeness centrality**	**Betweenness centrality**	**Eigenvector centrality**	**Information centrality**
Degree centrality	**-**	0.5704	0.7040	0.6703	0.4034
Closeness centrality	0.5704	**-**	0.4418	0.5870	0.1960
Betweenness centrality	0.7040	0.4418	**-**	0.3743	**0.7546**
Eigenvector centrality	0.6703	0.5870	0.3743	**-**	0.0090
Information centrality	0.4034	0.1960	**0.7546**	0.0090	**-**

Table 17. Centrality correlation result of the Osaka subway network

Kyushu railway network					
	Degree centrality	**Closeness centrality**	**Betweenness centrality**	**Eigenvector centrality**	**Information centrality**
Degree centrality	**-**	0.6199	0.7423	0.7174	0.5944
Closeness centrality	0.6199	**-**	0.5882	**0.7970**	0.4270
Betweenness centrality	0.7423	0.5882	**-**	0.4693	0.7729
Eigenvector centrality	0.7174	**0.7970**	0.4693	**-**	0.2282
Information centrality	0.5944	0.4270	0.7729	0.2282	**-**

The result showed that information centrality and betweenness centrality had the highest corresponding coefficient in both the Kyushu network and Tokyo subway network, with the coefficient of 0.8837 and 0.7546, respectively. Although the Osaka subway case showed this pair of criteria is not the highest value of correlation coefficient, it still held the second rank of the coefficient of 0.7729. This condition corresponds with the result of Crucitti et al. (2006), who explained that information centrality is more corresponding with the betweenness centrality after analyzing the scatter plot graph.

5.8 Summary of the Centrality Analyses

The comparison can be concluded from gird testing and case studies analyses, as illustrated in Table 18. The comparison shows that degree centrality is the simplest method and uses the least processing time. However, this method is inaccurate and can point to the critical node only in the local area, and difficult to manage the priority if several of the most critical nodes have the same value. The eigenvector centrality can show the most critical node more clearly than the degree centrality, but it focuses more on the influence of connected links and the influence of neighbor nodes than the network flow. The closeness, betweenness, and information centralities are suitable for analyzing the network with the flow, such as the passenger or traffic flow. However, the closeness centrality still has a weak point in that the range of variation is narrow, and very high-level centrality nodes are located as a large cluster, which makes it hard to classify the important level. The information centrality spends the longest computing time than other methods because it requires a more complex algorithm of change of the global efficiency to calculate centrality. For this reason, the betweenness centrality was usually selected and used in the past several railway analysis research works.

Table 18. Summary of centrality's performance by each type

Centrality Type	Advantage	Disadvantage
Degree centrality	- Most simple method due to considering only the number of connected links. - Used least computing time.	- Not accurate and considered a critical node only in the local area. - If there are several same highest degree nodes, it is difficult to identify the real most influential node at the global level.
Closeness centrality	- Can analyze the critical node when considering the influence of network flow.	- The range of centrality is narrow, so it is difficult to classify the importance by each level. - The very critical nodes are located as the large cluster group.
Betweenness centrality	- Can analyze the critical node more clearly than closeness centrality when considering the influence of network flow. - Easy to apply with the passenger flow and traffic flow in the future.	- The calculating method is more complex than closeness centrality because it does not calculate the distance directly.
Eigenvector centrality	- Can identify the most influential node more clearly than the degree centrality method.	- Mainly focused on the connected links of measured nodes and influence from neighbor nodes, not the flow.
Information centrality	- Can analyze the critical node as well as the betweenness centrality when considering the influence of network flow. - Corresponding with the betweenness centrality.	- The algorithm is more complex than betweenness and closeness centrality because it needs to calculate the global efficiency both before and after node deactivation. - Consume much processing time.

References

[1] Crucitti, P., Latora, V., & Porta, S. (2006). Centrality in networks of urban streets. *Chaos*, 16. https://doi.org/10.1063/1.2150162

[2] Howell, D. C. (2010) *Statistical methods for psychology*. 7th Ed. Wadsworth. Cengage Learning, California

[3] Rodrigues, F. A. (2019) Network centrality: an introduction, *A Mathematical Modeling Approach from Nonlinear Dynamics to Complex Systems*. Vol.22, New York: Springer.

Chapter 6

Algebraic Connectivity- and Global Efficiency-based Vulnerability Analyses

6.1 Overview of Vulnerability Analysis

The vulnerability can be evaluated by measuring the change in algebraic connectivity and network efficiency after removing or attacking the link that needs to evaluate. This research mainly focuses on using the algebraic connectivity-based vulnerability to analyze the network and then compare the performance with traditional global efficiency-based vulnerability analysis. The main reason for this comparison is that very few research works to study the performance comparison between both vulnerability analyses. The advantage and disadvantages of both methods will also be considered. In addition, this research will compare both vulnerability results with the edge betweenness centrality (transform into a percentage by multiplying each link's value by 100), which has been used to analyze the critical line alongside the global efficiency-based vulnerability analysis, which helps to focus on the area that expects to have vulnerable links (Sun and Guan, 2016).

All results are expected to scope the vulnerable sections on both algebraic connectivity and global efficiency views, which help operators scope and manage the priority for preventive planning easier, especially the very vulnerable section on both criteria.

6.1.1 Equations and algorithms for vulnerability analysis

One of the methods for evaluating the network vulnerability is to simulate an attack scenario by removing some node or link and then calculate the change of indicator after attacking. This research aims to use this method to apply the algebraic connectivity and global efficiency analyses to analyze the Kyushu railway, Tokyo subway, and Osaka subway networks.

The basic equation of the algebraic connectivity-based vulnerability ($V_{A,ij}$) analysis can be illustrated by

$$V_{A,ij} = \frac{|\lambda_2(G)_{base} - \lambda_2(G)_{ij}|}{\lambda_2(G)_{base}} \times 100\%. \tag{6.1}$$

From Eq. 6.1, $V_{A,ij}$ is the algebraic connectivity-based vulnerability of the section (link) between nodes i and j, $\lambda_2(G)_{base}$ is the algebraic connectivity entire the network before attacking any link, and $\lambda_2(G)_{ij}$ is the algebraic connectivity after attacking or removing the link between nodes i and j. The example is illustrated in Fig. 27.

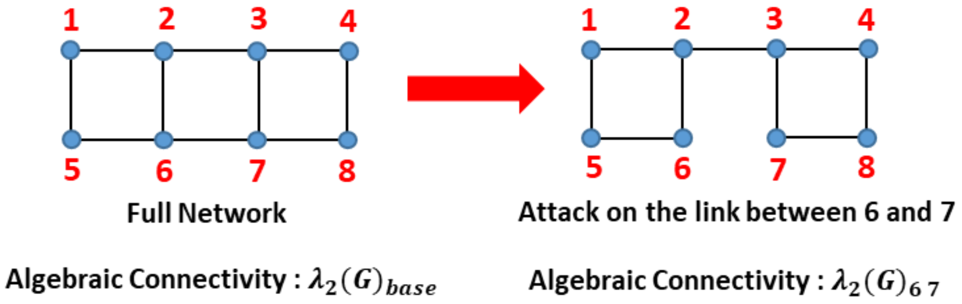


Figure 27. Example of an attack scenario on the link between nodes 6 and 7

In the case of global efficiency-based vulnerability ($V_{G,ij}$) analysis, the evaluation method is very similar, as shown in Eq. 6.2

$$V_{G,ij} = \frac{E_{glob} - E_{glob}^{(ij)}}{E_{glob}} \times 100\%, \tag{6.2}$$

where $V_{G,ij}$ is the global efficiency-based vulnerability of the link between nodes *i* and *j*, E_{glob} represents the global efficiency of the entire network before removing any link, and $E_{glob}^{(ij)}$ denotes the global efficiency after removing the link between nodes *i* and *j*.

From both types of vulnerability evaluation, the more vulnerability volume means that section is more vulnerable and sensitive if it is attacked. From Eq. 6.1 and 6.2, the algorithm can be concluded by Table 19.

Table 19. The algorithm for link-based vulnerability evaluation

```
Algorithm: Vulnerability evaluation
Require: Adjacency matrix ($(A(G))$), $a_{ij} \in \{0,1\}$, and its length ($n$)
Set up the vulnerability matrix ($(V(G))$) = ($(A(G))$) for the beginning value
Calculate $L(G)$ and $\lambda_2(G)_{base}$ or $E_{glob}$
for $i = 1{:}n$
    for $j = 1{:}n$
        if $A(i,j) = 1$ for checking adjacent element $a_{ij}$ is connected
            let $A(i,j) = 0$ and $A(j,i) = 0$ for simulating an attack on the link between
            nodes i and j when its adjacent element value becomes zero due to
            disconnection
            Calculate $\lambda_2(G)_{ij}$ or $E_{glob}^{(ij)}$ after removing the link between nodes i and j
            Calculate $V_{A,ij} = \frac{|\lambda_2(G)_{base} - \lambda_2(G)_{ij}|}{\lambda_2(G)_{base}} \times 100\%$ or $V_{G,ij} = \frac{E_{glob} - E_{glob}^{(ij)}}{E_{glob}} \times 100\%$,
            Update $V_{A,ij}$ or $V_{G,ij}$ into the element $a_{ij}$ of $V(G)$
            end
    Reset $A(G)$ as the beginning value to calculate the next measuring section
```

```
Algorithm: Vulnerability evaluation
12:     end
13: end
14: Transform $V_{A,ij}$ or $V_{G,ij}$ into string form, which shows all pair links and their
    vulnerability value.
```

This algorithm can apply both undirected and directed graphs in which the adjacency matrix is not symmetric.

6.1.2 The attack testing

From the algorithm in Table 19, The testing was divided into two main parts, similar to the centrality analysis testing. The first phase is the computing time testing with six different sizes of square grid network, 25, 100, 225, 400, 625, and 900 nodes matrix, in which each condition was testing by 30 rounds. The purpose is to compare the calculation time and characteristics between both algebraic connectivity- and global efficiency-based vulnerability methods on different sizes of dense and large networks. The second phase of testing is the case study networks by evaluating the Kyushu railway network (comprised of 671 nodes and 692 links), Tokyo subway network (composed of 201 nodes and 252 links), and Osaka subway network (composed of 106 nodes and 121 links). The main objective is to compare the results and their performances between both vulnerability methods.

All testing still was conducted by MATLAB program on the same computer with Intel(R) Core(TM) i5-4570 CPU @ 3.20GHz, and 16.0 GB of RAM due to more convenience for linking data with spreadsheet software and ease of the testing and checking the correctness of algorithm.

6.2 Result of Comparison Between Algebraic Connectivity- and Global Efficiency-based Vulnerability Analyses

6.2.1 The calculation time testing of vulnerability analyses

After testing the vulnerability algorithm on a variety of square grid networks by the MATLAB programming, the computing time result is shown in Table 20 and Fig. 28. In Fig. 28, the result was divided into two stages. When considering 25 and 100 nodes-networks, the average computing time of both vulnerability methods was not different so much. But since the 225 nodes network, the average computing time of the global efficiency-based vulnerability was increasing rapidly. The algebraic connectivity-based vulnerability used less average computing time than the global efficiency-based vulnerability analysis, especially when considering the larger network, such as the 900 nodes-network, which the latter spent about 11 times longer than the algebraic connectivity-based vulnerability. The main reason for this condition is the definition of algebraic connectivity, which considers the second smallest eigenvalue, while the global efficiency must calculate the shortest path of every pair node. In addition, edge betweenness centrality was also tested. The result showed that the edge betweenness centrality spent a little calculating time similarly to the algebraic connectivity-based vulnerability in the early stage and used the least amount of time to measure the 625- and 900-node square grid networks.

Table 20. The average processing time of the vulnerabilities and edge betweenness centrality analyses on various sizes of square grid networks

	Average calculating time (seconds)		
Grid Network Size	Algebraic connectivity-based vulnerability	Global efficiency-based vulnerability	Edge betweenness centrality
5 × 5 (25 nodes)	1.2950	1.3364	1.7782
10 × 10 (100 nodes)	1.6878	2.0392	3.0703
15 × 15 (225 nodes)	4.7121	10.5306	8.7112
20 × 20 (400 nodes)	22.8520	113.8533	25.7419
25 × 25 (625 nodes)	80.2609	673.4744	67.0400
30 × 30 (900 nodes)	258.7356	3012.7478	172.5238

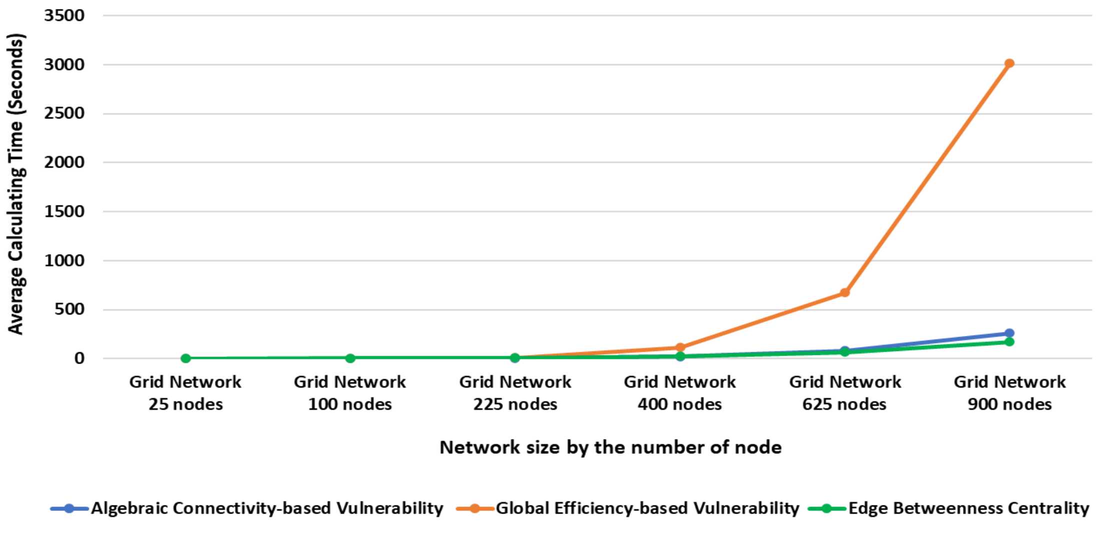


Figure 28. Calculation time results of vulnerabilities and edge betweenness centrality on the six different sizes of square grid networks

6.2.2 Comparison of the vulnerability of the square grid network

The comparison of both vulnerability analyses and edge betweenness centrality on the square grid network was also conducted on the 625 nodes network with 1,200 links, similar to the centrality's performance comparison. The result of each case was divided into five levels and independent from each other.

The result from Fig. 29 showed that both vulnerability results were not high because of the topology of the dense grid network. However, after dividing the vulnerability volume into five levels, the algebraic connectivity-based case showed the most vulnerable sections at the central outermost part of the network in a specific area. It was different from the global efficiency-based case in that the very vulnerable sections were extensively located in the central area, similarly to the edge betweenness centrality. The main reason is the algebraic connectivity-based vulnerability obtained from the second smallest eigenvalue instead of the shortest paths, which are used for the global efficiency-based vulnerability and edge betweenness centrality. Moreover, the outermost sections have fewer connected links than the inner area.

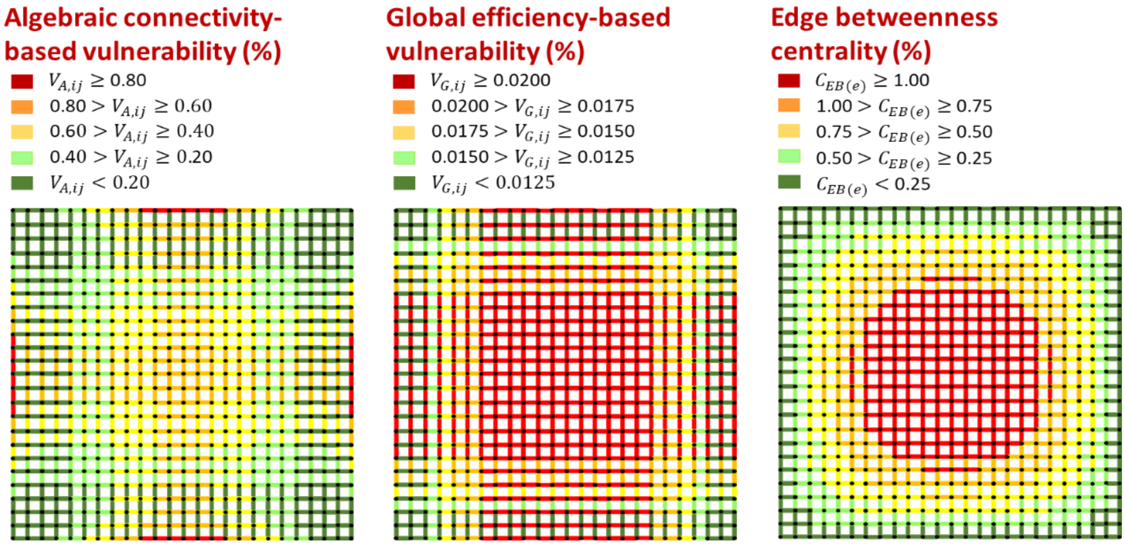


Figure 29. Comparison between the algebraic connectivity-based vulnerability (left), global efficiency-based vulnerability (center), and edge betweenness centrality (right)

6.2.3 Case studies vulnerability results

The algebraic connectivity- and global efficiency-based vulnerability analysis results of the Kyushu railway, Tokyo subway, and Osaka subway networks can conclude in Tables 21 - 23.

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network

No.	From node	To node	Line	Algebraic connectivity-based vulnerability (%)	Global efficiency-based vulnerability (%)	Edge betweenness centrality (%)
1	1	2	Kagoshima Main Line	100	0.3901	0.1490
2	2	3	Kagoshima Main Line	100	0.8017	0.2976
3	3	4	Kagoshima Main Line	100	1.6411	0.5935
4	4	5	Kagoshima Main Line	0.3625	4.5153	8.1033

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	From node	To node	Line	Algebraic connectivity-based vulnerability (%)	Global efficiency-based vulnerability (%)	Edge betweenness centrality (%)
5	5	6	Kagoshima Main Line	0.0157	1.7008	3.1272
6	6	7	Kagoshima Main Line	0.0133	1.4818	2.9913
7	7	8	Kagoshima Main Line	0.0110	1.3016	2.8558
8	8	9	Kagoshima Main Line	0.0091	1.1533	2.7212
9	9	10	Kagoshima Main Line	0.0073	1.0329	2.5875
10	10	11	Kagoshima Main Line	0.0057	0.9384	2.4548
11	11	12	Kagoshima Main Line	0.0043	0.8691	2.3233
12	12	13	Kagoshima Main Line	0.0031	0.8267	2.1970
13	13	14	Kagoshima Main Line	0.0001	0.3629	0.6529
14	14	15	Kagoshima Main Line	0.0004	0.2787	0.5150
15	15	16	Kagoshima Main Line	0.0009	0.2184	0.3802
16	16	17	Kagoshima Main Line	0.0018	0.1784	0.2482
17	17	18	Kagoshima Main Line	0.0028	0.1568	0.1201
18	18	19	Kagoshima Main Line	0.0041	0.1528	0.2117
19	19	20	Kagoshima Main Line	0.0057	0.1661	0.3137
20	20	21	Kagoshima Main Line	0.0075	0.1967	0.4428
21	21	22	Kagoshima Main Line	0.0095	0.2449	0.5744
22	22	23	Kagoshima Main Line	0.0118	0.3115	0.7091
23	23	24	Kagoshima Main Line	0.0144	0.3977	0.8450
24	24	25	Kagoshima Main Line	0.0172	0.5053	0.9813
25	25	26	Kagoshima Main Line	0.0203	0.6370	1.1181
26	26	27	Kagoshima Main Line	0.0237	0.7972	1.2554
27	27	28	Kagoshima Main Line	0.0273	0.9935	1.3931
28	28	29	Kagoshima Main Line	0.0061	0.9219	2.8104
29	29	30	Kagoshima Main Line	0.0217	1.4479	3.7418
30	30	31	Kagoshima Main Line	0.0233	1.5624	3.8762
31	31	32	Kagoshima Main Line	0.0679	3.2025	6.0142
32	32	33	Kagoshima Main Line	0.0017	0.8766	0.9684
33	33	34	Kagoshima Main Line	0.0024	0.6370	0.8239
34	34	35	Kagoshima Main Line	0.0032	0.4539	0.6846
35	35	36	Kagoshima Main Line	0.0041	0.3149	0.5467
36	36	37	Kagoshima Main Line	0.0051	0.2133	0.4115
37	37	38	Kagoshima Main Line	0.0062	0.1453	0.2793
38	38	39	Kagoshima Main Line	0.0075	0.1084	0.1491
39	39	40	Kagoshima Main Line	0.0088	0.1020	0.1547
40	40	41	Kagoshima Main Line	0.0103	0.1261	0.2392
41	41	42	Kagoshima Main Line	0.0119	0.1826	0.3568
42	42	43	Kagoshima Main Line	0.1900	0.4729	2.1123
43	43	44	Kagoshima Main Line	0.1960	0.5282	2.2400
44	44	45	Kagoshima Main Line	0.2021	0.6205	2.3694
45	45	46	Kagoshima Main Line	0.2083	0.7518	2.5006
46	46	47	Kagoshima Main Line	0.2146	0.9273	2.6337
47	47	48	Kagoshima Main Line	0.1898	0.0159	1.0247
48	48	49	Kagoshima Main Line	0.1926	0.0492	1.1056
49	49	50	Kagoshima Main Line	0.1711	0.1213	0.2263
50	50	51	Kagoshima Main Line	0.1734	0.0343	0.0782
51	51	52	Kagoshima Main Line	0.1757	0.0254	0.0713
52	52	53	Kagoshima Main Line	0.1780	0.0931	0.2194
53	53	54	Kagoshima Main Line	23.5781	2.5010	10.9713
54	54	55	Kagoshima Main Line	23.4943	2.2787	10.8607
55	55	56	Kagoshima Main Line	23.4108	2.1015	10.7502
56	56	57	Kagoshima Main Line	23.3276	1.9613	10.6396
57	57	58	Kagoshima Main Line	23.2447	1.8540	10.5291
58	58	59	Kagoshima Main Line	23.1620	1.7778	10.4185

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
59	59	60	Kagoshima Main Line	6.8385	0.3104	0.6045
60	60	61	Kagoshima Main Line	6.8586	0.2266	0.4601
61	61	62	Kagoshima Main Line	6.8788	0.1692	0.3651
62	62	63	Kagoshima Main Line	6.8991	0.1356	0.3252
63	63	64	Kagoshima Main Line	6.9197	0.1249	0.2914
64	64	65	Kagoshima Main Line	6.9403	0.1367	0.3302
65	65	66	Kagoshima Main Line	6.9612	0.1711	0.4408
66	66	67	Kagoshima Main Line	6.9823	0.2287	0.5513
67	67	68	Kagoshima Main Line	7.0037	0.3109	0.6619
68	68	69	Kagoshima Main Line	7.0253	0.4196	0.7724
69	69	70	Kagoshima Main Line	7.0472	0.5578	0.8830
70	70	71	Kagoshima Main Line	7.0694	0.7307	1.0104
71	71	72	Kagoshima Main Line	7.0919	0.9469	1.1548
72	72	73	Kagoshima Main Line	0.0516	0.5121	1.6308
73	73	74	Kagoshima Main Line	0.0494	0.3598	1.4873
74	74	75	Kagoshima Main Line	0.0473	0.2607	1.3438
75	75	76	Kagoshima Main Line	0.0453	0.2085	1.2004
76	76	77	Kagoshima Main Line	0.0286	0.0541	0.2354
77	77	78	Kagoshima Main Line	0.0271	0.0524	0.2719
78	78	79	Kagoshima Main Line	0.0255	0.0934	0.4154
79	79	80	Kagoshima Main Line	0.0240	0.1785	0.5589
80	80	81	Kagoshima Main Line	0.0226	0.3136	0.7023
81	81	82	Kagoshima Main Line	2.0623	2.0748	3.3654
82	83	84	Kagoshima Main Line	0.0210	0.4692	0.8507
83	84	85	Kagoshima Main Line	0.0186	0.3236	0.7043
84	85	86	Kagoshima Main Line	0.0165	0.2124	0.5580
85	86	87	Kagoshima Main Line	0.0144	0.1305	0.4116
86	87	88	Kagoshima Main Line	0.0126	0.0754	0.2653
87	88	89	Kagoshima Main Line	0.0108	0.0453	0.1189
88	89	90	Kagoshima Main Line	0.0092	0.0396	0.0368
89	90	91	Kagoshima Main Line	0.0077	0.0583	0.1832
90	91	92	Kagoshima Main Line	0.0063	0.1018	0.3295
91	92	93	Kagoshima Main Line	0.0051	0.1718	0.4759
92	93	94	Kagoshima Main Line	0.0040	0.2708	0.6223
93	94	95	Kagoshima Main Line	0.0030	0.4042	0.7686
94	95	96	Kagoshima Main Line	3.6640	3.2144	10.3898
95	97	3	Sanyo Main Line	100	0.4255	0.1490
96	5	98	Nippō Main Line	0.3812	3.2733	5.7176
97	98	99	Nippō Main Line	0.3674	3.0886	5.6102
98	99	100	Nippō Main Line	1.6391	3.0148	4.1143
99	100	101	Nippō Main Line	1.5336	2.7712	3.9971
100	101	102	Nippō Main Line	1.4349	2.5529	3.8808
101	102	103	Nippō Main Line	1.3424	2.3551	3.7653
102	103	104	Nippō Main Line	1.2557	2.1747	3.6508
103	104	105	Nippō Main Line	1.1741	2.0094	3.5371
104	105	106	Nippō Main Line	1.0974	1.8575	3.4243
105	106	107	Nippō Main Line	1.0250	1.7177	3.3125
106	107	108	Nippō Main Line	0.9567	1.5889	3.2015
107	108	109	Nippō Main Line	0.8922	1.4701	3.0914
108	109	110	Nippō Main Line	0.8311	1.3606	2.9821
109	110	111	Nippō Main Line	0.7733	1.2597	2.8738
110	111	112	Nippō Main Line	0.7185	1.1670	2.7664
111	112	113	Nippō Main Line	0.6665	1.0820	2.6598
112	113	114	Nippō Main Line	0.6172	1.0043	2.5542

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
113	114	115	Nippō Main Line	0.5703	0.9335	2.4494
114	115	116	Nippō Main Line	0.5258	0.8693	2.3455
115	116	117	Nippō Main Line	0.4835	0.8116	2.2426
116	117	118	Nippō Main Line	0.4433	0.7601	2.1405
117	118	119	Nippō Main Line	0.4051	0.7146	2.0392
118	119	120	Nippō Main Line	0.3689	0.6751	1.9452
119	120	121	Nippō Main Line	0.3344	0.6413	1.8582
120	121	122	Nippō Main Line	0.3018	0.6133	1.7732
121	122	123	Nippō Main Line	0.2708	0.5909	1.6896
122	123	124	Nippō Main Line	0.2415	0.5742	1.6073
123	124	125	Nippō Main Line	0.2139	0.5632	1.5263
124	125	126	Nippō Main Line	0.1879	0.5580	1.4467
125	126	127	Nippō Main Line	0.1634	0.5586	1.3684
126	127	128	Nippō Main Line	0.1405	0.5651	1.2914
127	128	129	Nippō Main Line	0.1192	0.5779	1.2173
128	129	130	Nippō Main Line	0.0995	0.5973	1.1463
129	130	131	Nippō Main Line	0.0814	0.6236	1.1319
130	131	132	Nippō Main Line	0.0649	0.6576	1.1207
131	132	133	Nippō Main Line	0.0501	0.7002	1.1534
132	133	134	Nippō Main Line	0.0370	0.7533	1.2317
133	134	135	Nippō Main Line	0.0257	0.8202	1.3123
134	135	136	Nippō Main Line	17.0174	1.4648	3.3439
135	136	137	Nippō Main Line	15.9505	1.3664	3.2362
136	137	138	Nippō Main Line	14.8942	1.2806	3.1290
137	138	139	Nippō Main Line	13.8516	1.2037	3.0222
138	139	140	Nippō Main Line	12.8262	1.1341	2.9159
139	140	141	Nippō Main Line	11.8219	1.0707	2.8100
140	141	142	Nippō Main Line	10.8428	1.0127	2.7046
141	142	143	Nippō Main Line	9.8933	0.9596	2.5996
142	143	144	Nippō Main Line	8.9782	0.9110	2.4986
143	144	145	Nippō Main Line	8.1022	0.8667	2.3981
144	145	146	Nippō Main Line	7.2698	0.8263	2.2980
145	146	147	Nippō Main Line	6.4852	0.7897	2.1983
146	147	148	Nippō Main Line	5.7520	0.7567	2.0991
147	148	149	Nippō Main Line	5.0729	0.7272	2.0004
148	149	150	Nippō Main Line	4.4492	0.7010	1.9021
149	150	151	Nippō Main Line	3.8814	0.6781	1.8042
150	151	152	Nippō Main Line	3.3683	0.6583	1.7248
151	152	153	Nippō Main Line	2.9079	0.6416	1.6483
152	153	154	Nippō Main Line	2.4973	0.6280	1.5722
153	154	155	Nippō Main Line	2.1329	0.6173	1.4966
154	155	156	Nippō Main Line	1.8109	0.6097	1.4258
155	156	157	Nippō Main Line	1.5273	0.6049	1.3658
156	157	158	Nippō Main Line	1.2784	0.6031	1.3442
157	158	159	Nippō Main Line	1.0604	0.6042	1.4084
158	159	160	Nippō Main Line	0.8703	0.6082	1.4803
159	160	161	Nippō Main Line	0.7049	0.6150	1.5587
160	161	162	Nippō Main Line	0.5618	0.6248	1.6416
161	162	163	Nippō Main Line	0.4387	0.6376	1.7282
162	163	164	Nippō Main Line	0.3337	0.6533	1.8166
163	164	165	Nippō Main Line	0.2452	0.6720	1.9067
164	165	166	Nippō Main Line	0.1719	0.6937	1.9987
165	166	167	Nippō Main Line	0.1126	0.7185	2.0921
166	167	168	Nippō Main Line	0.0664	0.7466	2.1869

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
167	168	169	Nippō Main Line	0.0327	0.7778	2.2830
168	169	170	Nippō Main Line	0.0109	0.8124	2.3842
169	170	171	Nippō Main Line	0.0008	0.8505	2.4863
170	171	172	Nippō Main Line	0.0022	0.8921	2.5893
171	172	173	Nippō Main Line	0.0151	0.9375	2.6931
172	173	174	Nippō Main Line	0.0396	0.9867	2.7979
173	174	175	Nippō Main Line	0.0762	1.0402	2.9036
174	175	176	Nippō Main Line	0.1253	1.0981	3.0101
175	176	177	Nippō Main Line	0.1878	1.1608	3.1176
176	177	178	Nippō Main Line	0.2646	1.2289	3.2259
177	178	179	Nippō Main Line	0.3570	1.3030	3.3351
178	179	180	Nippō Main Line	0.4664	1.3843	3.4452
179	180	181	Nippō Main Line	0.5949	1.4751	3.5562
180	181	182	Nippō Main Line	26.6335	2.7066	6.6202
181	182	183	Nippō Main Line	27.6350	2.7423	6.7206
182	183	184	Nippō Main Line	28.6159	2.7882	6.8218
183	184	185	Nippō Main Line	29.5767	2.8423	6.9239
184	185	186	Nippō Main Line	30.5179	2.9042	7.0268
185	186	187	Nippō Main Line	31.4400	2.9737	7.1307
186	187	188	Nippō Main Line	32.3437	3.0510	7.2355
187	188	189	Nippō Main Line	33.2294	3.1370	7.3411
188	189	190	Nippō Main Line	34.0978	3.2335	7.4477
189	190	191	Nippō Main Line	1.1804	1.6643	7.8655
190	191	192	Nippō Main Line	1.1804	1.6794	7.9645
191	192	193	Nippō Main Line	1.2395	1.7084	8.0697
192	193	194	Nippō Main Line	1.3637	1.7499	8.1754
193	194	195	Nippō Main Line	1.4290	1.8034	8.2815
194	195	196	Nippō Main Line	1.4967	1.8690	8.3880
195	196	197	Nippō Main Line	1.5670	1.9473	8.4950
196	197	198	Nippō Main Line	1.6399	2.0394	8.6025
197	198	199	Nippō Main Line	1.7156	2.1472	8.7103
198	199	200	Nippō Main Line	2.8609	2.4817	9.7032
199	200	201	Nippō Main Line	2.9654	2.5287	9.7999
200	201	202	Nippō Main Line	3.0730	2.5940	9.8971
201	202	203	Nippō Main Line	3.1839	2.6770	9.9948
202	203	204	Nippō Main Line	3.2982	2.7783	10.0929
203	204	205	Nippō Main Line	3.4161	2.8993	10.1914
204	205	96	Nippō Main Line	3.5380	3.0430	10.2904
205	206	207	Chikuhō Main Line	100	0.2582	0.1490
206	207	208	Chikuhō Main Line	100	0.5157	0.2976
207	208	209	Chikuhō Main Line	100	0.7794	0.4458
208	209	210	Chikuhō Main Line	100	1.0533	0.5935
209	210	13	Chikuhō Main Line	100	1.3426	0.7407
210	13	211	Chikuhō Main Line	0.0000	0.5048	0.9266
211	211	212	Chikuhō Main Line	0.0000	0.4103	0.7954
212	212	213	Chikuhō Main Line	0.0003	0.3383	0.6659
213	213	214	Chikuhō Main Line	0.0008	0.2851	0.5433
214	214	215	Chikuhō Main Line	0.0016	0.2490	0.4297
215	215	216	Chikuhō Main Line	0.0026	0.2288	0.3325
216	216	217	Chikuhō Main Line	0.0038	0.2242	0.2393
217	217	218	Chikuhō Main Line	0.0052	0.2351	0.3374
218	218	219	Chikuhō Main Line	0.0069	0.2616	0.4469
219	219	220	Chikuhō Main Line	0.0089	0.3044	0.5793
220	220	221	Chikuhō Main Line	0.0111	0.3647	0.7136

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
221	221	222	Chikuhō Main Line	0.0136	0.4450	0.8484
222	222	223	Chikuhō Main Line	0.0759	0.5567	1.2761
223	223	224	Chikuhō Main Line	0.0808	0.5937	1.3592
224	224	225	Chikuhō Main Line	0.0859	0.6565	1.4587
225	225	226	Chikuhō Main Line	0.1036	0.3866	1.4019
226	226	227	Chikuhō Main Line	0.1086	0.3867	1.5074
227	227	228	Chikuhō Main Line	0.1137	0.4170	1.6218
228	228	42	Chikuhō Main Line	0.1189	0.4775	1.7422
229	31	229	Sasaguri Line	0.0095	0.9382	2.1342
230	229	230	Sasaguri Line	0.0084	0.7915	1.9992
231	230	231	Sasaguri Line	0.0073	0.7058	1.8684
232	231	232	Sasaguri Line	0.0000	0.5841	0.9478
233	232	233	Sasaguri Line	0.0000	0.4549	0.8165
234	233	234	Sasaguri Line	0.0001	0.3613	0.6871
235	234	235	Sasaguri Line	0.0003	0.2986	0.5594
236	235	236	Sasaguri Line	0.0007	0.2644	0.4352
237	236	237	Sasaguri Line	0.0012	0.2581	0.3261
238	237	225	Sasaguri Line	0.0018	0.2807	0.4175
239	238	239	Kashii Line	100	0.3107	0.1490
240	239	240	Kashii Line	100	0.6278	0.2976
241	240	241	Kashii Line	100	0.9605	0.4458
242	241	242	Kashii Line	100	1.3170	0.5935
243	242	28	Kashii Line	0.0007	0.4747	1.1829
244	28	243	Kashii Line	0.0005	0.2137	0.3952
245	243	244	Kashii Line	0.0009	0.1316	0.2621
246	244	245	Kashii Line	0.0013	0.1009	0.1324
247	245	246	Kashii Line	0.0019	0.1172	0.2658
248	246	231	Kashii Line	0.0025	0.1819	0.4002
249	231	247	Kashii Line	100	1.6408	0.7407
250	247	248	Kashii Line	100	1.2765	0.5935
251	248	249	Kashii Line	100	0.9369	0.4458
252	249	250	Kashii Line	100	0.6150	0.2976
253	250	251	Kashii Line	100	0.3053	0.1490
254	99	252	Hitahikosan Line	0.0005	1.0625	1.8664
255	252	253	Hitahikosan Line	0.0002	0.8888	1.7311
256	253	254	Hitahikosan Line	0.0000	0.7448	1.5970
257	254	255	Hitahikosan Line	0.0001	0.6260	1.4738
258	255	256	Hitahikosan Line	0.0004	0.5296	1.3523
259	256	257	Hitahikosan Line	0.0009	0.4540	1.2328
260	257	258	Hitahikosan Line	0.0016	0.3981	1.1224
261	258	259	Hitahikosan Line	0.0026	0.3615	1.0218
262	259	260	Hitahikosan Line	0.0038	0.3443	0.9244
263	260	261	Hitahikosan Line	0.0052	0.3478	0.8386
264	261	262	Hitahikosan Line	0.0080	0.4777	0.9345
265	262	263	Hitahikosan Line	0.0058	0.4119	0.8255
266	263	264	Hitahikosan Line	0.0040	0.3633	0.7179
267	264	265	Hitahikosan Line	0.0025	0.3297	0.6115
268	265	266	Hitahikosan Line	0.0014	0.3101	0.6393
269	266	267	Hitahikosan Line	0.0006	0.3038	0.7078
270	267	268	Hitahikosan Line	0.0001	0.3105	0.7777
271	268	269	Hitahikosan Line	0.0000	0.3302	0.8756
272	269	270	Hitahikosan Line	0.0002	0.3629	0.9799
273	270	271	Hitahikosan Line	0.0008	0.4090	1.0909
274	271	272	Hitahikosan Line	0.0017	0.4691	1.2074

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
275	272	273	Hitahikosan Line	0.0030	0.5444	1.3369
276	273	274	Hitahikosan Line	0.0046	0.6368	1.4677
277	222	275	Gotōji Line	0.0358	0.4252	0.8133
278	275	276	Gotōji Line	0.0322	0.3787	0.7209
279	276	277	Gotōji Line	0.0287	0.3543	0.6326
280	277	278	Gotōji Line	0.0254	0.3507	0.5600
281	278	261	Gotōji Line	0.0223	0.3684	0.5632
282	32	279	Chikuhi Line	0.1115	1.7853	3.6630
283	279	280	Chikuhi Line	0.1064	1.5925	3.5273
284	280	281	Chikuhi Line	0.7380	2.2742	2.8806
285	281	282	Chikuhi Line	0.6717	2.0173	2.7427
286	282	283	Chikuhi Line	0.6115	1.7918	2.6052
287	283	284	Chikuhi Line	0.5567	1.5913	2.4677
288	284	285	Chikuhi Line	0.5066	1.4120	2.3347
289	285	286	Chikuhi Line	0.4607	1.2509	2.2017
290	286	287	Chikuhi Line	0.4187	1.1060	2.0687
291	287	288	Chikuhi Line	0.3800	0.9756	1.9357
292	288	289	Chikuhi Line	0.3444	0.8585	1.8026
293	289	290	Chikuhi Line	0.3115	0.7536	1.6696
294	290	291	Chikuhi Line	0.2812	0.6600	1.5366
295	291	292	Chikuhi Line	0.2532	0.5771	1.4036
296	292	293	Chikuhi Line	0.2272	0.5043	1.2706
297	293	294	Chikuhi Line	0.2033	0.4411	1.1620
298	294	295	Chikuhi Line	0.1811	0.3871	1.0535
299	295	296	Chikuhi Line	0.1606	0.3420	0.9449
300	296	297	Chikuhi Line	0.1416	0.3057	0.8364
301	297	298	Chikuhi Line	0.1241	0.2779	0.7278
302	298	299	Chikuhi Line	0.1080	0.2585	0.6193
303	299	300	Chikuhi Line	0.0932	0.2474	0.5107
304	300	301	Chikuhi Line	0.0796	0.2447	0.4022
305	301	302	Chikuhi Line	0.0672	0.2504	0.2937
306	302	303	Chikuhi Line	0.0559	0.2648	0.3648
307	303	304	Chikuhi Line	0.0457	0.2881	0.4404
308	304	305	Chikuhi Line	0.0366	0.3207	0.5758
309	305	306	Chikuhi Line	0.0285	0.3634	0.7118
310	306	307	Chikuhi Line	0.0214	0.4177	0.8497
311	307	308	Chikuhi Line	100	0.2520	0.1490
312	307	309	Chikuhi Line	0.0102	0.5330	1.1250
313	309	310	Chikuhi Line	0.0062	0.6011	1.2625
314	310	311	Chikuhi Line	100	1.9922	1.4703
315	311	312	Chikuhi Line	100	1.7699	1.3253
316	312	313	Chikuhi Line	100	1.5571	1.1798
317	313	314	Chikuhi Line	100	1.3514	1.0339
318	314	315	Chikuhi Line	100	1.1513	0.8875
319	315	316	Chikuhi Line	100	0.9557	0.7407
320	316	317	Chikuhi Line	100	0.7635	0.5935
321	317	318	Chikuhi Line	100	0.5738	0.4458
322	318	319	Chikuhi Line	100	0.3853	0.2976
323	319	320	Chikuhi Line	100	0.1959	0.1490
324	47	321	Nagasaki Main Line	0.0260	0.2271	1.7996
325	321	322	Nagasaki Main Line	39.0297	6.9241	11.4465
326	322	323	Nagasaki Main Line	38.5105	6.6353	11.3379
327	323	324	Nagasaki Main Line	37.9841	6.3907	11.2294
328	324	325	Nagasaki Main Line	37.4500	6.1805	11.1208

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	From node	To node	Line	Algebraic connectivity-based vulnerability (%)	Global efficiency-based vulnerability (%)	Edge betweenness centrality (%)
329	325	326	Nagasaki Main Line	36.9080	5.9989	11.0123
330	326	327	Nagasaki Main Line	36.3574	5.8422	10.9037
331	327	328	Nagasaki Main Line	35.7980	5.7084	10.7952
332	328	329	Nagasaki Main Line	35.2293	5.5963	10.6866
333	329	330	Nagasaki Main Line	34.6506	5.5065	10.5781
334	330	331	Nagasaki Main Line	100	9.4531	7.5361
335	331	332	Nagasaki Main Line	100	9.2360	7.4111
336	332	333	Nagasaki Main Line	0.5023	1.6042	4.4476
337	333	334	Nagasaki Main Line	0.4509	1.4497	4.3128
338	334	335	Nagasaki Main Line	0.4045	1.3126	4.1780
339	335	336	Nagasaki Main Line	0.3626	1.1901	4.0432
340	336	337	Nagasaki Main Line	0.3245	1.0806	3.9084
341	337	338	Nagasaki Main Line	0.2900	0.9830	3.7736
342	338	339	Nagasaki Main Line	0.2586	0.8963	3.6388
343	339	340	Nagasaki Main Line	0.2299	0.8201	3.5040
344	340	341	Nagasaki Main Line	0.2037	0.7539	3.3692
345	341	342	Nagasaki Main Line	0.1798	0.6974	3.2358
346	342	343	Nagasaki Main Line	0.1580	0.6506	3.1023
347	343	344	Nagasaki Main Line	0.1380	0.6136	2.9688
348	344	345	Nagasaki Main Line	0.1198	0.5867	2.8354
349	345	346	Nagasaki Main Line	0.1033	0.5709	2.7019
350	346	347	Nagasaki Main Line	0.0882	0.5679	2.5685
351	347	348	Nagasaki Main Line	100	2.3923	2.0460
352	348	349	Nagasaki Main Line	100	2.2241	1.9027
353	349	350	Nagasaki Main Line	0.0054	0.2049	0.9559
354	350	351	Nagasaki Main Line	0.0037	0.1521	0.8098
355	351	352	Nagasaki Main Line	0.0024	0.1138	0.6636
356	352	353	Nagasaki Main Line	0.0014	0.0887	0.5175
357	353	354	Nagasaki Main Line	100	0.1995	0.1490
358	349	355	Nagasaki Main Line (old)	0.0048	0.1866	0.8093
359	355	356	Nagasaki Main Line (old)	0.0033	0.1318	0.6627
360	356	357	Nagasaki Main Line (old)	0.0020	0.0910	0.5164
361	357	358	Nagasaki Main Line (old)	0.0011	0.0620	0.3702
362	358	359	Nagasaki Main Line (old)	0.0005	0.0437	0.2241
363	359	360	Nagasaki Main Line (old)	0.0001	0.0358	0.0780
364	360	361	Nagasaki Main Line (old)	0.0000	0.0383	0.0782
365	361	353	Nagasaki Main Line (old)	0.0002	0.0521	0.2248
366	330	362	Karatsu Line	0.1633	1.9219	3.8143
367	362	363	Karatsu Line	0.1402	1.7667	3.6813
368	363	364	Karatsu Line	0.1195	1.6305	3.5482
369	364	365	Karatsu Line	0.1009	1.5106	3.4152
370	365	366	Karatsu Line	0.0843	1.4054	3.2822
371	366	367	Karatsu Line	0.0695	1.3138	3.1492
372	367	368	Karatsu Line	0.0564	1.2354	3.0162
373	368	369	Karatsu Line	0.0448	1.1704	2.8832
374	369	310	Karatsu Line	0.0347	1.1200	2.7501
375	332	370	Sasebo Line	0.4198	1.5054	2.9355
376	370	371	Sasebo Line	0.3765	1.3500	2.7958
377	371	372	Sasebo Line	0.3374	1.2121	2.6561
378	372	373	Sasebo Line	0.3019	1.0889	2.5164
379	373	374	Sasebo Line	0.2696	0.9788	2.3829
380	374	375	Sasebo Line	0.2402	0.8807	2.2495
381	375	376	Sasebo Line	0.2134	0.7939	2.1160
382	376	377	Sasebo Line	0.1890	0.7181	1.9826

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
383	377	378	Sasebo Line	0.1666	0.6533	1.8491
384	378	379	Sasebo Line	0.1462	0.6005	1.7156
385	379	380	Sasebo Line	100	0.6671	0.4458
386	380	381	Sasebo Line	100	0.4428	0.2976
387	381	382	Sasebo Line	100	0.2228	0.1490
388	379	383	Ōmura Line	0.0811	0.3976	1.1791
389	383	384	Ōmura Line	0.0684	0.3408	1.0443
390	384	385	Ōmura Line	0.0571	0.2951	0.9095
391	385	386	Ōmura Line	0.0468	0.2592	0.7747
392	386	387	Ōmura Line	0.0377	0.2324	0.6399
393	387	388	Ōmura Line	0.0297	0.2142	0.5051
394	388	389	Ōmura Line	0.0227	0.2047	0.3704
395	389	390	Ōmura Line	0.0167	0.2040	0.2356
396	390	391	Ōmura Line	0.0116	0.2122	0.1008
397	391	392	Ōmura Line	0.0075	0.2299	0.2405
398	392	393	Ōmura Line	0.0042	0.2583	0.3801
399	393	347	Ōmura Line	0.0019	0.2993	0.5198
400	49	394	Kyūdai Main Line	0.9391	3.0496	6.4064
401	394	395	Kyūdai Main Line	0.9143	2.7642	6.2905
402	395	396	Kyūdai Main Line	0.8900	2.5232	6.1759
403	396	397	Kyūdai Main Line	0.8664	2.3164	6.0627
404	397	398	Kyūdai Main Line	0.8432	2.1379	5.9513
405	398	399	Kyūdai Main Line	0.8206	1.9839	5.8416
406	399	400	Kyūdai Main Line	0.7984	1.8517	5.7337
407	400	401	Kyūdai Main Line	0.7766	1.7395	5.6276
408	401	402	Kyūdai Main Line	0.7553	1.6462	5.5258
409	402	403	Kyūdai Main Line	0.7343	1.5715	5.4261
410	403	274	Kyūdai Main Line	0.7137	1.5159	5.3287
411	274	404	Kyūdai Main Line	1.5342	1.8465	4.7893
412	404	405	Kyūdai Main Line	1.4590	1.7053	4.6788
413	405	406	Kyūdai Main Line	1.3873	1.5780	4.5691
414	406	407	Kyūdai Main Line	1.3187	1.4624	4.4604
415	407	408	Kyūdai Main Line	1.2530	1.3572	4.3525
416	408	409	Kyūdai Main Line	1.1900	1.2613	4.2455
417	409	410	Kyūdai Main Line	1.1296	1.1742	4.1394
418	410	411	Kyūdai Main Line	1.0716	1.0952	4.0342
419	411	412	Kyūdai Main Line	1.0158	1.0240	3.9298
420	412	413	Kyūdai Main Line	0.9622	0.9602	3.8264
421	413	414	Kyūdai Main Line	0.9105	0.9034	3.7239
422	414	415	Kyūdai Main Line	0.8607	0.8536	3.6222
423	415	416	Kyūdai Main Line	0.8126	0.8105	3.5215
424	416	417	Kyūdai Main Line	0.7663	0.7740	3.4340
425	417	418	Kyūdai Main Line	0.7215	0.7441	3.3475
426	418	419	Kyūdai Main Line	0.6782	0.7206	3.2619
427	419	420	Kyūdai Main Line	0.6364	0.7037	3.1771
428	420	421	Kyūdai Main Line	0.5960	0.6933	3.0933
429	421	422	Kyūdai Main Line	0.5570	0.6898	3.0103
430	422	423	Kyūdai Main Line	0.5192	0.6933	2.9282
431	423	424	Kyūdai Main Line	0.4827	0.7043	2.8470
432	424	425	Kyūdai Main Line	0.4473	0.7234	2.7667
433	425	426	Kyūdai Main Line	0.4132	0.7515	2.6916
434	426	427	Kyūdai Main Line	0.3802	0.7902	2.6457
435	427	135	Kyūdai Main Line	0.3484	0.8432	2.6377
436	72	428	Hōhi Main Line	2.6972	3.5790	4.7137

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	From node	To node	Line	Algebraic connectivity-based vulnerability (%)	Global efficiency-based vulnerability (%)	Edge betweenness centrality (%)
437	428	429	Hōhi Main Line	2.5736	3.2763	4.5933
438	429	430	Hōhi Main Line	2.4572	3.0107	4.4739
439	430	431	Hōhi Main Line	2.3473	2.7731	4.3553
440	431	432	Hōhi Main Line	2.2433	2.5583	4.2376
441	432	433	Hōhi Main Line	2.1448	2.3626	4.1209
442	433	434	Hōhi Main Line	2.0512	2.1834	4.0050
443	434	435	Hōhi Main Line	1.9623	2.0188	3.8900
444	435	436	Hōhi Main Line	1.8776	1.8673	3.7759
445	436	437	Hōhi Main Line	1.7967	1.7277	3.6626
446	437	438	Hōhi Main Line	1.7195	1.5989	3.5503
447	438	439	Hōhi Main Line	1.6456	1.4801	3.4389
448	439	440	Hōhi Main Line	1.5747	1.3706	3.3283
449	440	441	Hōhi Main Line	1.5068	1.2698	3.2187
450	441	442	Hōhi Main Line	1.4415	1.1772	3.1099
451	442	443	Hōhi Main Line	1.3786	1.0924	3.0020
452	443	444	Hōhi Main Line	1.3181	1.0150	2.8950
453	444	445	Hōhi Main Line	1.2597	0.9447	2.7889
454	445	446	Hōhi Main Line	1.2033	0.8812	2.6837
455	446	447	Hōhi Main Line	1.1489	0.8243	2.5794
456	447	448	Hōhi Main Line	1.0962	0.7738	2.4760
457	448	449	Hōhi Main Line	1.0451	0.7296	2.3734
458	449	450	Hōhi Main Line	0.9956	0.6916	2.2718
459	450	451	Hōhi Main Line	0.9477	0.6596	2.1730
460	451	452	Hōhi Main Line	0.9011	0.6336	2.0776
461	452	453	Hōhi Main Line	0.8558	0.6137	1.9835
462	453	454	Hōhi Main Line	0.8118	0.5998	1.9032
463	454	455	Hōhi Main Line	0.7689	0.5921	1.8247
464	455	456	Hōhi Main Line	0.7272	0.5906	1.7479
465	456	457	Hōhi Main Line	0.6866	0.5956	1.6730
466	457	458	Hōhi Main Line	0.6470	0.6074	1.5998
467	458	459	Hōhi Main Line	0.6084	0.6264	1.5331
468	459	460	Hōhi Main Line	0.5708	0.6532	1.4681
469	460	461	Hōhi Main Line	0.5341	0.6889	1.4143
470	461	462	Hōhi Main Line	0.4983	0.7351	1.4064
471	462	135	Hōhi Main Line	0.4634	0.7954	1.4628
472	76	463	Misumi Line	100	2.1924	1.1798
473	463	464	Misumi Line	100	1.8794	1.0339
474	464	465	Misumi Line	100	1.5833	0.8875
475	465	466	Misumi Line	100	1.3006	0.7407
476	466	467	Misumi Line	100	1.0286	0.5935
477	467	468	Misumi Line	100	0.7652	0.4458
478	468	469	Misumi Line	100	0.5084	0.2976
479	469	470	Misumi Line	100	0.2553	0.1490
480	82	471	Hisatsu Line	1.9853	1.8158	3.2293
481	471	472	Hisatsu Line	1.9115	1.5926	3.0936
482	472	473	Hisatsu Line	1.8407	1.3972	2.9584
483	473	474	Hisatsu Line	1.7727	1.2249	2.8236
484	474	475	Hisatsu Line	1.7073	1.0726	2.6892
485	475	476	Hisatsu Line	1.6443	0.9380	2.5553
486	476	477	Hisatsu Line	1.5836	0.8194	2.4359
487	477	478	Hisatsu Line	1.5250	0.7155	2.3330
488	478	479	Hisatsu Line	1.4683	0.6254	2.2322
489	479	480	Hisatsu Line	1.4135	0.5483	2.1319
490	480	481	Hisatsu Line	1.3605	0.4835	2.0320

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
491	481	482	Hisatsu Line	1.3090	0.4306	1.9326
492	482	483	Hisatsu Line	1.2591	0.3894	1.8336
493	483	484	Hisatsu Line	1.2105	0.3597	1.7351
494	484	485	Hisatsu Line	1.1633	0.3416	1.6370
495	485	486	Hisatsu Line	1.1174	0.3354	1.5482
496	486	487	Hisatsu Line	1.0727	0.3419	1.4704
497	487	488	Hisatsu Line	1.0291	0.3626	1.3941
498	488	489	Hisatsu Line	0.0039	0.2412	0.2834
499	489	490	Hisatsu Line	0.0022	0.2295	0.2921
500	490	491	Hisatsu Line	0.0009	0.2333	0.4260
501	491	492	Hisatsu Line	0.0002	0.2510	0.5604
502	492	493	Hisatsu Line	0.0000	0.2822	0.6951
503	493	494	Hisatsu Line	0.0003	0.3270	0.8304
504	494	495	Hisatsu Line	0.0010	0.3859	0.9661
505	495	496	Hisatsu Line	0.0023	0.4598	1.1022
506	496	199	Hisatsu Line	0.0040	0.5509	1.2388
507	488	497	Kitto Line	0.6653	0.4347	1.2970
508	497	498	Kitto Line	0.6261	0.3680	1.1891
509	498	499	Kitto Line	0.5886	0.3140	1.0817
510	499	500	Kitto Line	0.5527	0.2709	0.9747
511	500	501	Kitto Line	0.5184	0.2376	0.8681
512	501	502	Kitto Line	0.4854	0.2137	0.7620
513	502	503	Kitto Line	0.4539	0.1988	0.6564
514	503	504	Kitto Line	0.4236	0.1927	0.5512
515	504	505	Kitto Line	0.3946	0.1953	0.4623
516	505	506	Kitto Line	0.3667	0.2066	0.4918
517	506	507	Kitto Line	0.3400	0.2267	0.6257
518	507	508	Kitto Line	0.3144	0.2560	0.7600
519	508	509	Kitto Line	0.2898	0.2947	0.8948
520	509	510	Kitto Line	0.2662	0.3435	1.0301
521	510	511	Kitto Line	0.2436	0.4035	1.1657
522	511	190	Kitto Line	0.2219	0.4765	1.3019
523	181	512	Nichinan Line	100	3.7836	4.0047
524	512	513	Nichinan Line	100	3.4570	3.7302
525	513	514	Nichinan Line	100	3.2919	3.5923
526	514	515	Nichinan Line	100	3.1341	3.4540
527	515	516	Nichinan Line	100	2.9816	3.3152
528	516	517	Nichinan Line	100	2.8335	3.1759
529	517	518	Nichinan Line	100	2.6891	3.0362
530	518	519	Nichinan Line	100	2.5479	2.8961
531	519	520	Nichinan Line	100	2.4095	2.7555
532	520	521	Nichinan Line	100	2.2736	2.6145
533	521	522	Nichinan Line	100	2.1400	2.4730
534	522	523	Nichinan Line	100	2.0085	2.3311
535	523	524	Nichinan Line	100	1.8787	2.1888
536	524	525	Nichinan Line	100	1.7506	2.0460
537	525	526	Nichinan Line	100	1.6239	1.9027
538	526	527	Nichinan Line	100	1.4985	1.7590
539	527	528	Nichinan Line	100	1.3742	1.6149
540	528	529	Nichinan Line	100	1.2509	1.4703
541	529	530	Nichinan Line	100	1.1283	1.3253
542	530	531	Nichinan Line	100	1.0062	1.1798
543	531	532	Nichinan Line	100	0.8845	1.0339
544	532	533	Nichinan Line	100	0.7628	0.8875

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
545	533	534	Nichinan Line	100	0.6409	0.7407
546	534	535	Nichinan Line	100	0.5184	0.5935
547	535	536	Nichinan Line	100	0.3946	0.4458
548	536	537	Nichinan Line	100	0.2688	0.2976
549	537	538	Nichinan Line	100	0.1390	0.1490
550	512	539	Nichinan Line	100	0.2296	0.1490
551	95	540	Ibusuki Makurazaki Line	100	6.3758	4.9514
552	540	541	Ibusuki Makurazaki Line	100	6.0630	4.8175
553	541	542	Ibusuki Makurazaki Line	100	5.7741	4.6831
554	542	543	Ibusuki Makurazaki Line	100	5.5034	4.5483
555	543	544	Ibusuki Makurazaki Line	100	5.2474	4.4131
556	544	545	Ibusuki Makurazaki Line	100	5.0039	4.2774
557	545	546	Ibusuki Makurazaki Line	100	4.7712	4.1413
558	546	547	Ibusuki Makurazaki Line	100	4.5481	4.0047
559	547	548	Ibusuki Makurazaki Line	100	4.3333	3.8677
560	548	549	Ibusuki Makurazaki Line	100	4.1261	3.7302
561	549	550	Ibusuki Makurazaki Line	100	3.9258	3.5923
562	550	551	Ibusuki Makurazaki Line	100	3.7316	3.4540
563	551	552	Ibusuki Makurazaki Line	100	3.5432	3.3152
564	552	553	Ibusuki Makurazaki Line	100	3.3598	3.1759
565	553	554	Ibusuki Makurazaki Line	100	3.1812	3.0362
566	554	555	Ibusuki Makurazaki Line	100	3.0070	2.8961
567	555	556	Ibusuki Makurazaki Line	100	2.8368	2.7555
568	556	557	Ibusuki Makurazaki Line	100	2.6703	2.6145
569	557	558	Ibusuki Makurazaki Line	100	2.5072	2.4730
570	558	559	Ibusuki Makurazaki Line	100	2.3472	2.3311
571	559	560	Ibusuki Makurazaki Line	100	2.1902	2.1888
572	560	561	Ibusuki Makurazaki Line	100	2.0357	2.0460
573	561	562	Ibusuki Makurazaki Line	100	1.8837	1.9027
574	562	563	Ibusuki Makurazaki Line	100	1.7339	1.7590
575	563	564	Ibusuki Makurazaki Line	100	1.5861	1.6149
576	564	565	Ibusuki Makurazaki Line	100	1.4401	1.4703
577	565	566	Ibusuki Makurazaki Line	100	1.2955	1.3253
578	566	567	Ibusuki Makurazaki Line	100	1.1523	1.1798
579	567	568	Ibusuki Makurazaki Line	100	1.0101	1.0339
580	568	569	Ibusuki Makurazaki Line	100	0.8687	0.8875
581	569	570	Ibusuki Makurazaki Line	100	0.7277	0.7407
582	570	571	Ibusuki Makurazaki Line	100	0.5867	0.5935
583	571	572	Ibusuki Makurazaki Line	100	0.4451	0.4458
584	572	573	Ibusuki Makurazaki Line	100	0.3018	0.2976
585	573	574	Ibusuki Makurazaki Line	100	0.1552	0.1490
586	577	578	Nishitetsu-Tenjin Ōmuta Line	100	0.1689	0.1490
587	578	579	Nishitetsu-Tenjin Ōmuta Line	100	1.8796	2.4730
588	579	580	Nishitetsu-Tenjin Ōmuta Line	100	1.9714	2.6145
589	580	581	Nishitetsu-Tenjin Ōmuta Line	100	2.0683	2.7555
590	581	582	Nishitetsu-Tenjin Ōmuta Line	100	2.1683	2.8961
591	582	583	Nishitetsu-Tenjin Ōmuta Line	100	2.2706	3.0362

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
592	583	584	Nishitetsu-Tenjin Ōmuta Line	100	2.3750	3.1759
593	584	585	Nishitetsu-Tenjin Ōmuta Line	100	2.4815	3.3152
594	585	586	Nishitetsu-Tenjin Ōmuta Line	100	2.5901	3.4540
595	586	587	Nishitetsu-Tenjin Ōmuta Line	100	2.7011	3.5923
596	587	588	Nishitetsu-Tenjin Ōmuta Line	100	2.8150	3.7302
597	588	589	Nishitetsu-Tenjin Ōmuta Line	100	2.9331	3.8677
598	589	590	Nishitetsu-Tenjin Ōmuta Line	100	3.2601	4.2774
599	590	591	Nishitetsu-Tenjin Ōmuta Line	100	3.3643	4.4131
600	591	592	Nishitetsu-Tenjin Ōmuta Line	100	3.4730	4.5483
601	592	593	Nishitetsu-Tenjin Ōmuta Line	100	3.5851	4.6831
602	593	594	Nishitetsu-Tenjin Ōmuta Line	100	3.7003	4.8175
603	594	595	Nishitetsu-Tenjin Ōmuta Line	100	3.8185	4.9514
604	595	596	Nishitetsu-Tenjin Ōmuta Line	100	3.9399	5.0849
605	596	597	Nishitetsu-Tenjin Ōmuta Line	100	4.0647	5.2179
606	597	598	Nishitetsu-Tenjin Ōmuta Line	100	4.1935	5.3504
607	598	599	Nishitetsu-Tenjin Ōmuta Line	100	4.3270	5.4826
608	599	600	Nishitetsu-Tenjin Ōmuta Line	100	4.4671	5.6143
609	600	601	Nishitetsu-Tenjin Ōmuta Line	100	5.7709	7.1597
610	601	602	Nishitetsu-Tenjin Ōmuta Line	100	5.8719	7.2856
611	602	603	Nishitetsu-Tenjin Ōmuta Line	100	5.9796	7.4111
612	603	604	Nishitetsu-Tenjin Ōmuta Line	100	6.0926	7.5361
613	604	605	Nishitetsu-Tenjin Ōmuta Line	100	6.2102	7.6607
614	605	606	Nishitetsu-Tenjin Ōmuta Line	100	6.3321	7.7848
615	606	607	Nishitetsu-Tenjin Ōmuta Line	100	6.4580	7.9084
616	607	608	Nishitetsu-Tenjin Ōmuta Line	100	6.5880	8.0317
617	608	609	Nishitetsu-Tenjin Ōmuta Line	100	6.7221	8.1545
618	609	610	Nishitetsu-Tenjin Ōmuta Line	100	6.8605	8.2768

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
619	610	611	Nishitetsu-Tenjin Ōmuta Line	100	7.0032	8.3987
620	611	612	Nishitetsu-Tenjin Ōmuta Line	100	7.1504	8.5201
621	612	613	Nishitetsu-Tenjin Ōmuta Line	100	7.3024	8.6411
622	613	614	Nishitetsu-Tenjin Ōmuta Line	100	7.4595	8.7617
623	614	615	Nishitetsu-Tenjin Ōmuta Line	100	7.6220	8.8818
624	615	616	Nishitetsu-Tenjin Ōmuta Line	100	7.7902	9.0015
625	616	617	Nishitetsu-Tenjin Ōmuta Line	100	7.9647	9.1207
626	617	618	Nishitetsu-Tenjin Ōmuta Line	100	8.1458	9.2395
627	618	619	Nishitetsu-Tenjin Ōmuta Line	100	8.3343	9.3578
628	619	620	Nishitetsu-Tenjin Ōmuta Line	100	8.5309	9.4757
629	620	621	Nishitetsu-Tenjin Ōmuta Line	100	8.7364	9.5932
630	621	622	Nishitetsu-Tenjin Ōmuta Line	100	8.9519	9.7102
631	622	623	Nishitetsu-Tenjin Ōmuta Line	100	9.1787	9.8267
632	623	624	Nishitetsu-Tenjin Ōmuta Line	100	9.4188	9.9428
633	624	59	Nishitetsu-Tenjin Ōmuta Line	100	9.6752	10.0585
634	589	625	Nishitetsu-Dazaifu Line	100	0.3480	0.2976
635	626	626	Nishitetsu-Dazaifu Line	100	0.1758	0.1490
636	600	627	Nishitetsu-Amagi Line	100	1.7223	1.6149
637	627	628	Nishitetsu-Amagi Line	100	1.5527	1.4703
638	628	629	Nishitetsu-Amagi Line	100	1.3893	1.3253
639	629	630	Nishitetsu-Amagi Line	100	1.2302	1.1798
640	630	631	Nishitetsu-Amagi Line	100	1.0743	1.0339
641	631	632	Nishitetsu-Amagi Line	100	0.9208	0.8875
642	632	633	Nishitetsu-Amagi Line	100	0.7690	0.7407
643	633	634	Nishitetsu-Amagi Line	100	0.6182	0.5935
644	634	635	Nishitetsu-Amagi Line	100	0.4676	0.4458
645	635	636	Nishitetsu-Amagi Line	100	0.3163	0.2976
646	636	637	Nishitetsu-Amagi Line	100	0.1622	0.1490
647	652	638	Nishitetsu-Kaizuka Line	0.0000	0.1019	0.1743
648	638	639	Nishitetsu-Kaizuka Line	0.0001	0.1788	0.3102
649	639	29	Nishitetsu-Kaizuka Line	0.0003	0.3040	0.4565
650	29	640	Nishitetsu-Kaizuka Line	0.0013	0.2061	0.4351
651	640	641	Nishitetsu-Kaizuka Line	0.0009	0.0941	0.2874
652	641	642	Nishitetsu-Kaizuka Line	0.0006	0.0369	0.1424
653	642	643	Nishitetsu-Kaizuka Line	0.0004	0.0300	0.0106
654	643	242	Nishitetsu-Kaizuka Line	0.0002	0.0751	0.1583
655	242	644	Nishitetsu-Kaizuka Line	100	0.7219	0.2976
656	644	645	Nishitetsu-Kaizuka Line	100	0.3520	0.1490

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
657	32	646	Fukuoka City Subway-Kūkō Line	100	0.9249	0.2976
658	646	647	Fukuoka City Subway-Kūkō Line	100	0.4431	0.1490
659	280	648	Fukuoka City Subway-Hakozaki Line	0.0013	0.3217	0.5773
660	648	649	Fukuoka City Subway-Hakozaki Line	0.0008	0.1946	0.4310
661	649	650	Fukuoka City Subway-Hakozaki Line	0.0004	0.1123	0.2957
662	650	651	Fukuoka City Subway-Hakozaki Line	0.0002	0.0702	0.1614
663	651	652	Fukuoka City Subway-Hakozaki Line	0.0000	0.0666	0.0388
664	653	654	Fukuoka City Subway-Nanakuma Line	100	0.1580	0.1490
665	654	578	Fukuoka City Subway-Nanakuma Line	100	0.3130	0.2976
666	578	655	Fukuoka City Subway-Nanakuma Line	100	1.4673	1.9027
667	655	656	Fukuoka City Subway-Nanakuma Line	100	1.3489	1.7590
668	656	657	Fukuoka City Subway-Nanakuma Line	100	1.2351	1.6149
669	657	658	Fukuoka City Subway-Nanakuma Line	100	1.1238	1.4703
670	658	659	Fukuoka City Subway-Nanakuma Line	100	1.0140	1.3253
671	659	660	Fukuoka City Subway-Nanakuma Line	100	0.9050	1.1798
672	660	661	Fukuoka City Subway-Nanakuma Line	100	0.7965	1.0339
673	661	662	Fukuoka City Subway-Nanakuma Line	100	0.6880	0.8875
674	662	663	Fukuoka City Subway-Nanakuma Line	100	0.5791	0.7407
675	663	664	Fukuoka City Subway-Nanakuma Line	100	0.4694	0.5935
676	664	665	Fukuoka City Subway-Nanakuma Line	100	0.3582	0.4458
677	665	666	Fukuoka City Subway-Nanakuma Line	100	0.2447	0.2976
678	666	667	Fukuoka City Subway-Nanakuma Line	100	0.1271	0.1490
679	668	4	Shinkansen	100	0.4667	0.1490
680	4	32	Shinkansen	0.4266	5.5100	8.6972
681	32	669	Shinkansen	1.4890	7.8886	15.7639
682	669	321	Shinkansen	1.5070	7.9003	15.8382
683	321	49	Shinkansen	1.1955	2.0617	21.5246
684	49	53	Shinkansen	3.3307	3.1099	22.5021
685	53	670	Shinkansen	5.5235	8.5196	19.9903
686	670	671	Shinkansen	5.4664	8.4341	19.9505
687	671	72	Shinkansen	5.4093	8.4062	19.9166
688	72	81	Shinkansen	4.8107	4.6423	17.7172

Table 21. The results of vulnerabilities and edge betweenness centrality by the link of the Kyushu railway network (Cont.)

No.	From node	To node	Line	Algebraic connectivity-based vulnerability (%)	Global efficiency-based vulnerability (%)	Edge betweenness centrality (%)
689	81	575	Shinkansen	20.3273	7.0547	14.8505
690	575	576	Shinkansen	19.9439	6.8748	14.7680
691	576	83	Shinkansen	19.5610	6.7367	14.6877
692	83	95	Shinkansen	1.4368	3.7285	14.0830

Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network

No.	From node	To node	Line	Algebraic connectivity-based vulnerability (%)	Global efficiency-based vulnerability (%)	Edge betweenness centrality (%)
1	1	2	Tokyo Metro Ginza Line	0.0001	0.1065	0.3587
2	2	3	Tokyo Metro Ginza Line	0.0000	0.0471	0.0411
3	3	4	Tokyo Metro Ginza Line	0.0010	0.1550	0.4564
4	4	5	Tokyo Metro Ginza Line	0.1102	1.0068	4.6560
5	5	6	Tokyo Metro Ginza Line	0.0772	0.3536	1.6426
6	6	7	Tokyo Metro Ginza Line	0.0041	0.0963	0.5044
7	7	8	Tokyo Metro Ginza Line	0.0025	0.1260	0.7084
8	8	9	Tokyo Metro Ginza Line	0.0167	0.2044	2.7922
9	9	10	Tokyo Metro Ginza Line	0.0012	0.2314	1.1453
10	10	11	Tokyo Metro Ginza Line	0.0039	0.2219	1.2302
11	11	12	Tokyo Metro Ginza Line	0.0152	0.0529	0.1610
12	12	13	Tokyo Metro Ginza Line	0.0424	0.3515	0.5913
13	13	14	Tokyo Metro Ginza Line	0.0554	0.1780	0.3873
14	14	15	Tokyo Metro Ginza Line	0.0701	0.3092	0.4771
15	15	16	Tokyo Metro Ginza Line	0.1797	1.3625	3.4251
16	16	17	Tokyo Metro Ginza Line	0.0655	0.3666	0.8761
17	17	18	Tokyo Metro Ginza Line	0.0470	0.1699	0.4927
18	18	19	Tokyo Metro Ginza Line	0.0317	0.1751	0.2732
19	20	21	Tokyo Metro Marunouchi Line	100	0.5824	0.4975
20	21	22	Tokyo Metro Marunouchi Line	100	1.1688	0.9900
21	22	23	Tokyo Metro Marunouchi Line	100	1.7861	1.4776
22	23	24	Tokyo Metro Marunouchi Line	100	2.4545	1.9602
23	24	28	Tokyo Metro Marunouchi Line	100	3.2022	2.4378
24	25	26	Tokyo Metro Marunouchi Line	100	0.6789	0.4975
25	26	27	Tokyo Metro Marunouchi Line	100	1.3872	0.9900
26	27	28	Tokyo Metro Marunouchi Line	100	2.1680	1.4776
27	28	29	Tokyo Metro Marunouchi Line	1.3483	0.8820	7.6990
28	29	30	Tokyo Metro Marunouchi Line	1.7420	1.0749	8.1070
29	30	31	Tokyo Metro Marunouchi Line	0.1736	0.6545	5.9671

Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
30	31	32	Tokyo Metro Marunouchi Line	0.0116	0.2450	0.4100
31	32	33	Tokyo Metro Marunouchi Line	0.0200	0.1523	0.4516
32	33	34	Tokyo Metro Marunouchi Line	0.0307	0.3459	0.7376
33	34	5	Tokyo Metro Marunouchi Line	0.0402	0.3167	1.5725
34	6	35	Tokyo Metro Marunouchi Line	0.0003	0.3018	1.2048
35	35	9	Tokyo Metro Marunouchi Line	0.0001	0.1395	0.6341
36	9	36	Tokyo Metro Marunouchi Line	0.0463	0.1555	1.4605
37	36	37	Tokyo Metro Marunouchi Line	0.0580	0.2023	1.6131
38	37	38	Tokyo Metro Marunouchi Line	0.1041	0.5650	2.4971
39	38	39	Tokyo Metro Marunouchi Line	0.3770	0.2525	0.6527
40	39	40	Tokyo Metro Marunouchi Line	0.3956	0.1626	0.5682
41	40	41	Tokyo Metro Marunouchi Line	4.0000	1.3660	5.6191
42	41	42	Tokyo Metro Marunouchi Line	1.1255	1.1331	4.0374
43	42	43	Tokyo Metro Marunouchi Line	1.0957	0.7475	3.6668
44	43	44	Tokyo Metro Marunouchi Line	1.0687	0.6134	3.3976
45	45	46	Tokyo Metro Hibiya Line	100	0.7740	0.4975
46	46	47	Tokyo Metro Hibiya Line	100	1.6010	0.9900
47	47	48	Tokyo Metro Hibiya Line	100	2.5357	1.4776
48	48	49	Tokyo Metro Hibiya Line	0.0038	0.1904	0.8089
49	49	7	Tokyo Metro Hibiya Line	0.0094	0.2288	0.9036
50	7	35	Tokyo Metro Hibiya Line	0.0065	0.1122	0.6198
51	35	50	Tokyo Metro Hibiya Line	0.0133	0.2692	1.3348
52	50	9	Tokyo Metro Hibiya Line	0.0097	0.1350	1.6178
53	9	51	Tokyo Metro Hibiya Line	0.0066	0.1066	0.3200
54	51	52	Tokyo Metro Hibiya Line	0.0078	0.0613	0.5676
55	52	53	Tokyo Metro Hibiya Line	0.0000	0.1599	0.2859
56	53	54	Tokyo Metro Hibiya Line	0.0013	0.1661	0.3683
57	54	55	Tokyo Metro Hibiya Line	0.0320	0.0803	0.5203
58	55	56	Tokyo Metro Hibiya Line	0.0529	0.1436	0.9307
59	56	57	Tokyo Metro Hibiya Line	0.0657	0.1436	1.0224
60	57	15	Tokyo Metro Hibiya Line	0.0721	0.5025	2.9233
61	16	58	Tokyo Metro Hibiya Line	0.0908	1.0314	2.3401
62	58	59	Tokyo Metro Hibiya Line	0.0484	0.6513	1.8824
63	59	60	Tokyo Metro Hibiya Line	0.0221	0.4148	1.4247
64	60	61	Tokyo Metro Hibiya Line	0.0069	0.3031	0.9670
65	62	63	Tokyo Metro Tozai Line	100	0.7050	0.4975
66	63	64	Tokyo Metro Tozai Line	100	1.4382	0.9900
67	64	65	Tokyo Metro Tozai Line	100	2.2352	1.4776
68	65	66	Tokyo Metro Tozai Line	100	3.1309	1.9602

Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
69	66	67	Tokyo Metro Tozai Line	100	4.1801	2.4378
70	67	68	Tokyo Metro Tozai Line	0.3651	0.4407	2.9745
71	68	69	Tokyo Metro Tozai Line	0.0913	0.1697	2.2677
72	69	37	Tokyo Metro Tozai Line	0.0786	0.2097	2.2644
73	37	11	Tokyo Metro Tozai Line	0.0852	0.4893	5.3462
74	11	54	Tokyo Metro Tozai Line	0.0501	0.3493	3.4068
75	54	70	Tokyo Metro Tozai Line	0.1643	0.2952	3.3987
76	70	71	Tokyo Metro Tozai Line	100	6.2871	5.1990
77	71	72	Tokyo Metro Tozai Line	100	5.4191	4.7512
78	72	73	Tokyo Metro Tozai Line	100	4.6729	4.2985
79	73	74	Tokyo Metro Tozai Line	100	4.0104	3.8408
80	74	75	Tokyo Metro Tozai Line	100	3.4093	3.3781
81	75	76	Tokyo Metro Tozai Line	100	2.8550	2.9104
82	76	77	Tokyo Metro Tozai Line	100	2.3368	2.4378
83	77	78	Tokyo Metro Tozai Line	100	1.8463	1.9602
84	78	79	Tokyo Metro Tozai Line	100	1.3765	1.4776
85	79	80	Tokyo Metro Tozai Line	100	0.9201	0.9900
86	80	81	Tokyo Metro Tozai Line	100	0.4680	0.4975
87	82	83	Tokyo Metro Chiyoda Line	100	0.8148	0.4975
88	83	84	Tokyo Metro Chiyoda Line	100	1.6998	0.9900
89	84	2	Tokyo Metro Chiyoda Line	0.0001	0.2651	1.0613
90	2	85	Tokyo Metro Chiyoda Line	0.0000	0.1742	0.3276
91	85	86	Tokyo Metro Chiyoda Line	0.0011	0.1325	0.4197
92	86	6	Tokyo Metro Chiyoda Line	0.0052	0.3733	0.7641
93	50	87	Tokyo Metro Chiyoda Line	0.0239	0.0924	0.1339
94	87	37	Tokyo Metro Chiyoda Line	0.0324	0.1599	0.3636
95	38	88	Tokyo Metro Chiyoda Line	0.2207	1.7855	2.8738
96	88	89	Tokyo Metro Chiyoda Line	0.1415	1.1602	2.4161
97	89	90	Tokyo Metro Chiyoda Line	0.0885	0.7268	1.9584
98	90	91	Tokyo Metro Chiyoda Line	0.0520	0.4393	1.5007
99	91	92	Tokyo Metro Chiyoda Line	0.0269	0.2755	1.0570
100	92	61	Tokyo Metro Chiyoda Line	0.0106	0.2275	0.7097
101	61	93	Tokyo Metro Chiyoda Line	100	1.2681	0.9900
102	93	94	Tokyo Metro Chiyoda Line	100	0.6257	0.4975
103	95	96	Tokyo Metro Yurakucho Line	100	0.5122	0.4975
104	96	97	Tokyo Metro Yurakucho Line	100	1.0146	0.9900
105	97	98	Tokyo Metro Yurakucho Line	100	1.5287	1.4776
106	98	99	Tokyo Metro Yurakucho Line	100	2.0656	1.9602
107	99	100	Tokyo Metro Yurakucho Line	100	2.6354	2.4378
108	100	101	Tokyo Metro Yurakucho Line	100	3.2494	2.9104
109	101	102	Tokyo Metro Yurakucho Line	100	3.9232	3.3781
110	102	44	Tokyo Metro Yurakucho Line	100	4.6811	3.8408
111	44	103	Tokyo Metro Yurakucho Line	0.0021	0.2016	0.4825
112	103	104	Tokyo Metro Yurakucho Line	0.0008	0.1752	0.6330

Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
113	104	105	Tokyo Metro Yurakucho Line	0.0001	0.3639	1.0036
114	105	67	Tokyo Metro Yurakucho Line	0.0001	0.7958	1.4340
115	67	106	Tokyo Metro Yurakucho Line	0.5872	0.6270	5.9476
116	106	107	Tokyo Metro Yurakucho Line	0.0923	0.1801	1.3189
117	107	5	Tokyo Metro Yurakucho Line	0.0787	0.1692	1.1966
118	5	108	Tokyo Metro Yurakucho Line	0.0266	0.5663	3.1225
119	108	9	Tokyo Metro Yurakucho Line	0.0180	0.5898	3.1005
120	9	52	Tokyo Metro Yurakucho Line	0.0297	0.4123	3.6285
121	52	109	Tokyo Metro Yurakucho Line	0.0846	0.5827	3.6921
122	109	110	Tokyo Metro Yurakucho Line	100	2.5320	1.4776
123	110	111	Tokyo Metro Yurakucho Line	100	1.5997	0.9900
124	111	112	Tokyo Metro Yurakucho Line	100	0.7736	0.4975
125	2	4	Tokyo Metro Hanzomon Line	0.0009	0.3526	1.6741
126	5	113	Tokyo Metro Hanzomon Line	0.1412	0.2957	1.6335
127	113	68	Tokyo Metro Hanzomon Line	0.1590	0.3174	1.7782
128	68	114	Tokyo Metro Hanzomon Line	0.0241	0.4063	4.0756
129	114	37	Tokyo Metro Hanzomon Line	0.3700	0.5504	6.9693
130	37	12	Tokyo Metro Hanzomon Line	0.0337	0.2083	2.0788
131	12	55	Tokyo Metro Hanzomon Line	0.0291	0.1014	1.6859
132	55	115	Tokyo Metro Hanzomon Line	0.1059	0.5483	4.7682
133	115	116	Tokyo Metro Hanzomon Line	0.0411	1.0092	5.0340
134	116	117	Tokyo Metro Hanzomon Line	0.0208	0.4591	0.9507
135	117	118	Tokyo Metro Hanzomon Line	0.0377	0.2335	0.5654
136	119	120	Tokyo Metro Namboku Line	100	0.7889	0.4975
137	120	121	Tokyo Metro Namboku Line	100	1.6417	0.9900
138	121	122	Tokyo Metro Namboku Line	0.0387	0.5052	2.0628
139	122	123	Tokyo Metro Namboku Line	0.0251	0.1562	0.7084

Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
140	123	6	Tokyo Metro Namboku Line	0.0373	0.2920	1.0182
141	34	106	Tokyo Metro Namboku Line	0.1172	0.2353	1.6044
142	67	41	Tokyo Metro Namboku Line	3.9638	1.5742	6.9334
143	41	124	Tokyo Metro Namboku Line	100	5.6625	3.8408
144	124	125	Tokyo Metro Namboku Line	100	4.6537	3.3781
145	125	126	Tokyo Metro Namboku Line	100	3.7974	2.9104
146	126	127	Tokyo Metro Namboku Line	100	3.0425	2.4378
147	127	128	Tokyo Metro Namboku Line	100	2.3599	1.9602
148	128	129	Tokyo Metro Namboku Line	100	1.7302	1.4776
149	129	130	Tokyo Metro Namboku Line	100	1.1383	0.9900
150	130	131	Tokyo Metro Namboku Line	100	0.5696	0.4975
151	44	132	Tokyo Metro Fukutoshin Line	0.6746	0.6679	0.6679
152	132	133	Tokyo Metro Fukutoshin Line	0.6346	0.6558	0.6558
153	133	134	Tokyo Metro Fukutoshin Line	0.5964	0.8416	0.8416
154	134	31	Tokyo Metro Fukutoshin Line	0.0250	0.2221	0.2221
155	31	135	Tokyo Metro Fukutoshin Line	0.0385	0.5851	0.5851
156	135	84	Tokyo Metro Fukutoshin Line	0.0248	0.4383	0.4383
157	84	1	Tokyo Metro Fukutoshin Line	0.0004	0.0598	0.0598
158	136	137	Toei Asakusa Line	100	0.5390	0.5390
159	137	138	Toei Asakusa Line	100	1.0732	1.0732
160	138	139	Toei Asakusa Line	100	1.6258	1.4776
161	139	140	Toei Asakusa Line	100	2.2105	1.9602
162	140	141	Toei Asakusa Line	100	2.8412	2.4378
163	141	142	Toei Asakusa Line	100	3.5365	2.9104
164	142	143	Toei Asakusa Line	100	4.3263	3.3781
165	143	144	Toei Asakusa Line	0.0320	0.5038	2.7928
166	144	8	Toei Asakusa Line	0.0628	0.5867	3.3419
167	8	51	Toei Asakusa Line	0.0026	0.0885	0.9571
168	51	145	Toei Asakusa Line	0.0080	0.1451	0.6217
169	145	11	Toei Asakusa Line	0.0143	0.2526	0.8100
170	11	55	Toei Asakusa Line	0.0017	0.2029	2.7976
171	55	146	Toei Asakusa Line	0.0111	0.2445	1.4770
172	146	147	Toei Asakusa Line	0.0004	0.2481	0.7616
173	147	148	Toei Asakusa Line	0.0001	0.1282	0.4477
174	148	19	Toei Asakusa Line	0.0129	0.3278	1.0836
175	19	149	Toei Asakusa Line	0.0863	0.2929	0.6613

Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
176	149	118	Toei Asakusa Line	0.0593	0.1808	0.4561
177	121	143	Toei Mita Line	0.0049	0.3673	1.8596
178	143	150	Toei Mita Line	0.0343	0.2615	0.7882
179	150	151	Toei Mita Line	0.0545	0.2037	0.8846
180	151	152	Toei Mita Line	0.0795	0.3701	1.2356
181	152	50	Toei Mita Line	0.1100	0.7820	1.6568
182	50	37	Toei Mita Line	0.0756	0.4903	4.0381
183	114	153	Toei Mita Line	1.9377	0.6250	5.2418
184	153	41	Toei Mita Line	1.9474	0.5852	5.1514
185	41	154	Toei Mita Line	100	8.0896	6.9403
186	154	155	Toei Mita Line	100	7.1230	6.5124
187	155	156	Toei Mita Line	100	6.3099	6.0796
188	156	157	Toei Mita Line	100	5.6002	5.6418
189	157	158	Toei Mita Line	100	4.9658	5.1990
190	158	159	Toei Mita Line	100	4.3889	4.7512
191	159	160	Toei Mita Line	100	3.8571	4.2985
192	160	161	Toei Mita Line	100	3.3612	3.8408
193	161	162	Toei Mita Line	100	2.8944	3.3781
194	162	163	Toei Mita Line	100	2.4510	2.9104
195	163	164	Toei Mita Line	100	2.0261	2.4378
196	164	165	Toei Mita Line	100	1.6155	1.9602
197	165	166	Toei Mita Line	100	1.2148	1.4776
198	166	167	Toei Mita Line	100	0.8190	0.9900
199	167	168	Toei Mita Line	100	0.4204	0.4975
200	31	169	Toei Shinjuku Line	0.1450	0.9790	7.0009
201	169	106	Toei Shinjuku Line	0.1672	1.1451	7.2679
202	106	68	Toei Shinjuku Line	0.0216	0.4373	3.7468
203	114	38	Toei Shinjuku Line	0.0944	0.3457	2.1723
204	38	57	Toei Shinjuku Line	0.0975	0.5060	1.9860
205	57	146	Toei Shinjuku Line	0.1072	0.2771	1.7822
206	146	170	Toei Shinjuku Line	0.0457	0.2206	0.6158
207	170	171	Toei Shinjuku Line	0.0328	0.1034	0.3016
208	171	172	Toei Shinjuku Line	0.0366	0.0766	0.3843
209	172	116	Toei Shinjuku Line	0.0236	0.0608	0.2376
210	116	173	Toei Shinjuku Line	100	4.7446	3.8408
211	173	174	Toei Shinjuku Line	100	3.9611	3.3781
212	174	175	Toei Shinjuku Line	100	3.2722	2.9104
213	175	176	Toei Shinjuku Line	100	2.6489	2.4378
214	176	177	Toei Shinjuku Line	100	2.0733	1.9602
215	177	178	Toei Shinjuku Line	100	1.5327	1.4776
216	178	179	Toei Shinjuku Line	100	1.0164	0.9900
217	179	180	Toei Shinjuku Line	100	0.5128	0.4975
218	181	30	Toei Oedo Line	0.4723	0.3872	0.9030
219	30	134	Toei Oedo Line	0.3638	0.3106	1.6002
220	134	182	Toei Oedo Line	0.2210	0.3260	0.4577
221	182	183	Toei Oedo Line	0.2491	0.2311	0.5549
222	183	184	Toei Oedo Line	0.2792	0.3710	0.8090
223	184	67	Toei Oedo Line	0.3120	0.7661	1.1559
224	40	15	Toei Oedo Line	1.6006	1.2034	5.1138
225	15	185	Toei Oedo Line	0.1537	0.5158	1.7747
226	185	148	Toei Oedo Line	0.1320	0.3215	1.4840
227	148	186	Toei Oedo Line	0.0478	0.2306	0.6804
228	186	171	Toei Oedo Line	0.0342	0.2393	0.6432
229	171	115	Toei Oedo Line	0.0132	0.2539	0.9385

Table 22. The results of vulnerabilities and edge betweenness centrality by the link of the Tokyo subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
230	115	70	Toei Oedo Line	0.0862	0.4535	2.1828
231	70	109	Toei Oedo Line	0.0144	0.7061	3.5314
232	109	187	Toei Oedo Line	0.0024	0.4574	1.0835
233	187	188	Toei Oedo Line	0.0000	0.2207	0.6996
234	188	189	Toei Oedo Line	0.0029	0.2026	0.6992
235	189	144	Toei Oedo Line	0.0112	0.4016	0.9878
236	144	190	Toei Oedo Line	0.0003	0.2342	0.6501
237	190	122	Toei Oedo Line	0.0028	0.2277	0.6393
238	122	48	Toei Oedo Line	0.0114	0.6298	2.3449
239	48	4	Toei Oedo Line	0.0370	1.0227	3.5462
240	4	191	Toei Oedo Line	0.0164	0.9247	3.8164
241	191	192	Toei Oedo Line	0.0069	0.7084	3.5266
242	192	30	Toei Oedo Line	0.0015	0.7611	3.5380
243	181	193	Toei Oedo Line	0.3700	0.1429	0.4950
244	193	28	Toei Oedo Line	0.2856	0.1321	0.0871
245	28	194	Toei Oedo Line	100	4.4433	3.8408
246	194	195	Toei Oedo Line	100	3.7266	3.3781
247	195	196	Toei Oedo Line	100	3.0914	2.9104
248	196	197	Toei Oedo Line	100	2.5120	2.4378
249	197	198	Toei Oedo Line	100	1.9729	1.9602
250	198	199	Toei Oedo Line	100	1.4632	1.4776
251	199	200	Toei Oedo Line	100	0.9734	0.9900
252	200	201	Toei Oedo Line	100	0.4927	0.4975

Table 23. The results of vulnerabilities and edge betweenness centrality by the link of the Osaka subway network

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
1	1	2	Midosuji Line	100	1.2310	0.9434
2	2	3	Midosuji Line	100	2.4380	1.8688
3	3	4	Midosuji Line	100	3.7006	2.7763
4	4	5	Midosuji Line	100	5.0761	3.6658
5	5	6	Midosuji Line	100	6.6439	4.5373
6	6	7	Midosuji Line	5.0777	1.3688	6.2279
7	7	8	Midosuji Line	5.7129	1.7012	6.7670
8	8	9	Midosuji Line	1.3611	0.8541	5.8567
9	9	10	Midosuji Line	1.7878	0.8021	5.7396
10	10	11	Midosuji Line	7.2422	2.1105	10.0958
11	11	12	Midosuji Line	10.2562	1.5134	6.5397
12	12	13	Midosuji Line	26.2520	1.2310	0.9434
13	13	14	Midosuji Line	100	8.2512	6.2264
14	14	15	Midosuji Line	100	6.7537	5.3908
15	15	16	Midosuji Line	100	5.4527	4.5373
16	16	17	Midosuji Line	100	4.2757	3.6658
17	17	18	Midosuji Line	100	3.1791	2.7763
18	18	19	Midosuji Line	100	2.1295	1.8688
19	19	20	Midosuji Line	100	1.0921	0.9434
20	21	22	Tanimachi Line	100	1.2745	0.9434
21	22	23	Tanimachi Line	100	2.5709	1.8688
22	23	24	Tanimachi Line	12.7716	1.4258	3.1866

Table 23. The results of vulnerabilities and edge betweenness centrality by the link of the Osaka subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
23	24	25	Tanimachi Line	18.2786	1.3043	3.4382
24	25	26	Tanimachi Line	23.5687	1.3875	3.8874
25	26	27	Tanimachi Line	28.6376	1.6626	4.5073
26	27	28	Tanimachi Line	33.4899	2.1502	5.1453
27	28	29	Tanimachi Line	1.5564	0.6320	3.8215
28	29	6	Tanimachi Line	2.1151	0.9183	4.5073
29	6	30	Tanimachi Line	0.2093	0.7033	2.7628
30	30	31	Tanimachi Line	0.7294	0.9321	4.4602
31	31	32	Tanimachi Line	0.9327	1.1942	5.0082
32	32	33	Tanimachi Line	1.4233	0.6140	3.6331
33	33	34	Tanimachi Line	4.8194	1.0302	4.1870
34	34	35	Tanimachi Line	22.8613	1.8339	8.1063
35	35	13	Tanimachi Line	22.0155	1.6096	7.7140
36	13	36	Tanimachi Line	100	9.5716	7.8437
37	36	37	Tanimachi Line	100	8.1265	7.0440
38	37	38	Tanimachi Line	100	6.8803	6.2264
39	38	39	Tanimachi Line	100	5.7625	5.3908
40	39	40	Tanimachi Line	100	4.7328	4.5373
41	40	41	Tanimachi Line	100	3.7633	3.6658
42	41	42	Tanimachi Line	100	2.8322	2.7763
43	42	43	Tanimachi Line	100	1.9185	1.8688
44	43	44	Tanimachi Line	100	0.9949	0.9434
45	45	46	Yotsubashi Line	100	1.8117	0.9434
46	46	8	Yotsubashi Line	100	3.8238	1.8688
47	11	47	Yotsubashi Line	63.8955	4.7541	7.3094
48	47	48	Yotsubashi Line	60.3184	3.7135	6.5636
49	48	49	Yotsubashi Line	56.1511	2.9117	5.8179
50	49	50	Yotsubashi Line	51.2606	2.2824	5.0722
51	50	51	Yotsubashi Line	45.4782	1.7885	4.3264
52	52	53	Chūō Line	44.9205	1.8086	4.3169
53	53	54	Chūō Line	50.8300	2.3120	5.0626
54	54	55	Chūō Line	55.8129	2.9560	5.8083
55	55	56	Chūō Line	60.0486	3.7828	6.5541
56	56	57	Chūō Line	63.6771	4.8707	7.2998
57	57	8	Chūō Line	6.4101	2.6320	8.9849
58	8	58	Chūō Line	0.1542	1.3777	5.6167
59	58	32	Chūō Line	0.4474	1.7822	5.7730
60	32	59	Chūō Line	1.8387	2.0117	7.7046
61	59	60	Chūō Line	0.8081	1.5386	5.1153
62	60	61	Chūō Line	100	4.7944	2.7763
63	61	62	Chūō Line	100	3.0509	1.8688
64	62	63	Chūō Line	100	1.4973	0.9434
65	64	65	Sennichimae Line	100	1.6834	1.8688
66	65	57	Sennichimae Line	100	3.5130	1.8688
67	57	66	Sennichimae Line	0.6290	0.8722	3.1319
68	66	67	Sennichimae Line	0.7119	0.4987	2.5636
69	67	10	Sennichimae Line	0.7516	0.7149	3.0027
70	10	68	Sennichimae Line	0.0066	0.6584	3.9480
71	68	34	Sennichimae Line	0.0691	0.9655	4.7138
72	34	69	Sennichimae Line	10.0460	1.9593	8.2719
73	69	70	Sennichimae Line	9.2298	1.6740	7.8407
74	70	71	Sennichimae Line	100	5.6940	3.6658
75	71	72	Sennichimae Line	100	4.0940	2.7763
76	72	73	Sennichimae Line	100	2.6675	1.8688

Table 23. The results of vulnerabilities and edge betweenness centrality by the link of the Osaka subway network (Cont.)

No.	**From node**	**To node**	**Line**	**Algebraic connectivity-based vulnerability (%)**	**Global efficiency-based vulnerability (%)**	**Edge betweenness centrality (%)**
77	73	74	Sennichimae Line	100	1.3333	0.9434
78	28	75	Sakaisuji Line	2.1657	0.5151	2.0515
79	75	30	Sakaisuji Line	2.7914	0.8158	2.7433
80	30	76	Sakaisuji Line	1.4876	0.3808	1.0370
81	76	58	Sakaisuji Line	1.7550	0.6026	1.5881
82	58	77	Sakaisuji Line	1.7609	0.3623	1.9828
83	77	68	Sakaisuji Line	2.3903	0.3547	1.6933
84	68	78	Sakaisuji Line	5.1318	0.5309	1.4007
85	78	12	Sakaisuji Line	4.8626	0.3524	1.0090
86	12	79	Sakaisuji Line	100	1.8951	0.9434
87	80	81	Nagahori Tsurumi-ryokuchi Line	100	1.6326	0.9434
88	81	66	Nagahori Tsurumi-ryokuchi Line	100	3.3876	1.8688
89	66	82	Nagahori Tsurumi-ryokuchi Line	0.0019	0.2745	0.2448
90	82	9	Nagahori Tsurumi-ryokuchi Line	0.0002	0.4562	0.8917
91	9	77	Nagahori Tsurumi-ryokuchi Line	0.0285	0.1960	1.1208
92	77	83	Nagahori Tsurumi-ryokuchi Line	0.2097	0.3517	0.5076
93	83	33	Nagahori Tsurumi-ryokuchi Line	0.1714	0.3306	0.5750
94	33	84	Nagahori Tsurumi-ryokuchi Line	2.0258	0.3019	1.0885
95	84	59	Nagahori Tsurumi-ryokuchi Line	1.7839	0.2637	0.9582
96	59	85	Nagahori Tsurumi-ryokuchi Line	6.0205	0.9437	3.0758
97	85	86	Nagahori Tsurumi-ryokuchi Line	4.7473	0.4447	2.4468
98	86	87	Nagahori Tsurumi-ryokuchi Line	3.6553	0.4366	1.9167
99	87	88	Nagahori Tsurumi-ryokuchi Line	100	5.2009	3.6658
100	88	89	Nagahori Tsurumi-ryokuchi Line	100	3.7655	2.7763
101	89	90	Nagahori Tsurumi-ryokuchi Line	100	2.4709	1.8688
102	90	91	Nagahori Tsurumi-ryokuchi Line	100	1.2442	0.9434
103	92	93	Imazatosuji Line	100	1.1667	0.9434
104	93	94	Imazatosuji Line	100	2.3098	1.8688
105	94	23	Imazatosuji Line	100	3.5241	2.7763
106	23	95	Imazatosuji Line	16.4069	1.7995	3.7975
107	95	96	Imazatosuji Line	22.7676	1.8490	4.3576
108	96	97	Imazatosuji Line	28.6294	2.1271	5.0135
109	97	87	Imazatosuji Line	34.0123	2.6514	5.7173
110	87	98	Imazatosuji Line	6.6630	1.1907	7.2866
111	98	60	Imazatosuji Line	8.1233	1.4622	7.8976
112	60	70	Imazatosuji Line	5.6489	2.0341	8.3798
113	52	99	Nankō Port Town Line	37.8571	1.4208	3.5711
114	99	100	Nankō Port Town Line	29.3699	1.1321	2.8254

Table 23. The results of vulnerabilities and edge betweenness centrality by the link of the Osaka subway network (Cont.)

No.	From node	To node	Line	Algebraic connectivity-based vulnerability (%)	Global efficiency-based vulnerability (%)	Edge betweenness centrality (%)
115	100	101	Nankō Port Town Line	19.2513	0.9321	2.0797
116	101	102	Nankō Port Town Line	8.4833	0.8141	1.6819
117	102	103	Nankō Port Town Line	4.1630	0.7743	1.4124
118	103	104	Nankō Port Town Line	9.6057	0.8115	1.3795
119	104	105	Nankō Port Town Line	20.5164	0.9266	2.0892
120	105	106	Nankō Port Town Line	30.3471	1.1231	2.8350
121	106	51	Nankō Port Town Line	38.5916	1.4071	3.5807

From the result in Tables 21 - 23, if any section (the link between stations) has a higher vulnerability value, this section considers it more vulnerable or sensitive to the risk of extensive disruption within the network. All results were illustrated in the network map and divided into five levels to manage priority easier.

1) Vulnerable sections of the algebraic connectivity-based vulnerability analysis

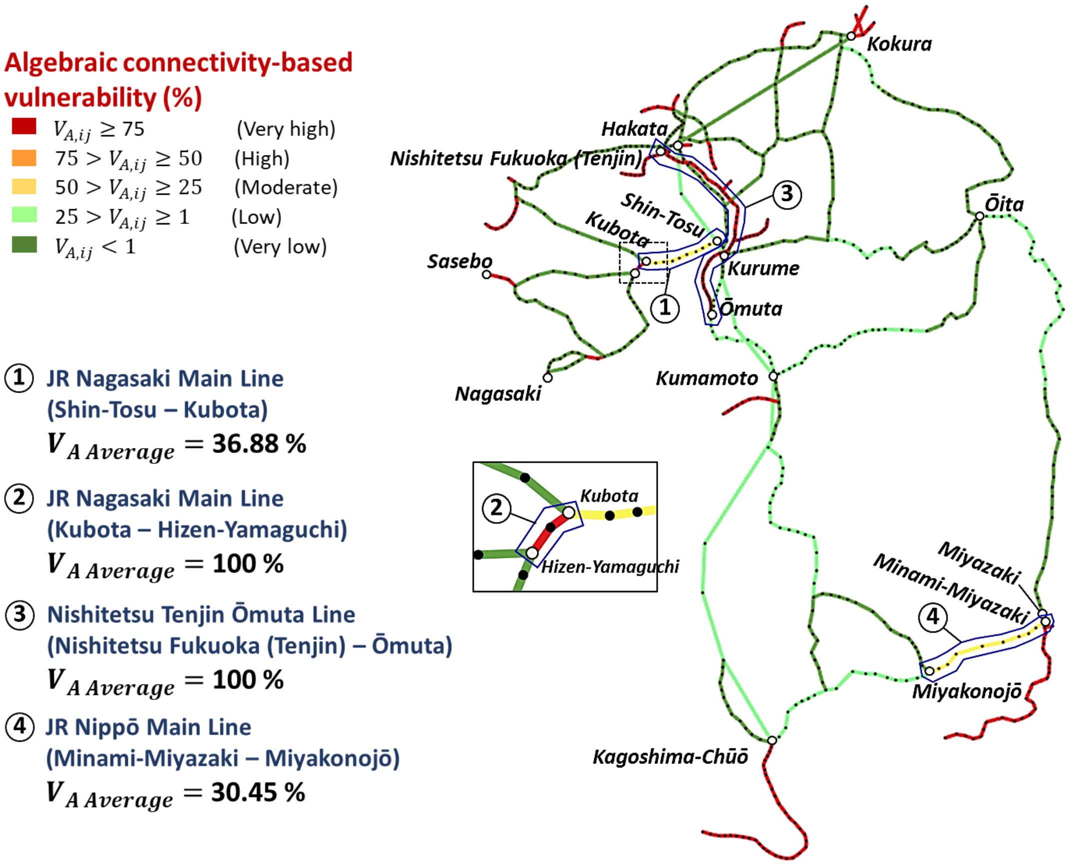


Figure 30. The algebraic connectivity-based vulnerability of the Kyushu railway network

From the algebraic connectivity-based vulnerability evaluation by Eq. 6.1, The result was concluded as Figs. 30 - 32 that the branch lines with no detour route were very vulnerable with a 100% vulnerability. This condition was also shown in some sections of the mainline that have no alternate route, such as the section between Kubota and Hizen-Yamaguchi stations of the JR Kyushu Nagasaki Main Line in the Kyushu railway network in Fig. 30. The main reason is the property of algebraic connectivity that its value will drop to zero if any railway section is cut off or attacked and making the network completely separate from each other. In other words, if after cut-algebraic connectivity is zero, the vulnerability value will be 100%.

In addition, some moderate vulnerable sections on the mainline in the Kyushu network still had the detour route for traveling but consumed very long distances and travel time. In the example, the section between Shin-Tosu and Kubota stations of the JR Kyushu Nagasaki Main Line and the section between Minami-Miyazaki and Miyakonojō stations of the JR Kyushu Nippō Main Line had a 36.88% and 30.45% of average vulnerability, respectively.

In The Tokyo network, the inner area is dense and has strong robustness, so the algebraic connectivity in each section still high, making the algebraic connectivity-based vulnerability in this area is very low when compared with the branch line that has no alternate route. For considering the inner area, which has the grid similarly network, we downscaled the range of a vulnerability value for each level for easier to consider and classify priorities of the inner area as well as the global efficiency-based case.

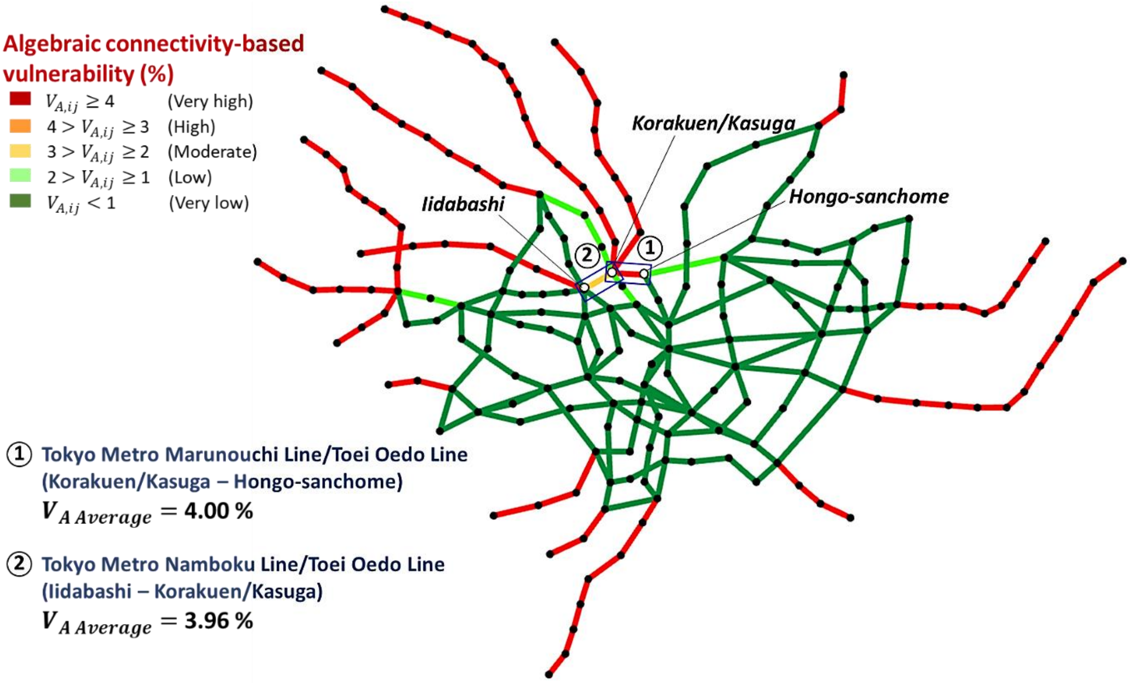


Figure 31. The algebraic connectivity-based vulnerability of the Tokyo subway network

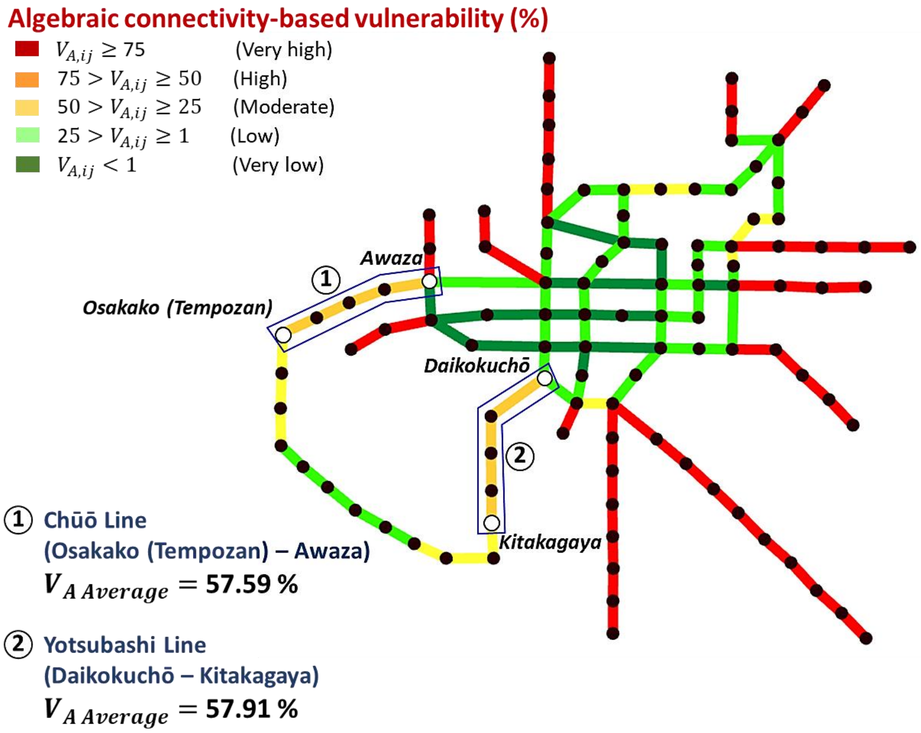


Figure 32. The algebraic connectivity-based vulnerability of the Osaka subway network

In the Tokyo subway network shown in Fig. 31, the no-alternate-route sections always had 100% algebraic connectivity-based vulnerability. Considering the inner area, the section between Korakuen/Kasuga and Hongo-sanchome stations of the Toei Oedo Line had the highest average vulnerability value of 4.00%. It was followed by the section between Iidabashi and Korakuen/Kasuga of the same line, with 3.96% of the average vulnerability. Both sections are not only located on the Toei Oedo Line but also share with the Tokyo Metro Namboku Line and Marunouchi Line, respectively. Moreover, these highly vulnerable sections are important because they connect at least three no-alternate-route sections, and the Korakuen/Kasuga station also had the highest information and betweenness centralities.

In the Osaka network (Fig. 32), if not consider the no-alternate-route sections, the vulnerable sections were located on the Chūō Line between Osakako (Tempozan) and Awaza stations and the Yotsubashi Line between Daikokuchō and Kitakagaya stations. It can notice that these sections are located at the loop line, which has a long-distance detour traveling similar to the Kyushu railway network if it is disrupted or cut off. Both sections had a vulnerability of 57.59% and 57.91% on average, respectively.

2) Vulnerable sections of the global efficiency-based vulnerability analysis

The global efficiency-based vulnerability results were illustrated as shown in Figs. 33 - 35. In the case of the Kyushu railway network (Fig. 33), if considered the mainline section, the most significantly vulnerable section located on the entire Shinkansen line, the JR Kyushu Nagasaki Main Line between Shin-Tosu and Hizen-Yamaguchi stations, and the southern part of the Nishitetsu Tenjin Ōmuta Line between Miyanojin and Ōmuta stations. All three sections had an average vulnerability of 6.22%, 6.68%, and 7.45%, respectively. Unlike the algebraic connectivity-based case, the most vulnerable sections, in this case, were located on the mainline section, especially in the central area of the network, not on the branch lines. This characteristic was similar to the closeness centrality and betweenness centrality analysis of the Kyushu network, which showed the critical stations in the same area. The main reason is the global efficiency-based vulnerability measured on the average shortest route, similar to both closeness and betweenness centrality analyses.

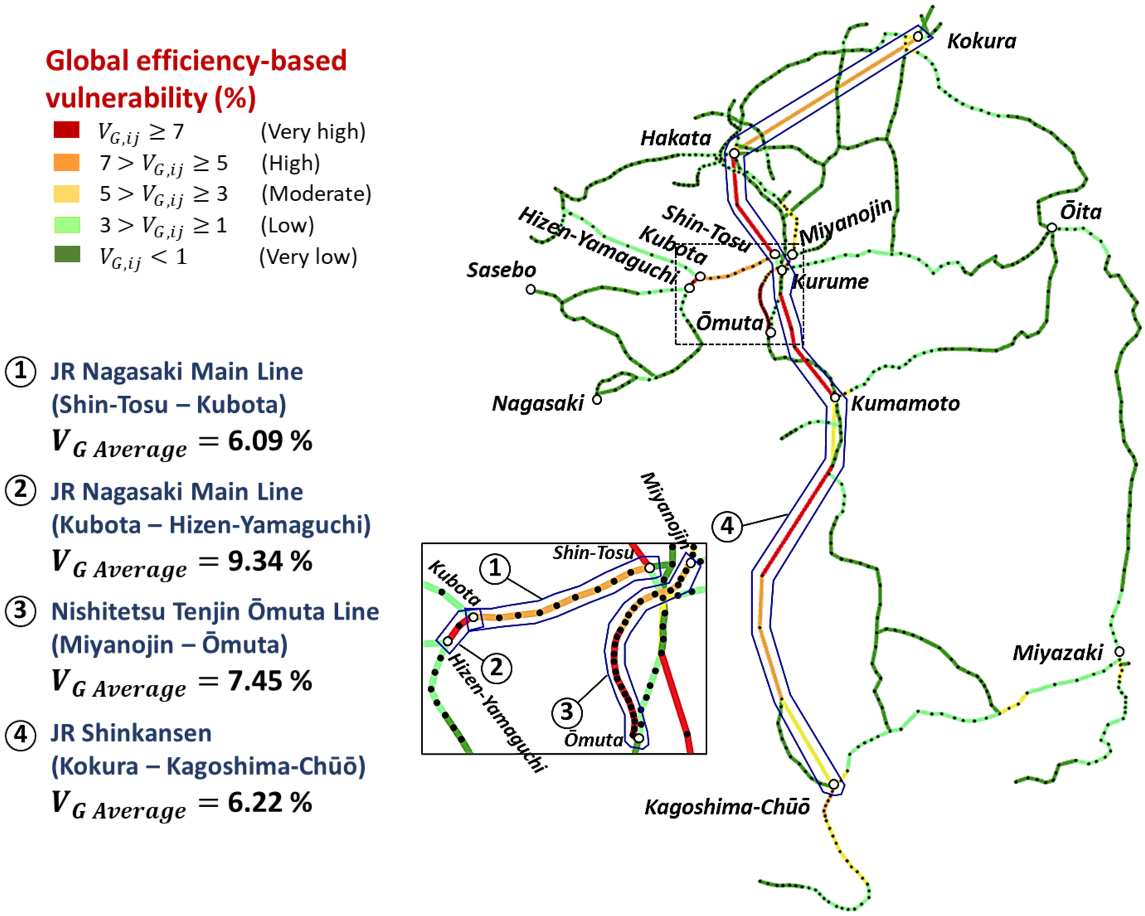


Figure 33. The global efficiency-based vulnerability of the Kyushu railway network

In the Tokyo subway network (Fig. 34), the condition was different from the case of the Kyushu railway network because most highly vulnerable sections were located on the branch section with no detour route. Considering the inner area, which is a dense network, the significant vulnerable section was the section between the Iidabashi and Korakuen/Kasuga stations of the Toei Oedo Line. It shares the route with the Tokyo Metro Namboku Line, which had 1.57% of the vulnerability. Another section was the section between Korakuen/Kasuga and Ueno-hirokoji/Naka-okachimachi stations which had a 1.28% of average vulnerability. This section also shares the route with Tokyo Metro Marunouchi Line. These results are similar to the algebraic connectivity-based vulnerability case, in which the vulnerable sections were located in the same area.

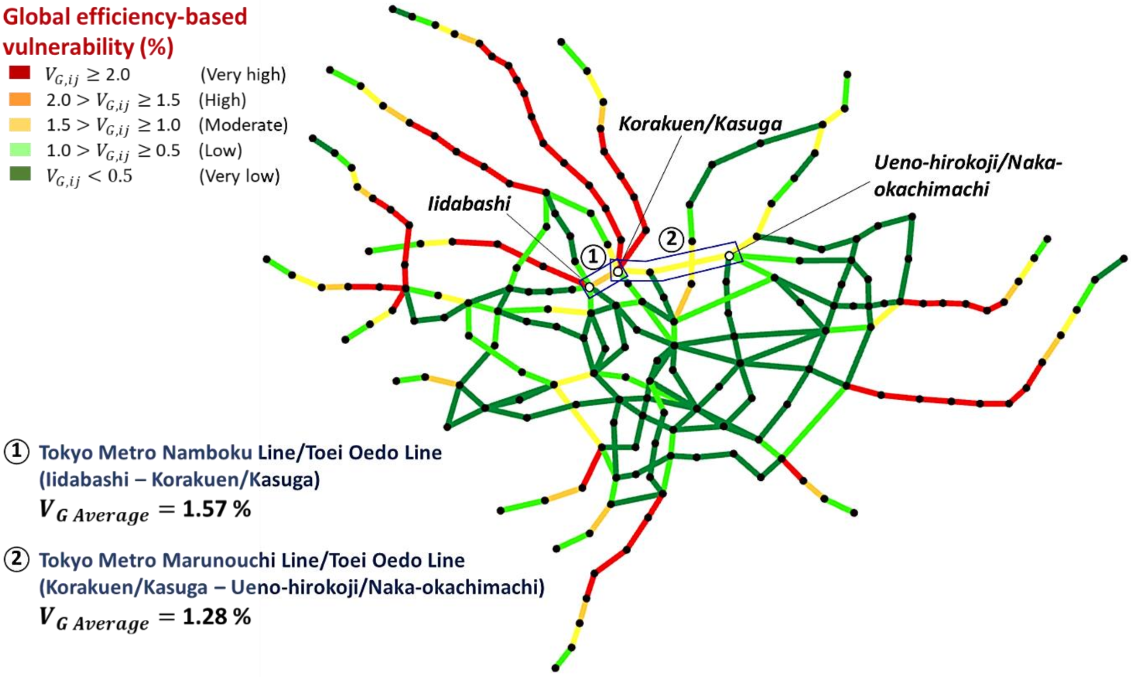


Figure 34. The global efficiency-based vulnerability of the Tokyo subway network

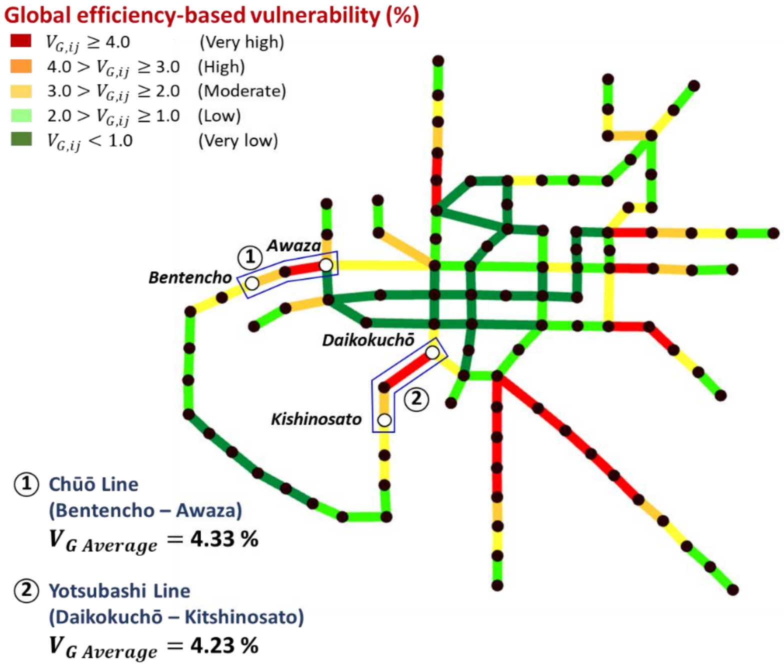


Figure 35. The global efficiency-based vulnerability of the Osaka subway network

In the Osaka network shown in Fig. 35, the result was still similar to the algebraic connectivity-based case in that the significant vulnerable links on the mainline were located in the same area on the loop section. The high vulnerability sections were the section between Bentencho and Awaza stations of the Chūō Line and the section between Daikokuchō and Kitshinosato stations of the Yotsubashi Line. These sections had an average vulnerability of 4.33% and 4.23%, respectively.

3) Critical sections of the edge betweenness centrality analysis

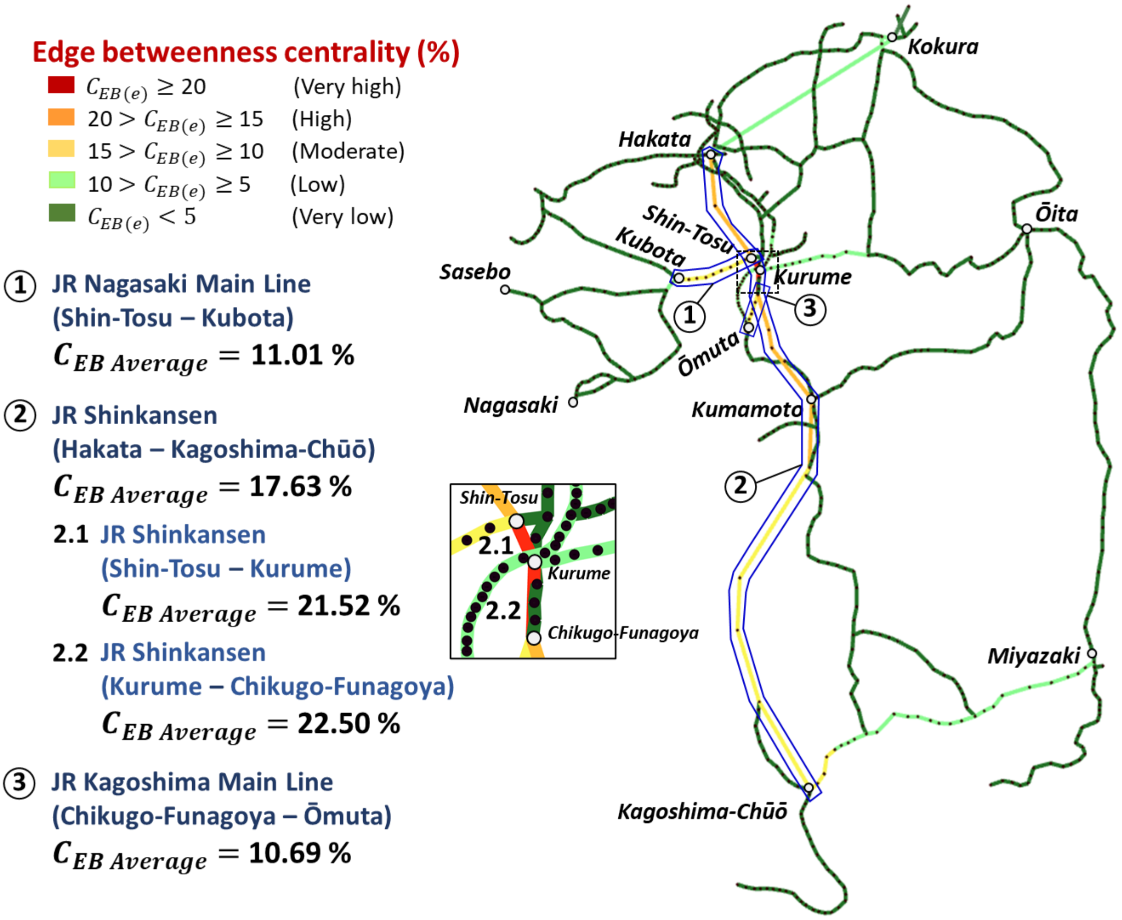


Figure 36. Edge betweenness centrality of the Kyushu railway network

From Fig. 36, The edge betweenness results in the Kyushu railway network showed that the most critical sections were located in the central area of the JR Shinkansen, the eastern part of the JR Kyushu Nagasaki Main Line, and some parts of the JR Kyushu Kagoshima Main Line. This condition has corresponded to the global efficiency-based vulnerability, which considers the average shortage route. The very important sections were the section between Kurume and Chikugo-Funagoya stations of the JR Shinkansen lines, with a centrality value of 22.50%. This was followed by the section between Shin-

Tosu and Kurume stations of the same line, with a value of 21.52%. In addition, the section between Shin-Tosu and Kubota stations of the JR Kyushu Nagasaki Main Line had a moderate edge betweenness centrality value of 11.01% on average, as well as the section between Chikugo-Funagoya and Ōmuta stations on the JR Kyushu Kagoshima Main Line had a value of 10.69% on average.

In the case of the Tokyo subway network, as shown in Fig. 37, The highest average edge betweenness centrality section was located between Nakano-sakaue and Shinjuku/Shinjuku-nishiguchi stations of Tokyo Metro Marunouchi Line with 7.90% of the average centrality. However, the section between Iidabashi and Korakuen/Kasuga stations and between Korakuen/Kasuga and Ueno-hirokoji/Naka-okachimachi stations of the Toei Oedo Line still had significant value on the edge betweenness centrality. Both sections had average centrality of 6.93% and 5.37%, respectively.

The edge betweenness centrality of the Osaka metro network in Fig. 38 showed the most critical sections located in the central area, which is dense and highly robust on a topology view. The most critical section was the section between the Namba and Daikokuchō stations of the Midosuji Line that share the route with the Yotsubashi Line and had 10.10% of centrality. It was followed by the section between Shigino and Imazato stations of the Imazatosuji Line with 8.14% of average centrality.

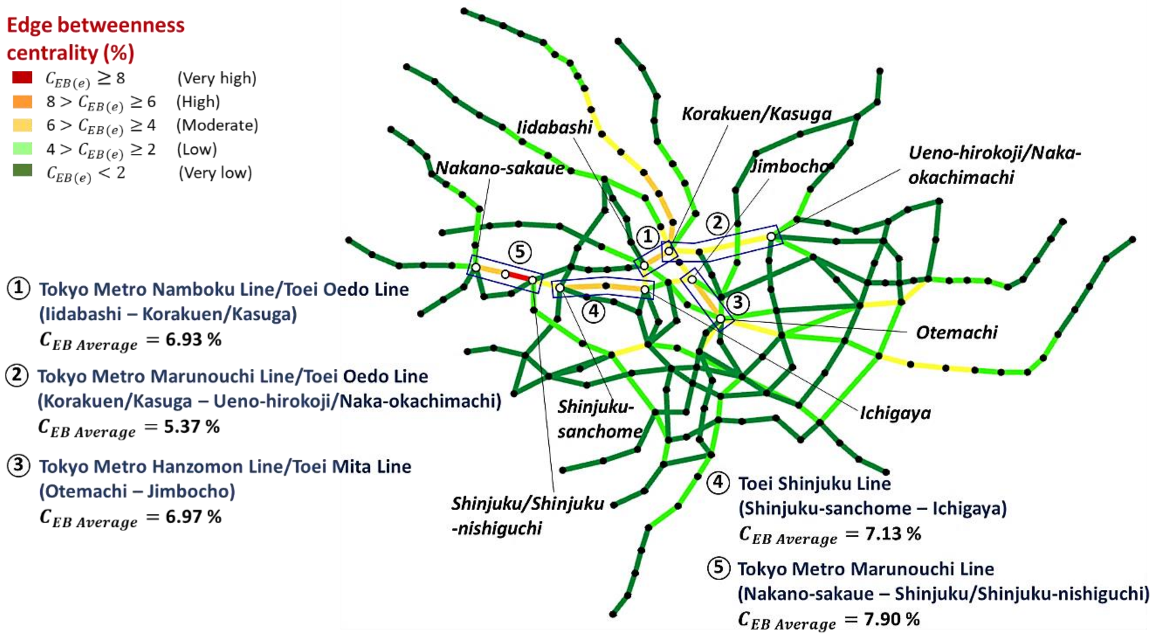


Figure 37. Edge betweenness centrality of the Tokyo subway network

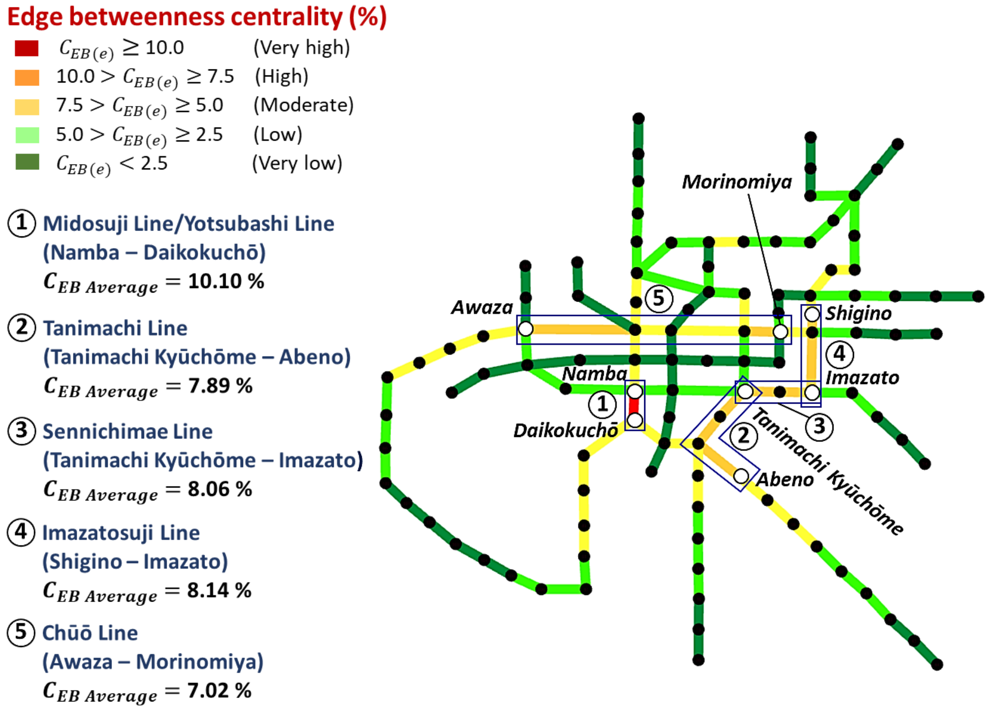


Figure 38. Edge betweenness centrality of the Osaka subway network

6.2.4 Case studies correlation analysis

The algebraic connectivity-based vulnerability, global efficiency-based vulnerability, and edge betweenness centrality were analyzed by correlation analysis with the scatter plot graph. The main purpose is to determine the relationships between indicators from all three criteria and then find important sections with significant value for all criteria.

From a case study of the Kyushu railway network in Figs. 39 - 41, The correlation between each pair of both vulnerabilities and edge betweenness centrality did not correspond to each other overall on a global scale because of a nonlinear relationship. However, on a local scale, considering each group of links or sections, the global efficiency-based vulnerability corresponded more with the edge betweenness centrality even for no-detour-route sections, as shown in Fig. 41. The main reason is both methods are calculated based on the shortest patch in the network. The algebraic connectivity-based vulnerability showed the no-detour-route sections do not correspond with other criteria on a local level because the vulnerability of this section type is steadily at 100%.

In the case of the Tokyo subway network in Figs. 42 - 43, there were large differences between the algebraic connectivity-based vulnerability of no-alternate-route sections in the right-hand side scatter plot and the sections within a grid or dense area on the left-hand side. For measuring and comparing the sections within the inner area, which is denser, the easier method is to omit to calculate the branch section or decrease the range of vulnerability value in each level, as explained in subsection 6.2.3. Although the global efficiency-based vulnerability corresponded more with the edge betweenness centrality on a local scale, as shown in Fig. 44, similar to the Kyushu railway network case, the plot was more scattered than the previous network. This condition showed the probability that the pair of these criteria are less corresponding to each other on a global scale.

In the Osaka subway network case, the correlation pattern between algebraic connectivity-based vulnerability and global efficiency-based vulnerability/edge betweenness centrality in Figs. 45 – 46 was similar to the Kyushu railway network. But the correlation between the global efficiency-based vulnerability and edge betweenness centrality in Fig. 47 was similar to the case of the Tokyo subway network, which was scattered.

The correlation between the global efficiency-based vulnerability and edge betweenness centrality of both Tokyo and Osaka subway networks was different from the Kyushu railway network because the topology of both subway networks has a dense grid network in the central area. In contrast, the inner area of the Kyushu railway network is lighter, and each section between junction nodes has a longer distance and composes of several nodes (smaller stations) between the junction. These conditions are related to the global efficiency definition that calculates the times of nodes between both target nodes.

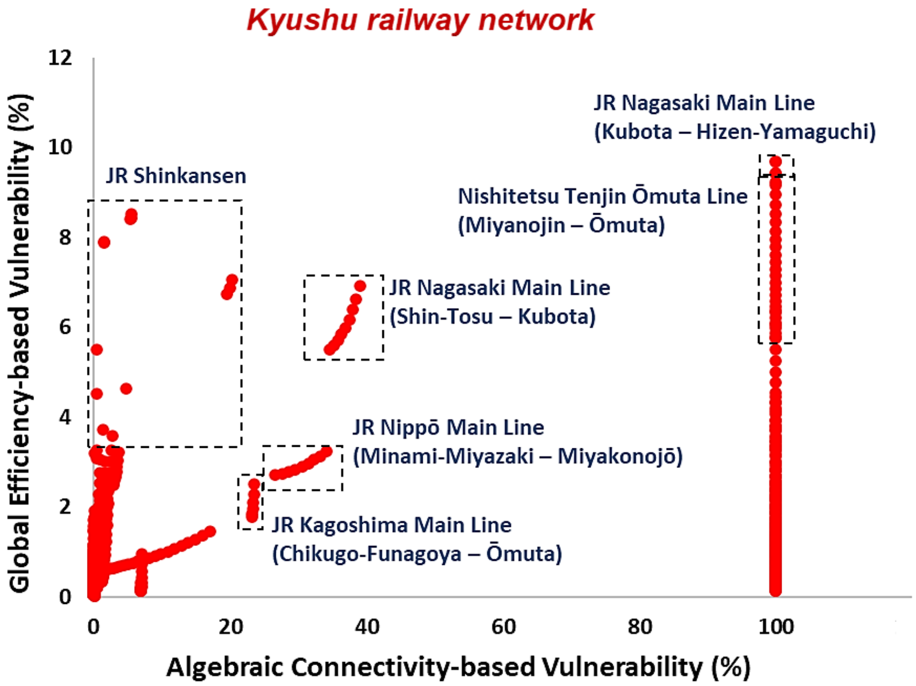


Figure 39. Correlation between the algebraic connectivity- and global efficiency-based vulnerabilities of the Kyushu railway network

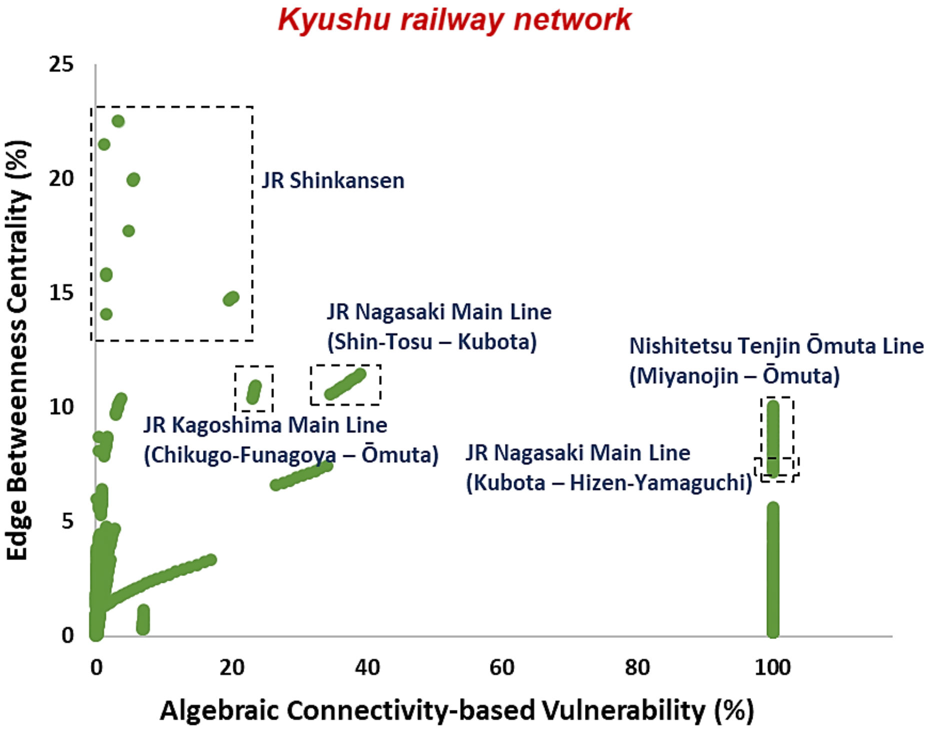


Figure 40. Correlation between the algebraic connectivity-based vulnerability and edge betweenness centrality of the Kyushu railway network

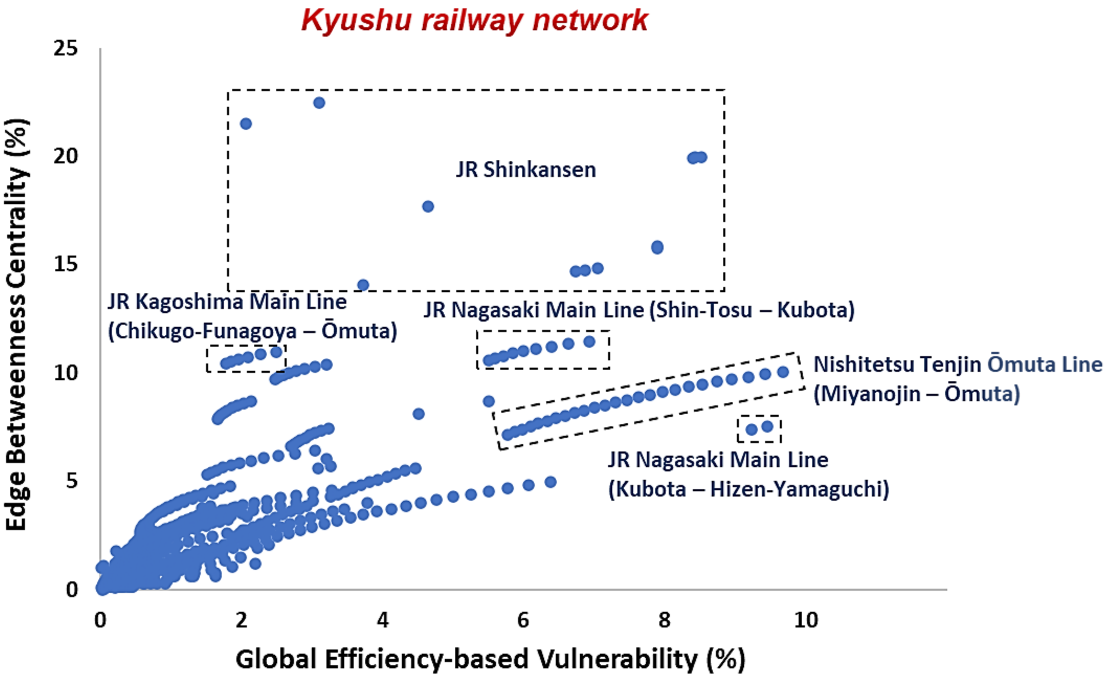


Figure 41. Correlation between the global efficiency-based vulnerability and edge betweenness centrality of the Kyushu railway network

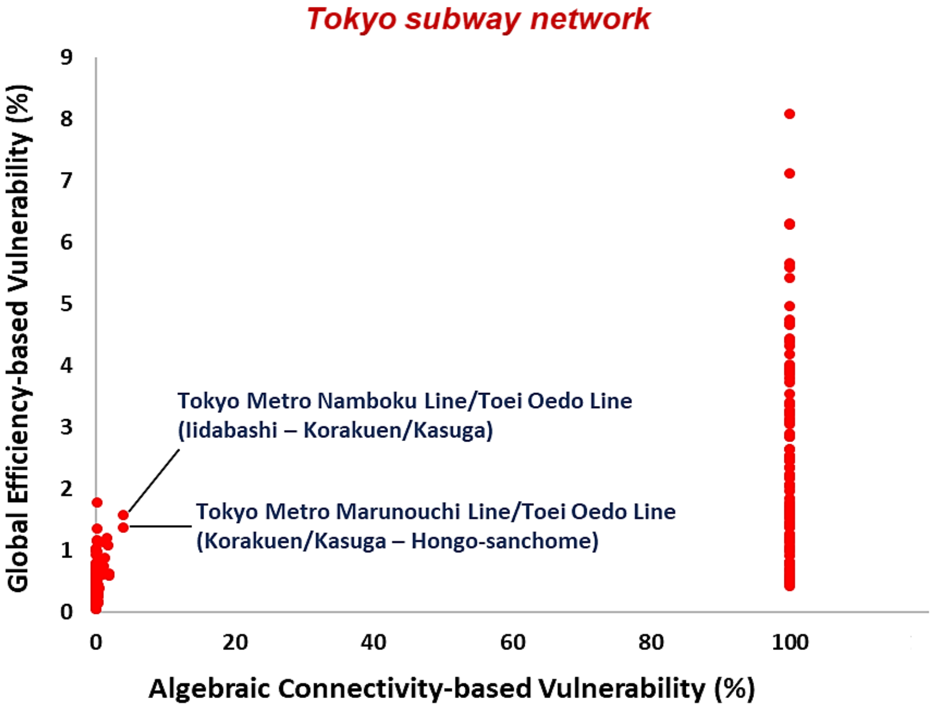


Figure 42. Correlation between the algebraic connectivity- and global efficiency-based vulnerabilities of the Tokyo subway network

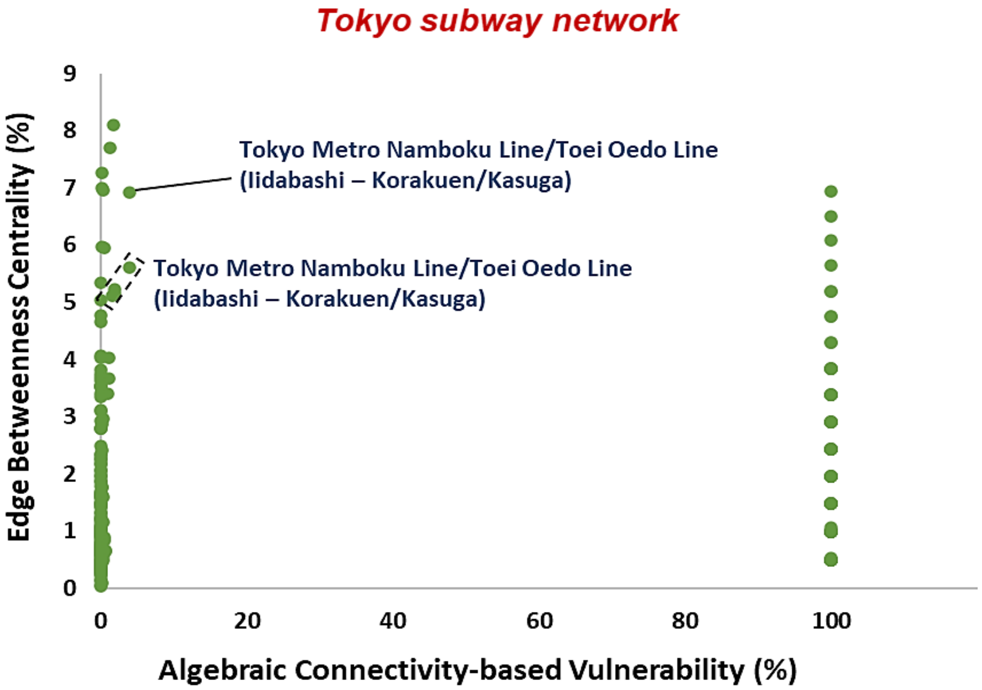


Figure 43. Correlation between the algebraic connectivity-based vulnerability and edge betweenness centrality of the Tokyo subway network

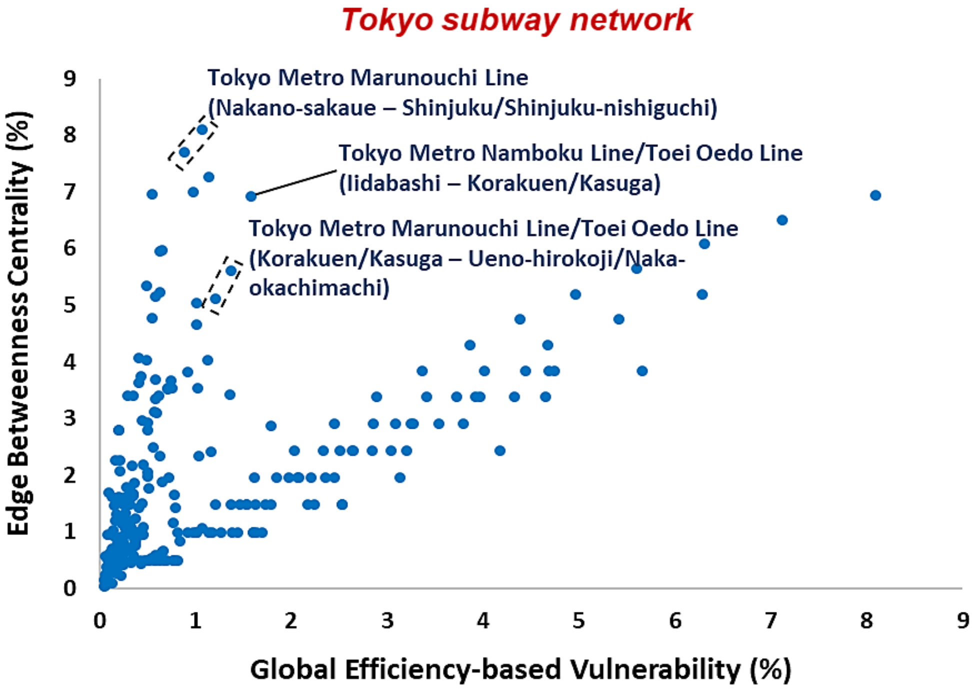


Figure 44. Correlation between the global efficiency-based vulnerability and edge betweenness centrality of the Tokyo subway network

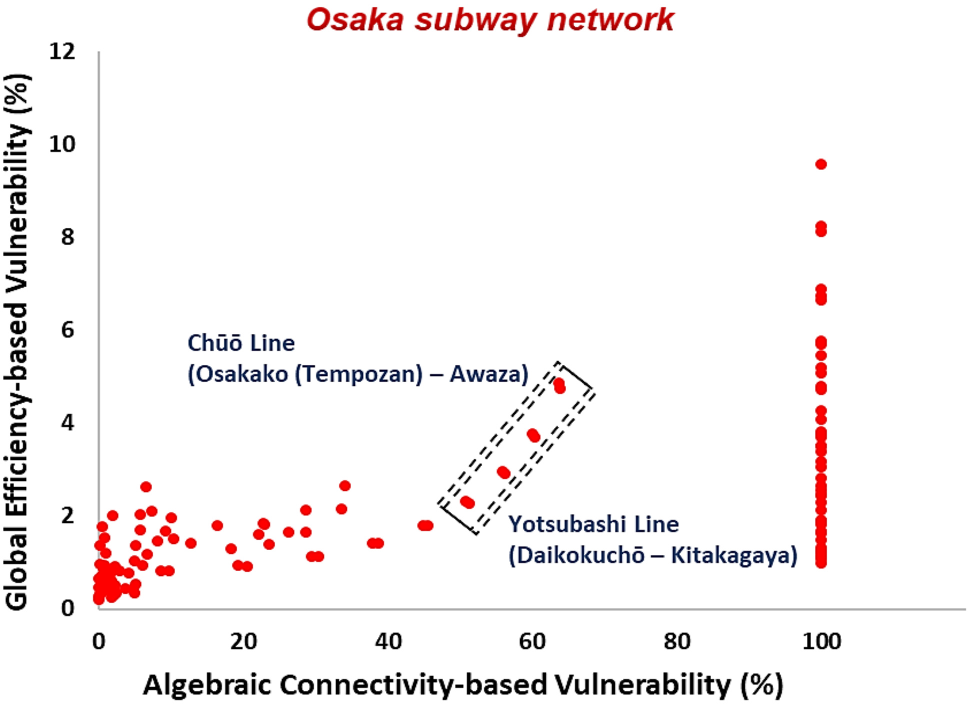


Figure 45. Correlation between the algebraic connectivity- and global efficiency-based vulnerabilities of the Osaka subway network

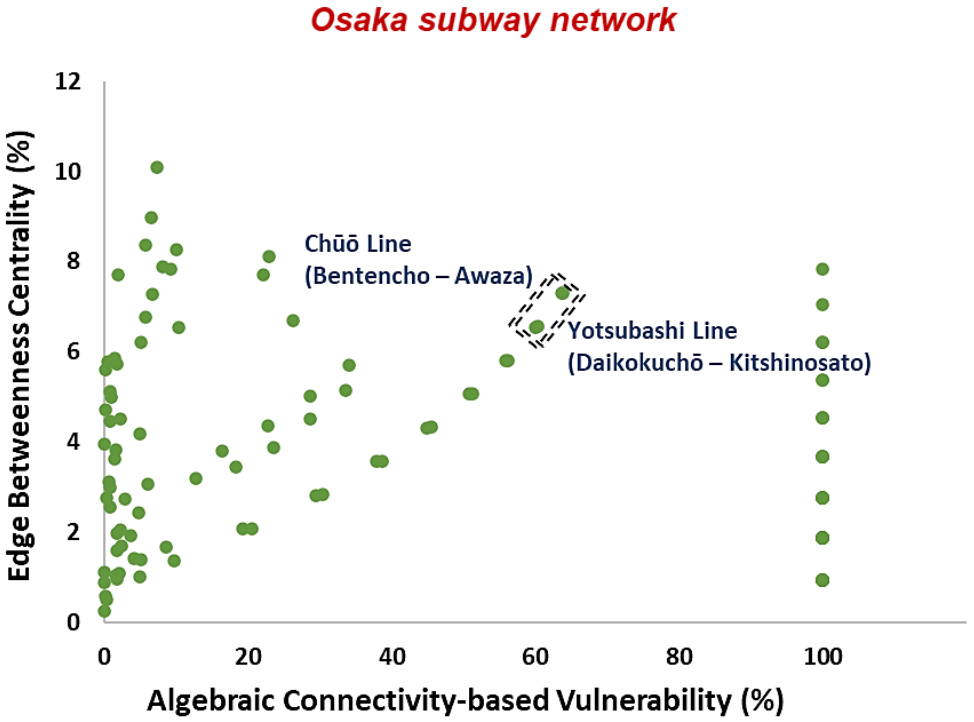


Figure 46. Correlation between the algebraic connectivity-based vulnerability and edge betweenness centrality of the Osaka subway network

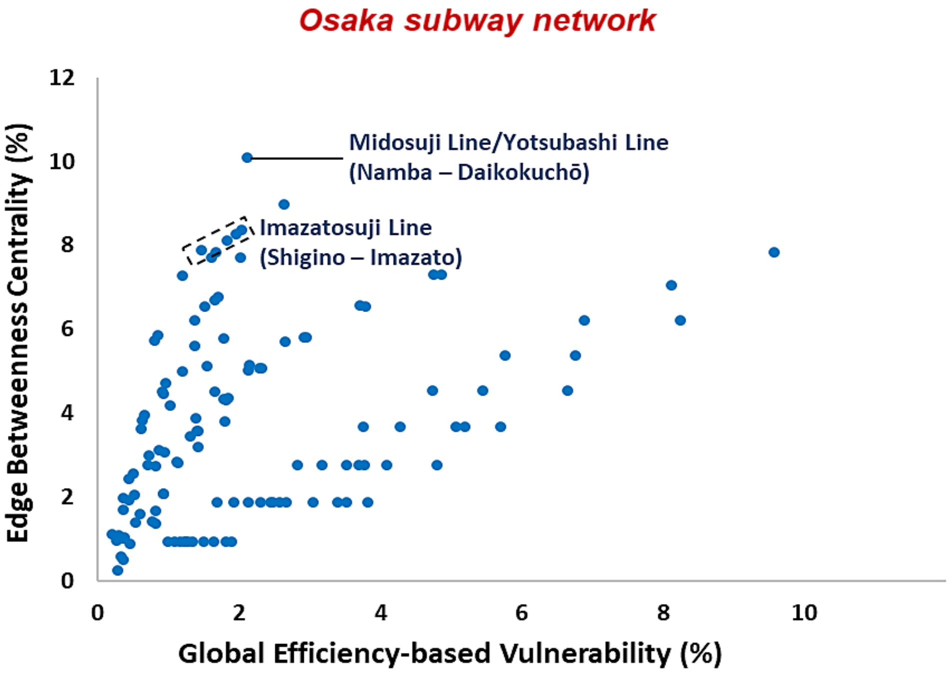


Figure 47. Correlation between the global efficiency-based vulnerability and edge betweenness centrality of the Osaka subway network

6.3 Summary of the Topology-based Comparison Between Algebraic Connectivity- and Global Efficiency-based Vulnerability Analyses

After testing with six different-sized square grid networks, the result showed that the algebraic connectivity-based vulnerability used less processing time than the global efficiency-based vulnerability analysis. In addition, the algebraic connectivity-based method identified the very vulnerable links in a more specific area.

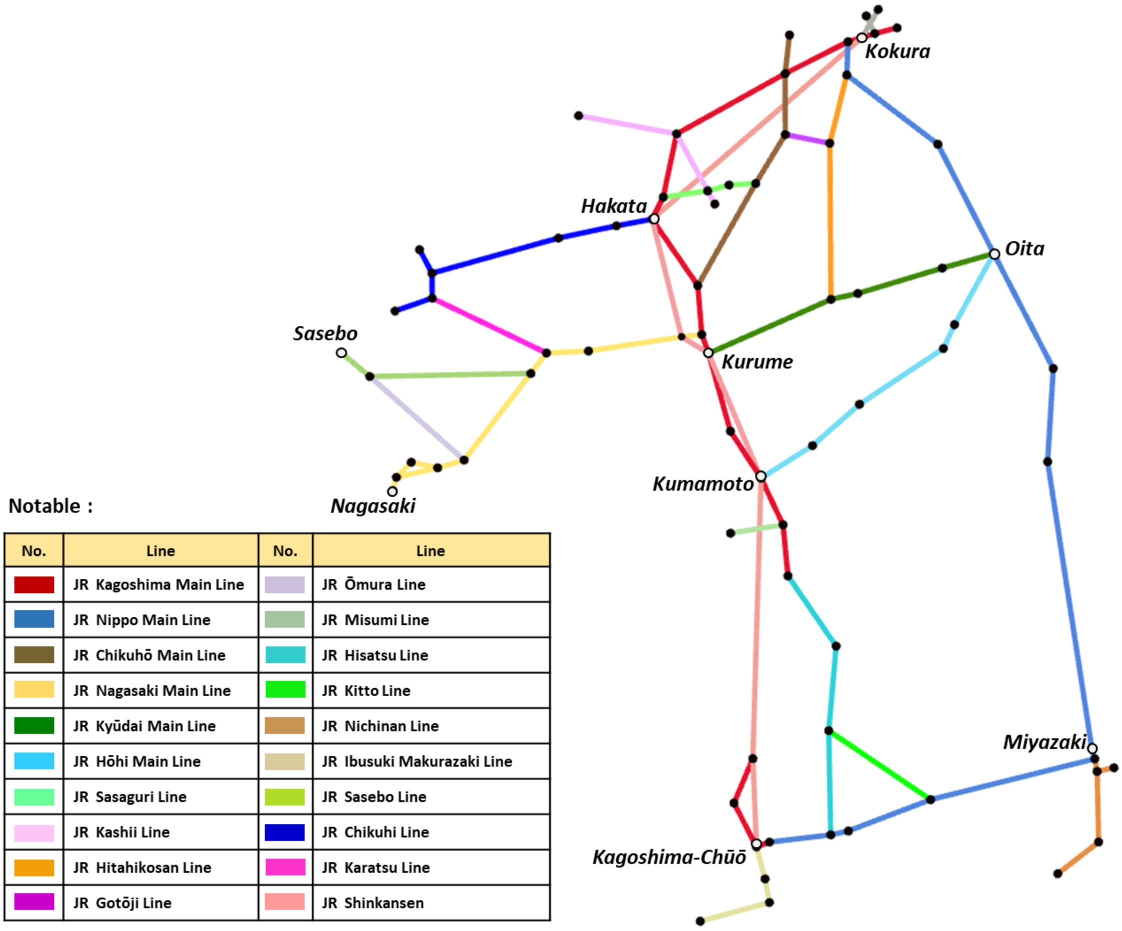
When comparing both vulnerabilities in all three case study networks, the algebraic connectivity-based vulnerability showed the very vulnerable sections on the branch line or non-detour route section. Moreover, it also showed the moderated sections that use very long detour routes, which did not show clearly on the global efficiency-based vulnerability result. The main reason is the property of algebraic connectivity, which is sensitive to the change of topology on the non-detour route link and is considered the second smallest eigenvalue.

The result from both vulnerability analyses showed several interesting sections that need to manage as the priority for preventive strategy planning. In the Kyushu railway network, the very critical sections were located on the JR Kyushu Nagasaki Main Line between Shin-Tosu and Kubota stations and between Kubota and Hizen-Yamaguchi stations. These sections had a significant vulnerability on both algebraic connectivity- and global efficiency-based vulnerabilities. Moreover, the section between Shin-Tosu and Kubota stations also had a significant value of edge betweenness centrality. For this reason, the sections between Shin-Tosu and Hizen-Yamaguchi stations need priority for inspection, repair, or maintenance. If these sections are cut off, it has an extensive effect on the passenger who travels from Nagasaki and Saga prefecture to the other part of Kyushu. The second most important railway was the JR Shinkansen line because of the high level of global efficiency-based vulnerability and edge betweenness centrality. However, this line was not vulnerable when considering the algebraic connectivity viewpoint because of the strong robustness of several interchange stations. Moreover, the southern part of the Nishitetsu Tenjin Ōmuta Line between Miyanojin and Ōmuta stations was also important because this section connects the JR Kyushu railway network at the Ōmuta station, which is the only interchange station.

Another example is the section between Iidabashi and Korakuen/Kasuga stations of the Toei Oedo Line, which shares the route with the Tokyo Metro Namboku Line in the Tokyo subway network. This section had a significantly high value on both vulnerabilities and edge betweenness centrality, even though it did not have the highest value for all three criteria. In addition, this section also connected the large no-detour-route sections from at least three lines, so it is essential to keep these branch lines still connected to the main network and need to manage as the first priority section for preventive planning.

The comparison can conclude that algebraic connectivity-based vulnerability can be the alternative method used to analyze the robust network, which has large dense nodes and links such as the subway and urban railway network in the large city, and the urban street network. However, it needs to adjust the data range of each level in some cases if the network has a combination between branch sections and a large dense grid network. In the global efficiency-based vulnerability case, this method is more corresponding with the edge betweenness centrality than the algebraic connectivity-based case on a local scale. From this result, the global efficiency-based vulnerability can evaluate the network alongside the betweenness centrality due to similarly measuring methods that help scope the critical section or area easier (Sun and Guan, 2016).

6.4 Algebraic Connectivity-based Vulnerability Analysis with the Passenger-weighted Link in the Kyushu Railway Network



No.	Line	No.	Line
🟥	JR Kagoshima Main Line	🟪	JR Ōmura Line
🟦	JR Nippo Main Line	🟩	JR Misumi Line
🟫	JR Chikuhō Main Line	🟦	JR Hisatsu Line
🟨	JR Nagasaki Main Line	🟩	JR Kitto Line
🟩	JR Kyūdai Main Line	🟫	JR Nichinan Line
🟦	JR Hōhi Main Line	🟨	JR Ibusuki Makurazaki Line
🟩	JR Sasaguri Line	🟩	JR Sasebo Line
🟪	JR Kashii Line	🟦	JR Chikuhi Line
🟧	JR Hitahikosan Line	🟪	JR Karatsu Line
🟪	JR Gotōji Line	🟥	JR Shinkansen

Figure 48. The simple JR Kyushu railway network

After analyzing both pure topology-based vulnerability analyses, we applied the passenger-weight adjacency matrix to analyze the passenger-weight algebraic connectivity-based vulnerability. However, the actual passenger flow between each link (edge) is difficult to obtain, and flow data from the Nishitetsu railway and Fukuoka City subway is not available yet, so we focused on using the passenger flow data from the JR Kyushu network and some part of the JR West operated sections (JR Kyushu, 2021; JR

West, 2021). These data illustrate only the estimated daily ridership, not the actual data. Moreover, the testing network was analyzed only the important stations, such as junctions, interchange stations, terminal stations, and large stations, by the number of daily ridership.

From Fig. 48, the simple network is composed of 73 nodes and 90 links, then identified each node by Table 24.

Table 24. Node number and station name of the simple JR Kyushu network

No.	Station Name	Line	No.	Station Name	Line
1	Mojikō	Kagoshima Main Line	37	Umi	Kashii Line
2	Moji	Kagoshima Main Line	38	Tagawa-Gotōji	Hitahikosan Line
3	Kokura	Kagoshima Main Line	39	Yoake	Hitahikosan Line
4	Nishi-Kokura	Kagoshima Main Line	40	Meinohama	Chikuhi Line
5	Orio	Kagoshima Main Line	41	Chikuzen-Maebaru	Chikuhi Line
6	Kashii	Kagoshima Main Line	42	Karatsu	Chikuhi Line
7	Yoshizuka	Kagoshima Main Line	43	Nishi-Karatsu	Chikuhi Line
8	Hakata	Kagoshima Main Line	44	Yamamoto	Chikuhi Line
9	Haruda	Kagoshima Main Line	45	Imari	Chikuhi Line
10	Tosu	Kagoshima Main Line	46	Shin-Tosu	Nagasaki Main Line
11	Kurume	Kagoshima Main Line	47	Saga	Nagasaki Main Line
12	Ōmuta	Kagoshima Main Line	48	Kubota	Nagasaki Main Line
13	Kumamoto	Kagoshima Main Line	49	Hizen-Yamaguchi	Nagasaki Main Line
14	Uto	Kagoshima Main Line	50	Isahaya	Nagasaki Main Line
15	Yatsushiro	Kagoshima Main Line	51	Kikitsu	Nagasaki Main Line
16	Sendai	Kagoshima Main Line	52	Urakami	Nagasaki Main Line
17	Ijūin	Kagoshima Main Line	53	Nagasaki	Nagasaki Main Line
18	Kagoshima-Chūō	Kagoshima Main Line	54	Nagayo	Nagasaki Main Line (Old)
19	Kagoshima	Kagoshima Main Line	55	Haiki	Sasebo Line
20	Shimonoseki	Sanyo Main Line	56	Sasebo	Sasebo Line
21	Jōno	Nippō Main Line	57	Hita	Kyūdai Main Line
22	Nakatsu	Nippō Main Line	58	Yufuin	Kyūdai Main Line
23	Ōita	Nippō Main Line	59	Higo-Ōzu	Hōhi Main Line
24	Saiki	Nippō Main Line	60	Miyaji	Hōhi Main Line
25	Nobeoka	Nippō Main Line	61	Bungo-Taketa	Hōhi Main Line
26	Miyazaki	Nippō Main Line	62	Miemachi	Hōhi Main Line
27	Minami-Miyazaki	Nippō Main Line	63	Misumi	Misumi Line
28	Miyakonojō	Nippō Main Line	64	Hitoyoshi	Hisatsu Line
29	Kokubu	Nippō Main Line	65	Yoshimatsu	Hisatsu Line
30	Hayato	Nippō Main Line	66	Tayoshi	Nichinan Line
31	Wakamatsu	Chikuhō Main Line	67	Aburatsu	Nichinan Line
32	Shin Iizuka	Chikuhō Main Line	68	Shibushi	Nichinan Line
33	Keisen	Chikuhō Main Line	69	Miyazaki Airport	Miyazaki Kūkō Line
34	Chojabaru	Sasaguri Line	70	Kiire	Ibusuki Makurazaki Line
35	Sasaguri	Sasaguri Line	71	Ibusuki	Ibusuki Makurazaki Line
36	Saitozaki	Kashii Line	72	Makurazaki	Ibusuki Makurazaki Line
			73	Shin-Shimonoseki	San'yō Shinkansen

Moreover, the links ID, the number of passengers per day, and the result of algebraic connectivity-based vulnerability with the passenger-weighted link are shown in Table 25.

Table 25. The results of algebraic connectivity-based vulnerability with passenger-weighted edges

No.	From node	To node	Line	Daily Ridership (Passenger/day)	Algebraic connectivity-based vulnerability (%)
1	1	2	Kagoshima Main Line	57,706	100
2	2	3	Kagoshima Main Line	57,706	100
3	3	4	Kagoshima Main Line	115,367	0.2916
4	4	5	Kagoshima Main Line	115,367	0.2799
5	5	6	Kagoshima Main Line	115,367	0.1088
6	6	7	Kagoshima Main Line	115,367	0.0216
7	7	8	Kagoshima Main Line	115,367	0.0000
8	8	9	Kagoshima Main Line	100,520	0.0890
9	9	10	Kagoshima Main Line	100,520	0.1844
10	10	11	Kagoshima Main Line	100,520	0.7939
11	11	12	Kagoshima Main Line	42,131	0.4779
12	12	13	Kagoshima Main Line	40,630	0.5603
13	13	14	Kagoshima Main Line	44,254	79.1850
14	14	15	Kagoshima Main Line	44,254	54.6289
15	16	17	Kagoshima Main Line	41,008	1.8587
16	17	18	Kagoshima Main Line	41,008	1.8587
17	18	19	Kagoshima Main Line	45,464	32.7435
18	20	2	Kagoshima Main Line	19,049	100
19	4	21	Nippō Main Line	36,210	9.5527
20	21	22	Nippō Main Line	36,210	14.2265
21	22	23	Nippō Main Line	22,157	13.8341
22	23	24	Nippō Main Line	13,916	36.3534
23	24	25	Nippō Main Line	9,496	33.3208
24	25	26	Nippō Main Line	14,618	30.4126
25	26	27	Nippō Main Line	14,618	27.5841
26	27	28	Nippō Main Line	10,027	23.8936
27	28	29	Nippō Main Line	10,027	20.4303
28	29	30	Nippō Main Line	19,722	23.0324
29	30	19	Nippō Main Line	19,722	29.6962
30	31	5	Chikuhō Main Line	12,992	100
31	5	32	Chikuhō Main Line	11,693	0.0157
32	32	33	Chikuhō Main Line	11,693	0.0036
33	33	9	Chikuhō Main Line	16,192	0.0044
34	7	34	Sasaguri Line	55,605	0.0176
35	34	35	Sasaguri Line	55,605	0.0075
36	35	33	Sasaguri Line	36,718	0.0001
37	36	6	Kashii Line	8,247	100
38	6	34	Kashii Line	6,794	0.0044
39	34	37	Kashii Line	6,794	100
40	21	38	Hitahikosan Line	467	0.0433
41	38	39	Hitahikosan Line	467	0.0060
42	32	38	Gotōji Line	1,272	0.0097
43	8	40	Chikuhi Line	56,652	25.5463
44	40	41	Chikuhi Line	10,362	14.3275
45	41	42	Chikuhi Line	4,698	5.3519
46	42	43	Chikuhi Line	4,276	100
47	42	44	Chikuhi Line	14,691	0.5051
48	44	45	Chikuhi Line	14,691	100
49	10	46	Nagasaki Main Line	43,551	0.1395
50	46	47	Nagasaki Main Line	43,551	58.5357
51	47	48	Nagasaki Main Line	33,426	55.3505
52	48	49	Nagasaki Main Line	33,426	100
53	49	50	Nagasaki Main Line	21,514	6.5767
54	50	51	Nagasaki Main Line	31,097	100
55	51	52	Nagasaki Main Line	31,097	0.6879

Table 25. The results of algebraic connectivity-based vulnerability with passenger-weighted edges (Cont.)

No.	From node	To node	Line	Daily Ridership (Passenger/day)	Algebraic connectivity-based vulnerability (%)
56	52	53	Nagasaki Main Line	31,097	100
57	51	54	Nagasaki Main Line (Old)	4,484	0.0383
58	54	52	Nagasaki Main Line (Old)	4,484	0.0095
59	48	44	Karatsu Line	1,843	0.0000
60	49	55	Sasebo Line	5,994	0.2390
61	55	56	Sasebo Line	5,994	100
62	50	55	Ōmura Line	4,712	0.0037
63	11	39	Kyūdai Main Line	5,937	0.0426
64	39	57	Kyūdai Main Line	5,937	0.2975
65	57	58	Kyūdai Main Line	4,218	0.4727
66	58	23	Kyūdai Main Line	4,624	0.7151
67	13	59	Hōhi Main Line	15,049	0.0066
68	59	60	Hōhi Main Line	3,584	0.0788
69	60	61	Hōhi Main Line	3,680	0.1874
70	61	62	Hōhi Main Line	4,501	0.3564
71	62	23	Hōhi Main Line	7,355	0.7044
72	14	63	Misumi Line	1,187	100
73	15	64	Hisatsu Line	799	9.3204
74	64	65	Hisatsu Line	491	0.8246
75	65	30	Hisatsu Line	990	0.0868
76	65	28	Kitto Line	451	0.2842
77	27	66	Nichinan Line	4,474	100
78	66	67	Nichinan Line	1,874	100
79	67	68	Nichinan Line	940	100
80	66	69	Miyazaki Kūkō Line	1,854	100
81	18	70	Ibusuki Makurazaki Line	11,530	100
82	70	71	Ibusuki Makurazaki Line	5,589	100
83	71	72	Ibusuki Makurazaki Line	3,461	100
84	73	3	San'yō Shinkansen	56,365	100
85	3	8	Shinkansen	56,365	0.0644
86	8	46	Shinkansen	27,046	0.0056
87	46	11	Shinkansen	27,046	0.2543
88	11	13	Shinkansen	27,046	0.9035
89	13	16	Shinkansen	12,473	51.8802
90	16	18	Shinkansen	12,473	0.6823

From the results in Fig. 49, the algebraic connectivity-based vulnerability with passenger-weighted link analysis was similar to the pure topology analysis. The branch lines without detour routes still had a 100% vulnerability. The section between Shin-Tosu and Kubota stations and Kubota and Hizen-Yamaguchi stations of the JR Kyushu Nagasaki Main Line still had significant average vulnerability with a 56.94% and 100%, respectively. Moreover, this result also showed newly critical sections on the mainlines, such as the section between Kumamoto and Uto stations of the JR Kyushu Kagoshima Main Line, which had an average vulnerability of 79.19%, or the section between Kumamoto and Sendai stations of the Shinkansen line, which had an average vulnerability of 51.88%.

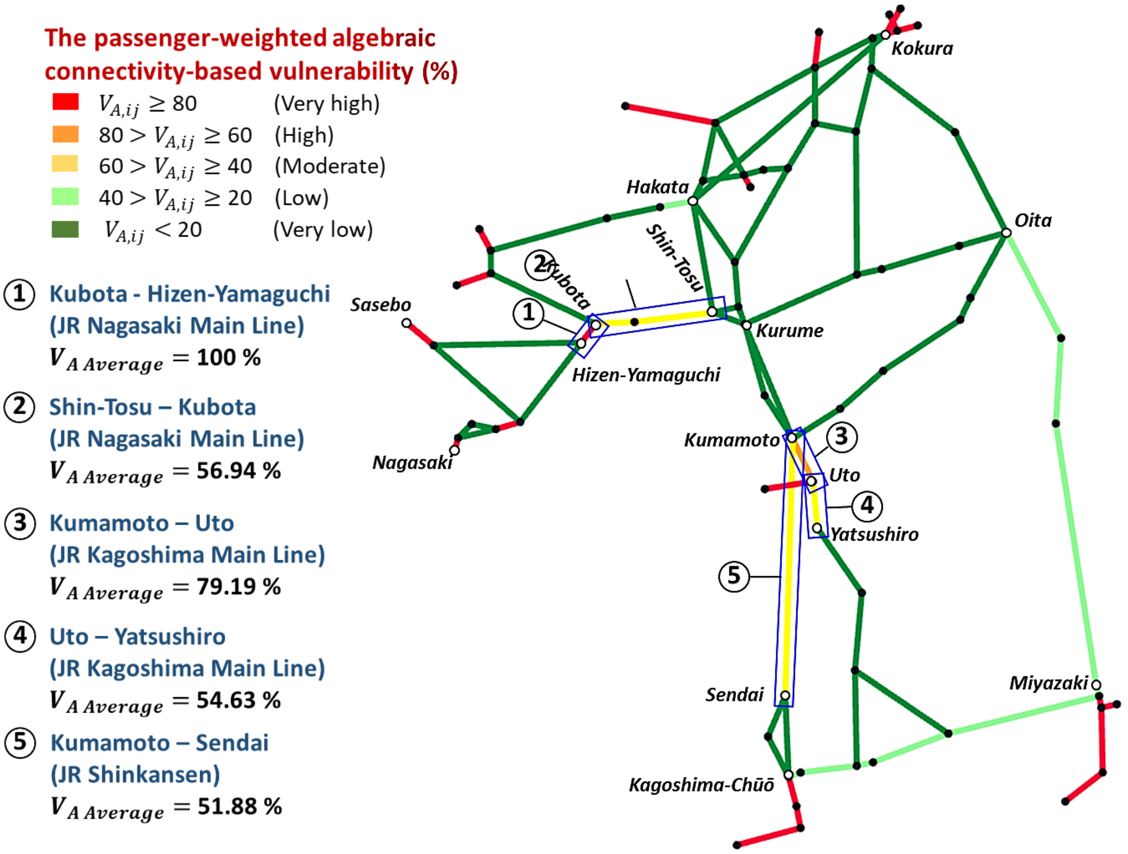


Figure 49. The algebraic connectivity-based vulnerability with passenger-weighted link of the simple JR Kyushu railway network

However, if we consider the relationship between daily ridership and vulnerability, both factors did not correspond due to the correlation coefficient of only -0.17. This condition shows the possibility that the topology characteristic has more influence than the number of passengers on each pair of nodes. In addition, this analysis still needs more accurate and cover data to analyze, especially the electronic data, which collect from IC cards and mobile phones from every operator in Kyushu.

References

[1] JR Kyushu (2021), 線区別ご利用状況. (https://www.jrkyushu.co.jp/company/info/data/pdf/2019senku.pdf; Accessed April 16, 2021) (In Japanese)

[2] JR West (2021), 区間別平均通過人員および旅客運輸収入（2019年度). (https://www.westjr.co.jp/company/info/issue/data/pdf/data2019_08.pdf; Accessed April 21, 2021) (In Japanese)

[3] Sun, D. J., & Guan, S. (2016) Measuring vulnerability of urban metro network from line operation perspective. *Transportation Research Part A: Policy and Practice,* 94, 348–359. https://doi.org/10.1016/j.tra.2016.09.024

Chapter 7

Stochastic Block Model Analysis

7.1 Overview of Weighted Stochastic Block Model (WSBM)

From previous reviews, the infinite relational model (IRM) illustrates the properties and concept of classifying the group of node clusters as a simple method. This model also is the basis of the stochastic block model (SBM), which has a similar concept and purpose in the larger scale network. However, the network clustering analysis can use more accurate methods.

In this research, we used an advanced model, the weighted stochastic block model (WSBM), which can apply to the passenger or flow on each edge of the network in the future (Aicher et al., 2015; Aicher et al., 2013; Aicher, 2014). This criterion has been explained by Aicher et al. (2013) that WSBM can be included as a special case in most standard distributional forms. Moreover, it can use weighted relations directly in recovering latent block structure, preventing the information loss caused by thresholding. However, the adjacency matrix in this research was obtained from the case study railway networks instead of the randomly created matrix, such as in another research. We expect the result to show the community structure, in which vertices in a group maintain the same probabilistic connectivity as the stochastic block model (SBM). If pairs of nodes come from different groups, the probability of its link will be low (Lee and Wilkinson, 2019).

The definition of the weighted stochastic block model can explain by giving a_{ij} as the element of the adjacency matrix A between nodes i and j that $a_{ij} \in \{0,1\}$. z is the vector obtained from the cluster of each vertex $z_i \in \{1, \dots, K\}$. K is the cluster group or block. $\theta_{z_i z_j}$ is the probability of the edge between z_i and z_j. The additional parameter of edge weights for WSBM is μ denoted as a random variable distributed. Unlike the SBM, which used Bernoulli distribution, WSBM can use other distribution, such as normal distribution. In the example, the basic likelihood function with edge weights model from a normal distribution that $\theta_{z_i z_j} = (\mu_{z_i z_j}, \sigma^2_{z_i z_j})$ can be

$$\mathrm{P}(A|z,\mu,\sigma^2) = \prod_{ij} exp\left(a_{ij}\frac{\mu_{z_i z_j}}{\sigma^2_{z_i z_j}} - a_{ij}^2\frac{1}{2\sigma^2_{z_i z_j}} - 1\frac{\mu^2_{z_i z_j}}{\sigma^2_{z_i z_j}}\right) \tag{7.1}$$

7.2 The Simple Result of Weighted Stochastic Block Model for Case Studies

The testing for the Kyushu railway, Tokyo subway, and Osaka subway networks still used a pure topology adjacency matrix. That is because the passenger flow data from every operator is still not yet available, and the research time is limited. To analyze the node clusters group, we tested using three and four-cluster group cases (k = 3 and k = 4) under the same adjacency matrix that was used for centrality and vulnerability analyses. The testing used a mixed model in which the edge's weight distribution is the normal distribution, while the edge distribution is the Bernoulli distribution. The calculating process was conducted by MATLAB with programming code based on the work of Aicher (2014). The primary result of the Kyushu railway network's cluster is illustrated in Table 26.

Table 26. Clusters group of the Kyushu railway network WSBM analysis

Node	**Station**	**Cluster k = 3**	**Cluster k = 4**	**Node**	**Station**	**Cluster k = 3**	**Cluster k = 4**
1	Mojikō	1	4	45	Yayoigaoka	2	3
2	Komorie	1	4	46	Tashiro	2	2
3	Moji	1	4	47	Tosu	2	1
4	Kokura	1	4	48	Hizen-Asahi	2	1
5	Nishi-Kokura	3	4	49	Kurume	2	1
6	Kyūshūkōdaimae	2	4	50	Araki	2	1
7	Tobata	2	4	51	Nishimuta	2	1
8	Edamitsu	2	4	52	Hainuzuka	2	1
9	Space World	2	4	53	Chikugo-Funagoya	3	1
10	Yahata	2	4	54	Setaka	1	1
11	Kurosaki	2	4	55	Minami-Setaka	1	1
12	Jinnoharu	3	4	56	Wataze	1	2
13	Orio	1	2	57	Yoshino	1	4
14	Mizumaki	1	4	58	Ginsui	1	4
15	Ongagawa	1	4	59	Ōmuta	3	4
16	Ebitsu	1	4	60	Arao	1	4
17	Kyōikudaimae	1	4	61	Minami-Arao	1	4
18	Akama	1	4	62	Nagasu	1	4
19	Tōgō	1	4	63	Ōnoshimo	1	4
20	Higashi-Fukuma	1	4	64	Tamana	1	4
21	Fukuma	1	2	65	Higo-Ikura	1	4
22	Chidori	1	3	66	Konoha	1	4
23	Koga	1	3	67	Tabaruzaka	1	4
24	Shishibu	1	3	68	Ueki	1	4
25	Shingū-Chūō	1	3	69	Nishisato	1	4
26	Fukkōdaimae	1	3	70	Sōjōdaigakumae	1	4
27	Kyūsandaimae	1	2	71	Kami-Kumamoto	1	4
28	Kashii	1	1	72	Kumamoto	1	4
29	Chihaya	1	1	73	Nishi-Kumamoto	1	4
30	Hakozaki	1	1	74	Kawashiri	1	4
31	Yoshizuka	1	1	75	Tomiai	1	4
32	Hakata	1	2	76	Uto	1	4
33	Takeshita	1	3	77	Matsubase	1	2
34	Sasabaru	1	3	78	Ogawa	1	1
35	Minami-Fukuoka	1	3	79	Arisa	1	1
36	Kasuga	1	3	80	Senchō	1	1
37	Ōnojō	1	3	81	Shin-Yatsushiro	3	2
38	Mizuki	1	3	82	Yatsushiro	2	4
39	Tofurōminami	1	3	83	Sendai	1	3
40	Futsukaichi	1	3	84	Kumanojō	1	3
41	Tenpaizan	1	3	85	Kobanchaya	1	3
42	Haruda	1	3	86	Kushikino	1	3
43	Keyakidai	3	3	87	Kamimuragakuenmae	1	3
44	Kiyama	2	3	88	Ichiki	1	3

Table 26. Clusters group of the Kyushu railway network WSBM analysis (Cont.)

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
89	Yunomoto	1	3	146	Usuki	2	3
90	Higashi-Ichiki	1	3	147	Tsukumi	2	3
91	Ijūin	1	3	148	Hishiro	2	3
92	Satsuma-Matsumoto	1	3	149	Azamui	2	3
93	Kami-Ijūin	1	3	150	Kariu	2	3
94	Hiroki	1	3	151	Kaizaki	2	3
95	Kagoshima-Chūō	3	3	152	Saiki	3	3
96	Kagoshima	1	3	153	Kamioka	1	3
97	Shimonoseki	1	4	154	Naomi	1	3
98	Minami-Kokura	1	4	155	Naokawa	1	3
99	Jōno	1	4	156	Shigeoka	1	3
100	Abeyamakōen	1	4	157	Sōtarō	1	3
101	Shimosone	1	4	158	Ichitana	1	3
102	Kusami	1	4	159	Kitagawa	1	3
103	Kanda	1	4	160	Hyūga-Nagai	1	3
104	Obase Nishikōdai-mae	1	4	161	Kita-Nobeoka	1	3
105	Yukuhashi	1	4	162	Nobeoka	1	3
106	Minami-Yukuhashi	1	4	163	Minami-Nobeoka	1	3
107	Shindenbaru	1	2	164	Asahigaoka	1	3
108	Tsuiki	1	1	165	Totoro	1	3
109	Shiida	1	1	166	Kadogawa	1	3
110	Buzen-Shōe	1	1	167	Hyūgashi	1	3
111	Unoshima	1	1	168	Zaikōji	1	3
112	Mikekado	1	1	169	Minami-Hyūga	1	3
113	Yoshitomi	1	1	170	Mimitsu	1	3
114	Nakatsu	1	1	171	Higashi-Tsuno	1	3
115	Higashi-Nakatsu	1	1	172	Tsuno	1	3
116	Imazu	1	1	173	Kawaminami	1	3
117	Amatsu	1	1	174	Takanabe	1	3
118	Buzen-Zenkōji	1	1	175	Hyūga-Shintomi	1	3
119	Yanagigaura	1	1	176	Sadowara	1	3
120	Buzen-Nagasu	1	1	177	Hyūga-Sumiyoshi	1	3
121	Usa	1	1	178	Hasugaike	1	3
122	Nishi-Yashiki	1	1	179	Miyazaki-Jingū	1	3
123	Tateishi	1	1	180	Miyazaki	1	3
124	Naka-Yamaga	1	1	181	Minami-Miyazaki	1	2
125	Kitsuki	3	1	182	Kanō	1	3
126	Ōga	2	2	183	Kiyotake	1	3
127	Hiji	2	4	184	Hyūga-Kutsukake	1	3
128	Yōkoku	2	4	185	Tano	1	3
129	Bungo-Toyooka	2	4	186	Aoidake	1	2
130	Kamegawa	2	4	187	Yamanokuchi	1	4
131	Beppu-Daigaku	2	4	188	Mochibaru	1	4
132	Beppu	2	4	189	Mimata	1	4
133	Higashi-Beppu	2	4	190	Miyakonojō	3	4
134	Nishi-Ōita	2	4	191	Nishi-Miyakonojō	1	4
135	Ōita	2	4	192	Isoichi	1	4
136	Maki	2	4	193	Takarabe	1	4
137	Takajō	2	4	194	Kitamata	1	4
138	Tsurusaki	2	4	195	Ōsumi-Ōkawara	1	4
139	Ōzai	2	4	196	Kita-Naganoda	1	4
140	Sakanoichi	2	2	197	Kirishima-Jingū	1	4
141	Kōzaki	2	3	198	Kokubu	1	4
142	Sashiu	2	3	199	Hayato	3	2
143	Shitanoe	2	3	200	Kajiki	1	4
144	Kumasaki	2	3	201	Kinkō	1	4
145	Kami-Usuki	2	3	202	Chōsa	1	4

Table 26. Clusters group of the Kyushu railway network WSBM analysis (Cont.)

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
203	Aira	1	4	260	Tagawa-Ita	1	3
204	Shigetomi	1	4	261	Tagawa-Gotōji	1	3
205	Ryūgamizu	1	2	262	Ikejiri	1	3
206	Wakamatsu	1	3	263	Buzen-Kawasaki	1	3
207	Fujinoki	1	3	264	Nishi-Soeda	1	3
208	Okudōkai	1	3	265	Soeda	1	3
209	Futajima	1	3	266	Kanyūsha-Hikosan	1	3
210	Honjō	1	3	267	Buzen-Masuda	1	3
211	Higashi-Mizumaki	1	1	268	Hikosan	1	2
212	Nakama	1	1	269	Chikuzen-Iwaya	1	4
213	Chikuzen-Habu	1	1	270	Daigyōji	1	4
214	Kurate	1	1	271	Hōshuyama	1	4
215	Chikuzen-Ueki	1	1	272	Ōtsuru	1	2
216	Shinnyū	1	1	273	Imayama	3	1
217	Nōgata	1	1	274	Yoake	2	1
218	Katsuno	1	1	275	Kami-Mio	1	3
219	Kotake	1	2	276	Shimo-Kamoo	1	3
220	Namazuta	1	4	277	Chikuzen-Shōnai	1	3
221	Urata	1	4	278	Funao	1	3
222	Shin Iizuka	1	2	279	Gion	1	4
223	Iizuka	1	4	280	Nakasu-Kawabata	1	4
224	Tentō	1	4	281	Tenjin	1	4
225	Keisen	1	4	282	Akasaka	1	4
226	Kami Honami	1	4	283	Ōhorikōen	1	4
227	Chikuzen Uchino	1	4	284	Tōjinmachi	1	4
228	Chikuzen Yamae	1	2	285	Nishijin	1	4
229	Yusu	1	2	286	Fujisaki	1	4
230	Harumachi	1	4	287	Muromi	1	2
231	Chojabaru	1	4	288	Meinohama	1	1
232	Kadomatsu	1	2	289	Shimoyamato	1	1
233	Sasaguri	1	1	290	Imajuku	1	1
234	Chikuzen-Yamate	1	1	291	Kyūdai-Gakkentoshi	1	1
235	Kido Nanzōin-mae	1	2	292	Susenji	1	1
236	Kurōbaru	1	4	293	Hatae	1	1
237	Chikuzen-Daibu	1	4	294	Itoshima-Kokomae	1	1
238	Saitozaki	1	3	295	Chikuzen-Maebaru	1	1
239	Umi-no-Nakamichi	1	3	296	Misakigaoka	1	1
240	Gannosu	1	3	297	Kafuri	1	1
241	Nata	1	2	298	Ikisan	1	1
242	Wajiro	1	1	299	Chikuzen-Fukae	1	1
243	Kashii-Jingū	1	1	300	Dainyū	1	1
244	Maimatsubara	1	1	301	Fukuyoshi	1	1
245	Doi	1	1	302	Shikaka	1	1
246	Iga	1	2	303	Hamasaki	1	1
247	Sakado	1	4	304	Nijinomatsubara	1	2
248	Sue	1	4	305	Higashi-Karatsu	1	3
249	Sue-Chūō	1	4	306	Watada	1	3
250	Shinbaru	1	4	307	Karatsu	1	3
251	Umi	1	4	308	Nishi-Karatsu	1	3
252	Ishida	1	2	309	Onizuka	1	3
253	Shii-Kōen	1	3	310	Yamamoto	3	3
254	Shii	1	3	311	Hizen-Kubo	2	3
255	Ishiharamachi	1	3	312	Nishi-Ōchi	2	3
256	Yobuno	1	3	313	Sari	2	3
257	Saidōsho	1	3	314	Komanaki	2	3
258	Kawara	1	3	315	Ōkawano	2	3
259	Ipponmatsu	1	3	316	Hizen-Nagano	2	2

Table 26. Clusters group of the Kyushu railway network WSBM analysis (Cont.)

Node	**Station**	**Cluster k = 3**	**Cluster k = 4**	**Node**	**Station**	**Cluster k = 3**	**Cluster k = 4**
317	Momonokawa	2	4	374	Nagao	2	1
318	Kanaishihara	2	4	375	Mimasaka	2	1
319	Kami-Imari	2	4	376	Kami-Arita	2	1
320	Imari	2	4	377	Arita	2	2
321	Shin-Tosu	2	1	378	Mikawachi	2	3
322	Hizen-Fumoto	2	1	379	Haiki	2	3
323	Nakabaru	2	1	380	Daitō	2	3
324	Yoshinogari-Kōen	2	1	381	Hiu	2	3
325	Kanzaki	2	1	382	Sasebo	2	3
326	Igaya	2	1	383	Huis Ten Bosch	2	3
327	Saga	2	1	384	Haenosaki	3	3
328	Nabeshima	2	1	385	Ogushigō	1	3
329	Balloon Saga (seasonal)	2	1	386	Kawatana	1	3
330	Kubota	2	1	387	Sonogi	1	3
331	Ushizu	2	1	388	Chiwata	1	3
332	Hizen-Yamaguchi	2	1	389	Matsubara	1	3
333	Hizen-Shiroishi	2	1	390	Takematsu	1	3
334	Hizen-Ryūō	2	1	391	Suwa	1	3
335	Hizen-Kashima	2	1	392	Ōmura	1	3
336	Hizen-Hama	2	1	393	Iwamatsu	1	2
337	Hizen-Nanaura	2	1	394	Kurume-Kōkōmae	2	1
338	Hizen-Iida	2	1	395	Minami-Kurume	2	1
339	Tara	2	1	396	Kurume-Daigakumae	2	1
340	Hizen-Ōura	2	1	397	Mii	2	1
341	Konagai	2	1	398	Zendōji	2	1
342	Nagasato	2	1	399	Chikugo-Kusano	2	2
343	Yue	2	1	400	Tanushimaru	2	4
344	Oe	2	1	401	Chikugo-Yoshii	2	4
345	Hizen-Nagata	2	1	402	Ukiha	2	4
346	Higashi-Isahaya	2	1	403	Chikugo-Ōishi	2	2
347	Isahaya	3	1	404	Teruoka	2	1
348	Nishi-Isahaya	1	1	405	Hita	2	1
349	Kikitsu	1	1	406	Bungo-Miyoshi	2	1
350	Ichinuno	1	1	407	Bungo-Nakagawa	2	1
351	Hizen-Koga	1	1	408	Amagase	2	1
352	Utsutsugawa	1	1	409	Sugikawachi	2	1
353	Urakami	1	2	410	Kita-Yamada	2	1
354	Nagasaki	1	1	411	Bungo-Mori	2	1
355	Higashisono	1	1	412	Era	2	1
356	Ōkusa	1	1	413	Hikiji	2	1
357	Honkawachi	1	1	414	Bungo-Nakamura	2	1
358	Nagayo	1	2	415	Noya	2	1
359	Kōda	1	4	416	Yufuin	2	1
360	Michinoo	1	4	417	Minami-Yufu	2	1
361	Nishi-Urakami	1	4	418	Yunohira	2	1
362	Ogi	2	1	419	Shōnai	2	1
363	Higashi-Taku	2	1	420	Tenjinyama	2	1
364	Naka-Taku	2	1	421	Onoya	2	1
365	Taku	2	1	422	Onigase	2	1
366	Kyūragi	2	1	423	Mukainoharu	2	1
367	Iwaya	2	1	424	Bungo-Kokubu	2	1
368	Ōchi	2	2	425	Kaku	2	2
369	Honmutabe	2	3	426	Minami-Ōita	2	4
370	Ōmachi	2	1	427	Furugō	2	4
371	Kitagata	2	1	428	Heisei	1	2
372	Takahashi	2	1	429	Minami-Kumamoto	1	3
373	Takeo-Onsen	2	1	430	Shin-Suizenji	1	3

Table 26. Clusters group of the Kyushu railway network WSBM analysis (Cont.)

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
431	Suizenji	3	3	488	Yoshimatsu	2	2
432	Tōkai-Gakuen-mae	2	3	489	Kurino	2	1
433	Tatsutaguchi	2	3	490	Ōsumi-Yokogawa	2	1
434	Musashizuka	2	3	491	Uemura	2	1
435	Hikari no Mori	2	3	492	Kirishima Onsen	2	1
436	Sanrigi	2	3	493	Kareigawa	2	1
437	Haramizu	2	3	494	Naka-fukura	2	1
438	Higo-Ōzu	2	2	495	Hyōkiyama	2	1
439	Seta	2	4	496	Hinatayama	2	1
440	Tateno	2	4	497	Tsurumaru	2	4
441	Akamizu	2	4	498	Kyōmachi Onsen	2	4
442	Ichinokawa	2	2	499	Ebino	2	4
443	Uchinomaki	2	1	500	Ebino Uwae	2	4
444	Aso	2	1	501	Ebino Iino	2	4
445	Ikoi-no-Mura	2	1	502	Nishi Kobayashi	2	4
446	Miyaji	2	1	503	Kobayashi	2	4
447	Namino	2	1	504	Hirowara	2	4
448	Takimizu	2	1	505	Takaharu	2	4
449	Bungo-Ogi	2	1	506	Hyūga Maeda	2	4
450	Tamarai	2	1	507	Takasaki Shinden	2	4
451	Bungo-Taketa	2	1	508	Higashi Takasaki	2	4
452	Asaji	2	1	509	Mangatsuka	2	4
453	Ogata	2	1	510	Tanigashira	2	4
454	Bungo-Kiyokawa	2	1	511	Hyūga Shōnai	2	4
455	Miemachi	2	1	512	Tayoshi	1	4
456	Sugao	2	1	513	Minamikata	3	4
457	Inukai	2	2	514	Kibana	2	4
458	Takenaka	2	3	515	Undōkōen	2	4
459	Nakahanda	2	3	516	Sosanji	2	4
460	Ōita-Daigaku-mae	2	3	517	Kodomonokuni	2	4
461	Shikido	2	3	518	Aoshima	2	4
462	Takio	2	2	519	Oryūzako	2	4
463	Midorikawa	1	4	520	Uchiumi	2	2
464	Sumiyoshi	1	4	521	Kouchiumi	2	1
465	Higo-Nagahama	1	4	522	Ibii	2	1
466	Ōda	1	4	523	Kitagō	2	1
467	Akase	1	4	524	Uchinoda	2	2
468	Ishiuchi Dam	1	4	525	Obi	2	3
469	Hataura	1	4	526	Nichinan	2	3
470	Misumi	1	4	527	Aburatsu	2	2
471	Dan	2	4	528	Ōdōtsu	2	1
472	Sakamoto	2	4	529	Nangō	2	1
473	Haki	2	4	530	Taninokuchi	2	1
474	Kamase	2	4	531	Yowara	2	1
475	Setoishi	2	4	532	Hyūga-Ōtsuka	2	1
476	Kaiji	2	4	533	Hyūga-Kitakata	2	1
477	Yoshio	2	4	534	Kushima	2	1
478	Shiroishi	2	4	535	Fukushima-Imamachi	2	1
479	Kyūsendō	2	4	536	Fukushima-Takamatsu	2	1
480	Isshōchi	2	4	537	Ōsumi-Natsui	2	1
481	Naraguchi	2	4	538	Shibushi	2	1
482	Watari	2	2	539	Miyazaki Airport	1	4
483	Nishi Hitoyoshi	2	3	540	Kōrimoto	2	3
484	Hitoyoshi	2	3	541	Minami-Kagoshima	2	3
485	Okoba	2	3	542	Usuki	2	3
486	Yatake	2	3	543	Taniyama	2	3
487	Masaki	2	3	544	Jigenji	2	3

Table 26. Clusters group of the Kyushu railway network WSBM analysis (Cont.)

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
545	Sakanoue	2	3	601	Kushiwara	2	1
546	Goino	2	3	602	Nishitetsu Kurume	2	2
547	Hirakawa	2	3	603	Hanabatake	2	4
548	Sesekushi	2	3	604	Shikenjōmae	3	4
549	Nakamyō	2	3	605	Tsubuku	1	4
550	Kiire	2	3	606	Yasutake	1	4
551	Maenohama	2	3	607	Daizenji	1	4
552	Nukumi	2	3	608	Mizuma	1	4
553	Satsuma-Imaizumi	2	3	609	Inuzuka	1	4
554	Miyagahama	2	3	610	Ōmizo	1	4
555	Nigatsuden	2	3	611	Hatchōmuta	1	4
556	Ibusuki	2	3	612	Kamachi	1	4
557	Yamakawa	2	3	613	Yakabe	1	4
558	Ōyama	2	3	614	Nishitetsu Yanagawa	1	4
559	Nishi-Ōyama	2	3	615	Tokumasu	3	4
560	Satsuma-Kawashiri	2	3	616	Shiotsuka	2	4
561	Higashi-Kaimon	2	3	617	Nishitetsu Nakashima	2	4
562	Kaimon	2	3	618	Enoura	2	4
563	Irino	2	2	619	Hiraki	2	4
564	Ei	2	4	620	Nishitetsu Wataze	2	4
565	Nishi-Ei	2	4	621	Kuranaga	2	4
566	Goryō	2	4	622	Higashi-Amagi	2	4
567	Ishikaki	2	4	623	Nishitetsu Ginsui	2	4
568	Mizunarikawa	2	4	624	Shin-Sakaemachi	2	4
569	Ei-Ōkawa	2	4	625	Nishitetsu Gojō	1	3
570	Matsugaura	2	4	626	Dazaifu	1	3
571	Satsuma-Shioya	2	4	627	Gorōmaru	2	1
572	Shirasawa	2	4	628	Gakkōmae	2	1
573	Satsuma-Itashiki	2	4	629	Koganchaya	2	1
574	Makurazaki	2	4	630	Kitano	2	1
575	Shin-Minamata	1	3	631	Ōki	2	2
576	Izumi	1	3	632	Kaneshima	2	4
577	Nishitetsu Fukuoka (Tenjin)	1	3	633	Ōzeki	2	4
578	Yakuin	1	3	634	Hongō	2	4
579	Nishitetsu Hirao	1	3	635	Kamiura	2	4
580	Takamiya	1	3	636	Mada	2	4
581	Ōhashi	1	3	637	Amagi	2	4
582	Ijiri	1	3	638	Kaizuka	1	1
583	Zasshonokuma	1	3	639	Najima	1	1
584	Kasugabaru	1	3	640	Kashii-Miyamae	1	1
585	Shirakibaru	1	3	641	Nishitetsu Kashii	1	1
586	Shimoōri	1	3	642	Kashii-Kaenmae	1	1
587	Tofurōmae	1	3	643	Tōnoharu	1	1
588	Nishitetsu Futsukaichi	1	3	644	Mitoma	1	1
589	Murasaki	1	3	645	Nishitetsu Shingū	1	1
590	Asakuragaidō	1	3	646	Higashi-Hie	1	1
591	Sakuradai	1	3	647	Fukuokakūkō (Airport)	1	1
592	Chikushi	1	3	648	Gofukumachi	1	4
593	Tsuko	1	3	649	Chiyo-Kenchōguchi	1	4
594	Mikunigaoka	1	2	650	Maidashi-Kyūdai-byōin-mae	1	2
595	Mitsusawa	3	4	651	Hakozaki-Miyamae	1	1
596	Ōho	2	4	652	Hakozaki-Kyūdai-mae	1	1
597	Nishitetsu Ogōri	2	4	653	Tenjin-Minami	1	3
598	Hatama	2	2	654	Watanabe-dōri	1	3
599	Ajisaka	2	1	655	Yakuin-ōdōri	1	3
600	Miyanojin	2	1	656	Sakurazaka	1	3

Table 26. Clusters group of the Kyushu railway network WSBM analysis (Cont.)

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
657	Ropponmatsu	1	3	665	Kamo	1	3
658	Befu	1	2	666	Jirōmaru	1	3
659	Chayama	1	1	667	Hashimoto	1	3
660	Kanayama	1	1	668	Shin-Shimonoseki	1	4
661	Nanakuma	1	1	669	Hakata-Minami	3	1
662	Fukudaimae	1	1	670	Shin-Ōmuta	1	2
663	Umebayashi	1	1	671	Shin-Tamana	1	4
664	Noke	1	2				

The WSBM testing result of the Tokyo subway network is illustrated in Table 27.

Table 27. Clusters group of the Tokyo subway network WSBM analysis

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
1	Shibuya	3	4	33	Yotsuya-sanchome	3	1
2	Omote-sando	3	4	34	Yotsuya	3	3
3	Gaiemmae	3	4	35	Kasumigaseki	3	3
4	Aoyama-itchome	3	1	36	Tokyo	3	3
5	Akasaka-mitsuke/Nagatacho	3	3	37	Otemachi	3	3
6	Tameike-sanno/Kokkai-gijidomae	3	1	38	Awajicho/Shin-ochanomizu/Ogawamachi	3	1
7	Toranomon/Toranomon-hills	3	3	39	Ochanomizu	3	3
8	Shimbashi	3	3	40	Hongo-sanchome	3	3
9	Ginza/Ginza-itchome	3	3	41	Korakuen/Kasuga	1	1
10	Kyobashi	3	3	42	Myogadani	3	3
11	Nihombashi	3	3	43	Shin-otsuka	3	3
12	Mitsukoshimae	3	3	44	Ikebukuro	1	1
13	Kanda	3	3	45	Naka-meguro	2	2
14	Suehirocho	3	3	46	Ebisu	2	2
15	Ueno-hirokoji/Ueno-Okachimachi/Naka-okachimachi	3	3	47	Hiro-o	1	1
16	Ueno	1	3	48	Roppongi	3	3
17	Inaricho	2	3	49	Kamiyacho	3	3
18	Tawaramachi	2	3	50	Hibiya/Yurakucho	3	1
19	Asakusa	1	1	51	Higashi-ginza	3	3
20	Ogikubo	2	2	52	Tsukiji/Shintomicho	3	3
21	Minami-asagaya	2	2	53	Hatchobori	3	3
22	Shin-koenji	2	2	54	Kayabacho	3	3
23	Higashi-koenji	2	2	55	Ningyocho/Suitengumae	3	3
24	Shin-nakano	2	1	56	Kodemmacho	3	3
25	Honancho	2	2	57	Akihabara/Iwamotocho	3	3
26	Nakano-fujimicho	2	2	58	Iriya	2	3
27	NakanoShimbashi	2	1	59	Minowa	2	3
28	Nakano-sakaue	1	4	60	Minami-senju	2	3
29	Nishi-shinjuku	3	4	61	Kita-senju	2	1
30	Shinjuku/Shinjuku-nishiguchi	3	4	62	Nakano	2	4
31	Shinjuku-sanchome	3	4	63	Ochiai	2	4
32	Shinjuku-gyoemmae	3	4	64	Takadanobaba	2	4

Table 27. Clusters group of the Tokyo subway network WSBM analysis (Cont.)

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
65	Waseda	2	4	120	Shirokanedai	3	3
66	Kagurazaka	1	4	121	Shirokane-takanawa	3	3
67	Iidabashi	3	1	122	Azabu-juban	3	3
68	Kudanshita	3	3	123	Roppongi-itchome	3	3
69	Takebashi	3	3	124	Todaimae	2	2
70	Monzen-nakacho	1	1	125	Hon-komagome	2	2
71	Kiba	2	4	126	Komagome	2	2
72	Toyocho	2	4	127	Nishigahara	2	2
73	Minami-sunamachi	2	4	128	Oji	2	2
74	Nishi-kasai	2	4	129	Oji-kamiya	2	2
75	Kasai	2	4	130	Shimo	2	2
76	Urayasu	2	4	131	Akabane-iwabuchi	2	2
77	Minami-gyotoku	2	4	132	Zoshigaya	3	4
78	Gyotoku	2	4	133	Nishi-waseda	3	4
79	Myoden	2	4	134	Higashi-shinjuku	3	4
80	Baraki-nakayama	2	4	135	Kita-sando	3	4
81	Nishi-funabashi	2	4	136	Nishi-magome	2	4
82	Yoyogi-uehara	3	4	137	Magome	2	4
83	Yoyogi-koen	3	4	138	Nakanobu	2	4
84	Meiji-jingumae	3	4	139	Togoshi	2	4
85	Nogizaka	3	4	140	Gotanda	2	4
86	Akasaka	3	4	141	Takanawadai	2	4
87	Nijubashimae	3	3	142	Sengakuji	1	4
88	Yushima	1	4	143	Mita	3	1
89	Nezu	2	4	144	Daimon	3	3
90	Sendagi	2	4	145	Takaracho	3	3
91	Nishi-nippori	2	4	146	Higashi-nihombashi/Bakuro-yokoyama	3	3
92	Machiya	2	4	147	Asakusabashi	3	3
93	Ayase	2	2	148	Kuramae	3	3
94	Kita-ayase	2	2	149	Honjo-azumabashi	2	4
95	Wakoshi	2	2	150	Shibakoen	3	2
96	Chikatetsu-narimasu	2	2	151	Onarimon	3	2
97	Chikatetsu-akatsuka	2	2	152	Uchisaiwaicho	3	2
98	Heiwadai	2	2	153	Suidobashi	3	3
99	Hikawadai	2	2	154	Hakusan	2	4
100	Kotake-mukaihara	2	2	155	Sengoku	2	4
101	Senkawa	2	2	156	Sugamo	2	4
102	Kanamecho	2	2	157	Nishi-sugamo	2	4
103	Higashi-ikebukuro	3	3	158	Shin-itabashi	2	4
104	Gokokuji	3	3	159	Itabashikuyakushomae	2	4
105	Edogawabashi	3	3	160	Itabashihoncho	2	4
106	Ichigaya	3	1	161	Motohasunuma	2	1
107	Kojimachi	3	3	162	Shimura-sakaue	2	2
108	Sakuradamon	3	3	163	Shimura-sanchome	2	2
109	Tsukishima	1	3	164	Hasune	2	2
110	Toyosu	2	1	165	Nishidai	2	2
111	Tatsumi	2	2	166	Takashimadaira	2	2
112	Shin-kiba	2	2	167	Shin-takashimadaira	2	2
113	Hanzomon	3	3	168	Nishi-takashimadaira	2	2
114	Jimbocho	3	3	169	Akebonobashi	3	4
115	Kiyosumi-shirakawa	3	3	170	Hamacho	3	3
116	Sumiyoshi	1	1	171	Morishita	3	3
117	Kinshicho	2	4	172	Kikukawa	3	3
118	Oshiage	2	4	173	Nishi-ojima	2	2
119	Meguro	3	3	174	Ojima	2	2

Table 27. Clusters group of the Tokyo subway network WSBM analysis (Cont.)

Node	**Station**	**Cluster k = 3**	**Cluster k = 4**	**Node**	**Station**	**Cluster k = 3**	**Cluster k = 4**
175	Higashi-ojima	2	2	189	Shiodome	3	3
176	Funabori	2	2	190	Akabanebashi	3	3
177	Ichinoe	2	2	191	Kokuritsu-Kyōgijō	3	4
178	Mizue	2	2	192	Yoyogi	3	4
179	Shinozaki	2	2	193	Nishi-shinjuku-gochome	3	4
180	Motoyawata	2	2	194	Higashi-Nakano	2	4
181	Tochomae	3	4	195	Nakai	2	4
182	Wakamatsu-kawada	3	4	196	Ochiai-minami-nagasaki	2	4
183	Ushigome-yanagicho	3	4	197	Shin-egota	2	4
184	Ushigome-kagurazaka	3	4	198	Nerima	2	4
185	Shin-okachimachi	3	3	199	Toshimaen	2	4
186	Ryōgoku	3	3	200	Nerima-kasugachō	2	4
187	Kachidoki	3	3	201	Hikarigaoka	2	4
188	Tsukijishijō	3	3				

The WSBM testing result of the Osaka subway network is illustrated in Table 28.

Table 28. Clusters group of the Osaka subway network WSBM analysis

Node	**Station**	**Cluster k = 3**	**Cluster k = 4**	**Node**	**Station**	**Cluster k = 3**	**Cluster k = 4**
1	Esaka	1	3	28	Tenjimbashisuji Rokuchōme	1	4
2	Higashi-Mikuni	1	3	29	Nakazakichō	1	4
3	Shin-Ōsaka	1	1	30	Minami-morimachi	1	1
4	Nishinakajima-Minamigata	1	4	31	Temmabashi	1	2
5	Nakatsu	1	4	32	Tanimachi Yonchōme	1	2
6	Umeda	1	4	33	Tanimachi Rokuchōme	1	2
7	Yodoyabashi (Osaka City Hall)	1	1	34	Tanimachi Kyūchōme	1	2
8	Hommachi (Semba-nishi)	1	2	35	Shitennōji-mae Yūhigaoka	1	2
9	Shinsaibashi	1	2	36	Abeno	3	3
10	Namba	1	2	37	Fuminosato	3	3
11	Daikokuchō	1	2	38	Tanabe	3	3
12	Dōbutsuen-mae (Shinsekai)	1	2	39	Komagawa-Nakano	3	3
13	Tennōji	2	1	40	Hirano	3	3
14	Shōwachō	3	4	41	Kire-Uriwari	3	3
15	Nishitanabe	3	4	42	Deto	3	3
16	Nagai	3	4	43	Nagahara	3	3
17	Abiko	3	4	44	Yaominami	3	3
18	Kitahanada	3	4	45	Nishi-Umeda	1	2
19	Shinkanaoka	3	4	46	Higobashi	1	2
20	Nakamozu	3	4	47	Hanazonochō	2	2
21	Dainichi	3	3	48	Kishinosato	3	2
22	Moriguchi	3	3	49	Tamade	3	1
23	Taishibashi-Imaichi	3	3	50	Kitakagaya	3	3
24	Sembayashi-Omiya	3	3	51	Suminoekōen	3	3
25	Sekime-Takadono	3	3	52	Cosmosquare	3	3
26	Noe-Uchindai	2	1	53	Osakako (Tempozan)	3	3
27	Miyakojima	1	4	54	Asashiobashi	3	3

Table 28. Clusters group of the Osaka subway network WSBM analysis (Cont.)

Node	Station	Cluster k = 3	Cluster k = 4	Node	Station	Cluster k = 3	Cluster k = 4
55	Bentencho	2	3	81	Dome-mae Chiyozaki (Kyocera Dome Osaka)	1	2
56	Kujo	1	1	82	Nishiōhashi	1	2
57	Awaza	1	2	83	Matsuyamachi	1	2
58	Sakaisuji-Hommachi (Semba-higashi)	1	2	84	Tamatsukuri	1	2
59	Morinomiya	2	1	85	Osaka Business Park (Osaka-jo Hall)	3	4
60	Midoribashi	2	4	86	Kyōbashi	3	4
61	Fukaebashi	3	4	87	Gamō-yonchōme	3	4
62	Takaida	3	4	88	Imafuku-Tsurumi	3	4
63	Nagata	3	4	89	Yokozutsumi	3	4
64	Nodahanshin	1	2	90	Tsurumi-ryokuchi	3	4
65	Tamagawa	1	2	91	Kadoma-minami	3	4
66	Nishi-Nagahori	1	2	92	Itakano	3	3
67	Sakuragawa	1	2	93	Zuikō Yonchōme	3	3
68	Nippombashi	1	2	94	Daidō-Toyosato	3	3
69	Tsuruhashi	1	1	95	Shimizu	3	1
70	Imazato	1	4	96	Shimmori-Furuichi	3	4
71	Shin-Fukae	1	4	97	Sekime-Seiiku	3	4
72	Shōji	1	4	98	Shigino	3	4
73	Kita-Tatsumi	1	4	99	Trade Center-mae	3	3
74	Minami-Tatsumi	1	4	100	Nakafuto	3	3
75	Ōgimachi	1	4	101	Port Town-nishi	3	3
76	Kitahama	1	2	102	Port Town-higashi	3	3
77	Nagahoribashi	1	2	103	Ferry Terminal	3	3
78	Ebisuchō (Nippombashi-suji)	1	2	104	Nankō-higashi	3	3
79	Tengachaya	1	2	105	Nankōguchi	3	3
80	Taishō	1	2	106	Hirabayashi	3	3

The results from all three case studies were concluded by the following, as shown in Figs. 50 – 58, which illustrated the clusters by each node.

7.2.1 The weighted stochastic block model results of the Kyushu railway network

From Figs. 50 and 51, the block model square matrix with edge-existence parameters between communities were mixed networks (Faskowitz et al., 2018). That means some node groups assemble as core clusters while some node group groups are random clusters, which have highly probably connected with another group. In the example, In the three-cluster cases (k = 3), the core cluster was groups 1 and 2, which clearly showed as a large group in Fig. 50. Nevertheless, cluster group 3 was scattered and usually located at the border between core clusters. Similarly, in the four-cluster cases (k = 4), clusters 1, 3, and 4 could be considered a core cluster, while group 2 was scattered and located at the border between two core cluster groups.

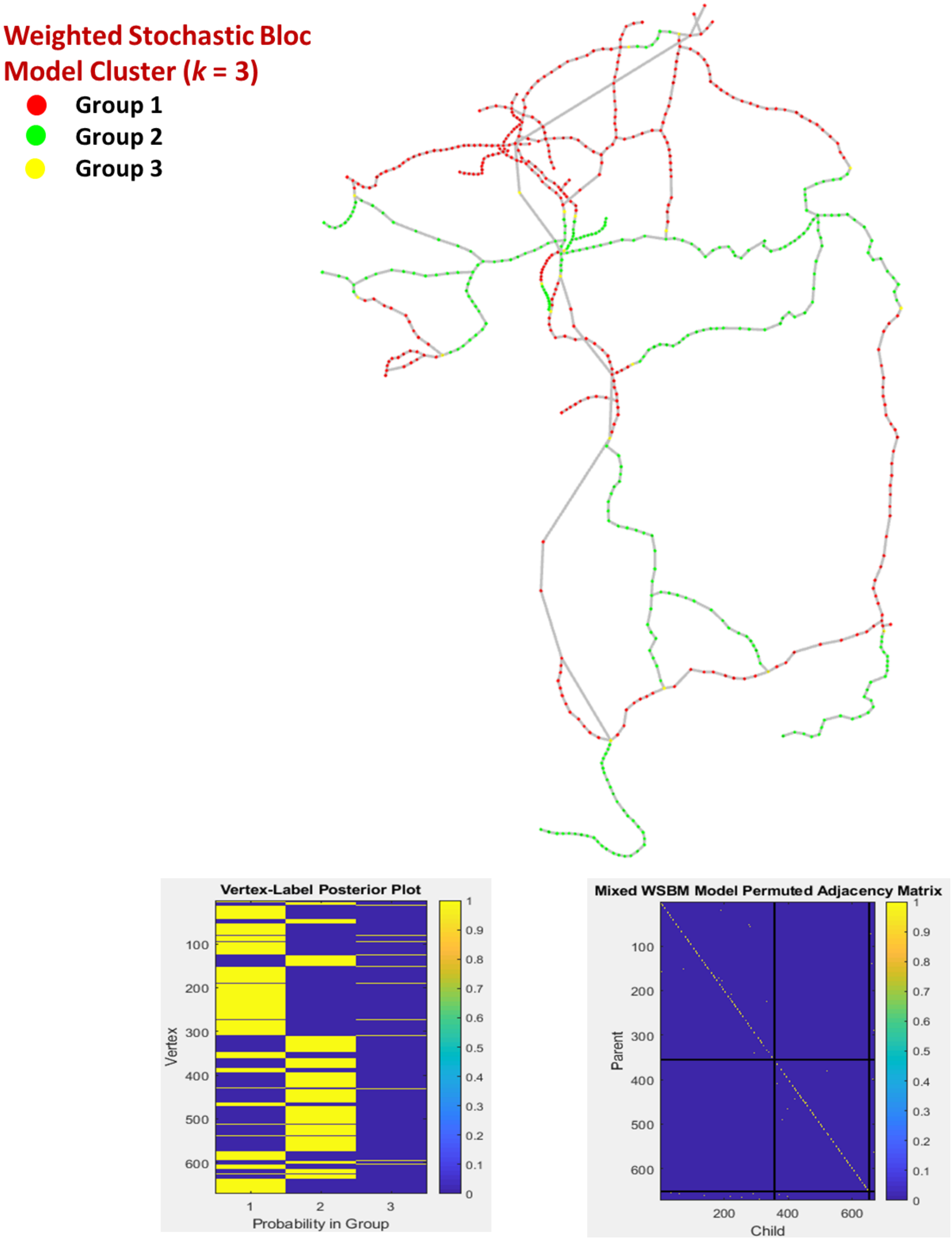


Figure 50. The weighted stochastic block model clustering of the Kyushu railway network with 3 clusters ($k = 3$)

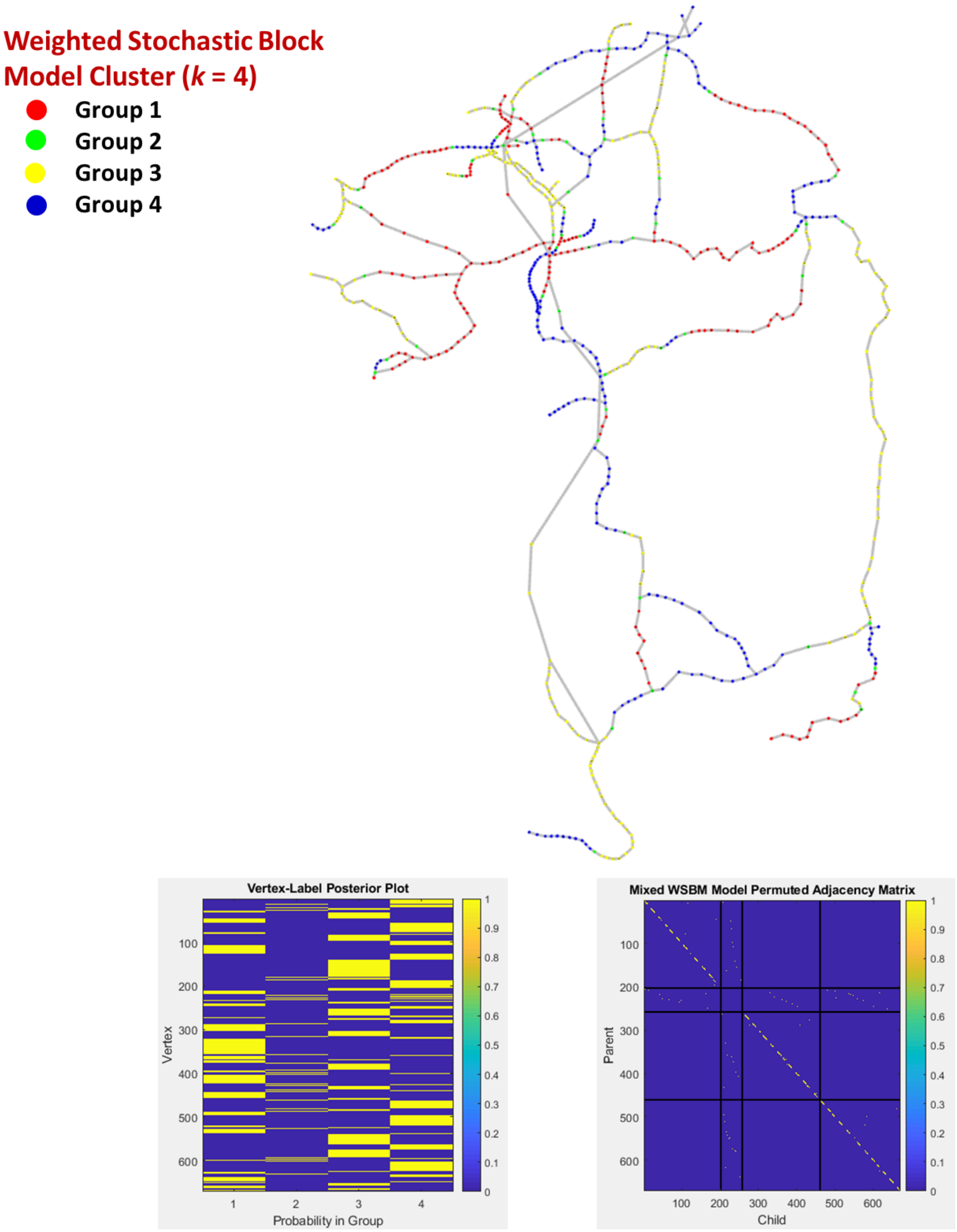


Figure 51. The weighted stochastic block model clustering of the Kyushu railway network with 4 clusters (*k* = 4)

In both cases, if there is at least one node cluster lays its position as the border nodes divide the main clusters or blocks. These nodes can be considered vulnerable nodes, and their links can also be vulnerable edges too. The example border nodes were shown in the k = 3 case model as Fig. 52, that many of the border nodes were located in the central area of the network. These areas compose of several mainline, including the Shinkansen line.

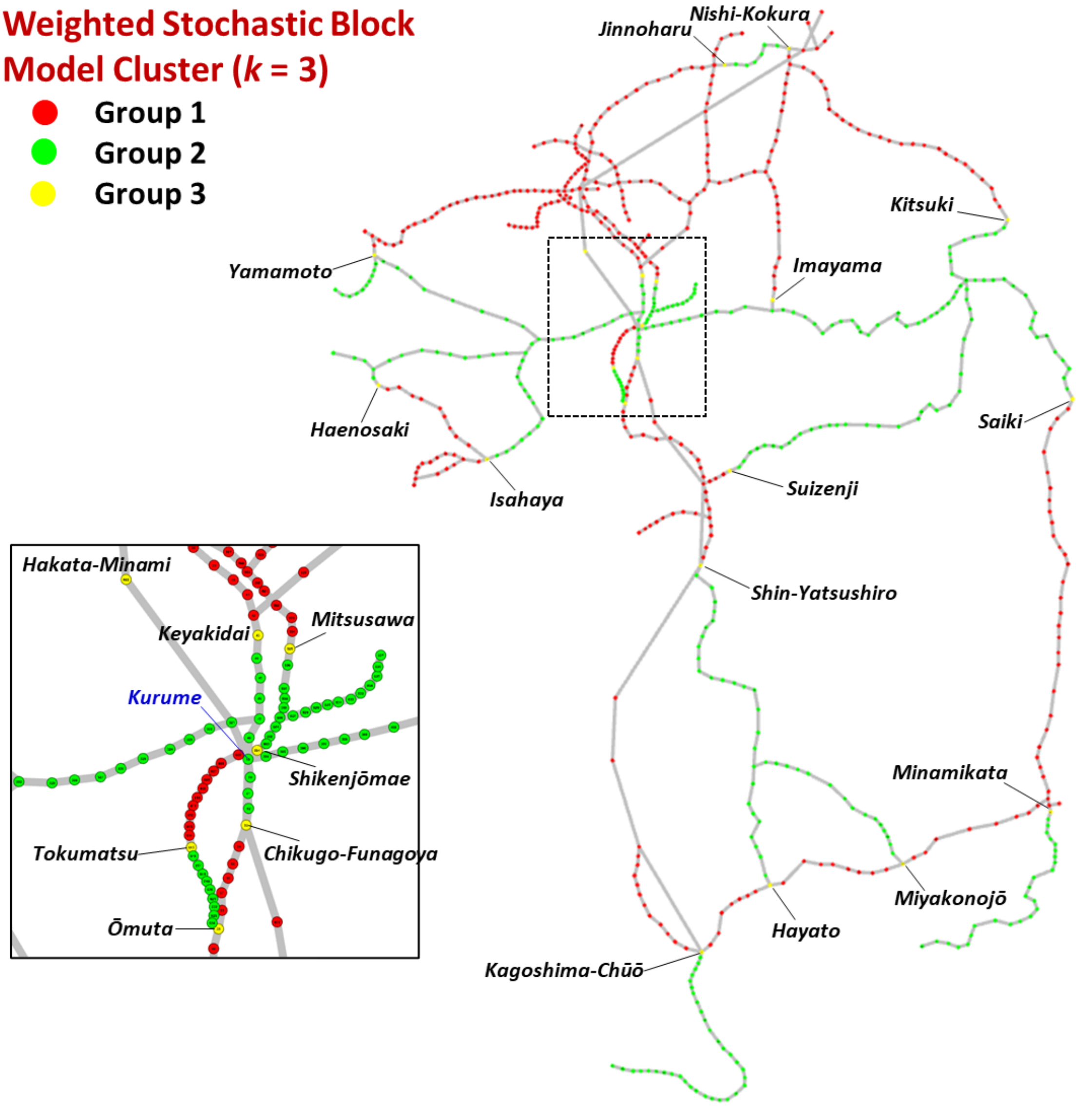


Figure 52. The location of cluster group 3 (in yellow) divided the core clusters 1 and 2 after analyzing the weighted stochastic block model with the three-cluster case of the Kyushu railway network

7.2.2 The weighted stochastic block model results of the Tokyo subway network

In the case of the Tokyo subway network, the block model showed its pattern as a mixed network similar to the Kyushu railway network case. Notice that at least one cluster was the main core cluster located in both three and four-cluster cases networks as illustrated in Figs. 53 – 54.

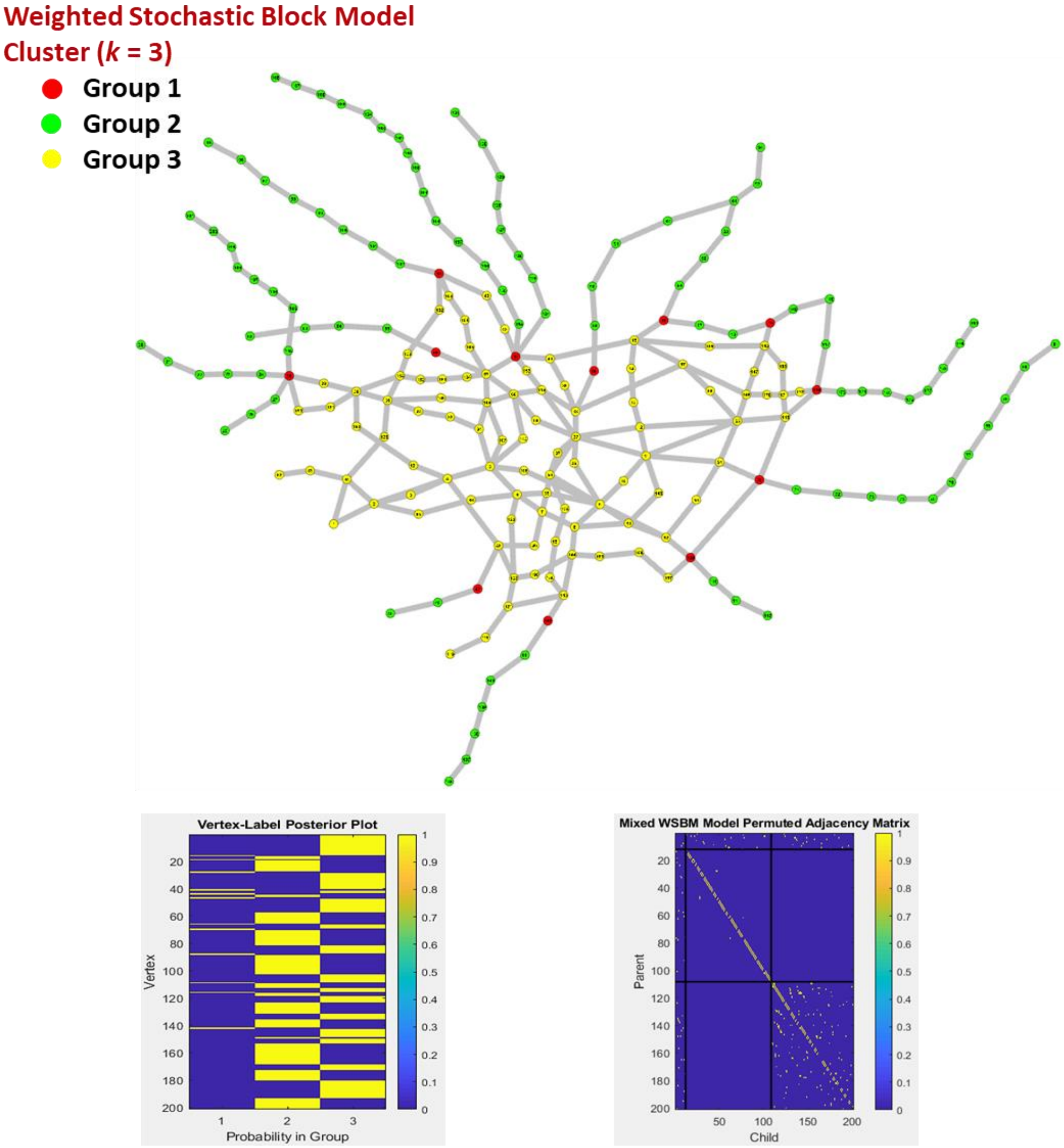


Figure 53. The weighted stochastic block model clustering of the Tokyo subway network with 3 clusters (*k* = 3)

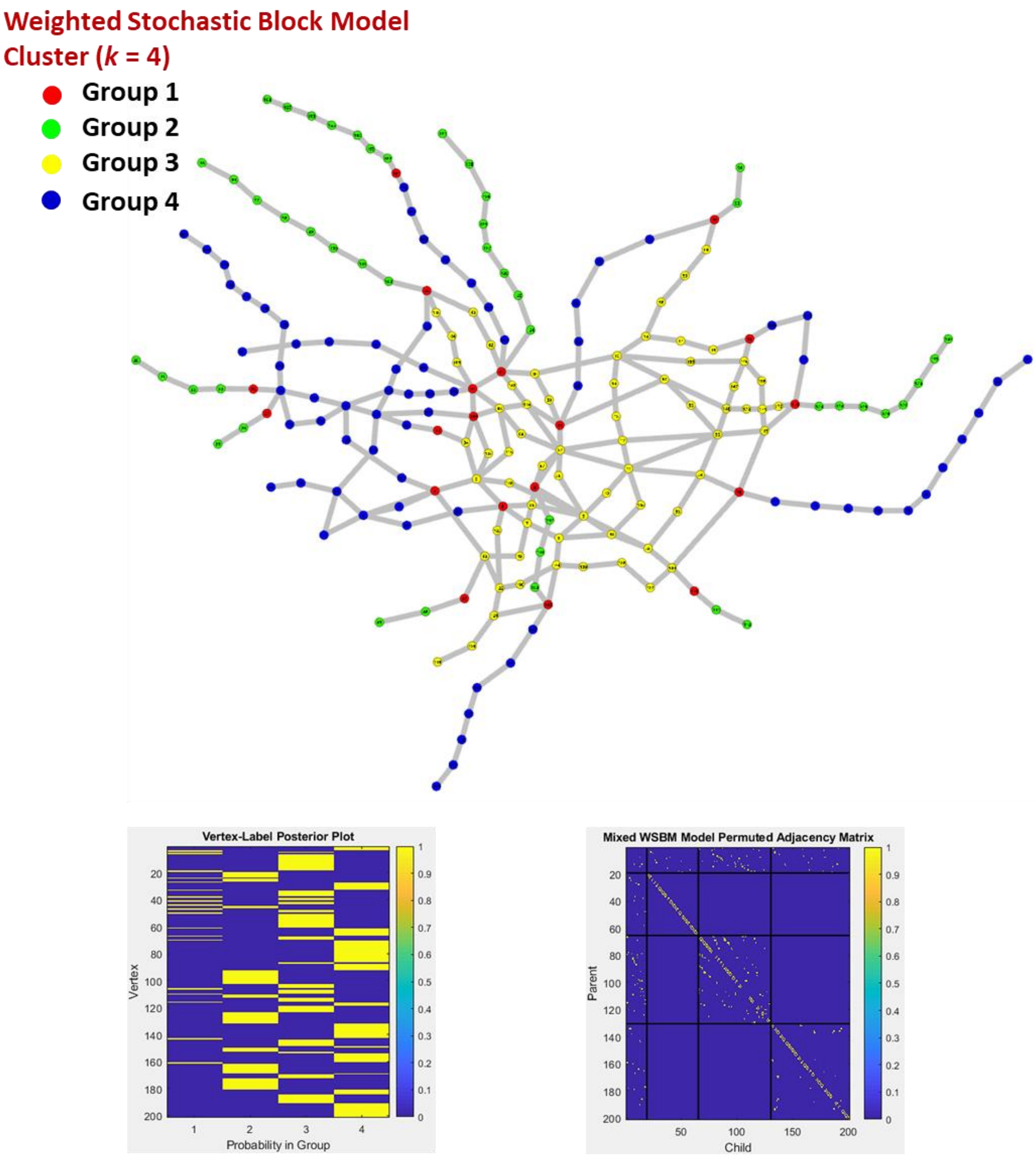


Figure 54. The weighted stochastic block model clustering of the Tokyo subway network with 4 clusters ($k = 4$)

Both cases in Figs. 53 – 54 also showed the border node that divides the main clusters or blocks. These nodes were located at the branches section, which has a no-detour route, and at the junction that connects the no-detour-route sections or the sections from another core cluster group (shown in the four-cluster case in Fig. 54). For example, the three clusters case (k = 3) showed the border nodes of cluster 1, which divides cluster 3, located in the inner area, and cluster 2, which was located on the branch sections in the outer area (shown in Fig. 55). These border nodes from cluster 3 can be considered as the vulnerable or critical nodes as well as their connecting links.

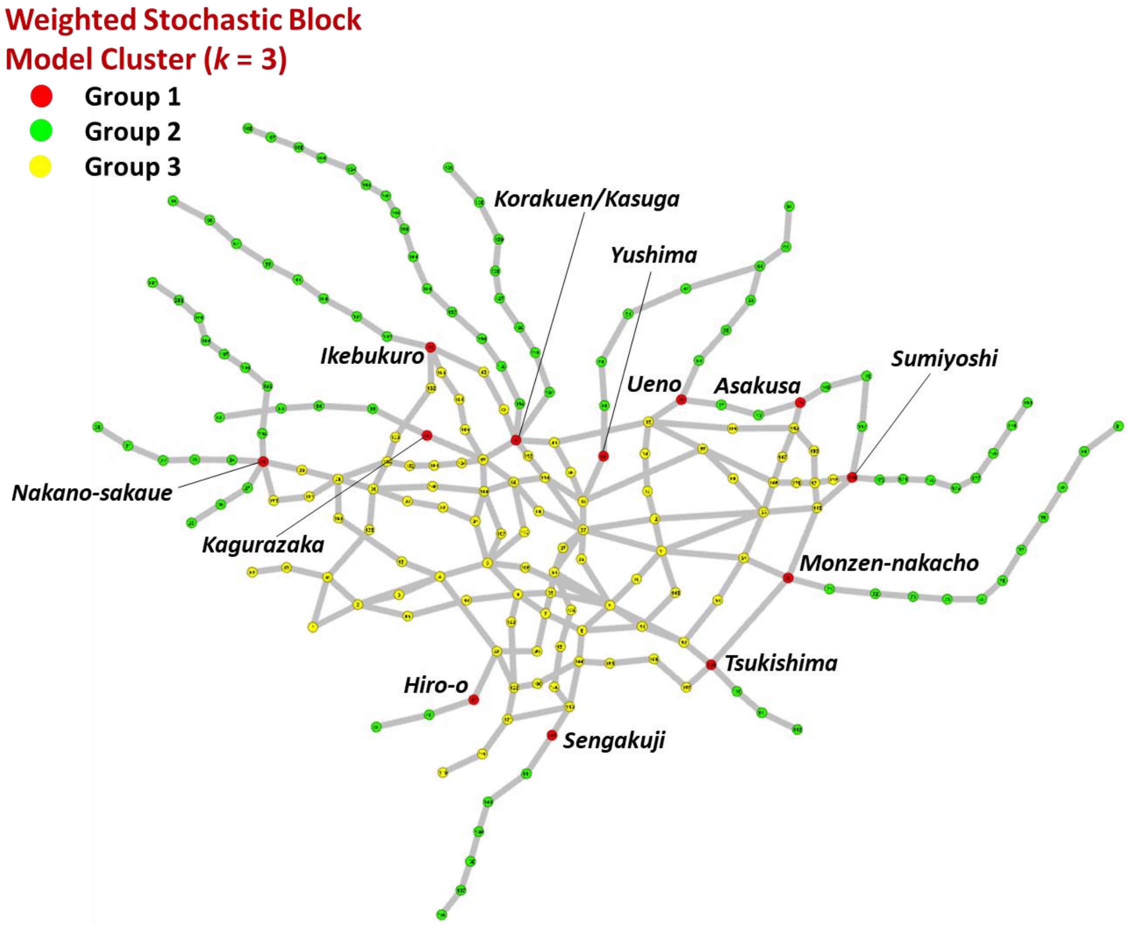


Figure 55. The location of cluster group 1 (in red) divided the core clusters 2 and 3 after analyzing the weighted stochastic block model with the three-cluster case of the Tokyo subway network

7.2.3 The weighted stochastic block model results of the Osaka subway network

The results characteristic of the Osaka subway network after being analyzed by the weighted stochastic block model were very similar to the Tokyo subway network. The results of the Osaka subway network are illustrated in Figs. 56 – 58. In three and four-cluster cases, the mixed network showed the border node between two core cluster groups.

Not only the pattern of the network but the location of border nodes was similar to the Tokyo subway case. The notice in both case studies is that the topology characteristic is composed of several branch sections in the outer area, but in the inner area, the network has strong robustness, and nodes and links are assembled as a grid similar network.

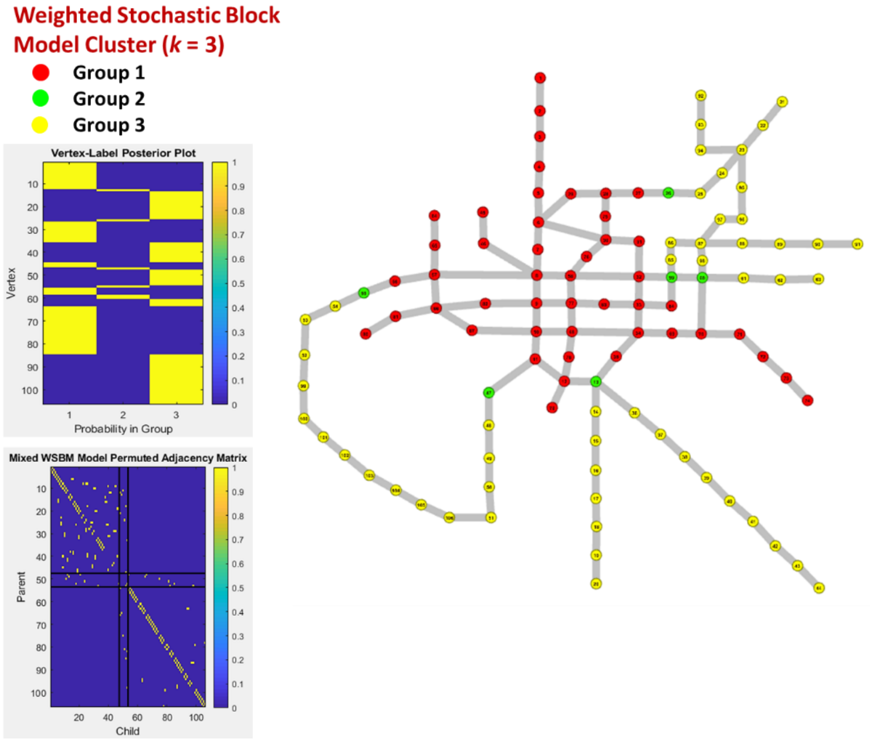


Figure 56. The weighted stochastic block model clustering of the Osaka subway network with 3 clusters (k = 3)

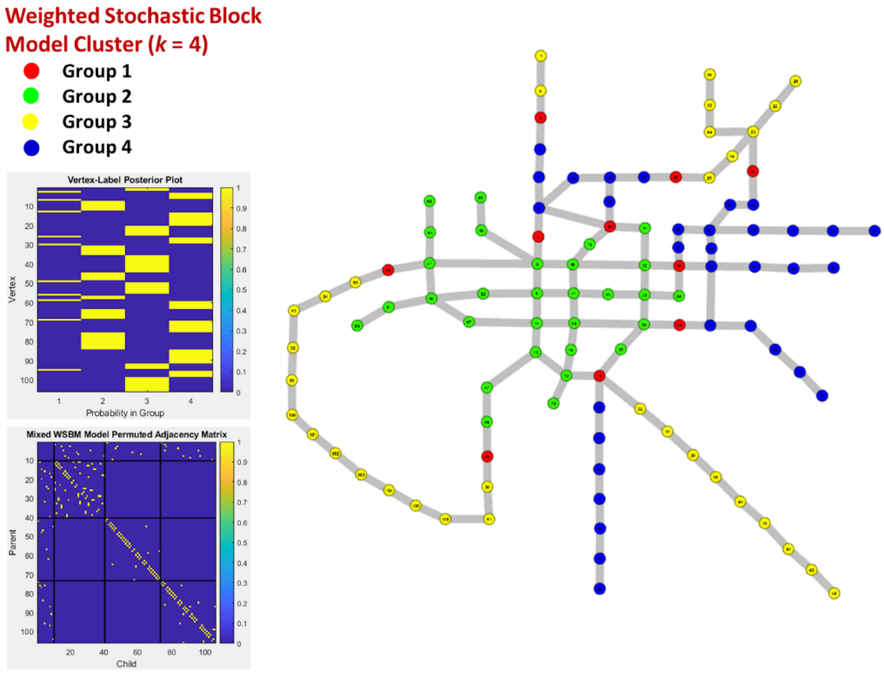


Figure 57. The weighted stochastic block model clustering of the Osaka subway network with 4 clusters (k = 4)

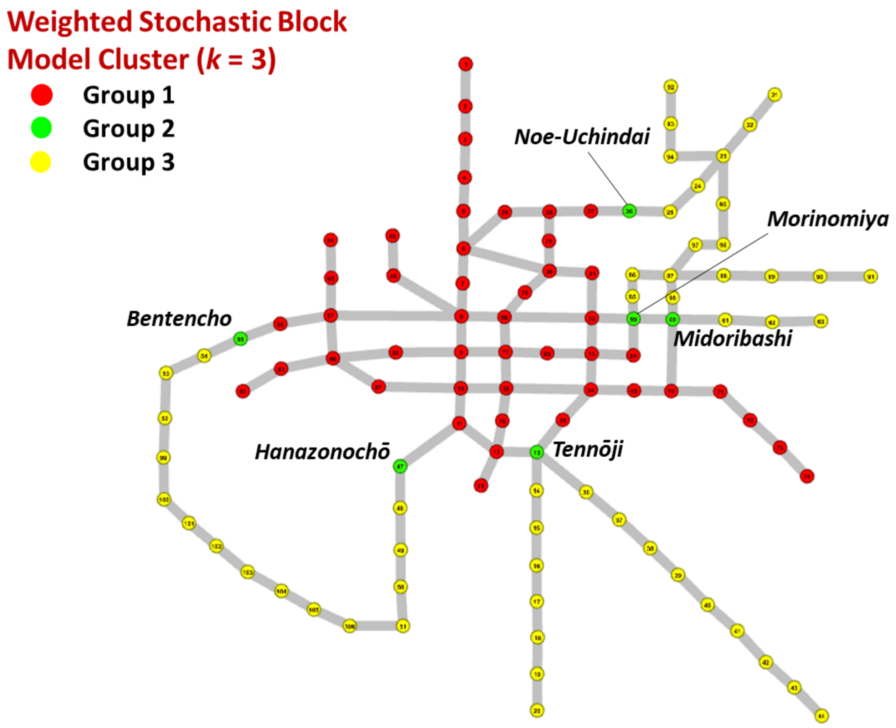


Figure 58. The location of cluster group 2 (in green) divided the core clusters 1 and 3 after analyzing the weighted stochastic block model with the three-cluster case of the Osaka subway network

The weighted stochastic block model illustrated the character of the node cluster group from all three case studies. However, this analysis shows only the primary result to find the simple block cluster. To find the optimized model, the next task in the future is finding the optimal number of cluster groups that can be measured from several methods. In the example, minimum description length, integrated likelihood, approximation thereof, and Bayes factor (Aicher et al., 2015; Aicher et al., 2013).

References

[1] Aicher, C., Jacobs, A. Z., & Clauset, A. (2015). Learning latent block structure in weighted networks. *Journal of Complex Networks*, 3, 221–248. https://doi.org/10.1093/comnet/cnu026

[2] Aicher, C. (2014). WSBMDEMO Demo Script for the WSBM (Version 1.0) [Computer software]. (https://aaronclauset.github.io/wsbm/; Accessed March 17, 2022)

[3] Aicher, C., Jacobs, A. Z., & Clauset, A. (2013). Adapting the Stochastic Block Model to Edge-Weighted Networks. (http://arxiv.org/abs/1305.5782; Accessed March 9, 2022)

[4] Faskowitz, J., Yan, X., Zuo, X. N., & Sporns, O. (2018). Weighted Stochastic Block Models of the Human Connectome across the Life Span. *Scientific Reports*, 8, 12997. https://doi.org/10.1038/s41598-018-31202-1

[5] Lee, C., & Wilkinson, D. J. (2019). A review of stochastic block models and extensions for graph clustering. *Applied Network Science*, 4. https://doi.org/10.1007/s41109-019-0232-2

Chapter 8

Discussions and Conclusions

8.1 Discussions

This thesis analyzed the network critical of the Kyushu railway, Tokyo subway, and Osaka subway networks, mainly using the centrality analysis, the algebraic connectivity-based vulnerability analysis, which compared with the global efficiency-based vulnerability analysis, and the stochastic block model for support vulnerability analysis. The results illustrated the difference and performance of each type of centrality and vulnerability, which showed several advantages and disadvantages, including the optimized condition.

In the case of centrality analysis in chapter 5, the degree centrality analysis did not accurately identify the actual critical node so much. That is because it can identify by only the number of links that connected the neighbor nodes, making it difficult to identify the most important node if there are several highest value centrality nodes with the same number of connected links. An example is the Ginza/Ginza-itchome and Otemachi stations in the Tokyo subway network, which had a degree centrality of 8. For this reason, the degree centrality can point only to the local critical node. If we consider the eigenvector centrality analysis, the most critical and influential station can be shown more clearly, even if it is based on the number of linked neighbors, like degree centrality.

To analyze the flow in the transportation network, closeness centrality, betweenness centrality, and information centrality are better choices for analyzing. However, the closeness centrality showed the disadvantage of the narrow range of centrality value, which is difficult to classify by range of centrality volume and hard to find the most important node if not considering the detail of calculated volume. In the example of the Kyushu network in Fig. 15, the very high centrality nodes located as the large group of cluster nodes, we can identify the specifically important node only by considering the value of centrality. The better method is the betweenness centrality, which measures by number the shortest path of each pair node. The information centrality, which is based on the network efficiency, also showed more specific, very significant centrality nodes in the network, but it has a more complex algorithm and uses much computing time. In the case of closeness and information centralities of the Kyushu railway network, the Kurume station had the highest value of centrality, followed by the Chikugo-Funagoya station, but in the case of betweenness centrality, the rank of both stations was a swap. In addition, The Shin-Tosu station also had a significant value in all three centralities too. Another example case, the Tokyo subway network, also illustrated that the Korakuen/Kasuga station had the highest value in both betweenness and information centralities. The betweenness centrality and information centrality showed the highest correlation coefficient, which is a high probability corresponded to each other. This relation supports the result of Crucitti et al. (2006), who used information centrality to analyze the urban street network and then compared it with the other centrality analyses.

Considering the vulnerability analysis in chapter 6 showed that the algebraic connectivity-based vulnerability analysis presented the very vulnerable and moderate vulnerable section without a detour route. It also showed the significant vulnerable sections, which have a very long-distance detour route if it is cut off. For example, the section between Minami-Miyazaki and Miyakonojō stations of the JR Kyushu Nippō Main Line from the Kyushu network and the section between Osakako (Tempozan) and Awaza stations from the Osaka subway network's Chūō Line. The main reason is the algebraic connectivity property, which considers the second-smallest eigenvector is sensitive to decreasing network connection. Suppose any link is attacked that makes the network separated into at least two parts. In that case, the algebraic connectivity will drop to zero and make this link's vulnerability 100% as Eq. 6.1. This result corresponds to the research of Rodríguez-Núñez and García-Palomares (2014), who explained that the significant vulnerable link located on the position of both branch lines without a detour route and the lines connecting an alternate route but have a long detour distance.

If we tested algebraic connectivity-based vulnerability analysis with the passenger flow in a simple network of the JR Kyushu, the result was similar to the pure topology analysis case on no detour route sections. Also, it showed more significant vulnerable sections that are not shown in the pure topology analysis, such as some sections at the south of Kumamoto station. However, this testing had not shown a clear relationship between vulnerability and daily ridership on every link. This analysis still had a weak point due to the limited data access. Most of the data obtained from the JR Kyushu and JR West official websites are only estimated data and illustrate only the section between major stations. Suppose we need a more accurate result on the passenger-weighted algebraic connectivity-based vulnerability analysis, the next task in the future. In that case, we must use electronic data from all three case study networks that can cover every section on a larger scale.

The results of the global efficiency-based vulnerability from Figs. 33 - 35 were different; the Kyushu network showed the very vulnerable link in the central area, especially the Shinkansen line and the eastern part of the JR Kyushu Nagasaki Main Line between Shin-Tosu and Hizen-Yamaguchi stations. This result was similar to the edge betweenness centrality in that the most critical sections were located in the central area. However, the Tokyo and Osaka subway networks showed that the most vulnerable area was located in the branch sections in the outer area of its network. This condition is assumed because the topology of the Tokyo and Osaka networks is denser and more robust than the Kyushu case. In the Kyushu network, the section between junction nodes still has several nodes as the small station. The global efficiency definition still covers the number of nodes between pairs of measured nodes. Nevertheless, the Tokyo and Osaka subway network has grid pattern networks in the central area, while the branch sections in the outer area are composed of several nodes in the linear form.

The global efficiency-based vulnerability corresponded more with the edge betweenness centrality than algebraic connectivity-based vulnerability in local scale analysis, especially in the Kyushu railway network. The main reason is both global efficiency-based vulnerability, and edge betweenness centrality consider the average shortest path. However, the topology of the Tokyo and Osaka subway network is more robust, so the correlation between both criteria on a local scale was more scattering when analyzed with a scatter plot. In addition, the global efficiency-based vulnerability did not show some specific vulnerable links that are more clearly shown in the algebraic connectivity-based case, such as the section between Minami-Miyazaki and Miyakonojō stations of the JR Kyushu Nippō Main Line from the Kyushu railway network.

From the computing time testing analysis, the algebraic connectivity-based vulnerability used less computing time than global efficiency-based vulnerability, especially the large dense network. However, using the interpreted language program, such as MATLAB in this research, still consume more computing than other compiled languages such as C or C++. For this reason, we considered using the compiled languages for analyzing the computing time in future research.

In comparing the two vulnerability methods, the algebraic connectivity-based vulnerability showed the advantage of less computing time and pointed to the significant vulnerable section in a more specific area. However, this criterion is sensitive to the section with no alternate route; if these sections are removed or cut off, it always has a 100% vulnerability. From this issue, the algebraic connectivity-based vulnerability analysis had some feasibility and is suitable to analyze the network with robust connection and has a dense node and link, such as the urban/commuter railway network and urban street network. However, some network needs to adjust the range of vulnerable value in each level to make it easier to classify the importance of each section after evaluation.

In the stochastic block model (SBM) analysis, this research used the weighted stochastic block model (WSBM), which is easier applied to the passenger or traffic flow as the link weight in future tasks. Due to the research time limit, the testing was conducted with a pure topology adjacency matrix with the three- and four-cluster models. Both model results from all three-case study networks showed the clustering pattern in the mixed model that not every cluster is located in a large group, such as a core cluster. If considering the link between clusters as a vulnerable section, specific nodes from a small cluster were located between two large core clusters similar to the border node. It can consider as the critical node with a vulnerable link that connects the different core clusters.

The example is the three-cluster model results from the Tokyo and Osaka subway networks in Figs. 55 and 58. The results showed the border node divided into two main core clusters and also divided the robust inner area and sparer outer area. However, this research's WSBM result just considered the pure topology network as the primary result. The following solution in the future plan is to find the optimal WSBM model, which calculates the optimal number of clusters by several methods. In the example, minimum description length, integrated likelihood, approximation thereof, and Bayes factor.

From both centrality and vulnerability analyses, the railway operator can consider the priority section or area that needs to be inspected, maintained, and repaired by considering the section that has a high value on both centrality and vulnerability. This concept is similar to the research of Sun and Guan (2016), who used betweenness centrality alongside the global efficiency with passenger-weighted to find the vulnerability section on the critical line. In this thesis, we focused on the betweenness centrality, closeness centrality, and information centrality because they are based on the shortest route analysis and fit to analyze the flow of the railway network.

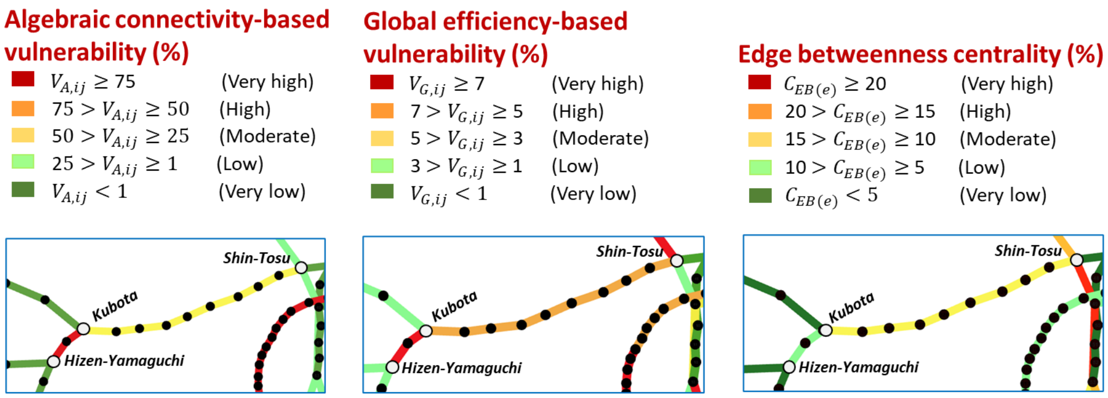


Figure 59. The vulnerability of the section between Shin-Tosu and Hizen-Yamaguchi stations of the JR Kyushu Nagasaki Main Line, Kyushu railway network

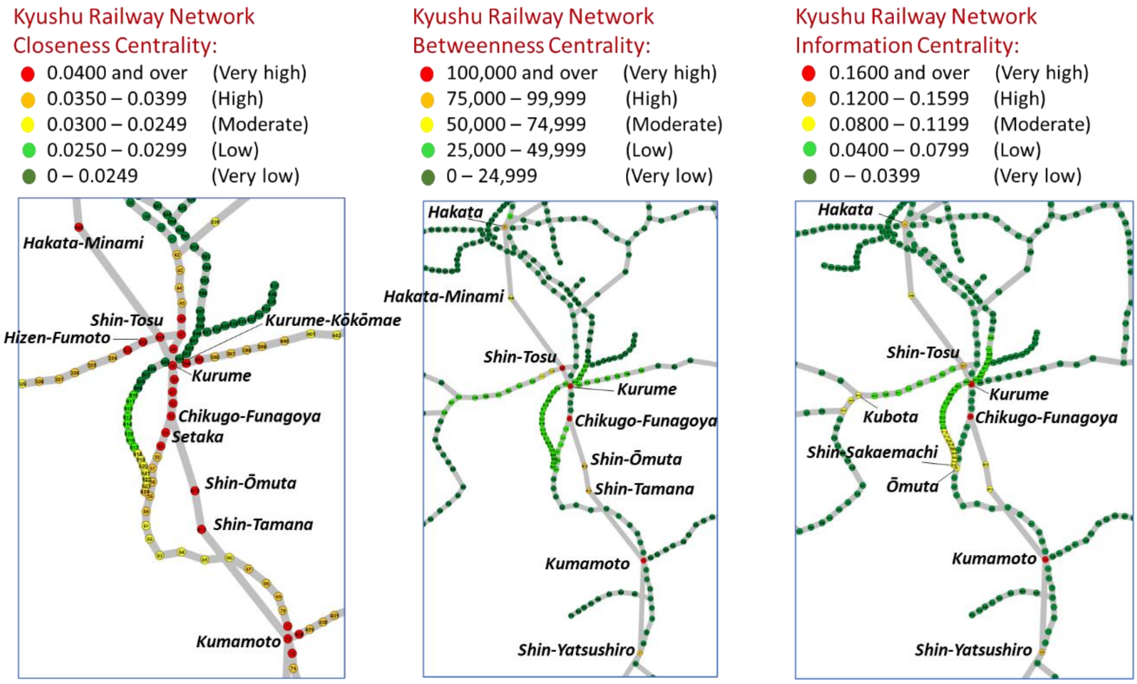


Figure 60. The very high centrality area at the Shin-Tosu, Kurume, and Chikugo-Funagoya stations when considering the closeness centrality, betweenness centrality, and information centrality

In the example, the section between Shin-Tosu and Hizen-Yamaguchi stations of the JR Kyushu Nagasaki Main Line can be considered a very important section for three main reasons. Firstly, this section had significant value on both algebraic connectivity- and global efficiency-based vulnerabilities, as shown in Fig. 59. Moreover, this section also was significantly vulnerable when considering the passenger-weighted algebraic connectivity-based vulnerability analysis. The second, the section between Shin-Tosu and Kubota stations, had a moderate value of edge betweenness centrality. Another reason, this section is connected with the Shin-Tosu station; although this station did not have the highest value in every type of centrality analysis, it still had very high centrality when considering the betweenness centrality and closeness centrality (shown in Fig. 60). In addition, this station had significant value on the information centrality analysis.

The next interesting line that also needs to consider as a high priority for managing is the JR Shinkansen line. This line was not in a significant vulnerable section when considering the algebraic connectivity-based vulnerability due to the high connection of several interchange stations connected with conventional railway lines. However, most of the entire line from Kokura station to Kagoshima-Chūō station had a high or very high volume of global efficiency-based vulnerability and significant edge betweenness centrality. Most of the very high centrality stations were located along this line too. An example is the Kurume station, which also connected the vulnerable section between Shin-Tosu and Hizen-Yamaguchi stations.

From the Kyushu network result, the important area that needs to be considered for management is the area surrounding the Kurume and Shin-Tosu stations. This area has several interchange stations that connect the route to most prefectures in Kyushu except Miyazaki prefecture. In addition, this summary can also be supported by the WSBM analysis, especially in the three-cluster case where many vulnerable nodes or links were located in this area too. The important secondary area is the area surrounding the Hakata station in Fukuoka, which had a very high degree and eigenvector centrality analyses. It is also connected to the JR Shinkansen line.

Another interesting example is the Tokyo subway network's section between the Iidabashi and Korakuen/Kasuga stations of the Toei Oedo Line (shared with the Tokyo Metro Namboku Line). This section also can be considered a high-priority link because it had significant value on both vulnerability analyses and edge betweenness centrality analysis. Moreover, the Korakuen/Kasuga station was the most important node when considering the betweenness and information centralities and also was the border node between cluster groups when analyzing with the WSBM (shown in Fig. 55), which means this station can be considered as the vulnerable node and also its link.

The multi-criteria analysis for railway network critical analysis shows benefit to helping the railway operators to plan the preventive strategy under the limit of resources such as materials, equipment, labor, and budget that make it difficult to conduct in several sections at the same period. The section with very high centrality nodes and vulnerability can be considered the very vulnerable section located in the critical area and needs to be managed as the first priority for improvement. However, this research mainly considered the pure topology analysis. In the future, we plan to apply more indicators such as passenger or traffic flow to improve the accuracy, such as the traffic-weighted edge for the algebraic connectivity-based vulnerability analysis (Liu et al., 2009).

8.2 Conclusions

From the railway network critical analysis for preventive strategy management, all the results can conclude by the following topic.

8.2.1 Railway network centrality analysis

After testing all centrality analyses on the case study railway networks, various characteristics depended on each centrality type's basic theory and purpose. In the case of degree and eigenvector centrality analyses, the node with the highest centrality is connected to several neighbor nodes, especially with the node that has a high value of centrality in the eigenvector case. However, if we need to consider the influence of network flow, the betweenness centrality is a better choice as well as the information centrality, which is also an alternative method even though it consumes much processing time. The main reasons are that both indicators are measured on the shortest path under the assumption that most of the flow follows the shortest route. These methods do not have the narrow range of centrality values found in the closeness centrality analysis. Moreover, the betweenness centrality corresponds with the information centrality when considering the correlation coefficient.

The result from betweenness centrality, closeness centrality, and information centrality are pointed to the most critical area within the railway network, which helps operators focus on finding the very vulnerable section or station and then managing priority for preventive plans and operations easier. In addition, we can apply this analysis with the passenger or traffic-weighted node as the future work for analyzing the most influential node, which has an extensive effect on passenger flow or traffic flow if it is disrupted.

8.2.2 The algebraic connectivity- and global efficiency-based vulnerability analyses for the railway network

This research analyzed the algebraic connectivity-based vulnerability, which compares the performance with the existing global efficiency-based vulnerability analysis to find the feasibility of using it as the alternative method. The result shows that the algebraic connectivity-based vulnerability method uses less computing time than the global efficiency-based vulnerability with the same computing algorithm because it measures only the second smallest eigenvalue, not the average shortest path. Moreover, it can point to some moderate vulnerable sections not shown clearly in the global efficiency-based vulnerability analysis. However, the algebraic connectivity-based vulnerability analysis also shows its disadvantage; if we measure the branch or no alternate route section, the vulnerability value always has 100% due to algebraic connectivity having zero if these sections are cut off and make the network completely separated. In addition, unlike the global efficiency-based case, the algebraic connectivity-based vulnerability less corresponds with the edge betweenness centrality on a local scale. For these reasons, the algebraic connectivity-based vulnerability has some feasibility in analyzing the large dense or grid network, such as the urban railway network or urban road network.

To manage the priority for preventive strategy, the operator can use the algebraic connectivity-based vulnerability alongside the global efficiency-based vulnerability to find the section that needs the first priority to inspect, maintain and repair, considering the high vulnerability of both methods. Moreover, centrality analysis can assist this evaluation in identifying the critical node or area, which probably has a very vulnerable section.

Although this research mainly analyzed the topology-based vulnerability, it can develop into passenger flow or traffic flow vulnerability analysis as a future task by adding the actual passenger or traffic-weighted edge. This concept is expected to improve the accuracy with more factors or indicators to help the operator improve the operation plan and schedule of the railway network under the constraints of the operation's resources.

8.2.3 The stochastic block model analysis for the railway network

This research used a weighted stochastic block model to recover the latent block structure and prevent the information loss caused by thresholding. The result shows the clustering pattern as a mixed model between core and random clusters. Most of the nodes from random clusters separate the group of core clusters so that it can consider these nodes as vulnerable nodes connected by vulnerable links. However, these results are just immediate results due to the research time limit. For accuracy improvement, this analysis still needs to calculate the optimal number of cluster groups, which can illustrate the optimal model.

8.3 Future Works

8.3.1 Analyze the network with the actual passenger or traffic flow by considering the weighted node or link within the network. The flow data can be obtained from electronic data such as IC cards or mobile phones, which has been applied to evaluate the passenger/traffic-weight centrality and vulnerability of the real network on a larger scale and help the operator manage the operation.

8.3.2 Use another compiled language such as C or C++ to analyze the network and compare the performance, including the processing time, which expect to use less than interpreted language programming.

8.3.3 Improve the weighted stochastic block model by calculating the optimal number of clusters, such as minimum description length, integrated likelihood, approximation thereof, and Bayes factor, to select the size of the cluster group.

Acknowledgments

The authors appreciate the reviewers and committees for their valuable comments and suggestions on the draft dissertation of this research. Besides, this research is inspired by the work of Mr. Takaaki Nakaminami. Some programming codes are obtained from MATLAB Tools for Network Analysis (2006-2011) and permitted to use by Apollo Program Professor of Astronautics and Engineering Systems, Massachusetts Institute of Technology. The open-source codes are obtained from the Brain Connectivity Toolbox, which Mika Rubinov and Jonathan Clayden developed, and from the Weighted Stochastic Block Model, which was developed by Christopher Aicher.

References

[1] Crucitti, P., Latora, V., & Porta, S. (2006). Centrality in networks of urban streets. *Chaos*, 16. https://doi.org/10.1063/1.2150162

[2] Liu, W., Sirisena, H., & Pawlikowski, K. (2009). Weighted algebraic connectivity metric for non-uniform traffic in reliable network design. *2009 International Conference on Ultra Modern Telecommunications and Workshops*. https://doi.org/10.1109/ICUMT.2009.5345573

[3] Rodríguez-Núñez, E., & García-Palomares, J. C. (2014). Measuring the vulnerability of public transport networks. *Journal of Transport Geography*, Vol., 50–63. https://doi.org/10.1016/j.jtrangeo.2014.01.008

[4] Sun, D. J., & Guan, S. (2016) Measuring vulnerability of urban metro network from line operation perspective. *Transportation Research Part A: Policy and Practice*, 94, 348–359. https://doi.org/10.1016/j.tra.2016.09.024

Appendixes

Appendix 1: MATLAB code for the algebraic connectivity-based vulnerability analysis

```
%*********** Algebraic connectivity-based vulnerability **************
% Input
% Adjacency_matrix, Adjacency matrix from the evaluated network

% Download adjacency matrix from MS Excel
filename = 'JR Subway Nishitetsu (colour).xlsx';
Range = 'A1:YU671'; % Range of adjacency matrix in MS Excel
Adjacency_matrix = xlsread(filename,'Sheet1',Range);
n = length(Adjacency_matrix);  % number of nodes
Vulnerability_matrix = Adjacency_matrix; % Setup the vulnerability
matrix as the beginning value

for loop = 1:1 % number of rounds to measure the average calculating
time
tic

% Preparing the result matrix
Vulnerability_matrix = Adjacency_matrix;

% Calculate algebraic connectivity of base network for reference
L_base = diag(sum(Adjacency_matrix))-Adjacency_matrix;
[V,D] = eig(L_base);
s = -sort(-diag(D));
AC_base = s(length(s)-1);

% Calculate algebraic connectivity of attacked link and vulnerability
for i = 1:n
    for j = 1:n
        % Attack the link between i and j by giving aij = 0
        if Adjacency_line(i,j) > 0;
           Adjacency_line(i,j) = 0;
           Adjacency_line(j,i) = 0;
           % Calculate the after-cut algebraic Connectivity of the
link between i and j
           L = diag(sum(Adjacency_line))-Adjacency_line;
           [V,D] = eig(L);
           s = -sort(-diag(D));
           AC_line = s(length(s)-1);
           % Calculate the vulnerability of link i and j
           Vulnerability = (abs(AC_base-AC_line)/AC_base)*100;
           % Write the vulnerability of each pair on the result table
           Vulnerability_matrix(i,j) = Vulnerability;
        end
        Adjacency_line = Adjacency_matrix;     % Reset for simulating
the attack on the next link
    end
```

```
end
toc
end

% Show the algebraic Connectivity-based vulnerability results of each
pair of nodes
Algebraic_Connectivity_based_Vulnerability =
adj2edgeL(Vulnerability_matrix)

% Export results to the MS Excel
xlswrite('The Vulnerability Result Comparison',
Algebraic_Connectivity_based_Vulnerability, 'Raw results', 'A3');

% Function for organizing pair of nodes and their vulnerability value
(Strategic Engineering Research Group (SERG), Massachusetts Institute
of Technology, Massachusetts Institute of Technology, 2011)

function el = adj2edgeL(Adjacency_matrix)

n = length(Adjacency_matrix); % number of nodes
edges = find(Adjacency_matrix>0); % indices of all edges

el = [];
for e = 1:length(edges)
  [i,j] = ind2sub([n,n],edges(e)); % node indices of edge e
  el = [el; i j Adjacency_matrix(i,j)];
end
end
```

Appendix 2: MATLAB code for the global efficiency-based vulnerability analysis

```
%************** Global efficiency-based vulnerability ****************
% Input
% Adjacency_matrix, adjacency matrix from the evaluated network

% Download adjacency matrix from MS Excel
filename = 'JR Subway Nishitetsu (colour).xlsx';
Range = 'A1:YU671'; % Range of Adjacency matrix in MS Excel
Adjacency_matrix = xlsread(filename,'Sheet1',Range);
n = length(Adjacency_matrix);  % number of nodes
Vulnerability_matrix = Adjacency_matrix; % Setup the vulnerability
matrix as the beginning value

for loop = 1:1 % number of rounds to measure the average calculating
time
tic

% Preparing the result matrix
Adjacency_line = Adjacency_matrix;

% Calculate global efficiency of base network for reference
E_glob = Global_Efficiency(Adjacency_matrix)

% Calculate global efficiency of attacked link and vulnerability
for I = 1:n
    for j = 1:n
        % Attack the link between i and j by giving aij = 0
        if Adjacency_line(i,j) > 0;
           Adjacency_line(i,j) = 0;
           Adjacency_line(j,i) = 0;
           % Calculate the after-cut global efficiency of the link
between i and j
           E_line = Global_Efficiency(Adjacency_line);
           % Calculate the vulnerability of link i and j
           Vulnerability = ((E_glob-E_line)/E_glob)*100;
           % Write the vulnerability of each pair on the result table
           Vulnerability_matrix(i,j) = Vulnerability;
        end
        Adjacency_line = Adjacency_matrix;    % Reset for simulating
the attack on the next link
    end
end
toc
end

% Show the global efficiency-based vulnerability results of each pair
of nodes
Global_Efficiency_based_Vulnerability =
adj2edgeL(Vulnerability_matrix)

% Export results to the MS Excel
xlswrite('The Vulnerability Result Comparison',
Global_Efficiency_based_Vulnerability, 'Raw results', 'E3');
```

```
% Global efficiency function (Rubinov and Clayden, 2013)

function E = Global_Efficiency(Adjacency_matrix)
 n = length(Adjacency_matrix);
 e = distance_inv(Adjacency_matrix);
 E = sum(e(:))./(n^2-n);                %global efficiency
end

% Calculate the shortest distance for the global efficiency function
(Rubinov and Clayden, 2013)

function D = distance_inv(Adjacency_matrix)
n = length(Adjacency_matrix);
D =  eye(length(Adjacency_matrix));     %identity matrix
n = 1;
nPATH = Adjacency_matrix;               %n-path matrix
L = (nPATH~=0);                         %shortest n-path matrix
(beginning with 1 time per link)

while find(L,1);
    D = D+n.*L;
    N = n+1;
    nPATH = nPATH*Adjacency_matrix;
    L = (nPATH~=0).*(D==0);
end

D(~D) = inf;                       %disconnected nodes are assigned
d=inf;
D = 1./D;                          %invert distance
D = D-eye(length(Adjacency_matrix));
end

% Function for organizing pair of nodes and their vulnerability value
(Strategic Engineering Research Group (SERG), Massachusetts Institute
of Technology, Massachusetts Institute of Technology, 2011)

function el = adj2edgeL(Adjacency_matrix)

n = length(Adjacency_matrix); % number of nodes
edges = find(Adjacency_matrix>0); % indices of all edges

el = [];
for e = 1:length(edges)
  [i,j] = ind2sub([n,n],edges(e)); % node indices of edge e
  el = [el; i j Adjacency_matrix(i,j)];
end
end
```

Appendix 3: MATLAB code for the edge betweenness centrality analysis

```
%******************** Edge betweenness Centrality ********************
% Input
% Adjacency_matrix, adjacency matrix from the evaluated network

% Download adjacency matrix from MS Excel
filename = 'JR Subway Nishitetsu (colour).xlsx';
Range = 'A1:YU671'; % Range of adjacency matrix in MS Excel
Adjacency_matrix = xlsread(filename,'Sheet1',Range);
n = length(Adjacency_matrix);  % number of nodes

% Calculate the edge betweenness centrality of every link
for loop = 1:1 % number of rounds to measure the average calculating
time
tic
Edge_Betweenness_Centrality = edge_betweenness(Adjacency_matrix)
toc
end

% Export results to the MS Excel
xlswrite('The Vulnerability Result
Comparison',Edge_Betweenness_Centrality, 'Raw results', 'I3');

% Function for edge betweenness centrality (Strategic Engineering
Research Group (SERG), Massachusetts Institute of Technology,
Massachusetts Institute of Technology, 2011)

function ew = edge_betweenness(Adjacency_matrix)

el = adj2edgeL(Adjacency_matrix);  % the corresponding edgelist
n = length(Adjacency_matrix);  % number of nodes
m = numedges(Adjacency_matrix);  % number of edges

ew = zeros(size(el,1),3); % edge betweenness - output


for s = 1:n % across all (source) nodes

    % compute the distances and weights starting at source node i
    d = inf(n,1); w = inf(n,1);
    d(s) = 0; w(s) = 1; % source node distance and weight
    queue = [s];      % add to queue
    visited = [];

    while not(isempty(queue))
        j = queue(1); % pop first member of node j
        visited = [visited j];
        neigh = kneighbors(Adjacency_matrix,j,1); % find all adjacent
nodes, 1 step away

        for x = 1:length(neigh)  % add to queue if unvisited
            nei = neigh(x);
```

```
            if isempty(find(visited==nei)) & sempty(find(queue==nei));
queue=[queue nei]; end

        end
        for x = 1:length(neigh)

            nei = neigh(x);
            if d(nei)==inf   % not assigned yet
                d(nei) = 1+d(j);
                w(nei) = w(j);
            elseif d(nei)<inf & d(nei)==d(j)+1  % assigned already,
add the new path
                w(nei) = w(nei)+w(j);
            elseif d(nei)<inf & d(nei)<d(j)+1
                'do nothing';
            end
        end
        queue=queue(2:length(queue));  % remove the first element
    end

    eww = zeros(size(el,1),3);   % edge betweenness for every source
node (iteration)

    % find every leaf - no path from "s" to other vertices goes
through the leaf
    leaves = find(d==max(d)); % farthest away from source
    for l =  1:length(leaves)
        leaf = leaves(l);
        neigh = kneighbors(Adjacency_matrix,leaf,1);
        nei2rem = [];
        for x = 1:length(neigh)

            if isempty(find(leaves==neigh(x))); nei2rem = [nei2rem
neigh(x)]; end

        end
        neigh = nei2rem;  % remove other leaves among the neighbors
        for x = 1:length(neigh)
            indi = find(el(:,1)==neigh(x));
            indj = find(el(:,2)==leaf);
            indij = intersect(indi,indj);   % should be only one
element at the intersection
            eww(indij,3) = w(neigh(x))/w(leaf);
        end
    end

    dsort = unique(d);
    dsort = -sort(-dsort);  % reverse sort of unique distance values

    for x = 1:length(dsort)
        leaves = find(d==dsort(x));
        for l = 1:length(leaves)
            leaf = leaves(l);
            neigh = kneighbors(Adjacency_matrix,leaf,1);
            up_neigh = []; down_neigh=[];
            for x = 1:length(neigh)
```

```
                if d(neigh(x))<d(leaf)
                    up_neigh = [up_neigh neigh(x)];
                elseif d(neigh(x))>d(leaf)
                    down_neigh = [down_neigh neigh(x)];
                end
            end
            sum_down_edges = 0;
            for x = 1:length(down_neigh)
                indi = find(el(:,1)==leaf);
                indj = find(el(:,2)==down_neigh(x));
                indij = intersect(indi,indj);
                sum_down_edges = sum_down_edges+eww(indij,3);
            end
            for x = 1:length(up_neigh)
                indi = find(el(:,1)==up_neigh(x));
                indj = find(el(:,2)==leaf);
                indij = intersect(indi,indj);

eww(indij,3)=w(up_neigh(x))/w(leaf)*(1+sum_down_edges);
            end
        end
    end

    for e = 1:size(ew,1); ew(e,3)  = ew(e,3)+eww(e,3); end

end

for e = 1:size(ew,1)
    ew(e,1) = el(e,1);
    ew(e,2) = el(e,2);
    ew(e,3) = ew(e,3)/n/(n-1);    % normalize by the total number of
paths
end
end

% Function for organizing pair of nodes and their vulnerability value
(Strategic Engineering Research Group (SERG), Massachusetts Institute
of Technology, Massachusetts Institute of Technology, 2011)

function el = adj2edgeL(Adjacency_matrix)

n = length(Adjacency_matrix); % number of nodes
edges = find(Adjacency_matrix>0); % indices of all edges

el = [];
for e = 1:length(edges)
  [i,j] = ind2sub([n,n],edges(e)); % node indices of edge e
  el = [el; i j Adjacency_matrix(i,j)];
end
end

% Function for neighbor node (Strategic Engineering Research Group
(SERG), Massachusetts Institute of Technology, Massachusetts Institute
of Technology, 2011)

function kneigh = kneighbors(Adjacency_matrix,ind,m)
```

```
adjk = Adjacency_matrix;
for i=1:m-1; adjk = adjk*Adjacency_matrix; end;

kneigh = find(adjk(ind,:)>0);
end
```

Appendix 4: MATLAB code for the degree, closeness, betweenness, eigenvector, and information centralities

```
%*********************** Centrality Analysis *************************
% Input
% Adjacency_matrix, adjacency matrix from the evaluated network

% Download adjacency matrix from MS Excel
filename = 'JR Subway Nishitetsu (colour).xlsx';
Range = 'A1:YU671'; % Range of adjacency matrix in MS Excel
Adjacency_matrix = xlsread(filename,'Sheet1',xlRange);
n = length(Adjacency_matrix);  % number of nodes

% Create a graph to calculate centrality
G = graph(Adjacency_matrix);
n = length(Adjacency_matrix);

% Calculate degree centrality
Degree_Centrality = centrality(G, 'degree')
% Calculate closeness centrality
Closeness_Centrality = (n-1)*centrality(G, 'closeness')
% Calculate betweenness centrality
Betweenness_Centrality = centrality(G, 'betweenness')
% Calculate eigenvector centrality
Eigenvector_Centrality = centrality(G, 'eigenvector')

% Calculate information centrality

% Based global efficiency
Information_C = zeros(n);
Adjacency_line = Adjacency_matrix;
E_base = Efficiency(Adjacency_matrix);

% Check every link on each node
for i=1:n
    for j=1:n
        if Adjacency_line(i,j)==1;
           Adjacency_line(i,j)=0; % Disconnect every link that
connected node i
           Adjacency_line(j,i)=0; % Disconnect every link that
connected node i
        end
    end
    % Calculate the after-cut global efficiency of each node
    G = graph(Adjacency_line);
    E_line = Efficiency(Adjacency_line);
    % Calculate information Centrality
    Information_C(i)=((E_base-E_line)/E_base);
    % Reset the adjacency matrix to calculate the next node
    Adjacency_line=Adjacency_matrix;
end

% Show the information centrality results of each node
Information_Centrality = Information_C(:,1)
```

```
% Export results to the MS Excel
filename = 'Centrality Comparison.xlsx';
xlswrite(filename, Degree_Centrality, 'Results', 'C3')
xlswrite(filename, Closeness_Centrality, 'Results', 'D3')
xlswrite(filename, Betweenness_Centrality, 'Results', 'E3')
xlswrite(filename, Eigenvector_Centrality, 'Results', 'F3')
xlswrite(filename, Information_Centrality, 'Results', 'G3')

% Global efficiency function (Rubinov and Clayden, 2013)

function E = Global_Efficiency(Adjacency_matrix)
 n = length(Adjacency_matrix);
 e = distance_inv(Adjacency_matrix);
 E = sum(e(:))./(n^2-n);                %global efficiency
end

% Calculate the shortest distance for the global efficiency function
(Rubinov and Clayden, 2013)

function D = distance_inv(Adjacency_matrix)
n = length(Adjacency_matrix);
D = eye(length(Adjacency_matrix));      %identity matrix
n = 1;
nPATH = Adjacency_matrix;               %n-path matrix
L = (nPATH~=0);                         %shortest n-path matrix
(beginning with 1 time per link)

while find(L,1);
    D = D+n.*L;
    n = n+1;
    nPATH = nPATH*Adjacency_matrix;
    L = (nPATH~=0).*(D==0);
end

D(~D) = inf;                       %disconnected nodes are assigned
d=inf;
D = 1./D;                          %invert distance
D = D-eye(length(Adjacency_matrix));
end
```

Appendix 5: MATLAB code for the weighted stochastic block model (WSBM)

Because of the limit of research time, The MATLAB code for the weighted stochastic block model is obtained from Christopher Aicher (Aicher, 2013; Aicher et al., 2015) under the agreement “This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.”

```
%% ***** 4 Group Weighted Stochastic Block Model (WSBM) example ******

%% Generate data from the adjacency matrix
True_Model1 = xlsread('JR Subway Nishitetsu (colour).xlsx','Sheet1'
,'A1:YU671');
plotWSBM(True_Model1);
title('Synthetic Data');

%% Create Fit Mixed Model, in which weighted distribution (W_Distr)
used the normal distribution, and edge distribution (E_Distr) used the
Bernoulli distribution. However, if you need to create a pure
Stochastic Block Model (SBM), you must set the weighted distribution
as “none”.

[~,Mixed_Model1] =
wsbm(True_Model1,4,'W_Distr','Normal','E_Distr','Bernoulli');
subplot(1,2,1);
plotMu(Mixed_Model1);
subplot(1,2,2);
plotWSBM(Mixed_Model1);
title('Mixed WSBM Model Permuted Adjacency Matrix')

%PLOTMU plots the posterior probability of vertex-label assignments.
function [] = plotMu(Model)
%
%   PLOTMU(Model) generates a IMAGESC plot of mu, the posterior
probability
%   of vertex-label assigments. Each row is a vertex and each column
is a
%   group. Ideally the posterior for each vertex is concentrated into
one
%   group. If a vertex has a uniform or dispersed posterior, this
indicates
%   a lack of fit or indecision in assigning a label.
%
%   Example:
%       [Label,Model] = wsbm(Raw_Data,k);
%       plotMu(Model)
%   Input:
%       mu    - kxn mat ~ mu(g,v) is the prob of vertex v belongs to
group g
%   Output:
```

```
%       A IMAGESC plot.
%
%   See also WSBM, PLOTWSBM
% PLOTMU
% Version 1.0 | December 2013 | Christopher Aicher
%
%   Copyright 2013-2014 Christopher Aicher
%
%   This program is free software: you can redistribute it and/or
modify
%   it under the terms of the GNU General Public License as published
by
%   the Free Software Foundation, either version 3 of the License, or
%   (at your option) any later version.
%   This program is distributed in the hope that it will be useful,
%   but WITHOUT ANY WARRANTY; without even the implied warranty of
%   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
%   GNU General Public License for more details.
%   You should have received a copy of the GNU General Public License
%   along with this program.  If not, see
<http://www.gnu.org/licenses/>

if isstruct(Model),
    mu = Model.Para.mu;
elseif isnumeric(Model),
    mu = Model;
else
    error('Unrecognized Input for Model');
end

opengl('software');
imagesc(mu');
title('Vertex-Label Posterior Plot');
xlabel('Probability in Group');
ylabel('Vertex');
colorbar
caxis([0 1]);
end

%PLOTWSBM plots the data (Raw_Data) sorted by the vertex-labels
(Labels).
function [] = plotWSBM(Raw_Data,Labels,style)
%
%   PLOTWSBM(Raw_Data,Labels) plots the Raw_Data as an adjacency
matrix
%   sorted by the vertex-labels. The adjacency matrix's rows and
columns
%   are permuted such that vertices in the same group are next to each
%   other. This allows us to visualize the `block' structure.
%   Alternative modes allow us to look at the edge count or weight on
a
%   log scale.
%
%   Examples:
%   Regular Plot
%       [Label,Model] = wsbm(Raw_Data,k);
%       plotWSBM(Raw_Data);
```

```
%       plotWSBM(Raw_Data,Labels);
%       plotWSBM(Raw_Data,Model.Para.mu);
%       plotWSBM(Model);
%   Log Plot
%       [Label,Model] = wsbm(Raw_Data,k);
%       plotWSBM(Raw_Data,Label,'log10');
%       plotWSBM(Model,'log10');
%   Edge Plot
%       [Label,Model] = wsbm(Raw_Data,k);
%       plotWSBM(Raw_Data,Label,'edge');
%       plotWSBM(Model,'edge');
%
%
%   Inputs:
%       Raw_Data - mx3 (edge list) or nxn (adjacency matrix) of the
network
%       Labels - 1xn vector of vertex-labels
%       mu  - kxn matrix ~ mu(g,v) is the prob of vertex v belonging
to
%                             group g
%       style - optional string for selecting
%                   logplot - 'log10' (base10) or 'log' (base e)
%                               Note: values <= 0 are discarded
%                   edge plot - 'edge'
%   Outputs:
%       Plot of Raw_Data ordered by group labels.
%   See also WSBM, PLOTMU

% PLOTWSBM
% Version 1.0 | December 2013 | Christopher Aicher
% Version 1.1 | April 2014 | Christopher Aicher | added different
options
%
%   Copyright 2013-2014 Christopher Aicher
%
%   This program is free software: you can redistribute it and/or
modify
%   it under the terms of the GNU General Public License as published
by
%   the Free Software Foundation, either version 3 of the License, or
%   (at your option) any later version.
%   This program is distributed in the hope that it will be useful,
%   but WITHOUT ANY WARRANTY; without even the implied warranty of
%   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
%   GNU General Public License for more details.
%   You should have received a copy of the GNU General Public License
%   along with this program.  If not, see
<http://www.gnu.org/licenses/>

% TODO: This is not written very intuitively due to cut and pasting
old code.

if isstruct(Raw_Data),
    if nargin == 2 && ischar(Labels),
        style = Labels;
    end
    Model = Raw_Data;
```

```
    Raw_Data = Model.Data.Raw_Data;
    Labels = Model.Para.mu;
end
[n,m] = size(Raw_Data);
if n == m,
    fprintf('Treating Raw_Data as an Adj Matrix\n');
elseif m == 3,
    fprintf('Treating Raw_Data as an Edge List\n');
    n = max([Raw_Data(:,1);Raw_Data(:,2)]);
else
    error('Invalid Raw_Data Format');
end
if ~exist('Labels','var'),
    mu = ones(1,n);
elseif ~isnumeric(Labels),
    if ischar(Labels),
        style = Labels;
        mu = ones(1,n);
    else
        error('Labels is not a numeric array.');
    end
elseif size(Labels,1) == 1,
    mu = zeros(max(Labels),size(Labels,2));
    for kk = 1:max(Labels),
        mu(kk,Labels == kk) = 1;
    end

else
    mu = Labels;
end
if ~exist('style','var'),
    style = 'default';
end

opengl('software');

mu = mu';
[n,k] = size(mu);
A_sort = zeros(n);
list = zeros(1,n);
breaks = zeros(1,k);

if sum(sum(mu > 0)) > n
    fprintf('\nRounding mu, randomly picking uniform\n');
    mu = mu./(sum(mu,2)*ones(1,size(mu,2)));
    e = [zeros(size(mu,1),1) cumsum(mu,2)];
    e(:,end) = 1;
    e(e>1) = 1;
    mu = diff(e,1,2);
    mu = mnrnd(1,mu)
end

cur = 1;
for ii = 1:k
    indicies = find(mu(:,ii));
    list(cur:cur+length(indicies)-1) = indicies;
    cur = cur + length(indicies);
```

```
    breaks(ii) = cur-1;
end

switch lower(style),
    case {'edge','edges'}
        if m == 3,
            [~,Raw_Data] = Edg2Adj(Raw_Data);
        else
            Raw_Data = ones(size(Raw_Data));
        end
    case {'log10'}
        if m == 3,
            [Raw_Data,~] = Edg2Adj(Raw_Data);
        end
        Raw_Data(Raw_Data <= 0) = NaN;
        Raw_Data = log10(Raw_Data);
    case {'log','loge'}
        if m == 3,
            [Raw_Data,~] = Edg2Adj(Raw_Data);
        end
        Raw_Data(Raw_Data <= 0) = NaN;
        Raw_Data = log(Raw_Data);
    otherwise
        if m == 3,
            [Raw_Data,~] = Edg2Adj(Raw_Data);
        end
end

for ii = 1:n
    A_sort(ii,:) = Raw_Data(list(ii),list);
end

%Plot the Matrix
h = imagesc(A_sort,[min(A_sort(:))-.00001,max(A_sort(:))]);
xlabel('Child');
ylabel('Parent');
set(h,'alphadata',~isnan(A_sort));
colorbar;
hold all;

for ii = 1:k-1
    plot([breaks(ii)+.5,breaks(ii)+.5],[-.5,breaks(k)+.5],'-
k','LineWidth',1.5);
    plot([-.5,breaks(k)+.5],[breaks(ii)+.5,breaks(ii)+.5],'-
k','LineWidth',1.5);
end
hold off;

end

%WSBM find latent community structure in weighted networks.
function [Labels,Model] = wsbm(E,R_Struct,varargin)
%
%   WSBM is the main driver program for finding community structure,
%   inferring the vertex-labels and edge-bundle parameters of a
%   Weighted Stochastic Block Model (WSBM).
%   This algorithm infers the parameters by approximating a posterior
```

```
%   distribution using an iterative variational Bayes algorithm.
%   See Aicher, Jacobs, Clauset (2013) for the theoretical derivation 
of
%   the algorithm
%
%   Syntax:
%       [Labels] = wsbm(E)
%       [Labels] = wsbm(E,k)
%       [Labels, Model] = wsbm(...,'ParaName',ParaValue)
%
%   Examples:
%       Raw_Data = generateEdges();
%     % Default
%       [Labels] = wsbm(Raw_Data);
%     % Infer 2 Groups
%       [Labels] = wsbm(Raw_Data,2);
%     % Change W_Distr to Exp
%       [Labels] = wsbm(Raw_Data,2,'W_Distr','Exp');
%     % Run Code in Parallel
%       [Labels] = wsbm(Raw_Data,2,'parallel',1);
%     % Ignore E_Distr
%       [Labels] = wsbm(Raw_Data,2,'E_Distr','None');
%       [Labels] = wsbm(Raw_Data,2,'alpha',0);
%     % Increase the number of trials
%       [Labels] = wsbm(Raw_Data,2,'numTrials','500');
%     % Multiple changes at once
%       [Labels] = 
wsbm(Raw_Data,2,'W_Distr','Exp','alpha',0,'parallel',1);
%   See WSBMDemo.m for more examples
%
%   Inputs:
%       E - an m by 3 network edge list (parent,child,weight)
%           (If E is an n by n network adjacency matrix, then it will 
be
%           converted into an edge list by ADJ2EDG)
%       k - number of blocks            (k = 4 default)
%
%   Outputs:
%       Labels - a n by 1 vector of edge labels (using the MAP 
estimates)
%                   - Ties are broken randomly (with a warning 
message)
%       Model - a MATLAB structure for advanced output
%
%   For more information, try 'type wsbm.m'
%
%   Copyright 2013-2014 Christopher Aicher
%
%   This program is free software: you can redistribute it and/or 
modify
%   it under the terms of the GNU General Public License as published 
by
%   the Free Software Foundation, either version 3 of the License, or
%   (at your option) any later version.
%   This program is distributed in the hope that it will be useful,
%   but WITHOUT ANY WARRANTY; without even the implied warranty of
%   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
%   GNU General Public License for more details.
```

```
%   You should have received a copy of the GNU General Public License
%   along with this program.  If not, see
<http://www.gnu.org/licenses/>.
%
%   See also SETUP_DISTR, INSTALLMEXFILES, WSBMDEMO, CALC_LOGEVIDENCE

% WSBM
% Version 1.0 | December 2013 | Christopher Aicher
%
%
% ADVANCED INPUT/OUTPUT:
% -Advanced Inputs: 'ParaName' - ParaValue      * indicates default
% --Model Inputs:
%   'W_Distr' - edge-weight distr    ('Normal' *)
%   'E_Distr' - edge-existence distr ('Bernoulli' *)
%       To select a distribution type it's name:
%       Weighted distributions:
%           Bernoulli, Binomial, Poisson, Normal, LogNormal,
%           Exponential, Pareto, None,
%       Edge distributions:
%           Bernoulli, Binomial, Poisson, None, DC (Degree Corrected)
%       See SETUP_DISTR for more information
%   'R_Struct' - kxk matrix of the block structure.
%   'alpha' - para in [0,1]. 0 only weight, 0.5* both, 1 only
existence
% --Inference Options:
%   'numTrials' - number of trials with different random initial
conditions
%   'algType' - 'vb'* naive bayes, 'bp' belief propagation
%   'networkType' - 'dir'* directed,'sym' = symmetric,'asym' =
asymmetric
%   'nanType' - 'missing'* nans are missing, 'nonedge' = nans are
nonedge
%   'mainMaxIter' - Maximum number of iterations in main_loop
%   'mainTol' - Minimum (Max Norm) convergence tolerance in main_loop
%   'muMaxIter' - Maximum number of iterations in mu_loop
%   'muTol' - Minimum (Max Norm) convergence tolerance in mu_loop
% --Extra Options:
%   'verbosity' - 0 silent, 1* default, 2 verbose, 3 very verbose
%   'parallel' - boolean, run in parallel? 0* No (Need Parallel
ToolBox)
%   'save' - boolean, save temp results? 0* No
%   'outputpath' - string to where to save temp results (Only if save
= 1)
%   'seed' - seed for algtype (mu_0 or mes_0) (Sets numTrials = 1)
%   'mexfile' - boolean, run using MEX files? 1* Yes (Need MEX Files)
% --Prior Options:
%   'mu_0' - kxn matrix prior vector for vertex-label parameters(sums
to 1)
%
% -Advanced Outputs:
%  'Model' - struct with the following fields
%       'name' - name of model <W_Distr-E_Distr-R_Struct>
%       'Data' - struct with Data related variables
%       'W_Distr' - struct from SETUP_DISTR
%       'E_Distr' - struct from SETUP_DISTR
%       'R_Struct' - struct with edge-bundle (R) variables
```

```
%       'Para' - struct with inferred hyperparamters (tau,mu) and
parameter
%                 estimates (theta)
%       'Options' - struct with inference option information
%       'Flags' - struct with convergence flags
%
%
% For an overview see the README.txt file
%

%----------------------------------------------------------------------
--%
% WSBM CODE
%----------------------------------------------------------------------
--%

% Call wsbm_driver.m
if nargin > 1,
    Model = wsbm_driver(E,R_Struct,varargin{:});
else
    Model = wsbm_driver(E);
end

Labels = Model.Para.mu';
[n,k] = size(Labels);
if sum(sum(Labels >= 1/k-10^-3)) > n
    fprintf('Breaking %u Ties Randomly\n',sum(sum(Labels >= 1/k-10^-
3,2) > 1));
    e = [zeros(size(Labels,1),1) cumsum(Labels,2)];
    e(:,end) = 1; e(e>1) = 1;
    Labels = diff(e,1,2);
    Labels = mnrnd(1,Labels);
end
[~,Labels] = max(Labels,[],2);

%----------------------------------------------------------------------
--%
% END OF WSBM CODE
%----------------------------------------------------------------------
--%
%EOF

% WSBM_Driver function
function [Model] = wsbm_driver(Raw_Data,R_Struct,varargin)
%
% See 'help wsbm.m' or 'type wsbm.m' for information
%----------------------------------------------------------------------
--%

% WSBM_Driver
% Version 1.0 | December 2013 | Christopher Aicher
%
%   Copyright 2013-2014 Christopher Aicher
%
%   This program is free software: you can redistribute it and/or
modify
```

```
%   it under the terms of the GNU General Public License as published
by
%   the Free Software Foundation, either version 3 of the License, or
%   (at your option) any later version.
%   This program is distributed in the hope that it will be useful,
%   but WITHOUT ANY WARRANTY; without even the implied warranty of
%   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
%   GNU General Public License for more details.
%   You should have received a copy of the GNU General Public License
%   along with this program.  If not, see
<http://www.gnu.org/licenses/>

%----------------------------------------------------------------------
---
% WSBM_DRIVER CODE
%----------------------------------------------------------------------
---


W_Distr = [];
E_Distr = [];
Data = [];
Options = [];
% Parse and Setup Input
format_input()
% 3 Cases:
% -Case 1: Once From Seed
if ~isempty(Options.seed),
    check_seed(Options.seed);
    % Run Inference
    [Para,Flags] =
main_alg(Data,W_Distr,E_Distr,R_Struct,Options.seed,Options);
    % Score Inference
    Para.LogEvidence =
calc_logEvidence(Data,W_Distr,E_Distr,R_Struct,Para,Options);

    % Print Info
    if Options.verbosity > 0,
        fprintf('Best Result LogEvidence: %2.4e
\n',Para.LogEvidence);
    end

% -Case 2: Trials in Serial
elseif ~Options.parallel,
    if Options.verbosity > 0,
        fprintf('Running %u Trials in Series \n', Options.numTrials);
    end
    LogEvidenceBest = -Inf;
    %Begin Trial Loop
    for trial_num = 1:Options.numTrials,
        % Set Seed
        seed = random_seed();
        % Run Inference
        [Para,Flags] =
main_alg(Data,W_Distr,E_Distr,R_Struct,seed,Options);
        % Score Inference
        Para.LogEvidence =
calc_logEvidence(Data,W_Distr,E_Distr,R_Struct,Para,Options);
```

```
        % Print Info
        if Options.verbosity > 0,
            fprintf('Trial %u of %u : %2.4e | Best: %2.4e \n',...

trial_num,Options.numTrials,Para.LogEvidence,LogEvidenceBest);
        end
        % Keep Track of Best Trial
        if Para.LogEvidence > LogEvidenceBest,
            ParaBest = Para;
            FlagsBest = Flags;
            seedBest = seed;
            LogEvidenceBest = Para.LogEvidence;
            if Options.verbosity > 0, % Indicate Better Model Found
                fprintf(' ** \n');
            end
        end
    end % End Trial Loop
    % Save Result
    Para = ParaBest;
    Flags = FlagsBest;
    Options.seed = seedBest;

% -Case 3: Trials in Parallel
else
    if Options.verbosity > 0,
        fprintf('Running %u Trials in Parallel \n',
Options.numTrials);
    end
    % Setup
    seeds = cell(Options.numTrials,1);
    Paras = cell(Options.numTrials,1);
    Flagss = cell(Options.numTrials,1);
    for trial_num = 1:Options.numTrials,
        % Set Seeds
        seeds{trial_num} = random_seed();
    end
    % Begin Para Trial Loop
    parfor trial_num = 1:Options.numTrials,
        % Run Inference
        [Para,Flags] =
main_alg(Data,W_Distr,E_Distr,R_Struct,seeds{trial_num},Options);
        % Score Inference
        Para.LogEvidence =
calc_logEvidence(Data,W_Distr,E_Distr,R_Struct,Para,Options);
        % Save Temp Results
        Paras{trial_num} = Para;
        Flagss{trial_num} = Flags;
        % Print Info
        if Options.verbosity > 0,
            fprintf('Trial %u complete \n',trial_num);
        end
    end % End Para Trial Loop
    % Find Best Trial
    LogEvidenceBest = zeros(Options.numTrials,1);
    for trial_num = 1:Options.numTrials,
        LogEvidenceBest(trial_num) = Paras{trial_num}.LogEvidence;
    end
```

```
    [~,bestTrial] = max(LogEvidenceBest);
    if Options.verbosity > 0,
        fprintf('Best Result LogEvidence: %2.4e
\n',LogEvidenceBest(bestTrial));
    end
    % Save Result
    Para = Paras{bestTrial};
    Flags = Flagss{bestTrial};
    Options.seed = seeds{bestTrial};
end % End Cases
% Save Model Struct
Model = struct('name',[W_Distr.name,'-',E_Distr.name,'-
',R_Struct.name],...
               'Data',Data,...
               'W_Distr',W_Distr,...
               'E_Distr',E_Distr,...
               'R_Struct',R_Struct,...
               'Para',Para,...
               'Flags',Flags,...
               'Options',Options);
if Options.verbosity > 0,
    fprintf('... wsbm.m Done\n');
end

%----------------------------------------------------------------------
---
% NESTED FUNCTIONS
%----------------------------------------------------------------------
---

% FORMAT_INPUT FUNC --------------------------------------------------
---

function format_input()
    % Parse Input
    % Setup + Check Options
    parse_varagin();
    % Setup + Check W_Distr
    setup_W_Distr();
    % Setup + Check E_Distr
    setup_E_Distr();
    % Check for BP cases
    if strcmpi(Options.algType,'bp'),
        if Options.mexFile == 0,
            error('BP only works with MEX files. Consider setting
algType = vb');
        end
        if isempty(E_Distr.tau_0),
            error('BP only works with E_Distr nonempty. Consider
setting alpha = 0');
        end
        if strncmpi(E_Distr.name,'dc',2),
            error('BP + Degree Correction has not been implemented.');
        end
    end

    % Setup + Check R_Struct
```

```
    setup_R_Struct();
    % Setup Data
    if Options.mexFile,
        setup_Data(); % With Mex files
    else
        fprintf('Running Code without MEX files.\n');
        fprintf('Installing MEX files leads to significantly\n');
        fprintf('faster performance and better scaling...\n');
        if strncmpi(W_Distr.name,'DC',2) ||
strncmpi(E_Distr.name,'DC',2)
            error('Degree Corrected Models have only been implemented
with MEX files.');
        end
        setup_Data_NoMex(); % Without Mex files
    end
    % Setup Mu_0
    if isempty(Options.mu_0),
        Options.mu_0 = ones(R_Struct.k,Data.n)/R_Struct.k;
    elseif size(Options.mu_0,1) ~= R_Struct.k ||...
            size(Options.mu_0,2) ~= Data.n,
        error('User specified mu_0 is not a k by n matrix');
    end
    % Setup Save Directory
    if Options.save,
        if ~exist(Options.outputPath,'dir'),
            disp('Making Save Dir');
            mkdir(Options.outputPath);
        else
            disp('Adding/Overwriting Save Dir');
        end
    end
    % Setup Parallel Computing
    if Options.parallel && matlabpool('size') == 0,
        try
            matlabpool;
        catch err,
            fprintf('Not Running in Parallel: %s\n', err.message);
            Options.parallel = 0;
        end
    end
end

% PARSE_INPUT FUNC ----------------------------------------------------
---


function parse_varagin()
    % Setup Default Options Struct
    Options = struct('algType','vb',...
                     'alpha',0.5,...
                     'networkType','directed',...
                     'nanType','missing',...
                     'parallel',0,...
                     'save',0,...

'outputPath',[cd,filesep,'WSBM_Temp_Output_MAT'],...
                     'verbosity',1,...
                     'numTrials',50,...
```

```
                     'mainMaxIter',80,...
                     'mainTol',0.001,...
                     'muMaxIter',50,...
                     'muTol',0.001,...
                     'mexFile',1,...
                     'seed',[],...
                     'mu_0',[]);
    % Parse Varagin
    for ii = 1:2:length(varargin)-1,
        argok = 1;
        if ischar(varargin{ii}),
            switch lower(varargin{ii}),
                case 'w_distr',
                    W_Distr = varargin{ii+1};
                case 'e_distr',
                    E_Distr = varargin{ii+1};
                case 'algtype',
                    Options.algType = varargin{ii+1};
                case 'alpha',
                    Options.alpha = varargin{ii+1};
                case 'networktype',
                    if ischar(varargin{ii+1}),
                        Options.networkType = varargin{ii+1};
                        if ~strcmpi(Options.networkType,'directed'),
                            fprintf('!WSBM.m currently only runs
directed models!\n');
                            error('Use networkType == directed');
                        end
                    else
                        error('networkType must be a string (in
varargin %u)',ii+1);
                    end
                case 'nantype',
                    if ischar(varargin{ii+1}),
                        Options.nanType = varargin{ii+1};
                        if ~strcmpi(Options.nanType,'missing'),
                            fprintf('!WSBM.m currently only allows
nans to be missing!\n');
                            error('Use nanType == missing');
                        end
                    else
                        error('nanType must be a string (in
varargin %u)',ii+1);
                    end
                case 'parallel',
                    Options.parallel = varargin{ii+1};
                case 'save',
                    Options.save = varargin{ii+1};
                case 'outputpath',
                    if ischar(varargin{ii+1}),
                        Options.outputpath = varargin{ii+1};
                    else
                        error('outputpath must be a string (in
varargin %u)',ii+1);
                    end
                case 'verbosity',
                    Options.verbosity = varargin{ii+1};
                case 'seed',
```

```
                    Options.seed = varargin{ii+1};
                case 'numtrials',
                    Options.numTrials = varargin{ii+1};
                case 'mainmaxiter',
                    if isnumeric(varargin{ii+1}),
                        Options.mainMaxIter = varargin{ii+1};
                    else
                        error('Invalid mainMaxIter parameter in 
varargin %u',ii+1);
                    end
                case 'maintol',
                    if isnumeric(varargin{ii+1}),
                        Options.mainTol = varargin{ii+1};
                    else
                        error('Invalid mainTol parameter in 
varargin %u',ii+1);
                    end
                case 'mumaxiter',
                    if isnumeric(varargin{ii+1}),
                        Options.muMaxIter = varargin{ii+1};
                    else
                        error('Invalid muMaxIter parameter in 
varargin %u',ii+1);
                    end
                case 'mutol',
                    if isnumeric(varargin{ii+1}),
                        Options.muTol = varargin{ii+1};
                    else
                        error('Invalid muTol parameter in 
varargin %u',ii+1);
                    end
                case 'mexfile',
                    if isnumeric(varargin{ii+1}),
                        Options.mexFile = varargin{ii+1};
                    else
                        error('Invalid mexFile parameter in 
varargin %u',ii+1);
                    end
                case 'mu_0',
                    Options.mu_0 = varargin{ii+1};
                otherwise,
                    argok = 0;
            end
        else
            error(['Invalid argument #',num2str(ii)]);
        end
        if ~argok,
            error('Unknown argument #%u: %s', ii,varargin{ii});
        end
    end % End Varagin For Loop

    % Check if mexFiles exist + compiled
    if Options.mexFile == 1,
        if ~(exist('calc_T_e_bra','file') == 3 &&...
             exist('calc_T_w_bra','file') == 3 &&...
             exist('vb_wsbm','file') == 3 &&...
             exist('create_T_e','file') == 3 && ...
```

```
              exist('create_T_w','file') == 3 && ...
              exist('create_T_bp','file') == 3 && ...
              exist('bp_wsbm','file') == 3),
             fprintf('Required MEX files not found... see
InstallMEXFiles.m\n');
             fprintf('Running with slower O(n^2) MATLAB code instead of
O(m)...\n');
             Options.mexFile = 0;
        end
    end

    % Check verbosity
    if Options.verbosity > 4,
        % Comment this out if you know what you are doing.
        error('Note: Options.verbosity > 4 is only used for debugging
purposes.\n');
    end

end

% SETUP_W_DISTR FUNC --------------------------------------------------
---

function setup_W_Distr()
    % Setup W_Distr Struct
    if isempty(W_Distr),
        W_Distr = setup_distr('normal');
        if Options.verbosity > 0,
            fprintf('W_Distr set to Normal (default)\n');
        end
    elseif ischar(W_Distr),
        W_Distr = setup_distr(W_Distr);
    elseif ~isstruct(W_Distr)
        error('Unrecognized W_Distr');
    end
    % Check W_Distr Struct
    check_distr(W_Distr);
end

% SETUP_E_DISTR FUNC --------------------------------------------------
---

function setup_E_Distr()
    % Setup E_Distr Struct
    if isempty(E_Distr),
        E_Distr = setup_distr('bernoulli');
        if Options.verbosity > 0,
            fprintf('E_Distr set to Bernoulli (default)\n');
        end
    elseif ischar(E_Distr),
        E_Distr = setup_distr(E_Distr);
    elseif ~isstruct(E_Distr)
        error('Unrecognized E_Distr');
    end
    % Check E_Distr Struct
    check_distr(E_Distr);
```

```
end

% SETUP_R_Struct FUNC -------------------------------------------------
---

function setup_R_Struct()
% Setup R_Struct
    if ~exist('R_Struct','var'),
        % Default
        R_Struct = SBM_Struct(4);
        if Options.verbosity > 0,
            fprintf('R_Struct set to 4 blocks (default)     \n');
        end
    elseif isnumeric(R_Struct),
        if numel(R_Struct) == 1,
            % Block Model
            R_Struct = SBM_Struct(R_Struct);
        else
            % Custom Matrix
            newR_Struct.R = R_Struct;
            newR_Struct.k = length(newR_Struct.R);
            newR_Struct.r = max(newR_Struct.R(:));
            newR_Struct.name = sprintf('Custom%u',newR_Struct.k);
            R_Struct = newR_Struct;
        end
    elseif ~isstruct(R_Struct),
        error('Unrecognized R_Struct');
    end
    % Check R_Struct
    check_r_struct(R_Struct);
    % Nested Functions:
    % SBM_Struct
    function [out] = SBM_Struct(k)
        if strcmpi(Options.networkType,'directed')
            r = 0;
            R = zeros(k);
            for ii = 1:k
                for jj = 1:k
                    r = r+1;
                    R(ii,jj) = r;
                end
            end
        else
            error('networkType = %s has not been implemented
yet',Options.networkType);
        end
        out.R = R;
        out.k = k;
        out.r = r;
        out.name = sprintf('SBM%u',k);
    end
end

% SETUP_DATA_NOMEX FUNC ----------------------------------------------
---

function setup_Data_NoMex()
```

```
%SETUP_DATA_NOMEX creates/formats the Data struct used in WSBM when 
MeX
%files are not loaded/complied.

    % Convert E into an Adj_Matrix
    [n,m] = size(Raw_Data);
    if (n ~= m),
        if m == 3 && exist('Edg2Adj','file') == 2,
            fprintf('Assuming Raw Data is an Edge List\n');
            fprintf('Converting Raw Data to an Adjacency Matrix\n');
            Adj_Mat = Edg2Adj(Raw_Data);
            n = size(Adj_Mat,1);
        else
            error('Raw Data is not a square n by n Adjancency 
Matrix');
        end
    else
        fprintf('Treating Raw Data as an Adjacency Matrix\n');
        Adj_Mat = Raw_Data;
    end

    % Create Sufficient Statistics
    T_w = cell(numel(W_Distr.T),1);
    for tw = 1:numel(W_Distr.T),
        temp = W_Distr.T{tw}(Adj_Mat);
        temp(isnan(temp)) = 0;
        T_w{tw} = temp;
    end
    T_e = cell(numel(E_Distr.T),1);
    for te = 1:numel(E_Distr.T),
        T_e{te} = E_Distr.T{te}(~isnan(Adj_Mat)*1);
    end

    % Calculate Additive LogLikelihood Constants
    logHw = sum(W_Distr.logh(Adj_Mat(~isnan(Adj_Mat))));
    logHe = sum(E_Distr.logh(isnan(Adj_Mat(:))));
    % COMMENT: This would need to be changed for symmetric / non-
Missing cases

    % Create the Struct
    Data = 
struct('n',n,'Raw_Data',Raw_Data,'T_w',{T_w},'T_e',{T_e},...
    'logHw',logHw,'logHe',logHe);
End

% SETUP_DATA FUNC ----------------------------------------------------
---

function setup_Data()
%SETUP_DATA creates/formats the Data struct used in WSBM

    % Convert E into an Edge_List
    [m,s1] = size(Raw_Data);
    if (s1 ~= 3),
        if s1 == m && exist('Adj2Edg','file') == 2,
            fprintf('Assuming Raw Data is an Adjacency Matrix\n');
```

```
            fprintf('Converting Raw Data to an Edge List\n');
            Edge_List = Adj2Edg(Raw_Data);
            [m,~] = size(Edge_List);
        else
            error('Raw Data is not an m by 3 Edge List\n');
        end
    else
        fprintf('Treating Raw Data as an Edge List\n');
        Edge_List = Raw_Data;
    end
    Edge_List = sortrows(Edge_List,[1,2]);
    n = max(max(Edge_List (:,1:2)));

    % Create Sufficient Statistics
    T_w = zeros(m,numel(W_Distr.T)+2);
    T_w(:,1:2) = Edge_List(:,1:2);
    if strncmpi(W_Distr.name,'DC',2)
        % Handle Degree Corrected Case
        for tw = 1:numel(W_Distr.T)-1,
            T_w(:,tw+2) = W_Distr.T{tw}(Edge_List(:,3));
        end
        A = Edg2Adj(Edge_List);
        T_w(:,end) =
W_Distr.T{end}(Edge_List,[nansum(A,2)';nansum(A,1)]);
    else
        % Handle General Corrected Case
        for tw = 1:numel(W_Distr.T),
            T_w(:,tw+2) = W_Distr.T{tw}(Edge_List(:,3));
        end
    end
    T_w = T_w'; % Convert to t_w+2 by m_E
    T_e = zeros(m,numel(E_Distr.T)+1);
    T_e(:,1:2) = Edge_List(:,1:2);
    edge_temp = ones(m,1);
    edge_temp(isnan(Edge_List(:,3))) = NaN;
    for te = 1:numel(E_Distr.T)-1,
        T_e(:,te+2) = E_Distr.T{te}(edge_temp);
    end
    T_e = T_e'; % Convert to t_e+1 by m_E

    % Calculate In- Out- Degrees
    % degrees(1,:) are the In Degrees
    % degrees(2,:) are the Out Degrees
    degrees_total = zeros(2,n); % Counts missing / NaN edges
    degrees_w = zeros(2,n); % Ignores missing edges
    for ee = 1:m,
        degrees_total(2,Edge_List(ee,1)) =
degrees_total(2,Edge_List(ee,1))+1;
        degrees_total(1,Edge_List(ee,2)) =
degrees_total(1,Edge_List(ee,2))+1;
        if ~isnan(Edge_List(ee,3)),
            degrees_w(2,Edge_List(ee,1)) =
degrees_w(2,Edge_List(ee,1))+1;
            degrees_w(1,Edge_List(ee,2)) =
degrees_w(1,Edge_List(ee,2))+1;
        end
    end
```

```
    % Create Sufficient Statistics Cell Arrays
    if Options.verbosity > 0,
        fprintf('Setting Up Weighted Statistics...\n');
    end
    if numel(W_Distr.T) > 0,
        [T_w_in,T_w_out] = create_T_w(T_w,degrees_w);
    else
        T_w_in = {};
        T_w_out = {};
    end
    if Options.verbosity > 0,
        fprintf('Setting Up Edge Statistics...\n');
    end
    if numel(E_Distr.T) > 0,
        [T_e_in,T_e_out] = create_T_e(T_e,degrees_total);
    else
        T_e_in = {};
        T_e_out = {};
    end
    if strcmpi(Options.algType,'bp'),
        if Options.verbosity > 0,
            fprintf('Setting Up BP Statistics...\n');
        end
        [T_w_in,T_w_out,T_e_in,T_e_out] = ...
            create_T_bp(T_w_in,T_w_out,T_e_in,T_e_out);
    end

    % Calculate Additive LogLikelihood Constants
    logHw = sum(W_Distr.logh(Edge_List(:,3)));
    logHe = m*E_Distr.logh(1); % Not exactly correct for Multigraphs
    logHe = logHe+(n*(n-1)-m)*E_Distr.logh(0);
    % COMMENT: This would need to be changed for symmetric / non-
Missing cases

    % Create the Struct
Data =
struct('n',n,'Raw_Data',Raw_Data,'T_w',T_w,'T_e',T_e,'degrees_w',degre
es_w,...

'degrees_total',degrees_total,'T_w_in',{T_w_in},'T_w_out',{T_w_out},..
.

'T_e_in',{T_e_in},'T_e_out',{T_e_out},'logHw',logHw,'logHe',logHe);
end

% RANDOM_SEED  --------------------------------------------------------
---

function theSeed = random_seed(tuning)
%RANDOM_SEED returns an appropriate random seed for the WSBM.m
inference
% tuning is a tuning parameter which biases the seed to concentrate
mass
%    tuning = 1 -> random/flat, 5 -> default (0 is a bad idea).
if nargin < 1, tuning = 5; end;
mu_seed = rand(R_Struct.k,Data.n).*Options.mu_0;
mu_seed = (mu_seed./(ones(R_Struct.k,1)*max(mu_seed,[],1))).^tuning;
```

```
theSeed = mu_seed./(ones(R_Struct.k,1)*sum(mu_seed,1));
end

% CHECK_SEED -----------------------------------------------------------
---

function check_seed(theSeed)
%CHECK_SEED checks to make sure seed is properly formatted.
if strncmpi(Options.algType,'vb',2),
    [k,n] = size(theSeed);
    %Check VB Seed Format (mu)
    if n ~= Data.n || k ~= R_Struct.k,
        error('Invalid Mu Seed Format:\nSeed is %u by %u needs to be k
by n',n,k);
    end
elseif strncmpi(Options.algType,'bp',2),
    error('BP SEED CHECKER needs to be fixed');
    %Check BP Seed Format (mu,mes)
    if  ~isstruct(theSeed),
        error('For BP, Seed needs to be struct of mu and mes initial
values');
    end
    [n,m] = size(theSeed.mu);
    if n ~= Data.n || m ~= R_Struct.k,
        error('Invalid Mu Seed Format:\nSeed is %u by %u needs to be n
by k',n,m);
    end
    [n,m] = size(theSeed.mes);
    if n ~= Data.m || m~= R_Struct.k,
        error('Invalide Mes Seed Format:\nSeed is %u by %u needs to be
m by k',n,m);
    end
else
    error('Unrecognized algType: %s',Options.algType);
end
end

%----------------------------------------------------------------------
--%
end % End of WSBM_DRIVER.M
%----------------------------------------------------------------------
--%


%----------------------------------------------------------------------
--%
% HELPER FUNCTIONS
%----------------------------------------------------------------------
--%

% CHECK DISTR STRUCT --------------------------------------------------
---

function [] = check_distr(Distr)
%CHECK_DISTR checks to make sure a distr struct has the necessary
fields
%for WSBM.m
```

```
%See `help SETUP_DISTR.m' for more details
    if ~isstruct(Distr),
        error('Distr needs to be a Distr Struct');
    end
    if ~isfield(Distr,'tau_0'),
        error('Distr is missing the field tau_0');
    end
    if ~isfield(Distr,'logh'),
        error('Distr is missing the field logh');
    end
    if ~isfield(Distr,'T'),
        error('Distr is missing the field T');
    end
    if size(Distr.tau_0,2) ~= size(Distr.T,1),
        error('size(tau_0,2) is not dimT in Distr');
    end
    if ~isfield(Distr,'Eta'),
        error('Distr is missing the field Eta');
    end
    if size(Distr.Eta,1) ~= size(Distr.T,1),
        error('size(Eta,1) is not size(T,1) in Distr');
    end
    if ~isfield(Distr,'logZ'),
        error('Distr is missing the field logZ');
    end
    if ~isfield(Distr,'Theta'),
        error('Distr is missing the field Theta');
    end
    if ~isfield(Distr,'Predict'),
        error('Distr is missing the field Predict');
    end
    if ~isfield(Distr,'name'),
        error('Distr is missing the field name');
    end
end

% CHECK R_Struct -----------------------------------------------------
---

function [] = check_r_struct(R_Struct)
%CHECK_R_STRUCT checks to make sure R_list has proper format. Throws
errors
% if R_Struct does not have proper format.
    if ~isstruct(R_Struct),
        error('R_Struct needs to be a struct');
    end
    if ~isfield(R_Struct,'R')
        error('R_Struct is missing the field R');
    end
    if size(R_Struct.R,1) ~= size(R_Struct.R,2),
        error('R is not square in R_Struct');
    end
    if ~isfield(R_Struct,'r')
        error('R_Struct is missing the field r');
    end
    if R_Struct.r ~= max(R_Struct.R(:)),
```

```
        error('R and r do not agree on the number of edge bundles\n r
~= max(R) in R_Struct');
    end
    if ~isfield(R_Struct,'k')
        error('R_Struct is missing the field k');
    end
    if size(R_Struct.R,1) ~= R_Struct.k,
        error('R is not a kxk matrix in R_Struct');
    end
    if ~isfield(R_Struct,'name')
        error('R_Struct is missing the field name');
    end
end
```

References

[1] Aicher, C. (2013). The Weighted Stochastic Block Model (Version 1.0) [Source code]. (https://aaronclauset.github.io/wsbm/; Accessed March 3, 2022)

[2] Aicher, C., Jacobs, A. Z., & Clauset, A. (2015). Learning latent block structure in weighted networks. *Journal of Complex Networks*, 3, 221–248. https://doi.org/10.1093/comnet/cnu026

[3] Rubinov, M., & Clayden, J. (2013). Brain Connectivity Toolbox. [Source code] (https://sites.google.com/site/bctnet/all-functions; Accessed October 26, 2020)

[4] Strategic Engineering Research Group (SERG), Massachusetts Institute of Technology, Massachusetts Institute of Technology. (2011). MATLAB Tools for Network Analysis (2006-2011). [Source code] (http://strategic.mit.edu/downloads.php?page=matlab_networks; Accessed March 25, 2020