Regular Paper

Route Estimation Algorithm in Wireless Sensor Network Using Time-Series Traffic Trend Similarity

Yui Kobayashi†, Ryozo Kiyohara‡, and Yoshiaki Terashima†

†Soka University, Japan
‡Kanagawa Institute of Technology, Japan

Abstract - Wireless sensor network (WSN) is utilized in various fields including agriculture. However, communication quality is often difficult to maintain owing to their complicated topologies. This study presents an approach for monitoring WSN based on analyzing of the traffic sent and received by the sensors. The proposed monitoring system is protocol-independent because only traffic amount data is used.

In this method, similarity that can be seen in traffic trends is exploited for communication route estimation. However, burst traffic often occurs during wireless communication which may result in errors. Thus, the proposed technique utilizes a new similarity measure called Similar Trend Time Ratio (STTR), which focuses on overall similarity rather than temporal differences. A random forest machine learning algorithm was utilized for the optimal threshold selection. This method improves accuracy in bursting conditions as the experiment using network simulation resulted route estimation accuracy around 42%.

Keywords: Network Monitoring, Traffic Analysis, Time Series Data, Similarity Measure

1 INTRODUCTION

Wireless sensor networks (WSNs) are expected to be introduced to the field of agriculture for acquiring temperature and humidity data. A survey on WSN applications [1] shows examples of diverse network varieties for smart farming. However, these networks often have complicated topologies and hinders the maintenance of network communication and quality. In particular, WSN systems for application to agriculture have various protocols or specifications depending on the required transmission capacity or frequency of data monitoring.

In order to keep the WSN maintained, the farmer needs to know the area or devices that are unstable and require maintenance. Especially, the information about the communication route used for data collection is useful for situation awareness. Network monitoring softwares typically analyze the headers of each transmitted packet to provide information on the network condition. However, developing softwares for specific networks is laborious, expensive, and inefficient.

Therefore, the proposed method utilizes traffic amount data to estimate the communication route, which makes it protocol-independent and suitable for any network environment. Similar researches on network monitoring which are based on physical layer characteristics or collected data similarity can be found and they indicate possibilities of black-box-like approaches [2] [3]. The proposed algorithm uses traffic trend similarity that can be found in time-series traffic amount data of communicating devices. However, in some cases, burst traffic or packet drop may cause temporal significant changes in traffic amount data, which introduces inaccuracies in our similarity based method.

Table 1: Packet log and time-series traffic amount data to be input and analyzed

(a) Packet log

Timestamp (sec)	Packet size (kbyte)
0.04	48
0.40	36
2.06	68
3.40	36
4.09	72
5.42	56
5.55	60

(b) Traffic amount data

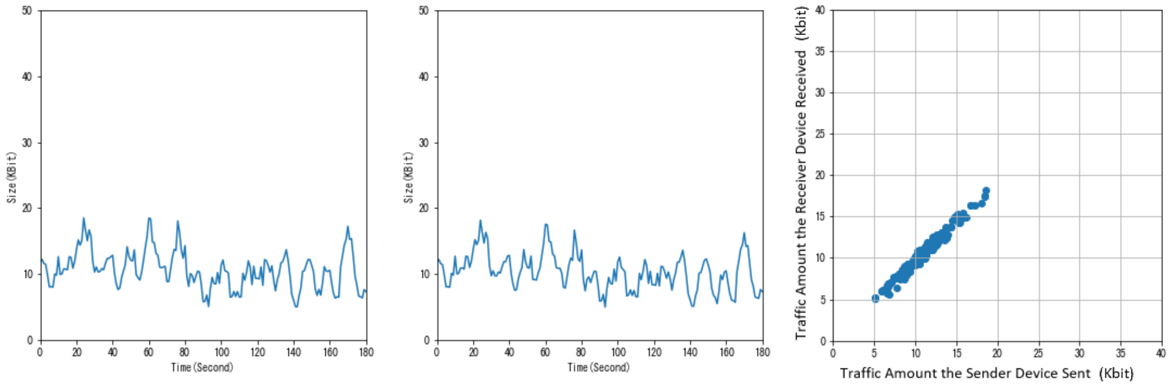
Time period (sec-sec))	Traffic amount (kbyte)
0 - 1	84
1 - 2	0
2 - 3	68
3 - 4	36
4 - 5	72
5 - 6	116

Therefore, a new similarity measurement method that is almost unaffected by burst traffic is required.

2 SIMILARITY-BASED APPROACH

The traffic data to be analyzed is the time-series data collected from each sensor, which represents the data sent and received by each device during each time interval. For example, if a sensor device has a recorded packet log as Table 1a, the time-series data are generated as shown in Table 1b. According to a survey on WSN utilizations, major protocols used for WSN are ZigBee, Wi-Fi or LoRaWAN. Each protocol has different packet format and different packet sending and receiving sequence, which hinders versatility of general network monitoring software. On the other hand, the data of traffic amount and timestamp of the packets the sensor device sent and received can be logged in the OS by adding some source codes hooked to packet events. The traffic amount data logged in each sensor device is expected to be collected every several hours via the network or by the farmers manually.

To estimate a communication route, the similarities between the traffic data of all possible pairs of devices are examined, which represent the possibilities of direct communications between the pair of the devices. Figure 1 shows an example of a pair of traffic data where the sender device sends packets to the receiver device. The similar shapes of the two graphs indicate similar traffic trends. In contrast, randomly selected pairs

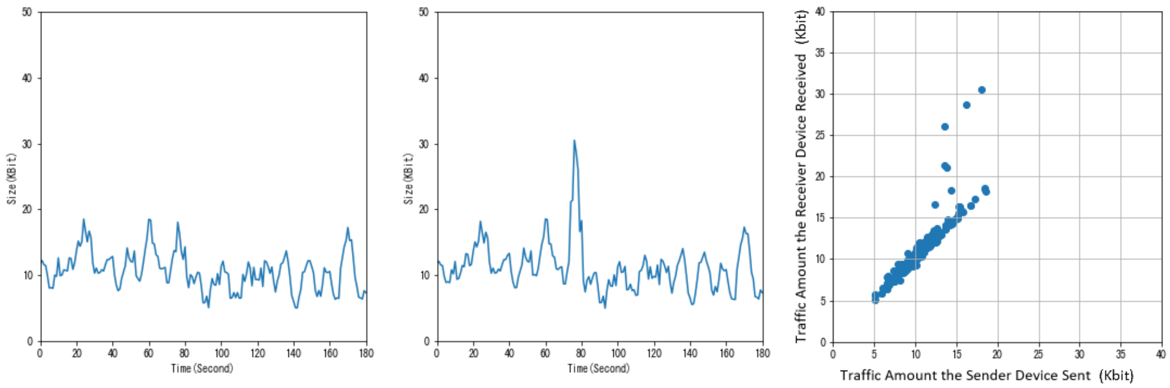


(a) Sent traffic amount data at sender device

(b) Recieved traffic amount data at reciever device

(c) Scatter plot of traffic amount at same timestamp

Figure 1: Traffic amount data of communicating devices in ideal condition



(a) Sent traffic amount data at sender device

(b) Recieved traffic amount data at reciever device

(c) Scatter plot of traffic amount at same timestamp

Figure 2: Traffic amount data of communicating devices in bursting condition

of traffic data are not similar. Figure 3b shows the correlations between the traffic data of all the devices that communicate based on a linear topology as Fig. 3a. For example, the value in the cell in row ”N2”, column ”N1”, indicates similarity between the traffic data transmitted and received by the sensor nodes ”N1” and ”N2”, respectively. Devices in the same communication route exhibit similar traffic trends, as shown by the boxes in Fig. 3b.

In this way, pair of devices which traffic amount data similar each other can be expected to be directly communicating devices. This idea can be the hints for communication route estimation.

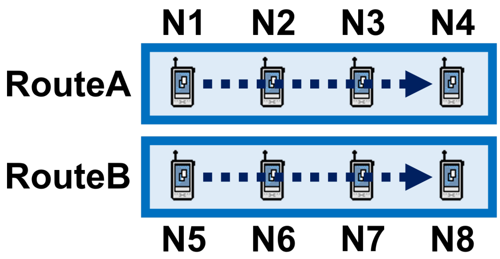
3 INACCURACY DUE TO BURST TRAFFIC

Burst traffic or packet drop, which often occurs during wireless communication, hinders accurate network estimation in the proposed similarity-based method. Figure 2, which examines temporal changes over the entire monitored interval, illustrates the significant effects of these events.

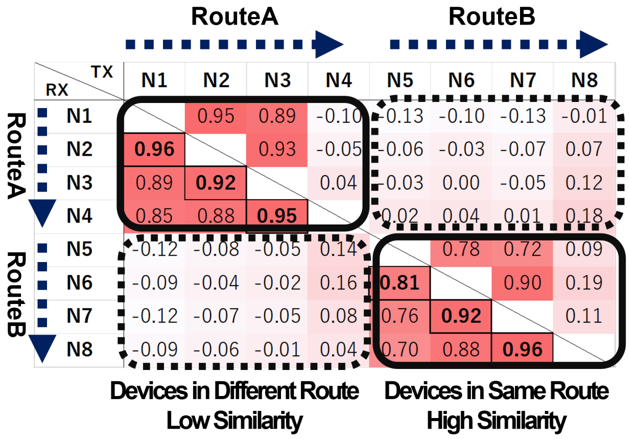
The calculated similarity scores based on basic similarity measures for time-series data, such as Euclidean distance or Dynamic Time Warping (DTW), can be adversely influenced despite the high degree of similarity observed during most of the monitored time. A general countermeasure to burst problems is smoothing using a moving average or rolling. However, smoothing eliminates traffic trend characteristics; therefore, it is not recommended for the proposed method because trends are utilized to identify communicating devices. A flexible similarity measure based on user feedback has been proposed [4], in which the user determines whether the effects of burst traffic should be retained for route estimation in certain cases, depending on the difficulty of estimation.

3.1 Pre-Experiment in Bursting Condition

This experiment examines how burst traffic affects the route estimation accuracy. The route estimation algorithm used in this experiment is explained in Sec.4 and coefficient coefficient is used as similarity measure. Estimation accuracy is the ratio of the number of devices that next hop destination



(a) Linear topology example



(b) Correlation distribution between TX and RX traffic amount data

Figure 3: Traffic Amount Data Similarity Comparison

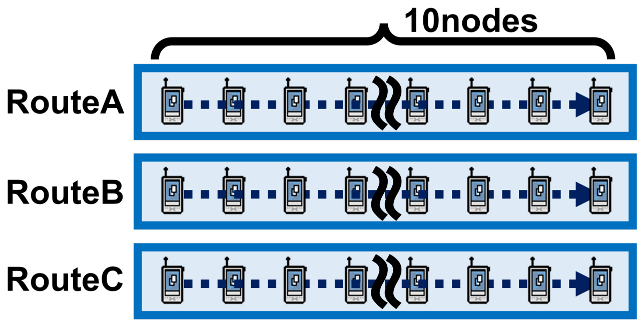


Figure 4: Network topology for the network simulation

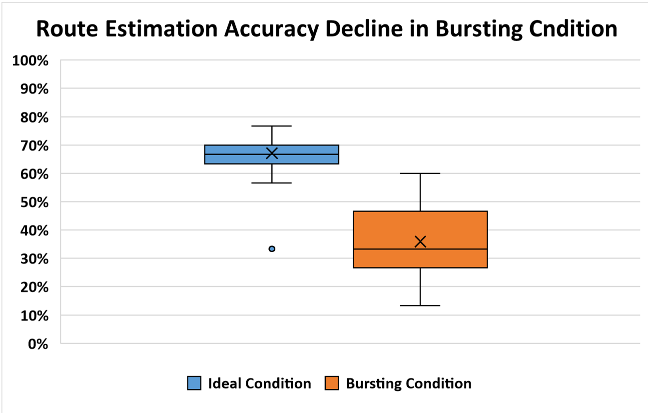


Figure 5: Pre-experiment result

is accurately estimated out of all devices and maximum results among Minimum Similarity Threshold set from 0.1, 0.2, ..., to 0.9 . The devices are placed as shown in Fig. 4 and each device sends packets constantly to the edge nodes. Network specification is set as shown in Table 2, which is designed for simulating realistic bursting condition. For ideal network condition, static communication route is specified and for realistic environment simulation, no communication route is set which makes devices decide route automatically by themselves.

One of the network simulation result is shown in Fig. 6 and Fig. 7 with their time-series traffic amount data placed in same order as Fig. 4. Several significant burst traffic can be seen in dynamic routing result and they are in different time and devices regardless of communication relations. Even not specifying communication route in bursting condition simulation, the route mainly used was same as Fig. 4 according to the simulation log.

The route estimation result (Fig. 5) indicates that satisfactory results are obtained in the ideal static routing case; however, the accuracy significantly declines in the real dynamic routing case. As explained in this section, bursting condition causes significant inaccuracy to this similarity-based route estimation method and requires proper similarity calculation method to prevent this problem.

Table 2: Parameters for pre-experiment in bursting condition

Item	Value
Network Simulation Parameters	
Trial Times	100
Simulation Time	180 seconds
Node Placement Spacing	150m
Transmission	IEEE 802.11b
Routing Protocol	OLSR INRIA
Routing	For Ideal Condition Static routing
	For Bursting Condition Dynamic routing
Application	Constant Bit Rate 1024byte/1sec
Route Estimation Parameters	
Similarity Measure	Correlation coefficient
Minimum Similarity Threshold	Optimum 0.1, 0.2, ..., 0.9

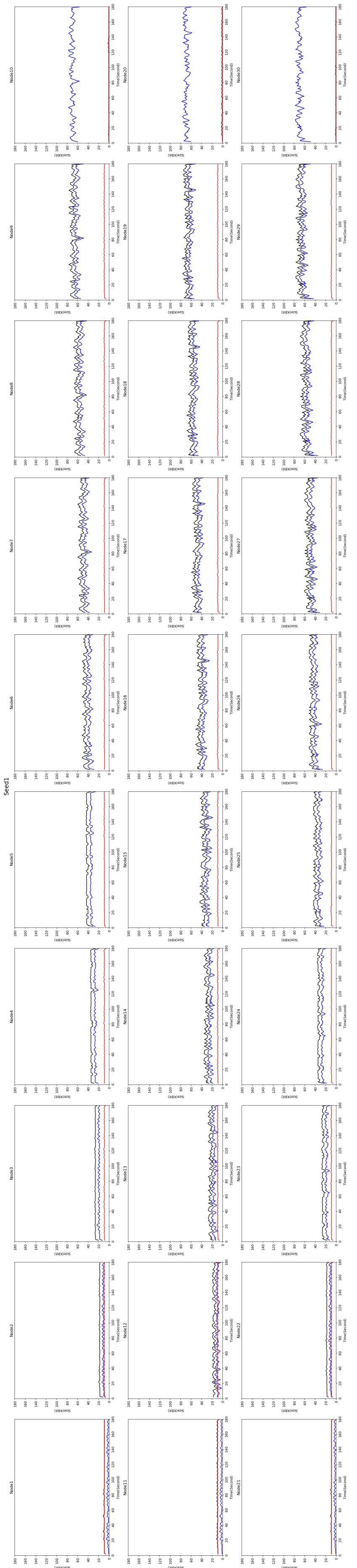


Figure 6: Ideal network condition simulation result

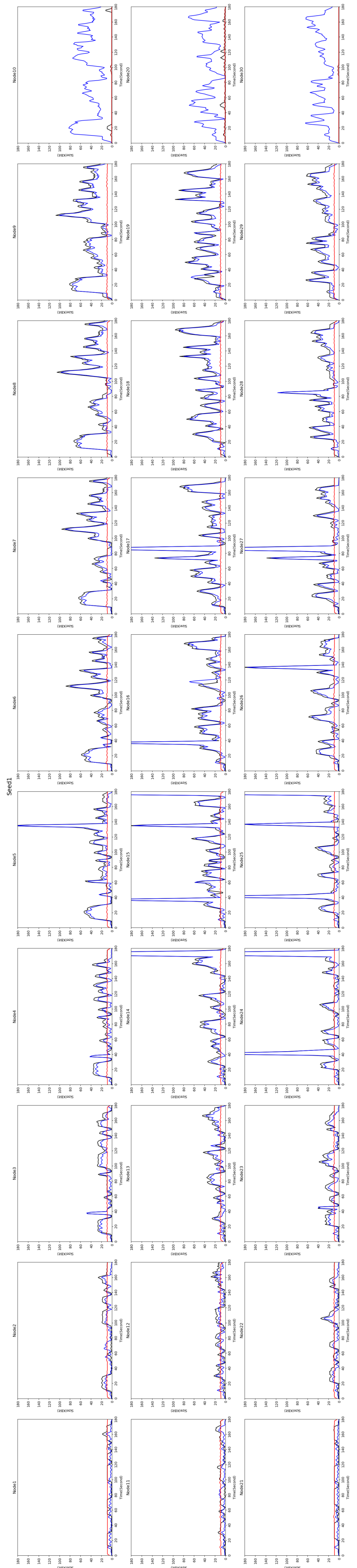


Figure 7: Bursting network condition simulation result

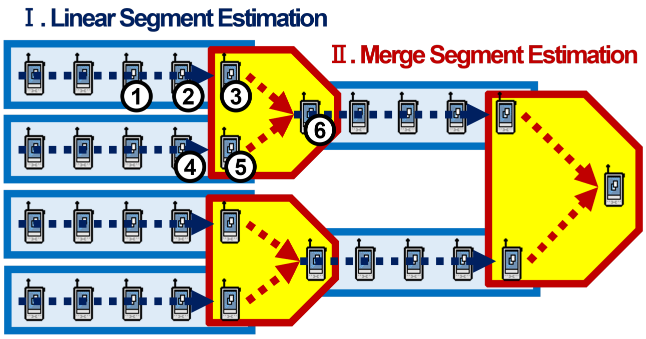


Figure 8: Communication route estimation order in tree-shaped network topology

4 PROPOSED METHOD

4.1 Overview

Figure 8 and Fig. 9 show basic process of the proposed method. When WSNs are utilized in agriculture, they are expected to have a tree-shaped network topology because they are designed to gather sensed data to the gateway device [5]. Figure 8 shows an example of a tree-shaped WSN topology with two types of segments: a simple linear segment and a merge segment. In this method, linear segments are firstly estimated, followed by merge segments estimation.

The proposed method analyzes traffic amount data to estimate the communication route and quality of each link, generating communication route map. This method can be used in any network environment regardless of its protocol type because the actual content of packets is not analyzed. After gathering time-series traffic amount data, the process of route estimation is processed through similarity calculation and route segment or merge relation estimation as shown in Fig. 9. The generated map includes network stability value for each link and is useful for farmers to understand network conditions and take countermeasures in the area where communications are not stable.

This section explains how each process contributes to the route estimation process.

4.2 STTR Similarity Measure

To address the inaccuracy problem caused by burst traffic, a new similarity measure called Similar Trend Time Ratio (STTR) is utilized, which examines the ratio of the time associated with a similar trend to the total time as a similarity criterion. Calculation process is shown in Fig. 11. STTR is the ratio of the similar time (highlighted area in the graph) to the total monitored time. The determination of whether trends are simultaneously similar is based on STTR Similarity Determination Threshold applied to the difference data. Large differences caused by burst traffic are represented as high or low values in the graph in Fig. 11. The periods in which the two data exhibit similar trends, as highlighted in the graph, are determined as similar time based on the threshold.

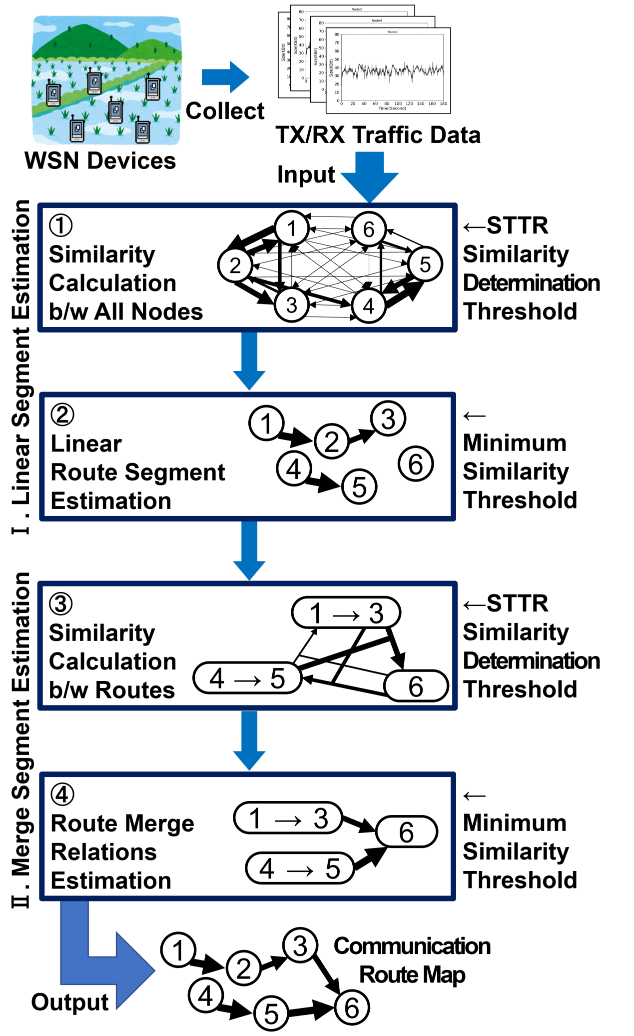


Figure 9: Network monitoring algorithm process

4.2.1 STTR Usage as Communication Stability Index

The STTR value can also be used as a metric of communication stability because higher STTR values are calculated when the burst occurrence time is shorter, and lower STTR values are calculated when it is longer. Once the communication route is estimated using the proposed method, the stability of each link can be determined using the STTR values.

Providing this value with the output communication route map enables farmers to understand the network status and easily implement measures to address any network failure.

4.2.2 Automatic STTR Similarity Determination Threshold Selection

STTR Similarity Determination Threshold is used to determine whether the traffic amount data compared at the same timestamp are similar. Figure 10 shows how different values are calculated depending on this threshold.

A lower threshold, as shown in Fig. 10a, is better in cases where most sensor devices have similar traffic because it is necessary to identify the communication relations among all the devices. However, if this threshold is too low, all calcu-

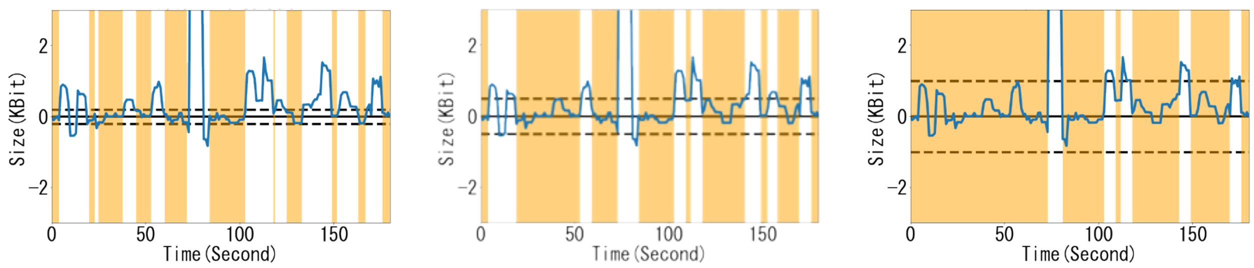


Figure 10: STTR Similarity Determination Threshold and STTR value

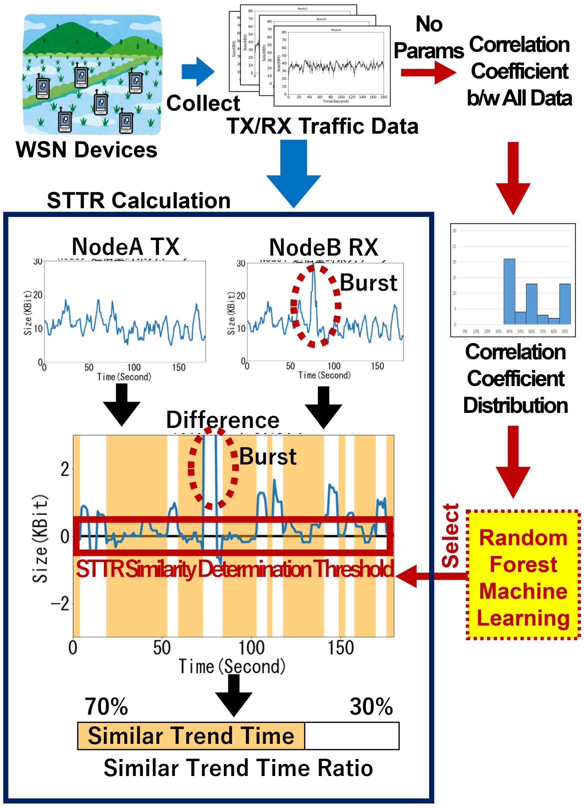


Figure 11: STTR calculation process

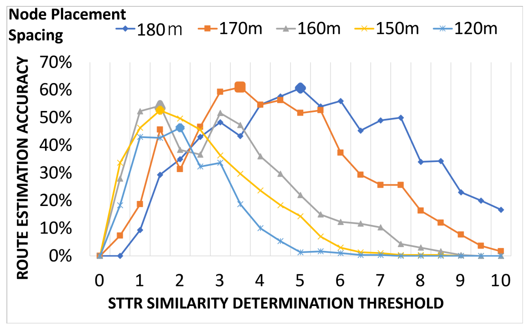


Figure 12: STTR Similarity Determination Threshold and estimation accuracy

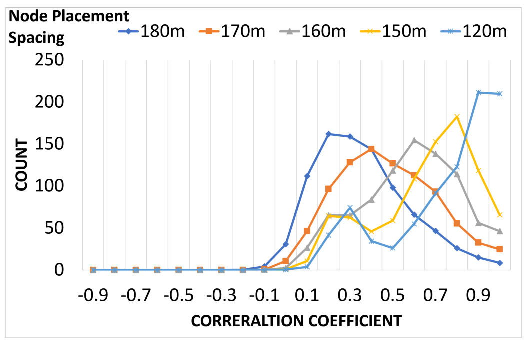


Figure 13: Correlation coefficient distribution comparison

lated values becomes too low that cannot be compared and makes it impossible to determine the communication relations. A higher threshold should be selected in cases where the general communication quality is low and burst traffic often occurs. However, when the threshold is set too high, the calculated values become too high to compare and the identification of the communicating links becomes challenging.

The effective value for this threshold differs depending on the quality and the similarity of the communication. Figure 12 shows relation between the threshold value and the route estimation accuracy, and the accuracy differs depending on the value selected for the threshold. This reveals that the optimum threshold differs according to the communication stability simulated by the node placement spacing as longer communication distance makes communication more unstable.

In order to set this threshold automatically, this method utilizes Random Forest Machine Learning. Figure 13 is a histogram of similarity values calculated by correlation coefficient and distribution differences can be seen depending on its communication distance. In order to focus on effects by STTR Similarity Determination Threshold, a fixed value is set for Minimum Similarity Threshold in this experiment.

The calculation of the correlation coefficient does not require the setting of any parameters, and it is easy to analyze the general similarity distribution. Although similarity evaluation using the correlation coefficient is not useful for route estimation using the proposed method, it is useful for determining the average communication quality and how similarly the sensors communicate. Figure 3b is an example of similarity distribution calculated using the correlation coefficient. For example, if the communication quality is low and a high similarity value is not found in Fig. 3b, this indicates that STTR Similarity Determination Threshold should be set lower. If most sensors send similar traffic, the average similarity in Fig. 3b will be high, which indicates that STTR Similarity Determination Threshold should be sufficiently high to identify the communicating links. To analyze these similarity distribution, making histogram is useful method and Fig. 13 clearly shows how differences in distribution can be revealed using histogram.

This is how this method uses Random Forest Machine Learning to estimate the optimum value for STTR Similarity Determination Threshold based on the similarity distribution calculated using the correlation coefficient.

4.3 Linear Route Segment Estimation

The process of estimating actual route segment from calculated similarity is performed as shown in Fig. 14. The figures in each step shows example sensor nodes, candidate communication links with STTR similarity values and the estimated communication routes. The sensor nodes are represented as circles with their node numbers inside and the STTR similarity values are represented as arrow thickness.

After similarity calculation step is done, there are too many candidate links that can be a part of actual communication route. Therefore, before starting route estimation, links are filtered by applying Minimum Similarity Threshold to the similarity value of each link. In Step1 of Fig. 14 as an example, the Minimum Similarity Threshold is set 50%, so the links with the similarity values below 50% were deleted from the candidate links. By applying this filter, candidate links are greatly reduced and calculation time is shortened.

After the filtering process, the candidate link with the highest STTR value is selected as the estimated route (Step2). Then, the estimated route is expanded from the both edge nodes, choosing the link with the highest similarity among all other possible links (Step3). Repeating this process until no possible link remains, the all linear communication segments are estimated (Step4 and Step5).

4.4 Route Merge Relations Estimation

After linear part estimation is done, merge parts are estimated. Merge part estimation is processed in mostly same way as linear part estimation. Looking at the estimated linear route segments, the edge devices are sending or receiving packets to other route segment. In a tree-shaped topology where merge parts gather packets from 2 routes into 1 route as shown in Fig. 8, the total sent traffic amounts of the proper pair of linear route segments should be similar to the traffic amount that have been received at the another linear route segment. Using this logic, similarity between all possible pairs of route edge devices are calculated and merge relations are estimated. Combining the results of linear route segment estimation and merge relation estimation, the whole communication routes of tree-shaped topology network are estimated and provided to the farmers for the network situation awareness.

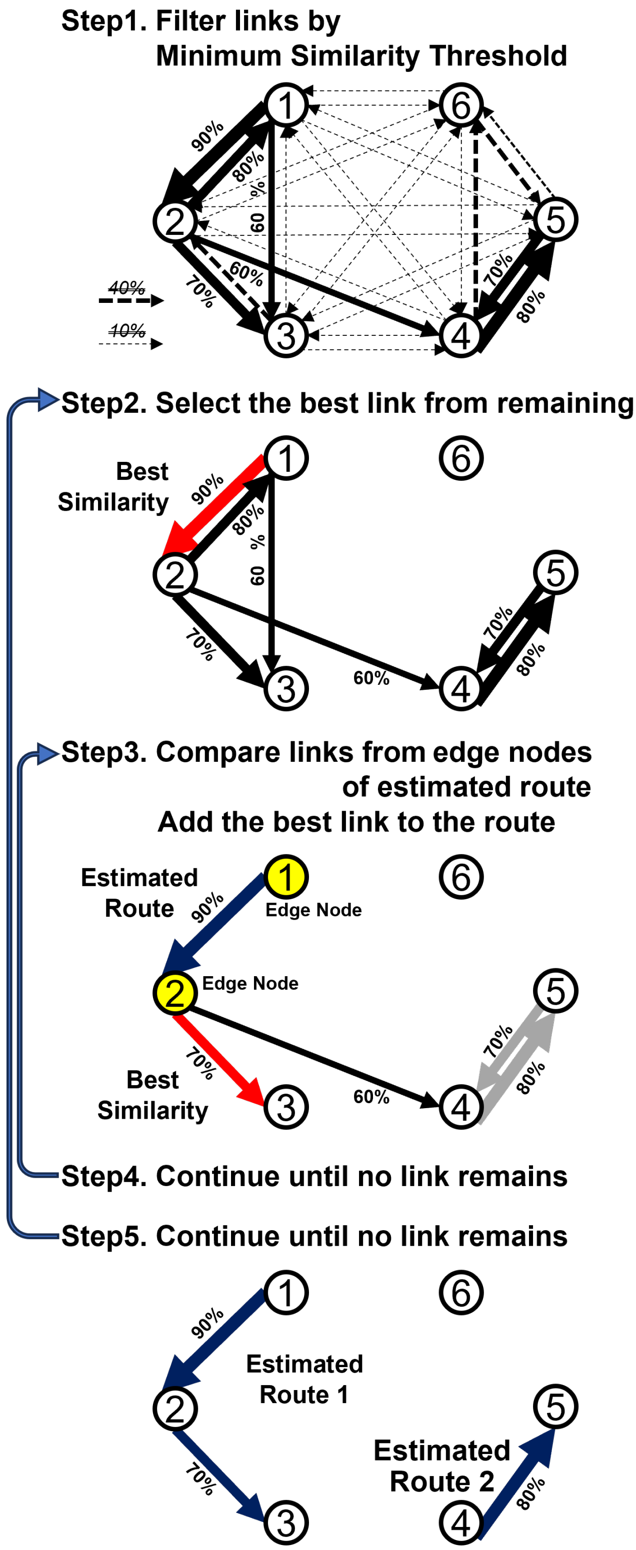


Figure 14: Linear route segment estimation process

Table 3: Parameters for overall performance experiment

Item	Value
Network Simulation Parameters	
Trial Times	100
Simulation Time	180 seconds
Node Placement Spacing	150m
Transmission	IEEE 802.11b
Routing Protocol	OLSR INRIA Dynamic routing
Application	Constant Bit Rate 1024byte/1sec
Route Estimation Parameters	
Similarity Measure	Correlation coefficient, STTR
Minimum Similarity Threshold	For Correlation Coefficient Optimum 0.1, 0.2, ..., 0.9
	For STTR 30%

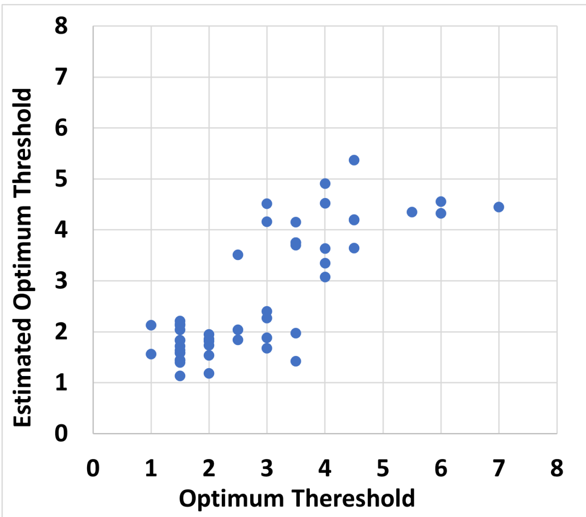


Figure 15: Relations between actual optimum threshold and estimated threshold

5 EVALUATION

This section evaluates the availability of the proposed method using the traffic amount data generated by network simulation using QualNet [6]. Following two experiments indicated the proposed algorithm worked properly and had contribution on accuracy improvement in bursting condition.

5.1 Automatic STTR Similarity Determination Threshold Selection Experiment

This experiment evaluates the accuracy of the optimum threshold estimation using machine learning.

The accuracy exhibited an average error of 0.68 for the relationship between the actual optimum and estimated thresholds. Figure 15 shows this relationship and indicates that this method typically yields thresholds close to the actual optimum value. Owing to this threshold error, the route estimation accuracy declines by approximately 10%. Figure 16 shows actual route estimation accuracy using the estimated STTR Similarity Determination Threshold. The result reveals its stable performance regardless of its communication stability simulated using different device spacing distances.

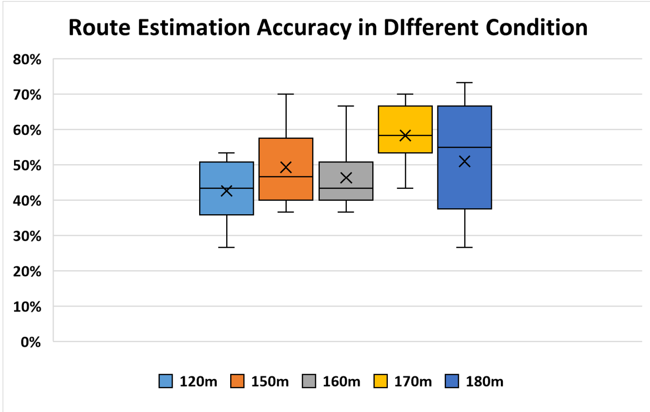


Figure 16: Automatic STTR Similarity Determination Threshold selection experiment result

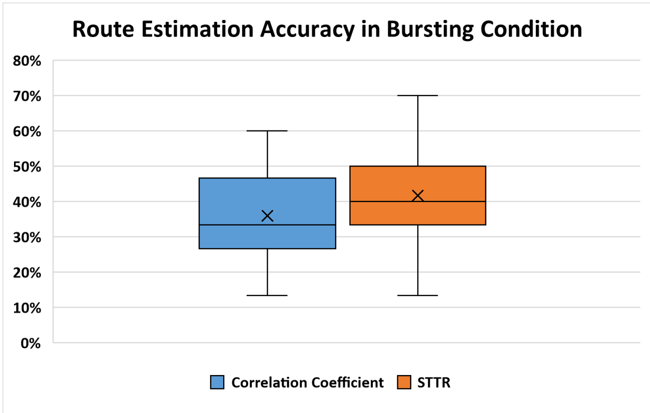


Figure 17: Overall performance experiment result

5.2 Overall Performance Experiment in Bursting Condition

This experiment examines route estimation accuracy improvements contributed by proposed similarity measure. Table 3 is the parameters set for this experiment. The result of network simulation is same as the pre-experiment in Sec.3.1 and shown in Fig. 7.

As for the route estimation result, average accuracy was improved from 36% to 42% and clearly reveals the contribution to the accuracy in unstable communication condition. The required accuracy level of the proposed method is expected to be around 70%-80% since the proposed method focuses on versatility just to notify farmers the area or sensors that have to be maintained. For this purpose, the result of 42% accuracy is still useful because the communication routes that were failed to be estimated were where the similarity in traffic data was low because of low communication stability and need some maintenance. Therefore, the proposed method has enough feasibility to provide useful information about the WSN conditions to the farmers.

One factor hindered the accuracy is that train data used for machine learning in this experiment is from the experiment of Sec.5.1 and does not include data of bursting condition. Therefore, the most optimal threshold was not necessarily selected for STTR Similarity Determination Threshold. Adding train data in various bursting condition will bring further improved accuracy. Another factor is that static threshold for Minimum Similarity Threshold was used for the STTR case unlike the correlation coefficient case took maximum result among several values. The accuracy is expected to be improved by implementing adjustment process of this threshold.

6 CONCLUSION

In this study, we propose a method for network route and quality estimation by analyzing the traffic of constituent sensors, which enables versatility in any network environment. This approach utilizes similarities that can be observed in the traffic amount trends of communicating devices. The similarity is calculated using the STTR for accurate route estimation even in the presence of burst traffic. For optimum threshold selection, we utilized a random forest machine learning algorithm using correlation distribution. The experiments have shown the proposed method using STTR increases accuracy in bursting condition compared to using similarity value calculated by correlation coefficient. This method is useful for providing network conditions with its versatility for assessing network quality in any environment, enabled by automatic threshold selection. In future work, further accuracy improvement through training by data in various network topologies and implementation of another threshold adjustment function will be investigated.

REFERENCES

[1] M. N. Mowla, N. Mowla, A. F. M. S. Shah, K. M. Rabie, and T. Shongwe, "Internet of Things and Wireless Sensor Networks for Smart Agriculture Applications: A Survey," *IEEE Access*, vol. 11, pp. 145813–145852, (2023).

[2] J. H. Tyler, D. R. Reising, F. I. Kandah, E. Kaplanoglu, and M. M. K. Fadul, "Physical Layer-Based IoT Security: An Investigation Into Improving Preamble-Based SEI Performance When Using Multiple Waveform Collections," *IEEE Access*, pp. 133601–133616, (2022).

[3] A. Muruganantham, M. Veluchamy, and S. Erulappan, "Distributed Similarity Based Clustering and Compressed Forwarding for Wireless Sensor Networks," *ISA Transactions*, vol. 59, pp. 180–192, (2015).

[4] C. Yin, H. Ishikawa, and Y. Takama, "Proposal of Time Series Data Retrieval with User Feedback," *2014 IEEE International Conference on Granular Computing (GrC)*, Noboribetsu, Japan, pp. 358–361, (2014).

[5] S. Carretero, J. Calderón, A. Marinescu, M. C. Guerra, and B. Borja, "A Novel Methodology for the Monitoring of the Agricultural Production Process Based on Wireless Sensor Networks," *Computers and Electronics in Agriculture*, vol. 76, pp. 252–265, (2011). DOI: 10.1016/j.compag.2011.02.004

[6] Japan QualNet Community Forums, "https://simweb.kke.co.jp/qualnet/forum/"

(Received: November 15, 2024)
(Accepted: March 13, 2025)

Yui Kobayashi Yui Kobayashi is a graduate student of Soka University. His currrent research intersts inculde network monitoring method using traffic amount analysis. He is also interested in Vehicle-to-Everything (V2X) network, drone network and Wireless Sensor Network (WSN), which are utilized for monitoring purpose. He is a member of the Information Processing Society of Japan (IPSJ).

Ryozo Kiyohara Ryozo Kiyohara received B.E., and M.E. degrees from Osaka University in 1983, and 1985. Since joining Mitsubishi Electric Corporation in 1985, he had been engaged in developing a machine translation system. From 1989, he had been at the Institute for the New Generation Computing Technologies engaged in the Fifth Generation Computing Project until he returned to Mitsubishi Electric Corporation in 1992. He received Ph.D. in 2008 from Osaka University in Information Science and Technology. He is a professor of Kanagawa Institute of Technology since 2012. His current research interests include software upgrade systems, Java processing in on-vehicle information devices, Driver Monitoring System and various Network applications. He is a member of IEEE(Senior Member), ACM, Fellow of IPSJ(Information Processing Society of Japan), and IEICE (The Institute of Electronics, Information and Communication Engineers).

Yosihaki Terashima Yoshiaki Terashima is a professor at Soka University in Japan. He worked at the Mitsubishi Electric Corporation's Information Technology R&D Center from 1984 to 2015 and received his Ph.D. in 2006. His research interests include automatic generation of communication software based on formal specifications, distributed processing architectures, and testing methodologies for wireless networks. He is a member of the IEEE, the Information Processing Society of Japan (IPSJ), and the Institute of Electronics, Information, and Communication Engineers (IEICE).