이학박사 학위논문

Multiscale Analysis of Spatio-Temporal Data

시공간 자료의 다중척도 분석

2019년 8월

서울대학교 대학원

통계학과

박 선 철

Multiscale Analysis of Spatio-Temporal Data

by

Seoncheol Park

A Thesis
submitted in fulfillment of the requirement
for the degree of
Doctor of Philosophy
in
Statistics

The Department of Statistics
College of Natural Sciences
Seoul National University
August, 2019

Abstract

Multiscale Analysis of Spatio-Temporal Data

Seoncheol Park
The Department of Statistics
The Graduate School
Seoul National University

This thesis presents a multiscale analysis of spatio-temporal data. The content of this thesis consists of three chapters.

First, we suggest an enhancement of the lifting scheme, one of the popular multiscale method, by using clustering-based network design. The proposed method is originally developed for enhancement of graph signal data, and the simulation and real data analysis results show that the proposed method has the advantage to reconstruct the noisy data compared to conventional lifting scheme method. Moreover, the advantage of the proposed method is not limited to the graph signal denoising. It is also shown that the proposed method the proposed neighborhood selection is able to combine with lifting one coefficient at a time (LOCAAT) algorithm, which is a lifting scheme algorithm frequently used in signal denoising.

Second, we suggest a new lifting scheme concept which could be applied for streamflow data. It is impossible to apply the original lifting scheme to streamflow data directly because of its complex structure. In this thesis, to adapt the concept of lifting scheme to streamflow data, we suggest a new lifting scheme algorithm for streamflow data with flow-adaptive neighborhood selection, flow proportional weight generation, and flow-length adaptive removal point selection. By using the proposed method, we can successfully construct a multiscale analysis of streamflow data. Simulation study supports the performance of the lifting scheme for streamflow data

is competitive for signal denoising. Besides, the proposed methods can visualize the multiscale structure of the network by adding or subtracting observations.

Third, multiscale analysis for particulate matter data in Seoul is provided as a case study. We suggest a new method, which is a novel combination of multiscale analysis and extreme value theory. The study starts from the idea that every climate event has its spatial or temporal event lengths. By changing the event area and duration time, we can estimate multiple extreme value parameters using generalized extreme value (GEV) distribution. Besides, we suggest a new property, called "piecewise scaling property" to combine multiple GEV estimators into a single equation. By using the proposed method, we can construct a return level map with arbitrary duration time and event area.

Keywords: Multiscale method, Lifting scheme, Streamflow data, Generalized extreme value distribution.

Student Number: 2013-20215

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Conventional statistics and data analysis usually have been developed under a single fixed scale. For example, statisticians only have considered yearly time series data or monthly time series data separately, not simultaneously. However, due to the accumulation of the data and appearance of complex data, it is not easy to catch all the information of the data using a single scale. Multiscale analysis solves the problem by proving a variety of detail feature of the data using various scales.

Multiscale methods are well explained in some references, including Daubechies (1992) and Ferreira and Lee (2007). Ferreira and Lee (2007) introduce a variety of multiscale methods. According to Ferreira and Lee (2007), multiscale methods include processes such as fractal, algorithms, and data. One of the popular multiscale techniques is wavelets, which have been researched in the 1990s and the 2000s. However, it is known that wavelets are not easy to adapt real data analysis when the length of the data is not dyadic, or it is not lying on the Euclidean domain.

Lifting scheme, first suggested by Sweldens (1996) and Sweldens (1998), can be understood as a generalization of wavelet methods to overcome such limitations. There are some works in statistics using lifting scheme such

as Nunes *et al.* (2006), Knight *et al.* (2009), Jansen *et al.* (2009), Knight *et al.* (2012), Knight *et al.* (2017), Hamilton *et al.* (2018) and Knight *et al.* (2019). They use an algorithm named lifting one coefficient at a time (LOCAAT), which decomposes the original signal by removing one coefficient at each level. In LOCAAT, it is not easy to choose the optimal remove order.

Recently, Martínez-Enríquez *et al.* (2018a) and Martínez-Enríquez *et al.* (2018b) suggest a new lifting scheme algorithm and apply it to the graph signal data. They make a new algorithm by starting from an empty set and adding important points under specific criteria. They show that one can data-adaptively find important points selection under the above algorithm with the simplest form of graph signal called generalized moving average (GMA) model.

There are many improvements to generalize the lifting scheme on more complex data. Nevertheless, Previous works on the lifting scheme have some limitations. In graph signal denoising, if the signal structure is more complicated than that of GMA model, there is a chance to improve the performance by suggesting a new algorithm. Besides, there are no previous studies about lifting scheme on streamflow dataset, whose domain is an example that conventional methods do not work well. In this thesis, we suggest new methods to solve these problems.

On the other hand, almost all climate events are also considered at a unique temporal and spatial scale. However, some climate events, especially extreme climate events, have their own spatial scale and temporal duration. In this thesis, we propose a case study of particulate matter (PM_{10}) extremes in Seoul with consideration of multiple scales.

In this thesis, there are three kinds of topics we focus on. In Chapter 3, we propose an enhancement method of lifting scheme on graph signal data via clustering-based network design. In Chapter 4, we provide a new lifting

scheme method for streamflow data called streamflow lifting scheme. In Chapter 5, we analyze real PM_{10} extremes data in Seoul with a combination of multiscale method and extreme value theory. Concluding remarks are provided in Chapter 6.

Chapter 2

Review: Multiscale analysis

In this chapter, we briefly review the concept of the multiscale analysis, focusing on wavelets and lifting scheme. Fourier analysis is the first method to understand signals using frequency information. Wavelets are developed to consider time and frequency information simultaneously. Wavelets have limitations caused by computation algorithms, which only works well on a dyadic dataset. Lifting scheme makes up these weak points of wavelets. Wavelets and lifting scheme can be understood under the concept of multiresolution analysis.

2.1 Wavelets

Wavelets are a system of orthonormal basis functions, are usually defined in $L^2(\mathbb{R})$. Wavelet bases are generated by translations and dilations of the father wavelet ($\phi(\cdot)$), and mother wavelet ($\psi(\cdot)$) functions (Haris *et al.*, 2018). Mother wavelet satisfies a zero average condition (Mallat, 1999),

$$\int_{-\infty}^{\infty} \psi(x)dx = 0. \tag{2.1}$$

Nason (2008) called (2.1) as an oscillation, implying that the wavelet goes up and down.

If one specifies a type of mother wavelet, one can generate wavelets using dilation and translation operations (Nason, 2008). Let us denote a resolution level, $j_0 \geq 0$. The concept of wavelets allows us to represent a function $f \in L^2(\mathbb{R})$ as

$$f(x) = \sum_{k=0}^{2^{j_0}-1} \alpha_{j_0 k}\phi_{j_0 k}(x) + \sum_{j=j_0}^{\infty}\sum_{k=0}^{2^j-1} \beta_{jk}\psi_{jk}(x), \tag{2.2}$$

where

$$\phi_{jk}(x) = 2^{j/2}\phi(2^j x - k), \qquad \psi_{jk}(x) = 2^{j/2}\psi(2^j x - k).$$

The coefficients $\alpha_{j_0 k}$ and β_{jk} are called the father and mother wavelet coefficients, respectively. The index j is called the resolution level.

The mother wavelet ψ and the dilation and translation parameters $a, b \in \mathbb{R}$, $a \neq 0$ generates called a wavelet family, $\{\psi_{jk}(x)\}_{j,k}$, where each element is represented by a rescaled and a translated version of the mother wavelet:

$$\psi_{jk}(x) = \frac{1}{\sqrt{|j|}}\psi\Big(\frac{x-k}{j}\Big).$$

The normalization parameter $\frac{1}{\sqrt{|j|}}$ is needed in order to preserve the L^2-norm of ψ : $\|\psi_{jk}\|_{L^2} = \|\psi\|_{L^2}$. More details about wavelets can be found in Daubechies (1992) or Nason (2008).

2.1.1 Haar transforms

The most straightforward and well-known wavelet bases are Haar bases. The construction of Haar wavelets starts with the simple scaling function

$$\phi(x) = \begin{cases} 1, & 0 \leq x < 1 \\ 0, & \text{otherwise.} \end{cases} \tag{2.3}$$

Consider a dyadic vector of data $\mathbf{x} = (x_1, \ldots, x_N)$, where the length of data N is dyadic, i.e., $N = 2^J$, for some non-negative integer J. Haar

wavelets extract "detail" information of data from the subtraction of every even and odd data.

$$d_k = x_{2k} - x_{2k-1}, \qquad k = 1, \ldots, \frac{n}{2}. \tag{2.4}$$

If odd and even values are similar, d_k will be small. However, if x_{2k} and x_{2k-1} have different values, the corresponding d_k will be large. The next step is gaining information using a summation:

$$c_k = x_{2k} + x_{2k-1}, \qquad k = 1, \ldots, \frac{n}{2}. \tag{2.5}$$

By repeating (2.4) and (2.5), we can get a coarser level of data.

2.2 Multiresolution analysis

Multiresolution analysis (MRA), introduced by Mallat (1989), explains the property of an approximation of $f \in L^2(\mathbb{R})$ at different resolutions by projecting f on a sequence of approximation spaces. Using MRA, wavelets are defined by a multiresolution analysis of $L^2(\mathbb{R})$. In this thesis, a summary of multiresolution analysis is provided, following notations of Knight (2006).

Definition 1 (Multiresolution analysis (MRA))**.** *An (orthogonal) multiresolution analysis (MRA) of $L^2(\mathbb{R})$ consists of a sequence of successive approximation of a space of functions V_j, with the following properties:*

1. *V_j is a closed subspace of $L^2(\mathbb{R}), \forall j \in \mathbb{Z}$,*
2. *Separation property: $\bigcap_{j\in\mathbb{Z}} V_j = \{0_{L^2(\mathbb{R})}\}$.*
3. *Density property: $\overline{\bigcup_{j\in\mathbb{Z}} V_j} = L^2(\mathbb{R})$.*
4. *Nesting property: $V_j \subset V_{j+1}, \forall j \in \mathbb{Z}$, hence $\{0\} \subset \cdots \subset V_{-1} \subset V_0 \subset V_1 \subset \cdots \subset L^2(\mathbb{R})$.*
5. *Scaling relation: $f(x) \in V_j \iff f(2x) \in V_{j+1}, \forall j \in \mathbb{Z}$.*

6. *There exists a scaling function* $\phi(x) \in V_0$ *with* $\int_{-\infty}^{\infty} \phi(x)dx = 1$, *such that* $\{\phi(x-k)\}_{k\in\mathbb{Z}}$ *forms an orthonormal basis of* V_0.

In this definition, large j means a fine-scale and a finer approximation space V_j. According to Knight (2006), MRA construction has important implications: (i) Condition 3 and 4 imply that any function $f \in L^2(\mathbb{R})$ can be gradually approximated by its projections on the $\{V_{j\in\mathbb{Z}}\}$ spaces. (ii) Condition 5 implies the multiresolution property: any V_j is a scaled version of V_0 - $f(x) \in V_0 \iff f(2^j x) \in V_j, \forall j \in \mathbb{Z}$.

2.3 Lifting scheme

It is known that wavelets are not appropriate methods when the data is not observed on regular grids or the number of observations is not dyadic, i.e., $n = 2^J$, for some $J \in \mathbb{Z}$. To overcome them, Sweldens (1996) and Sweldens (1998) proposed a new kind of second-generation wavelet called "lifting scheme". The lifting scheme has been extensively studied in signal processing and image analysis (Jansen and Oonincx, 2005).

Suppose that $\boldsymbol{x} = (x_1, \ldots, x_N)^T$ represents signal values for each node. The length of the data ($= N$) may not be dyadic. Assume that we want to construct a multiresolution transform at $j-1$-th level, given j-th level data $\boldsymbol{x}_j$. The following four steps describe the lifting scheme:

1. **Split**: At each level $j-1$, divide nodes of the data vector at j level $\boldsymbol{x}_j$ into two subsets, $\mathcal{P}_{j-1}$ and $\mathcal{U}_{j-1}$. We denote i for nodes in set $\mathcal{P}_{j-1}$ and k for nodes in set $\mathcal{U}_{j-1}$.

2. **Predict**: Predict every sample $x_{j,i} \in \mathcal{P}_{j-1}$ from $x_{j,k} \in \mathcal{U}_{j-1}$ with a prediction filter $\mathbf{p}_{j-1,i}$ and store the prediction error $d_{j-1,i}$

$$d_{j-1,i} = x_{j,i} - \hat{x}_{j,i} = x_{j,i} - \sum_{k\in\mathcal{N}_{j-1,i}\cap\mathcal{U}_{j-1}} p_{j-1,i,k}x_{j,k},$$

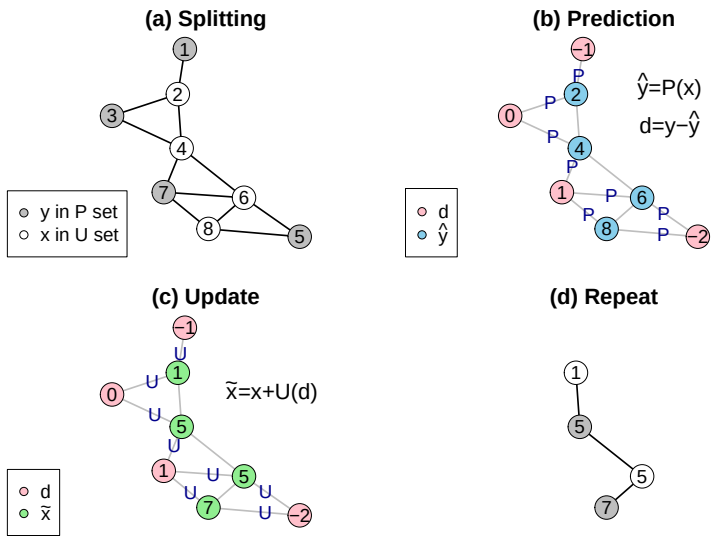

Figure 2.1: A brief description of the lifting scheme of graph signal data. (a) In the splitting step, we divide the data into two subsets, called $\mathcal{P}$ and $\mathcal{U}$ set. (b) In prediction step, we predict node values in $\mathcal{P}$ set with $\mathcal{U}$ set. (c) In the update step, we update values in $\mathcal{U}$ set with an appropriate update filter. (d) We repeat (a)$\sim$(c) at the next level with a coarser level of data.

where $\hat{x}_{j,i}$ represents the value of predicted values constructed from $\mathcal{U}_{j-1}$ neighbors of node i.

3. **Update**: Compute an update version of data at $j-1$ level $x_{j-1,k}$ in $\mathcal{U}_{j-1}$ with an appropriate update filter $\mathbf{u}_{j-1,k}$,

$$x_{j-1,k} = x_{j,k} + \sum_{i \in \mathcal{N}_{j-1,k} \cap \mathcal{P}_{j-1}} u_{j-1,k,i} d_{j-1,i}.$$

4. **Repeat**: Repeat the above steps until we reach the desired resolution level.

By iterating these steps, we can generate coarse signals from updated subsamples. Figure 2.1 briefly shows lifting scheme steps in a fixed level of an example graph signal data. On the other hand, the inverse version

of the lifting scheme algorithm can be easily obtained by undoing forward lifting scheme operations at each level $j-1$.

1. Undo update: $x_{j,k} = x_{j-1,k} - \sum_{i \in \mathcal{N}_{j-1,k} \cap \mathcal{P}_{j-1}} u_{j-1,k,i} d_{j-1,i}$.
2. Undo predict: $d_{j,i} = d_{j-1,i} + \sum_{k \in \mathcal{N}_{j-1,i} \cap \mathcal{U}_{j-1}} p_{j-1,i,k} x_{j,k}$.
3. Undo split.
4. Repeat: repeat the above steps at the next level.

Sweldens (1998) introduces the lifting algorithm also under the theory of a second-generation biorthogonal MRA. Knight (2006) summarizes that there are some differences between the lifting scheme and classical wavelet analysis: (i) The basis functions are no longer dilations and translations of a function. (ii) The filters are both location and scale-dependent, allowing for usage on irregular data. More details about the second generation multiresolution analysis, refer to Knight (2006).

Finally, we summarize some crucial ingredients for the construction of the lifting scheme.

- **The number of remove points at ones** ($|\mathcal{P}|$): The user should select how many points remain at the next (coarser) level. In conventional methods, including Martínez-Enríquez *et al.* (2018b), predetermined proportions which are similar to traditional wavelet methods, are used. (Jansen *et al.*, 2009) suggest a new removal scheme called lifting one coefficient at a time (LOCAAT). The main idea of the algorithm is removing only one coefficient at each level. That means, if we have length-n data, we can construct a variety of coarser data up to $n-1$ level. LOCAAT algorithm is more precisely explained in Section 2.3.1.

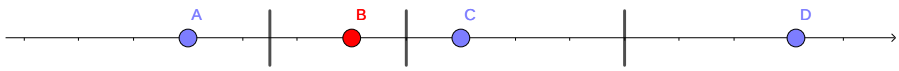

Figure 2.2: Illustration of removal order selection of lifting scheme in the real line. A, B, C, D denotes locations of data. One can define an area of each point by diving the real line into four blocks (vertical lines). LOCAAT algorithm chooses a point which has the smallest area among candidates. In this example, point B is selected to be removed.

- **Prediction filter**: It is essential to choose which prediction filter we will use. In lifting scheme literature, Haar (local constant), local linear, local polynomial or inverse distance weight Jansen *et al.* (2009) are frequently used. Nunes *et al.* (2006) suggest a unified approach, by selecting one of those filters minimizing the energy of detail coefficients, combined with LOCAAT algorithm.

- **Remove point selection**: when one decides the number of remove points, it is crucial to choose which location should be removed first. It is related to a question of which points are important or not to represent the underlying field. The simplest way to decide the order is by removing points from the densest sampled regions, finding the point which has the smallest integral, illustrated in Figure 2.2. ((Jansen *et al.*, 2009), (Nunes *et al.*, 2006)) Knight *et al.* (2009) suggest an enhanced approach similar to the ensemble method, popular in data mining. She generated many lifting schemes using predetermined, randomly drawn trajectories. After then, the denoising result is improved by computing ensemble mean of all lifting results. In signal processing, Martínez-Enríquez *et al.* (2018b) suggest an alternative approach by minimizing the expectation of the total prediction error (E_{tot}),

$$E_{tot} = \sum_{i \in \mathcal{P}} \mathbb{E}\{(y_i - \hat{y}_i)^2\} = \sum_{i \in \mathcal{P}} \mathbb{E}\{d_i^2\}.$$

- **Neighborhood selection**: It is crucial to select an appropriate number of neighbors to construct a prediction filter. Too many neighborhoods make it hard to catch a local behavior of the data, to small neighborhood makes a bias to predict each node. Therefore, it is important to choose an appropriate set of neighborhoods. The usual choice is the nearest n neighbors. Martínez-Enríquez *et al.* (2018b) suggest a data-adaptive method by finding an optimal neighborhood, minimizing the total prediction error from the set of all possible neighborhoods.

2.3.1 Lifting one coefficient at a time (LOCAAT)

In this subsection, we briefly introduce the lifting one coefficient at a time (LOCAAT) algorithm, suggested by Jansen *et al.* (2009). The idea of LOCAAT algorithm is constructing a removal order of data points and sequentially decomposes data with the predetermined order. Suppose we have the values $f_1, \ldots, f_n$ of a function f, sampled at n irregularly spaced points $x_1, \ldots, x_n$ on the real line. The usual nonparametric regression problem starts with an additive model setting:

$$f_i = g(x_i) + \varepsilon_i, \tag{2.6}$$

where $\varepsilon_1, \ldots, \varepsilon_n$ are random variables that denote the noise, usually assumed to be independently distributed, with zero mean and finite variance. Lifting scheme approximates the initial function f by

$$\tilde{f}(x) = \sum_{k=1}^{n} c_{n,k}\phi_{n,k}(x), \tag{2.7}$$

where $c_{n,i} := f(x_i)$ and $\phi_{n,k}(x_i) = \delta_{i,k}$ for $k, i \in \{1, \ldots, n\}$.

In the initial stage, LOCAAT algorithm defines the set of indices corresponding to the scaling coefficients is $\mathcal{U}_n = \{1, \ldots, n\}$, and the set of wavelet indices is empty, $\mathcal{P}_n = \emptyset$. At the next step, $n - 1$, we choose a

point to be lifted and denote its index by j_n. It will be the point that will be removed from the current set of scaling coefficients and converted into a detail coefficient. The new set of indices corresponding to the scaling coefficients is $\mathcal{U}_{n-1} = \mathcal{U}_n \backslash \{j_n\}$, while $\mathcal{P}_{n-1} = \{j_n\}$ is the set of wavelet indices constructed at this stage.

To choose the point to be lifted, Jansen *et al.* (2009) used the minimum of the integral of scaling function $\phi_{n,k}$ with respect to some suitable measure, denoted by I_{nk}. In this thesis, we consider the length or volume as suitable measures. The point to be lifted, (x_{j_n}, c_{n,j_n}) is chosen such that the point with the smallest scaling function integral, described in Figure 2.2, which corresponds to the point with the finest detail. To construct an update filter, Jansen *et al.* (2009) suggest minimum norm solution based update weights at level r for reasons of numerical stability,

$$b_j^r = I_{ri_r} I_{r-1,j} / \sum_{k \in \mathcal{N}_r} I_{r-1,k}^2, \tag{2.8}$$

where i_r is an index of remove candidate point, $\mathcal{N}_r$ is set of neighborhoods of node i_r at level r, and $j \in \mathcal{N}_r$ is a neighbor of i_r.

2.3.2 Other lifting scheme methods

With the suggestion of LOCAAT, there are some related articles concerning lifting scheme on statistics.

Nunes *et al.* (2006) propose a new lifting scheme method called "adaptive lifting". The main idea of adaptive lifting is twofold: (i) the data-adaptive selection of the order of the regression and (ii) neighborhood size in the lifting prediction step. Through these modifications, Nunes *et al.* (2006) give the flexibility to construct prediction filters under the one-dimensional signal denoising setting.

In the lifting scheme, finding an optimal removal order is not easy to decide. To overcome it, Knight *et al.* (2009) suggest a "nondecimated"

concept into the lifting transform. It borrows the idea from the nondecimated wavelet transform (NDWT), which improves the performance of the wavelet transform by using over-determined basis. Knight *et al.* (2009) produce many sequences of removal order called paths. They show that by combining multiple signal denoising results from different paths, one can reduce the error between the actual and estimated signal.

Recently, Hamilton *et al.* (2018) propose a complex-valued lifting scheme. The main idea of the complex-valued lifting (C-lifting) scheme allowing two prediction schemes at once. However, the current C-lifting method restricts its neighborhood construction up to two points. There are some researches for spectral estimation of long memory estimation with a lifting scheme. For spectral estimation, Knight *et al.* (2012) suggest a lifting scheme based spectral estimation method for locally stationary time series with missing observations. Knight *et al.* (2019) suggest long memory estimation for complex-valued time series.

Chapter 3

Enhancement of lifting scheme on graph signal data via clustering-based network design

Lifting scheme is a useful multiscale method when data are lying in an irregular domain, or the number of data is not dyadic. There are several previous works applying lifting scheme to high-dimensional or graph signal data to use its flexibility. Jansen *et al.* (2009) suggests the first attempt; they consider a multiscale concept of network data. To define neighborhoods of network data, they use minimal spanning trees. Furthermore, they use Voronoi polygons to make neighborhood structures on a spatial domain.

On the other hand, it is unknown that the general rule of minimizing prediction error in the lifting scheme algorithm, especially in LOCAAT setting. Knight *et al.* (2009) suggest an ensemble combination of different

removal orders to reduce the estimation error. Recently, Martínez-Enríquez *et al.* (2018a) and Martínez-Enríquez *et al.* (2018b) suggest an optimal greedy update/prediction assignment (UPA) under the assumption that the graph data are generated by a simple graph signal structure called generalized moving average (GMA) model with a bottom-up based assignment setting, different from LOCAAT method.

In this chapter, we expand the idea of Martínez-Enríquez *et al.* (2018b) to more general settings. Most of the signal values are not constant, rather complicated. In this thesis, we consider graph signals have piecewise constant signal values. To reflect more complicated characteristics of a data, we suggest a new multiscale analysis for spatio-temporal data by constructing a bottom-up network design to represent a natural multiresolution phenomenon of data. Besides, by adopting clustering information, we can have better results compared with previous studies.

This chapter is organized as follows. In Section 3.2, we briefly summarize previous works of lifting scheme on graph signal data. In Section 3.3, we introduce an extension of the GMA model called the piecewise generalized moving average model. We show that our proposed clustering-based neighborhood selection method is useful to reduce the expected value of the total prediction error (E_{tot}) under the piecewise generalized moving average model setting. In Section 3.4 and Section 3.5, simulation and real data analysis are provided, showing that our proposed method works well under certain situations.

3.1 Graph notations

In this section, we briefly summarize graph notations used in this paper, following notations of (Martínez-Enríquez *et al.*, 2018b). A undirected graph is denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, where $\mathcal{V} = \{1, \ldots, N\}$ is a set of nodes,

$\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ a set of edges and $\mathbf{W} = [w_{mn}]$ is the weighted adjacency matrix, with w_{mn} is a non-negative weight of the edge, where $mn \in \mathcal{E}$. In spatio-temporal graph with no spatio-temporal interactions, we classify edges into two classes, spatial ($\mathcal{S}$) and temporal ($\mathcal{T}$) edges, with $\mathcal{S} \cup \mathcal{T} = \mathcal{E}$. Let $\mathcal{N}_i^{\mathcal{S}} = \{j : ij \in \mathcal{S}\}$ ($\mathcal{N}_i^{\mathcal{T}} = \{j : ij \in \mathcal{T}\}$) denote spatial (temporal) set of neighborhoods of i for all nodes $i \in \mathcal{V}$. $|\mathcal{N}_i \cap \mathcal{U}|$ is the sum of weights of node m for all its nearby $\mathcal{U}$ set edges. The degree of a node m is the sum of weights of all its edges, $D_m = \sum_{n \in \mathcal{N}_m} w_{mn}$. If the graph is an unweighted graph, the degree is proportional to the complexity of the network.

Sometimes it is hard to choose a set of optimal weights, especially the Euclidean distance is not meaningful to understand the correlation of a set of data, especially in graph signal. In this chapter, we focus on the most straightforward case; every edge has the same weight, that is, equally weighted.

3.2 Previous works

To perform a lifting scheme on graphs, we need to decide which nodes should be removed in the lifting prediction step. It coincides solving the problem of how to split node in j level, i.e., $\mathcal{U}_j$ into two subsets, $\mathcal{P}_{j-1}$ and $\mathcal{U}_{j-1}$.

The weighted maximum-cut (WMC) problem is one of the classical approaches to solve the graph splitting problem by finding an optimal partition of the node set of a graph into two disjoint subsets which have the maximum sum of the weights between elements of both subsets. However, UPA based on WMC only depends on the edge structure, not signal values. That is, if the underlying network edge cannot reflect the differences of signals, it may cause a problem.

To get data-adaptive graph-partitions, Martínez-Enríquez *et al.* (2018a)

and Martínez-Enríquez *et al.* (2018b) suggest a greedy algorithm minimizing the energy of detail coefficients based on the minimization of predicted errors. That means when we fix the number of nodes in the prediction set as $|\mathcal{P}|$, they find the UPA minimizing the total prediction error,

$$E_{tot} = \sum_{i \in \mathcal{P}} \mathbb{E}\{(x_i - \hat{x}_i)^2\} = \sum_{i \in \mathcal{P}} E\{d_i^2\}. \tag{3.1}$$

According to Knight *et al.* (2009), it is not easy to find an optimal removal path if we use LOCAAT algorithm under the general functional regression structure. Martínez-Enríquez *et al.* (2018a) and Martínez-Enríquez *et al.* (2018b) solve the problem in graph signal data by designing a new algorithm starting from all nodes are in $\mathcal{P}$ set. In Martínez-Enríquez *et al.* (2018b), they analyze UPA problem under the assumption that data are generated from generalized moving average (GMA) model

$$x_m = c + \sum_{n \in \mathcal{N}_{[m]}} q_{m,n}\epsilon_n + \eta_m, \tag{3.2}$$

where c is a constant, ϵ_m and η_m are independent and identically distributed noise random variable and $q_{m,n}$ represents the contribution of neighbor n for the node m.

3.3 The use of clustering under the piecewise generalized moving average model

Although the GMA model is a basic model in the graph signal processing, it has a limitation due to its simplicity. One problem is that the signal part of the GMA model. If the original graph signal has K ($K \in \mathbb{N}$) distinct values, the edge structure cannot reflect the similarity of nodes due to the difference between c values. In such cases, a clustering-based network design method, which is suggested in this thesis, is helpful to reduce the expectation of total prediction error.

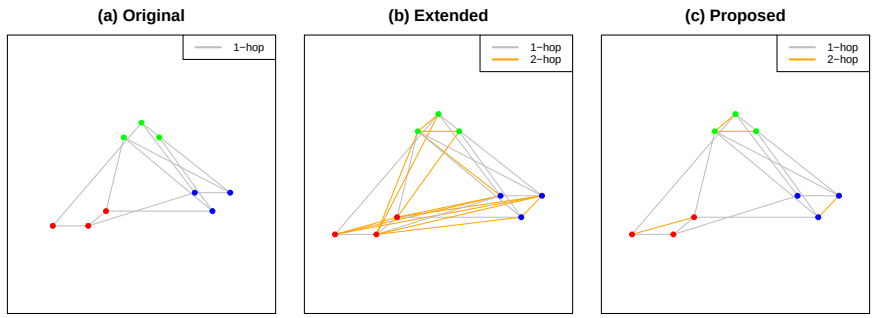

Figure 3.1: (a) Original graph signal network. (b) 2-hop graph signal network construction result. (c) Proposed clustering-based network designs.

When the data consists of a combination of piecewise values, there is a possibility that one can improve the performance of the lifting scheme, i.e., minimizing the energy of detail signals by adding or even subtracting edges along with cluster information. Figure 3.1 shows a description of the proposed method. Assume that the data are generated from a simple network structure in (a). Different colors mean signal values. We call edges generated from the original graph signal as "1-hop edges". We can create more edges by adding new edges of two-hop neighbors, i.e., neighbors of the neighbors. We call them as "2-hop edges", which are drawn in orange lines in Figure 3.1 (b). By eliminating 2-hop edges connecting nodes with difference signal values, we obtain the proposed network design, shown in Figure 3.1 (c).

3.3.1 Piecewise generalized moving average model

In this subsection, we extend the concept of GMA model to a more general setting while having piecewise constant signal values in each node. It brings the definition of a piecewise generalized moving average (PGMA) model.

Definition 2 (Piecewise generalized moving average (PGMA) model)**.**

$$x_m = c_m + \sum_{n \in \mathcal{N}_{[m]}} q_{m,n}\epsilon_n + \eta_m, \tag{3.3}$$

where c_m is a signal part of node m assuming that its value can be clustered into some values, i.e, for every $m \in \mathcal{V}$, $c_m \in \{c_{i_1}, \cdots, c_{i_K}\}$, where $i_1, \ldots, i_K$ are index of K clusters. $q_{m,n}$ is a constant representing correlation between node m and n. ϵ_n and η_m are zero-mean independent (and identically distributed) random variables, with fixed variances σ_ϵ^2 and σ_η^2. Equation (3.3) can be also written a vector form,

$$\mathbf{x} = \mathbf{c} + \mathbf{Q}\boldsymbol{\epsilon} + \boldsymbol{\eta}, \tag{3.4}$$

where $\mathbf{Q}$ is a matrix with components $q_{m,n}$ for $n \in \mathcal{N}_{[m]}$ and zero otherwise.

In the PGMA model, we can also define linear predictors and corresponding prediction error, following arguments of Martínez-Enríquez *et al.* (2018b).

Definition 3 (Linear predictors for the clustered version of PGMA model)**.** *Assume a given $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, $\mathbf{x}$ and UPA. Then linear predictors for PGMA model are defined as:*

$$\hat{x}_i = \sum_{k \in \mathcal{N}_i^* \cap \mathcal{U}} p_{i,k} x_k, \tag{3.5}$$

or using vector form,

$$\hat{\mathbf{x}} = \mathbf{P}^* \mathbf{x}, \tag{3.6}$$

where $\mathbf{P}^*$ is a matrix with components $p_{i,j}^*$ for $j \in \mathcal{N}_i^* \cap \mathcal{U}$ and zero otherwise and $\mathcal{N}_i$ is the set of neighbors of node i in $\mathcal{G}$.

Lemma 3.3.1 (PGMA prediction error)**.** *For a node $i \in \mathcal{P}$, signal $\mathbf{x}_i$ is defined by the definition of PGMA model. Then, the total prediction error*

is given by

$$E_{tot} = \sum_{i \in \mathcal{P}} \mathbb{E}\{(x_i - \hat{x}_i)^2\} = \sum_{i \in \mathcal{P}} E_{PGMA_i}, \qquad \textit{where} \tag{3.7}$$

$$\begin{aligned} E_{PGMA_i} &= \mathbb{E}\{x_i\}^2 - 2\mathbf{p}_i^T \mathbf{K}_{\mathcal{U}_i,i} + \mathbf{p}_i^T \mathbf{K}_{\mathcal{U}_i\mathcal{U}_i} \mathbf{p}_i \\ &= c_i^2 + \mu_{i,i} + \sigma_\eta^2 - 2\mathbf{p}_i^T (\mathbf{M}_{\mathcal{U}_i} \mathbf{e}_i + \mathbf{c}_{\mathcal{U}_i} c_i) \\ &+ \mathbf{p}_i^T (\mathbf{M}_{\mathcal{U}_i\mathcal{U}_i} + \sigma_\eta^2 \mathbf{I} + \mathbf{c}_{\mathcal{U}_i} \mathbf{c}_{\mathcal{U}_i}^T) \mathbf{p}_i, \end{aligned} \tag{3.8}$$

where $\mathcal{U}_i = \mathcal{N}_i \cap \mathcal{U}$ *is the set of* $\mathcal{U}$*-neighbors of node* i, $\mathbf{K} = \mathbb{E}\{\mathbf{x}\mathbf{x}^T\}$ *is the correlation matrix,* $\mathbf{p}_i$ *is the column vector that has components* $p_{i,k}, k \in \mathcal{N}_i \cap \mathcal{U}$, *and* $\mathbf{M} = \sigma_\epsilon^2 \mathbf{Q}\mathbf{Q}^T$ *is a matrix with components with* $\mu_{m,n} = \sigma_\epsilon^2 \sum_{l \in \mathcal{N}_{[m]} \cap \mathcal{N}_{[m]}} q_{m,l} q_{n,l}$.

Algorithm 1 Greedy UPA algorithm with clustering-based network design

---

1: Set an initial graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, length-$R$ cluster sets $\mathcal{B} = (B_1, \ldots, B_R)$, weights $\hat{w}_s$, $\hat{w}_t$, $\mathcal{U} = \{\emptyset\}$, $\mathcal{P} = \{\mathcal{V}\}$ and the optimal number of $\mathcal{P}$ nodes $|\mathcal{P}_{\text{opt}}|$.

2: Following $\mathcal{B}$, define a new neighborhood of node $i \in B_r$, $\mathcal{N}_i^* = \mathcal{N}_i \cap B_r, r = 1, \ldots, R.$, where station i is in a cluster B_r. It is identical to change the original matrix $\mathbf{W}$ to the neighborhood corrected version, $\mathbf{W}^*$.

3: For each node $v = 1, \ldots, |\mathcal{P}|$, repeat $(a) \sim (c)$ while $|\mathcal{P}| > |\mathcal{P}_{\text{opt}}|$.

- a Set $\mathcal{U} \leftarrow \mathcal{U} \cup \{v\}$, $\mathcal{P} \leftarrow \mathcal{P} \backslash \{v\}$ and estimate $\hat{y}_i = \sum_{k \in \mathcal{N}_i^*} \mathbf{p}_i(y_k)$.
- b Compute empirical version of the total prediction error $\hat{E}_{\text{tot}}$, $\hat{E}_{\text{tot}} = \sum_{i \in \mathcal{P}} (y_i - \hat{y}_i)^2$.
- c Find v^* that minimizes $\hat{E}_{\text{tot}}$. Set $\mathcal{U} \leftarrow \mathcal{U} \cup \{v^*\}$, $\mathcal{P} \leftarrow \mathcal{P} \backslash \{v^*\}$.

4: Return $\mathcal{U}$ and $\mathcal{P}$.

If one does not know all parameters in (3.3), we have to estimate $\hat{q}_{m,n}$. In this case, we can use the estimator in Martínez-Enríquez *et al.* (2018b).

$$\hat{q}_{m,n} = \left\{ \begin{array}{ll} \frac{\sum_{n\in\mathcal{N}_m} w_{m.n}}{|\mathcal{N}_m|} \frac{1}{\sum_{n'\in\mathcal{N}_m} w_{m,n'}(1+\frac{1}{|\mathcal{N}_m|})}, & \text{if } m = n, \\ w_{m,n} \frac{1}{\sum_{n'\in\mathcal{N}_m} w_{m,n'}(1+\frac{1}{|\mathcal{N}_m|})}, & \text{o.w.} \end{array} \right\} \tag{3.9}$$

To find an optimal partition, (Martínez-Enríquez *et al.*, 2018b) suggested a greedy algorithm minimizing E_{tot} for a given $|\mathcal{P}|$. In this thesis, we modify the original algorithm, suggested in Algorithm 1, by adding a step which computes clustering result and modifies $\mathbf{W}$ matrix. Note that Algorithm 1 starts from all nodes are in $\mathcal{P}$ set. It is different from conventional approaches such as LOCAAT, which starts from all nodes are in $\mathcal{U}$ set.

3.3.2 Optimal UPA assignment under the piecewise homogeneous model

To derive the optimal UPA for the PGMA model, we consider the piecewise homogeneous noise model, which is a PGMA model with $q_{m,n} = 0$, for all set of nodes $\{m, n\}$. Despite its simplicity, it can be easily used in nonparametric regression.

Definition 4 (Piecewise homogeneous noise model)**.** *A piecewise homogeneous noise model (PHNM) is a PGMA model with* $q_{m,n} = 0$*,*

$$x_m = c_m + \eta_m. \tag{3.10}$$

To compare the performance of our estimator, first we assume the same setting as in (Martínez-Enríquez *et al.*, 2018b). That means, we set length-m_i prediction vector $\mathbf{p}_i$ as $\mathbf{p}_i^T = (1/m_i, \ldots, 1/m_i)^T$. Thus, the linear predictor of x_i is

$$\hat{x}_i = \frac{1}{m_i} \sum_{k\in\mathcal{N}_i\cap\mathcal{U}} x_k. \tag{3.11}$$

By using the result of Lemma 3.3.1, we can easily derive the following proposition.

Proposition 3.3.2 (PHNM prediction error)**.** *The total prediction error of PHNM is given by*

$$E_{tot} = \sum_{i \in \mathcal{P}} \mathbb{E}\{(x_i - \hat{x}_i)^2\} = \sum_{i \in \mathcal{P}} E_{PHNM_i}, \tag{3.12}$$

where

$$\begin{aligned} E_{PHNM_i} &= c_i^2 + \sigma_\eta^2 - 2\mathbf{p}_i^T \mathbf{c}_{\mathcal{U}_i} c_i + \mathbf{p}_i^T \mathbf{c}_{\mathcal{U}_i} \mathbf{c}_{\mathcal{U}_i}^T \mathbf{p}_i + \sigma_\eta^2 \mathbf{p}_i^T \mathbf{p}_i \\ &= \sum_{i \in \mathcal{P}} \{(\mathbf{c}_{\mathcal{U}_i}^T \mathbf{p}_i - c_i)^2 + \sigma_\eta^2 (1 + \mathbf{p}_i^T \mathbf{p}_i)\} \\ &= \sum_{i \in \mathcal{P}} \{(bias)^2 + variance\}, \end{aligned} \tag{3.13}$$

where $\mathcal{U}_i = \mathcal{N}_i \cap \mathcal{U}$ *is the set of* $\mathcal{U}$*-neighbors of node* i*. When and appropriate clustering is given and removal order is fixed,* $\hat{y}_i^{Proposed}$ *gives a lower* E_{tot} *than* $\hat{y}_i^{Enriquez}$*.*

In PHNM setting, we can reduce E_{PHNM_i} by just changing the neighborhood structure like (c) in Figure 3.1. By doing that, we can reduce both bias and variance part. If we use clustering-based edge modification, we can get a new prediction filter, $\mathbf{p}_i^*$. Then

$$\begin{aligned} E_{\mathrm{PHNM}_i} &= c_i^2 + \sigma_\eta^2 - 2\mathbf{p}_i^T \mathbf{c}_{\mathcal{U}_i} c_i + \mathbf{p}_i^T \mathbf{c}_{\mathcal{U}_i} \mathbf{c}_{\mathcal{U}_i}^T \mathbf{p}_i + \sigma_\eta^2 \mathbf{p}_i^T \mathbf{p}_i \\ &= \sum_{i \in \mathcal{P}} \{(\mathbf{c}_{\mathcal{U}_i}^T \mathbf{p}_i - c_i)^2 + \sigma_\eta^2 (1 + \mathbf{p}_i^T \mathbf{p}_i)\} \\ &\geq \sum_{i \in \mathcal{P}} \{(\mathbf{c}_{\mathcal{U}_i}^T \mathbf{p}_i^* - c_i)^2 + \sigma_\eta^2 (1 + \mathbf{p}_i^{*T} \mathbf{p}_i^*)\}, \end{aligned} \tag{3.14}$$

since $|\mathbf{c}_{\mathcal{U}_i}^T \mathbf{p}_i^* - c_i| \leq |\mathbf{c}_{\mathcal{U}_i}^T \mathbf{p}_i - c_i|$ and $\mathbf{p}_i^T \mathbf{p}_i \geq \mathbf{p}_i^{*T} \mathbf{p}_i^*$.

We have an interest in the case of getting enhanced results with cluster-based network denoising while the number of edges in the graph signal is reduced, described in (b) and (c) of Figure 3.1. However, it depends on the

graph data structure since it is a bias-variance trade-off problem. In this thesis, we will show that our proposed network design, which is described in (c) in Figure 3.1 outperforms in the piecewise signal value setting through simulation studies.

3.3.3 Extension to the spatio-temporal data

One of the distinct characteristics of spatio-temporal expansion is that there are two (or many) different types of data. In spatio-temporal data setting, we can extend the concept of (3.3) into space-time version, called piecewise constant spatio-temporal model by considering an additive version of PGMA model. Suppose that data $\mathbf{y} = \{y_i\}_{i=1}^N$ are generated by

$$y_i = \Big(c_i + \frac{w_s}{|\mathcal{N}_{[i]}^s|} \sum_{n \in |\mathcal{N}_{[i]}^s|} \epsilon_n + \frac{w_t}{|\mathcal{N}_{[i]}^t|} \sum_{m \in |\mathcal{N}_{[i]}^t|} \epsilon_m \Big) + \eta_i, \tag{3.15}$$

where w_s, w_t are spatial and temporal weight, $w_s + w_t = 1$, $\mathcal{N}_{[i]}^s$ and $\mathcal{N}_{[i]}^t$ are spatial and temporal neighbors of node i including itself and ϵ_n and η_i are zero-mean independent random variables.

The most straightforward linear predictor is

$$\begin{aligned} \hat{y}_i^{\text{Proposed}} &= \sum_{k \in (\mathcal{N}_i^s \cap B_r) \cap \mathcal{U}} \mathbf{p}_s^*(y_k) + \sum_{k' \in (\mathcal{N}_i^t \cap B_r) \cap \mathcal{U}} \mathbf{p}_t^*(y_{k'}) \\ &= \frac{\hat{w}_s \sum_{k \in (\mathcal{N}_i^s \cap B_r) \cap \mathcal{U}} y_k}{|(\mathcal{N}_i^s \cap B_r) \cap \mathcal{U}|} + \frac{\hat{w}_t \sum_{k' \in (\mathcal{N}_i^t \cap B_r) \cap \mathcal{U}} y_{k'}}{|(\mathcal{N}_i^t \cap B_r) \cap \mathcal{U}|}. \end{aligned} \tag{3.16}$$

The weights w_s and w_t are usually estimated that minimize the quadratic prediction error over all the nodes $i \in \mathcal{V}$.

The idea of a new algorithm for the spatio-temporal data, lifting a one time series at a time (LOTAAT) borrows from the concept of functional geostatistics. Suppose that data are observed in regular time points like usual time series. If we can measure of similarity of each time series, one constructs a new lifting scheme algorithm by removing or adding one time

series at a time simultaneously, not one coefficient at a time. Note that in the LOTAAT setting, we do not need to estimate w_s or w_t since we remove or add one time series at a time.

3.4 Simulation study

In this section, we show that the proposed concept can be applied to a variety of dataset including stochastic block model, block signal test function of Donoho and Johnstone (1994), image dataset which has many piecewise distinct colors, etc. For each study, we use the empirical version of root mean square prediction error over all nodes in $\mathcal{P}$ as

$$\sqrt{\hat{E}_{tot}} = \sqrt{\frac{1}{|\mathcal{P}|}\sum_{i\in\mathcal{P}}\sum_{i\in\mathcal{P}}(x_i - \hat{x}_i)^2} = \sqrt{\frac{1}{|\mathcal{P}|}\sum_{i\in\mathcal{P}}\sum_{i\in\mathcal{P}}(d_i)^2}, \tag{3.17}$$

where x_i is the actual signal value of the node i and $\hat{x}_i$ the predicted value of the node i. For simplicity, we consider Haar-like filter (3.11) for all simulation study. In this setting, we can maximize the effect of clustering-based network design. We consider UPA methods, randomly selected (RA), (weighted) maximum-cut (MC), (Martínez-Enríquez *et al.*, 2018b)'s method (ME) and the proposed method (PM). In most cases, we compare PM to ME, which is the most competitive method. Especially, all simulation studies are conducted under the situation that ME model have more edges compared to PE model, described in Figure 3.1 (b). Therefore, we can get more effective UPA result through PM model under the piecewise constant signal or image data.

3.4.1 Stochastic block model

In this subsection, we assume that simulated signals are generated from the stochastic block model (SBM). SBM is a kind of model for random

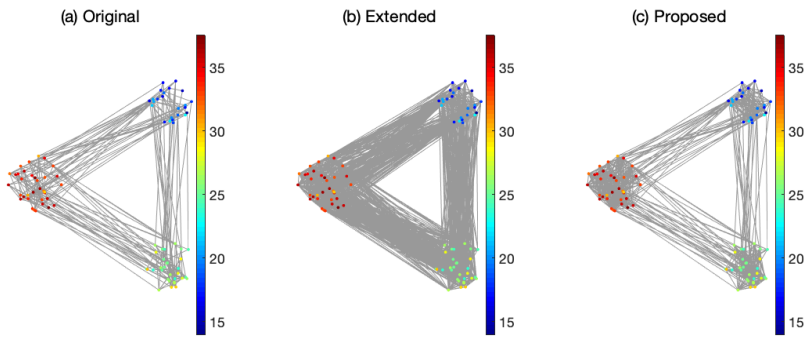

Figure 3.2: One realization of simulation data generated from spatial block model. (a) The original data generated from the given edge structure (gray lines). (b) The extended network design by adding length 2 (2-hop) edges and (c) the result of the proposed clustering-based edge reconstruction.

graphs. It consists of N nodes, and they include K disjoint subsets called communities. Predetermined edge probabilities decide edge structures.

In this thesis, we consider an SBM model with intra-cluster edge probability (the probability of having an edge between two nodes in the same cluster) of $1/8$ and inter-cluster edge probability (the probability of having an edge between two nodes in different clusters) of $1/64$. For simplicity, we consider Haar-like prediction filter, which gives local constant weights for its $\mathcal{U}$ neighbors. We generate $N = 90$ graph signals from the model with three clusters, while each cluster has constant values $10, 18$, and 26 (case $\mu_d = 8$) or $10, 14$, and 18 (case $\mu_d = 4$). We set $\sigma_\epsilon = 2$ or $\sigma_\epsilon = 3$ and fixed $\sigma_\eta = 1$. Figure 3.2 is one realization of the simulated data.

While assuming that we already know the number of clusters, we divide all nodes into three clusters. We compute $\sqrt{\hat{E}_{tot}}$ with 50 iterations, under various the number of predictor nodes $|\mathcal{P}|$ and time length $T \in \{1, 20, 40\}$. For $T = 20$ and $T = 40$ case, we use LOTAAT concept.

Table 3.1 and Figure 3.3 show the simulation results. They show that

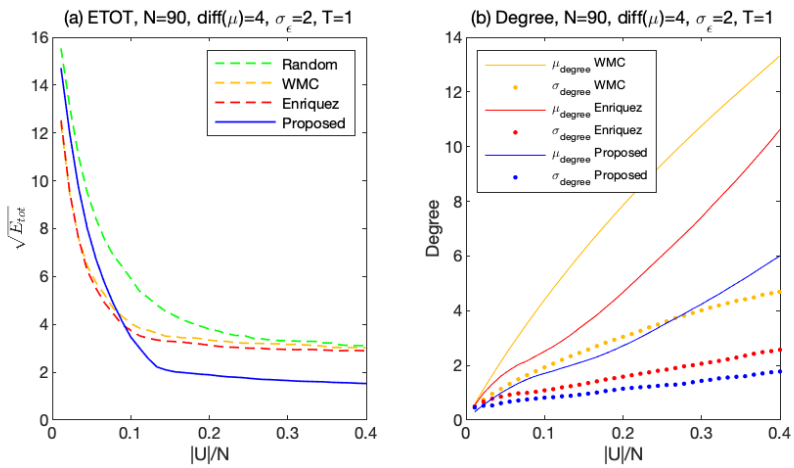

Figure 3.3: Simulation result of stochastic block model data.

the proposed neighborhood selection method outperforms compared to other candidate methods. The proposed method (blue) gives better $\sqrt{\hat{E}_{tot}}$ when $\frac{|\mathcal{U}|}{N}$ is bigger than 0.1. In addition, from Figure 3.3 (b), we can generate more effective UPA through clustering-based network design since both $\sqrt{\hat{E}_{tot}}$ and degree are smaller than Martínez-Enríquez *et al.* (2018b)'s method.

3.4.2 Image data analysis

In this subsection, we can apply the proposed method to image data analysis. We use the test image from Li *et al.* (2016). For image data application, we use a segmentation-based edge addition called statistical region merging (SRM), suggested by Nock (2004). SRM is described in Figure 3.4. SRM algorithm starts from an assumption that all nodes are in separated clusters. By applying appropriate statistical tests, nearby clusters are merged until the number of clusters reaches a predetermined number. In Figure 3.4, 34, 30, 14, and 9 clusters are used. Although the number of clusters is larger than the actual number of clusters, the proposed method still works

$\sqrt{\hat{E}_{tot}}$ ($\sqrt{\text{se}\hat{E}_{tot}}$)		$\lvert\mathcal{U}\rvert/N = 0.1$				$\lvert\mathcal{U}\rvert/N = 0.2$				$\lvert\mathcal{U}\rvert/N = 0.3$			
Methods		RA	MC	ME	PM	RA	MC	ME	PM	RA	MC	ME	PM
$\lvert T\rvert = 1$	$\mu_d = 4,$	7.3	5.2	4.0	**3.8**	4.5	3.4	2.8	**1.7**	3.2	2.8	2.4	**1.5**
	$\sigma_\epsilon = 2$	(0.8)	(1.0)	(0.6)	(0.9)	(0.8)	(0.7)	(0.4)	(0.3)	(0.6)	(0.5)	(0.3)	(0.2)
	$\mu_d = 4,$	7.4	4.9	3.7	**3.2**	4.5	3.2	2.8	**1.7**	3.2	2.7	2.5	**1.6**
	$\sigma_\epsilon = 3$	(1.3)	(0.9)	(0.6)	(0.9)	(1.3)	(0.7)	(0.4)	(0.3)	(0.9)	(0.6)	(0.2)	(0.2)
	$\mu_d = 8,$	10.9	8.2	7.0	**5.4**	6.7	6.0	5.2	**2.6**	5.3	5.0	4.5	**2.2**
	$\sigma_\epsilon = 2$	(1.5)	(1.4)	(1.4)	(1.3)	(1.6)	(1.0)	(0.6)	(0.5)	(0.9)	(0.8)	(0.4)	(0.3)
	$\mu_d = 8,$	11.1	8.0	7.3	**5.6**	6.7	5.8	5.2	**2.5**	5.4	5.1	4.6	**2.2**
	$\sigma_\epsilon = 3$	(1.9)	(1.2)	(1.1)	(1.1)	(1.0)	(0.8)	(0.6)	(0.4)	(1.0)	(0.8)	(0.5)	(0.3)
$\lvert T\rvert = 20$	$\mu_d = 4,$	7.6	4.9	4.0	**3.9**	4.4	3.4	2.8	**1.8**	3.3	3.0	2.5	**1.6**
	$\sigma_\epsilon = 2$	(1.2)	(0.8)	(0.7)	(0.8)	(1.1)	(0.9)	(0.3)	(0.2)	(0.7)	(0.5)	(0.2)	(0.1)
	$\mu_d = 4,$	7.2	5.3	4.3	**4.0**	4.2	3.4	3.0	**2.0**	3.2	2.9	2.6	**1.7**
	$\sigma_\epsilon = 3$	(1.1)	(1.3)	(0.9)	(1.0)	(1.0)	(0.8)	(0.3)	(0.1)	(0.7)	(0.4)	(0.2)	(0.1)
	$\mu_d = 8,$	11.2	8.5	7.3	**5.5**	7.0	5.8	5.0	**2.5**	5.4	5.1	4.5	**2.2**
	$\sigma_\epsilon = 2$	(1.9)	(2.1)	(1.3)	(1.8)	(1.3)	(1.0)	(0.6)	(0.4)	(0.9)	(0.7)	(0.4)	(0.2)
	$\mu_d = 8,$	11.4	8.3	7.4	**5.8**	7.2	5.9	5.1	**2.7**	5.9	5.3	4.6	**2.4**
	$\sigma_\epsilon = 3$	(2.0)	(1.3)	(1.5)	(1.5)	(1.2)	(0.7)	(0.6)	(0.4)	(0.9)	(0.8)	(0.5)	(0.3)
$\lvert T\rvert = 40$	$\mu_d = 4,$	7.3	5.1	4.2	**4.1**	4.4	3.6	2.8	**1.8**	3.4	2.8	2.5	**1.6**
	$\sigma_\epsilon = 2$	(1.4)	(0.9)	(0.7)	(1.1)	(0.8)	(0.7)	(0.3)	(0.2)	(0.7)	(0.4)	(0.2)	(0.1)
	$\mu_d = 4,$	7.3	5.2	4.2	**3.7**	4.5	3.6	2.9	**1.9**	3.4	2.9	2.6	**1.7**
	$\sigma_\epsilon = 3$	(1.4)	(0.9)	(0.6)	(0.9)	(1.0)	(0.6)	(0.3)	(0.1)	(0.9)	(0.5)	(0.2)	(0.1)
	$\mu_d = 8,$	10.8	8.2	7.2	**5.9**	6.8	5.4	5.0	**2.6**	5.3	4.8	4.5	**2.3**
	$\sigma_\epsilon = 2$	(2.1)	(1.4)	(1.1)	(1.2)	(1.5)	(0.9)	(0.6)	(0.4)	(0.8)	(0.7)	(0.4)	(0.3)
	$\mu_d = 8,$	10.5	7.8	6.8	**5.6**	6.8	5.7	5.2	**2.7**	5.5	5.1	4.5	**2.4**
	$\sigma_\epsilon = 3$	(2.0)	(1.3)	(1.1)	(1.4)	(1.1)	(0.9)	(0.7)	(0.4)	(1.1)	(0.9)	(0.4)	(0.3)

Table 3.1: $\sqrt{\hat{E}_{tot}}$ results of the simulated stochastic block model.

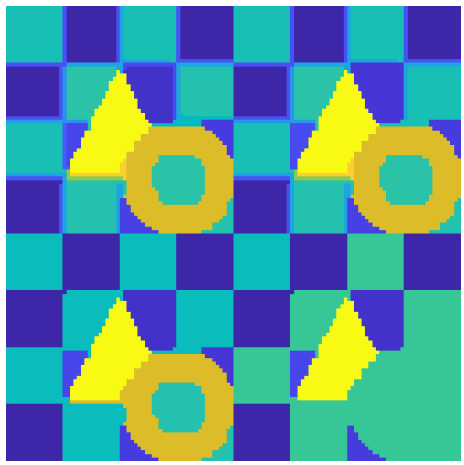

Figure 3.4: Statistical region mapping result of noisy simulation image. The number of segmentation are 34, 30, 14, and 9.

Ratio $\lvert U\rvert/N =$	0.05	0.2	0.4
$\sqrt{\hat{E}_{tot}}$ Enriquez (std. error)	89.92 (0.81)	38.02 (0.52)	35.20 (0.24)
$\sqrt{\hat{E}_{tot}}$ Proposed (std. error)	87.81 (2.03)	31.75 (0.99)	27.83 (1.16)

Table 3.2: $\sqrt{\hat{E}_{tot}}$ results of simulated blockwise image data.

for piecewise constant graph signal data. For the simulation study, we use 30 clusters of SRM.

Figure 3.5 and Table 3.2 show that the proposed method gives better reconstruction compared with the Martínez-Enríquez *et al.* (2018b)s model. Notably, the proposed method has an advantage of finding edges since the process chooses many nodes near edges into $\mathcal{U}$ set at the next level.

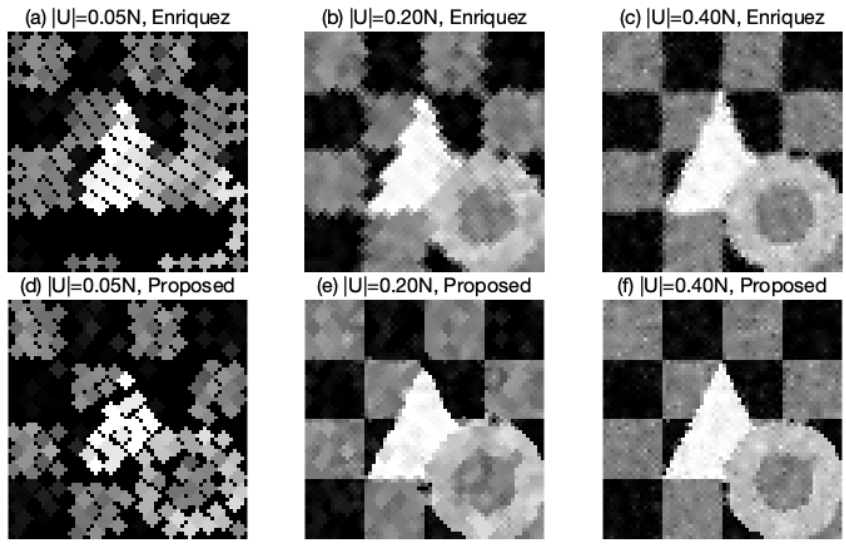

Figure 3.5: Results of graph image signal reconstruction with $100 \times |U|\%$ of nodes using (a), (b), (c) 2-hop graph edges (Enriquez) and (d), (e), (f) deleting all 2-hop inter-cluster edges from Enriquez (Proposed).

3.4.3 Blocks signal denoising

Moreover, the concept of clustering-based neighborhood selection suggested in this thesis, can be applied to LOCAAT setting without changing the removal order. Data-adaptive removal order selection does not guaranteed the lower root mean squared error in LOCAAT setting. We compare denoising performances of proposed neighborhood selection method with adaptive lifting (Adlift) Nunes *et al.* (2006) and nondecimated lifting (Nlift) Knight *et al.* (2009). For computation, we use `denoise` function in R package `adlift` and `denoiseperm` function in R package `nlt`, with prediction filter `AdaptPred`, respectively. The number of a neighborhood at each level is fixed to `5`. In this simulation, we consider block-signal simulation function of Donoho and Johnstone (1994) and independently generate it 100 times. In each iteration, the number of data points is fixed to `1024`.

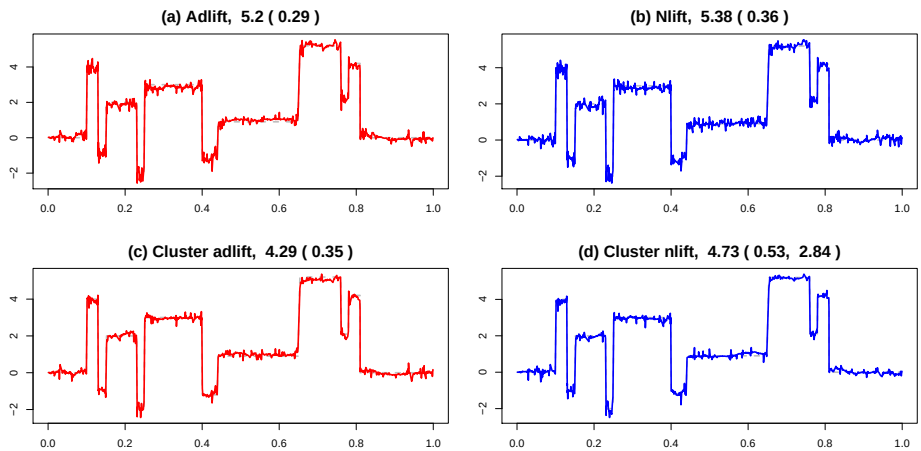

Figure 3.6: One denoising result of a simulated block signal Donoho and Johnstone (1994) with noise $\mathcal{N}(0, 0.2^2)$ and $n = 1024$. Note that numbers are the mean of square root total prediction error (and standard error), $\frac{1}{100}\sqrt{\sum_{i=1}^{1024}(y_i - \hat{y}_i)^2/1024}$.

To decide the cluster, We divide the simulated data into a few segments which have large gaps.

Figure 3.6 shows one realization of denoising results of 100 iterations. The result shows that the proposed neighborhood selection procedure, described in Figure 3.6 (c) and (d), works well under the simulated dataset combined with both Adlift and Nlift. In Nlift case, there are few cases that the proposed method is not worked. We conjecture that this phenomenon is closely related to the random removal order construction in Knight *et al.* (2009). We expect that we can control the worst cases by an appropriate removal order restriction.

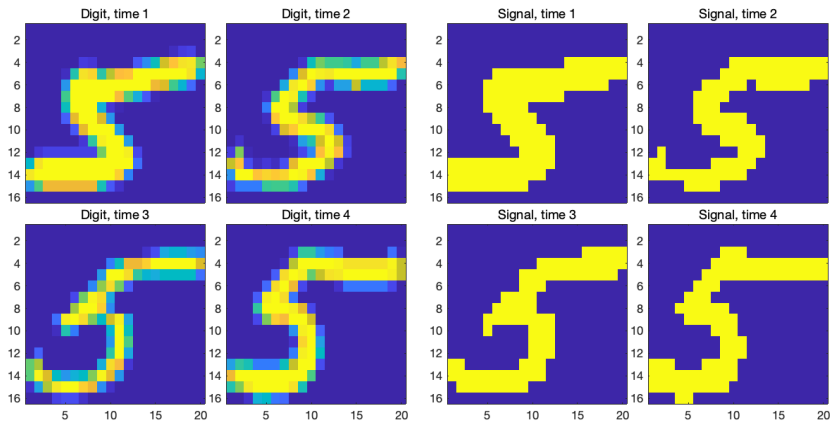

Figure 3.7: (Left) The original digit dataset and (Right) estimated digit signal values ($\hat{\mathbf{c}}$) through segmentation.

3.5 Real data analysis

In this section, we evaluate the UPA performance of the proposed algorithm with digit data analysis. In digit data analysis, we choose eight digit data from `MNIST handwritten` dataset with similar shapes. Figure 3.7 shows the original digit signal dataset and their estimated digit values, according to the proposed method. Note that Martínez-Enríquez *et al.* (2018b)'s method estimates every digit signal as a constant value. Every digit data has a long tail at the upper part of digit 5. We assume that the data as time-varying images. The order of images is defined by multidimensional scaling method. Each digit image has $16 \times 20 = 320$ nodes. Since we use 8 consecutive images, the number of total nodes is $N = 16 \times 20 \times 8 = 2560$ nodes. For digit data analysis, we use the Algorithm 1, not a LOTAAT method. Estimated spatial (nodes within an image) and temporal (nodes of different images) weights are $\hat{w}_s = 0.80$ and $\hat{w}_t = 0.20$.

In this data analysis, we select $|\mathcal{U}| = 0.1N$, $0.2N$ and $0.4N$ nodes according to Martínez-Enríquez *et al.* (2018b)'s and proposed method. Results are shown in Figure 3.8. It shows that the results are quite similar.

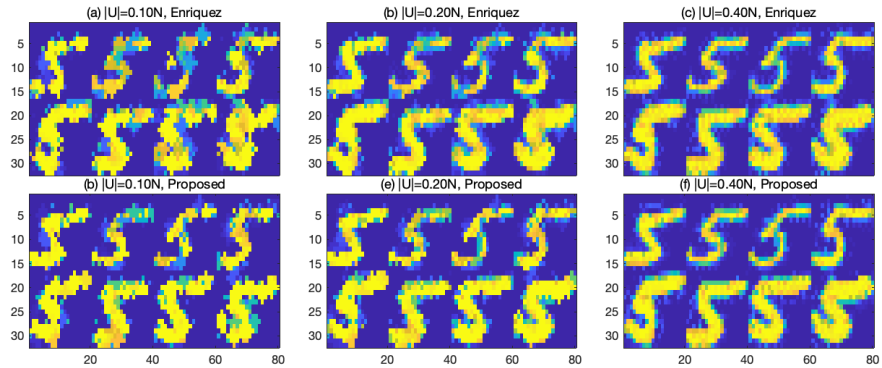

Figure 3.8: Image reconstructions from various UPA results of digit data signals with (a), (b), (c) Martínez-Enríquez *et al.* (2018b)'s method and (d), (e), (f) the proposed method.

However, the proposed method shows that it separates digit images from their backgrounds more clearly. Also, the proposed method produces the same result with relatively a small number of edges in the graph signal. Therefore, we can reduce computational time with the proposed method.

3.6 Summary and discussion

In this chapter, we suggest a new approach of lifting scheme for graphs signal data through update-prediction set assignment problem based on clustering methods are proposed to extend the concept of UPA to a piecewise-constant graph signal setting. The technique has some advantages: it is easily combined with any clustering or conventional lifting scheme algorithm such as LOCAAT. Also, under the piecewise-constant graph signal setting, we can get effective UPA results while reducing the number of edges of the graph signal.

For further research, we need to make a method of how to construct a coarser level of graph signal network with selected $\mathcal{U}$ nodes from the previous level of the data. The usual approach is making new edges within

a predetermined distance-based threshold. However, in the spatio-temporal setting, it may cause a problem because it makes many spatio-temporal distances, therefore directly applying Equation (3.15) is impossible.

Applying Kron reduction is also an option to generate a coarser level of graph signal network. Kron reduction is a kind of method for graph downsampling based on graph Laplacian, which saves both degree and local connectivity information. However, Kron reduction produces too many edges in practice, caused too many computation times. Therefore we have to consider additional measurements to apply it.

Chapter 4

Streamflow lifting scheme

This chapter aims to suggest a new lifting scheme method for streamflow data. Environmental monitoring is the collection of observations and studies for assessment (Artiola et al.*et al.*, 2004). It is crucial for our society because it is directly related to human and environmental health. Water quality monitoring, which is a branch of environmental monitoring, is also important for our society. One of the popular measurement for water quality monitoring is otal nitrogen (TN). TN increases with the increase of domestic, factory, and livestock wastewater. Therefore, TN is an important measure of water environment conservation.

One of the distinct characteristics of water quality index is that it is located on the streamflow network. That means observed values are correlated across the river network, not usual $\mathbb{R}^2$ domain. On the other hand, the majority of spatial statistical models have an interest in analyzing a spatial region, which is a subset of $\mathbb{R}^2$ where Euclidean distance works well. However, streamflow data is an example which Euclidean distance does not work well as a natural metric.

VerHoef *et al.* (2006) suggest the use of stream distance, which is defined as the shortest distance between two observation locations along with

the stream network. It is an attractive distance when we are working with river network data. They show that we can construct a large class of valid spatial autocovariance models with the use of stream distance. Moreover, they suggest the concept of "flow-connected", which is explained in Section 4.2.1.

O'Donnell *et al.* (2014) first proposed the use of nonparametric flexible regression models such as kernel methods and penalized splines to construct spatio-temporal models of river networks. They also suggest a piecewise simple regression approach by diving the network into a large number of small pieces called "streamflow segments" described in Figure 4.1, within which the underlying function value m is expected to change very little. They also suggest a spatio-temporal additive model based on penalized splines.

However, all the methods mentioned above do not consider the multiscale concept. From this perspective, we introduce a new lifting scheme method for streamflow data while expressing their complex structure. Applying conventional lifting scheme construction directly to the streamflow data is not easy due to the complex structure of the data. We are going to suggest a new lifting scheme method with careful consideration of the characteristics of streamflow data.

In this chapter, we briefly describe a lifting scheme on the river network data. In Section 4.1, we introduce the dataset used in this chapter. The main part of this chapter is in Section 4.2. In Section 4.2, we define a new method called streamflow lifting scheme. Simulation and real data analysis are provided in Section 4.3 and 4.4. We summarize the contents of this chapter in Section 4.5.

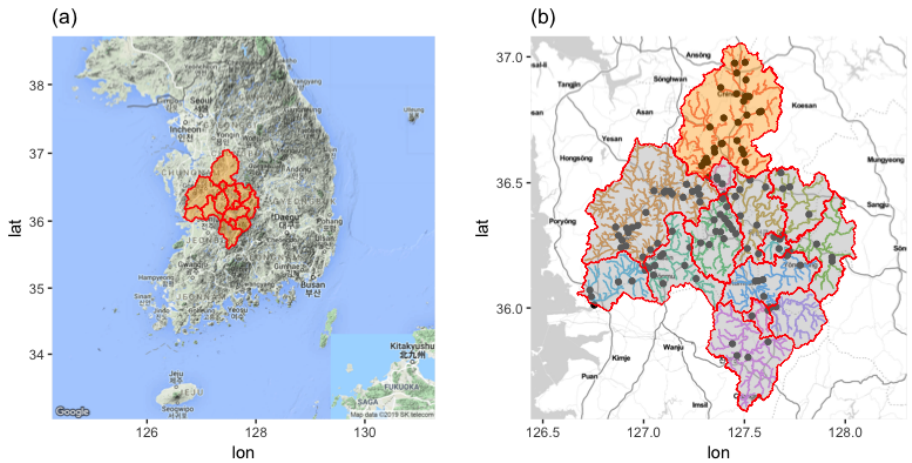

Figure 4.1: (a) Map of Geum-river basin in South Korea (orange area). (b) An enlarged figure of (a). Colored lines are river networks, and black dots are 127 observation points. The basin is divided by 14 catchments. Miho-cheon catchment, located in the north of Geum-river, is colored orange.

4.1 Dataset

The data for this paper are observed in Geum-river basin, which is located in the central part of South Korea, shown in Figure 4.1 (a). According to the Water Environment Information System, operated by Ministry of Environment, Geum-river basin is divided into 14 sub-regions called catchments. Among them, Miho-cheon catchment, which is an orange area in Figure 4.1 (b), is one of the sub-regions of Geum-river basin. It has relatively many observation stations compared to other catchments, and there are several cities and factories around it. Therefore, we believe that taking a closer look at this area is meaningful for the water quality study.

VerHoef *et al.* (2006) and VerHoef *et al.* (2010) define stream segments as lines between junctions in a stream network. In Figure 4.1 (b), orange lines mean streamflow segments. Additionally, black dots denote water quality observation stations. In particular, Miho-cheon catchment has 113

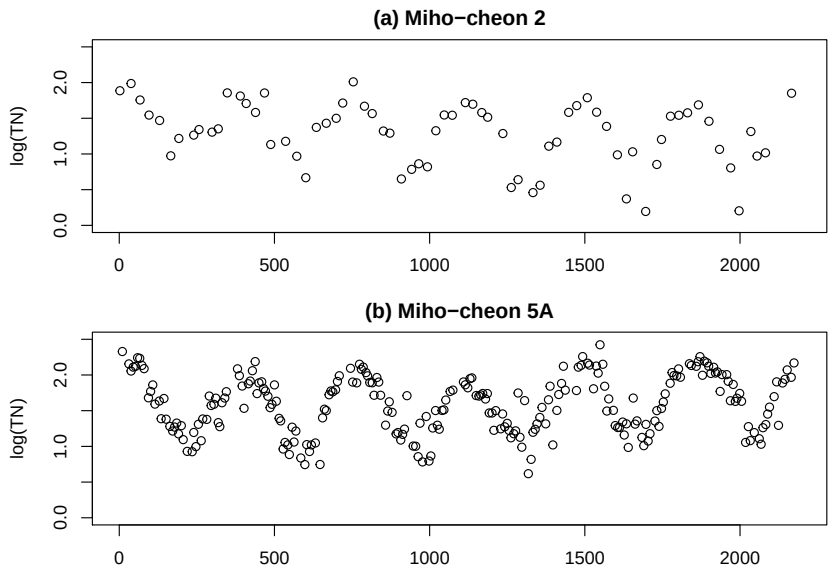

Figure 4.2: The log-transformed total nitrogen (TN) time series observed from two different observation stations, (a) Miho-cheon 2 station and (b) Miho-cheon 5A station.

streamflow segments and 28 observation stations.

Figure 4.2 shows a plot of two time-series observed in Miho-cheon 2 and Miho-cheon 5A station. Among 28 stations, TN is monthly observed in 15 stations such as Miho-cheon 2 station and weekly observed in 13 stations including Miho-cheon 5A station. According to Figure 4.2, TN data is complex and irregularly spaced both space and time. This dissonance makes it hard to analyze the spatio-temporal behavior of streamflow data.

4.2 Streamflow lifting scheme

In this section, we explain our modifications to construct a new lifting for streamflow data from the conventional lifting scheme, based on LOCAAT algorithm suggested in Jansen *et al.* (2009). The main idea of streamflow

lifting scheme is (i) choosing a network-adaptive neighborhood selection, (ii) constructing a prediction filter with flow-adaptive weighted averages and (iii) setting a removal order by defining an appropriate contribution measure of each observation point for the network.

Throughout this section, we consider a toy example, described in Figure 4.3. Suppose that there are five observation points (A, B, C, D, E), located in different segments of a river network. Assume that each segment has a flow volume of f. Let $f_A, f_B, \ldots, f_E$ denote flow volume of station $A, \ldots, E$. We also denote $y_A, \ldots, y_E$ as a water quality observation values at station $A, \ldots, E$.

4.2.1 Neighborhood selection

The concept of "flow-connected", introduced in VerHoef *et al.* (2006) is useful to construct a neighborhood set of streamflow data. VerHoef *et al.* (2006) define two locations are connected when the intersection of upstreams of two stations is not an empty set. In coincides with the water in a station can go to another location. We call A, C and B, C are "flow-connected" because the water in A and B can go to location C. On the other hand, C and D are not flow-connected since the water in C cannot go to station D, or vice versa.

We use the concept of "flow-connected" to decide whether two segments are neighbors or not. In this thesis, when two points are flow-connected, we define they are neighbors of each other. For example, suppose that we are interested in removing point C in Figure 4.3 at a specific resolution level. Following the concept of flow-connected, we define its neighborhood as A, B, and E (blue circles). On the other hand, D (green triangle) is excluded from the neighborhood of C.

One of the distinct characteristics of the proposed neighborhood selection is that it considers both upstream and downstream neighborhoods. By

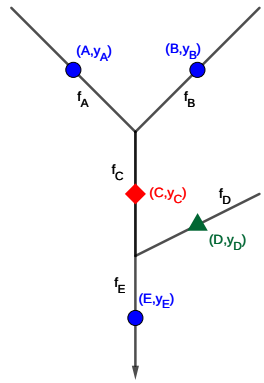

Figure 4.3: A simple streamflow data. There are five observation points from A to E, located in different segments. Each segment has its flows, called $f_A, f_B, \ldots, f_E$. We denote $y_A, \ldots y_E$ to represent water quality values.

doing so, we can reduce the number of boundary points. Including downstream points into neighborhood seems awkward. However, by combining an appropriate prediction filter construction, which is explained in 4.2.2, we can generate reasonable prediction filters.

4.2.2 Prediction filter construction

Although we define neighbors in Section 4.2.1, it is not desirable to give an equal weight for all neighbors. For example, if f_A is much bigger than f_B, then y_A has a larger impact on y_C, compared to y_B. Also, if f_D is bigger than f_C, then y_E is much more different from y_C. Besides, we need to fix the sum of all weights are equal to one for the stability problem. In this subsection, we adopt flow-adaptive weights construction to solve these problems.

In the previous example, since $f_C = f_A + f_B$, we define flow-adaptive

weights for point C as a ratio of flows,

$$w_A = \frac{f_A}{f_C}, w_B = \frac{f_B}{f_C} \text{ and } w_E = \frac{f_C}{f_E}. \tag{4.1}$$

Using 4.1, we can define a predictor of y_C, $\hat{y}_C = \tilde{w}_A y_A + \tilde{w}_B y_B + \tilde{w}_E y_E$, where

$$\begin{aligned} \tilde{w}_A &= \frac{f_A/f_C}{f_A/f_C + f_B/f_C + f_C/f_E}, \\ \tilde{w}_B &= \frac{f_B/f_C}{f_A/f_C + f_B/f_C + f_C/f_E}, \text{ and} \\ \tilde{w}_E &= \frac{f_C/f_E}{f_A/f_C + f_B/f_C + f_C/f_E}. \end{aligned} \tag{4.2}$$

Note that $\tilde{w}_A, \tilde{w}_B$, and $\tilde{w}_E$ normalized version of flow-adaptive weights to make the sum of weights is 1, i.e., $\tilde{w}_A + \tilde{w}_B + \tilde{w}_E = 1$. Following (4.1) and (4.2), we can construct a lifting scheme for streamflow data by combining flow-adaptive weights into the conventional lifting scheme.

In real data analysis, it is uncommon that one knows all f values throughout the whole streamlines. Therefore, we need to estimate flow values. In this chapter, we assume that flow volume f of the most upstream segments are proportional to their Shreve stream order and segment lengths. Stream order is a positive whole number frequently used in hydrology to define stream-based distance in a river network system. There are various kinds of stream order. Among them, the Shreve stream order, well explained in Cressie et al.*et al.* (2006) and VerHoef *et al.* (2010), is one of the most straightforward stream order.

Cressie et al.*et al.* (2006) define the stream order as merely a count of the number of sources in the upstream portion of a river network. Shreve stream order starts from setting all upper segments to 1. Magnitudes increase at all junctions in the river network system. For example, if streams have a magnitude 1 and 2 combined into one new stream, that has a magnitude 3. By doing so, we can construct all magnitudes of the given network.

Following the definition of Shreve stream order, If we let the flow of the upper-most segments, described in Figure 4.4, is proportional to their lengths, one can define a flow volume of the next upstream segments as a sum of their upstream segments. By repeating this approach, we can define all f values in the river network. In Figure 4.5, the volume of the gray line means the flow of each segment. In this thesis, following the O'Donnell *et al.* (2014), we use $\log(\sqrt{f})$ values, after normalizing all $\log(\sqrt{f})$ values are lying between 0.2 to 1.5.

4.2.3 Removal point selection

In the streamflow lifting scheme, it is also important to decide a removal order. If the data lie in the real line, such as Figure 2.2, we can easily adapt conventional approaches such as Nunes *et al.* (2006). It can be extended in two-dimensional data, suggested by Jansen *et al.* (2009). The basic idea of Jansen *et al.* (2009) to decide the removal point is setting an appropriate integral of a scaling function to find the densest observation in the Euclidean domain. In one-dimension, Nunes *et al.* (2006) use the length of a point in real line for the computation of an integral. In two-dimension, Jansen *et al.* (2009) suggests Voronoi-polygon based area measurement as a candidate of an appropriate integral. They choose a point which has the smallest integral as a removal point in LOCAAT algorithm.

However, these methods are not directly applied to the streamflow data since the network is not easily projected onto conventional one or two-dimensional data. In this subsection, we suggest a simple approach to distinguish which points is located in the densest region in the streamflow network by using the measure of the contribution of each segment in data. We define an integral as a contribution of each observation point to the network.

To define a contribution of each point in streamflow data, we use flow-

adaptive weights defined in (4.2). Consider the simple example described in Figure 4.3. Suppose that at j-th level, we want to remove point C with neighborhood points A, B and E. Let I_A^j denotes an integral of point A at j-th level. Then I_A^j is defined by the volume of segment observation A is located, V_A. In this chapter, we defined the volume V as a product of flow and length of the segment, l,

$$\begin{aligned} I_A^j &= V_A = f_A \times l_A, \\ I_B^j &= V_B = f_B \times l_B, \text{ and} \\ I_E^j &= V_E = f_E \times l_E. \end{aligned} \tag{4.3}$$

At the next level, $j-1$, after point C is removed, we need to update the integral of neighborhood points. The idea is to represent I_C^j as a weighted combination of neighborhoods. In this thesis, we use (4.2) as weights. Therefore, I_A^j, I_B^j, and I_E^j are changed to

$$\begin{aligned} I_A^{j-1} &= I_A^j + \tilde{w}_A \times V_C, \\ I_B^{j-1} &= I_B^j + \tilde{w}_B \times V_C, \text{ and} \\ I_E^{j-1} &= I_E^j + \tilde{w}_E \times V_C. \end{aligned} \tag{4.4}$$

Since $\tilde{w}_A \times V_C + \tilde{w}_B \times V_C + \tilde{w}_E \times V_C = I_C^j$, the sum of integrals is not changed. The next removed point at $j-1$ level has selected a point which has the minimum value of I^{j-1}. For the update filter, we use the minimum norm solution based filter, written in (2.8).

4.3 Simulation study

In this section, we provide a simple simulation analysis applied to a denoising problem of spatial streamflow data analysis. Suppose that the data are generated by the following spatial model,

$$y_i = m(x_i) + \varepsilon_i, \tag{4.5}$$

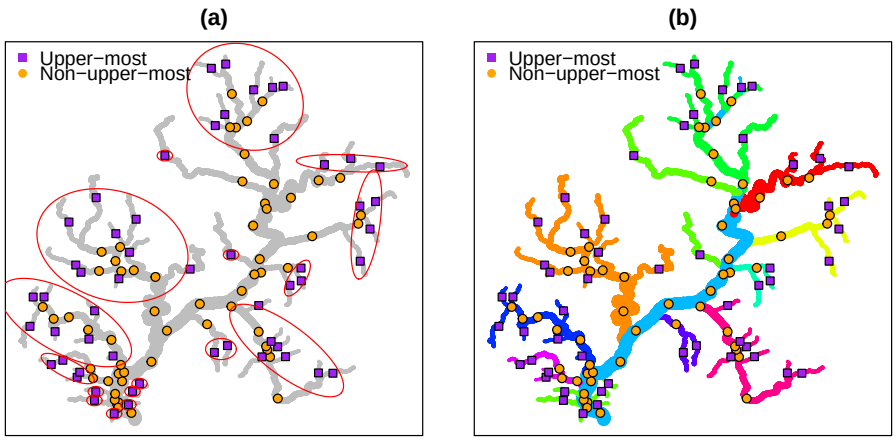

Figure 4.4: (a) Cluster construction (red circles) on the Miho-cheon river network. We classify streamflow segments into two, upper-most segments (purple squares) and non-upper-most segments (orange circles). (b) Colors represent substreams for the sampling procedure. The sampling probability is proportional to the number of streams of each substream.

where $m(x_i)$ is a signal value of i-th stream segment.

We compare the proposed method with B-spline smoothing method, proposed by O'Donnell *et al.* (2014). The main idea of O'Donnell *et al.* (2014) is to represent signal part m of (4.5) with an additive model based on penalized splines, minimizing

$$\|\mathbf{y} - B\boldsymbol{\beta}\|^2 + \boldsymbol{\beta}^T P \boldsymbol{\beta}, \tag{4.6}$$

where $\mathbf{y}$ is a data vector, B is a basis matrix, constructed by an additive combination of spatial, temporal, and seasonal basis, P is a penalty matrix, and $\boldsymbol{\beta}$ is a set of coefficients. O'Donnell *et al.* (2014) choose B-splines, constructed from polynomial pieces, for basis construction. They estimate m as a linear combination of spatial, temporal, or seasonal B-spline basis functions ϕ_j, i.e., $\hat{m}(x) = \sum_j \beta_j \phi_j(x)$.

In the simulation study, Miho-cheon streamflow segments are classi-

fied into two groups, upper-most segments and non-upper most segments, shown in Figure 4.4 (a). Note that there are 55 upper-most segments in the streamflow. If we assume that there are no intrinsic sources to change the simulated signal values, signal values of non-upper-most segments are generated from a weighted average of nearby upstream signal values. That means, it is enough for the simulation only to generate signal values of upper-most segments.

For simulation, we divide upper-most segments into a few clusters, shown in Figure 4.4. We assume that within a cluster, all segments have the same signal values. In each simulation, to give more variability, $m(x_i)$ value of upper-most segments is generated through the following procedures. First, we set all $m(x_i)$ value of upper-most segments 4. Second, we randomly select a cluster, which is described in Figure 4.4, and assign a value which is one of three: 7, 10, and 13. This procedure is repeated until at least 30 upper-most segments have a value bigger than 4.

Usually, we have a relatively small number of observations compared to the number of streams in a river network. To reflect it, we consider three scenarios. First, we assume that the data are available in 28 stations, where locations are the same as in real dataset. Among 28 stations, there are two cases that the observations are located in the same segment. Therefore, we only have observations in 26 segments. Second, we additionally select 32 segments and assume that we have available 60 data on 58 segments, which is almost half of the number of Miho-cheon streams. Within the sampling procedure, we consider the predetermined substreams, described in 4.4 (b), to prevent an awkward sampling result. One of the realizations is represented in Figure 4.4 (b). Third, we assume that we have at least one observation for all 113 segments, including 28 observations. In this case, we have 115 observations.

We assume that noises are generated from zero-mean independent and

RMSE (std. error)	Obs=28		Obs=60		Obs=115	
	σ=0.5	σ=1	σ=0.5	σ=1	σ=0.5	σ=1
O'Donnell	1.6917 (0.1819)	1.7432 (0.2000)	1.1356 (0.1016)	1.2427 (0.1198)	1.3992 (0.1299)	1.4847 (0.1594)
Proposed	**1.2373 (0.2563)**	**1.4740 (0.2957)**	**0.8127 (0.1417)**	**1.1240 (0.1842)**	**0.9376 (0.1539)**	**1.2971 (0.2404)**

Table 4.1: RMSE of simulation result (and their standard error). In each simulation, the number of iterations is 100.

identically distributed normal random variables, i.e., $\epsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$. For the simulation study, we consider two different standard deviations, (i) $\sigma = 0.5$ and (ii) $\sigma = 1$. For comparison, we consider the root mean square error (RMSE), i.e.,

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{113}(y_i - \hat{y}_i)^2}{113}},$$

where $\hat{y}_i$ is an estimated value for segment i.

Table 4.1 shows a summary of the simulation result. It shows that the proposed method works well under the simulation setting. Figure 4.5 shows one of the simulation results. From (c) of 4.5, we find that O'Donnell *et al.* (2014)'s method cannot make a denoised result when the variability of the data is inhomogeneous in the spatial domain. However, the proposed method produce smoothed output for all streamflow regions because of its local adaptability. Hence, the output of the streamflow lifting scheme is similar to true underlying streamflow signals. We also note that the standard error of the proposed method is always larger than B-spline based method. We conjecture that if the number of observation is larger than that of the simulation setting, it would be reduced.

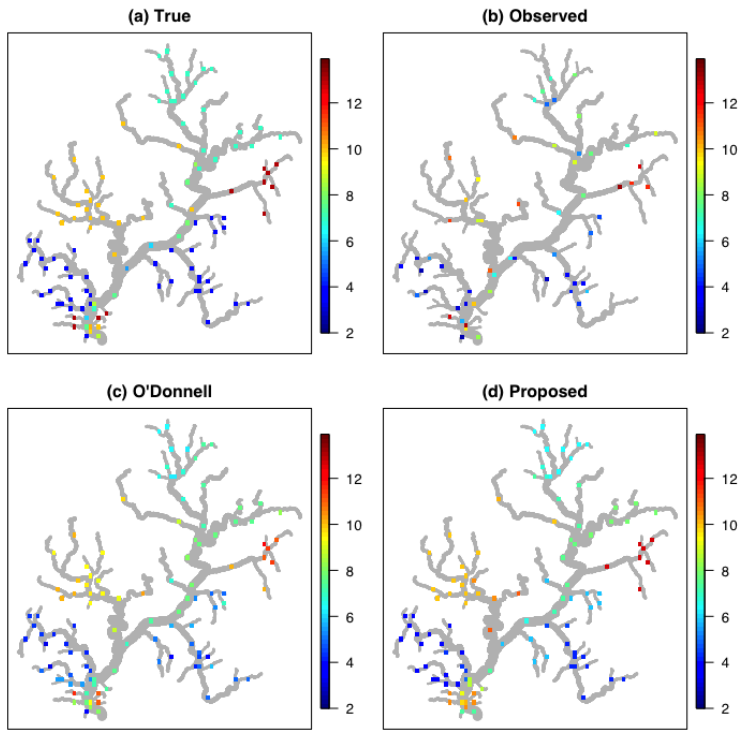

Figure 4.5: (a) True signals, (b) noisy observed values when Obs=60, (c) estimation results of O'Donnell *et al.* (2014) and (d) the proposed method.

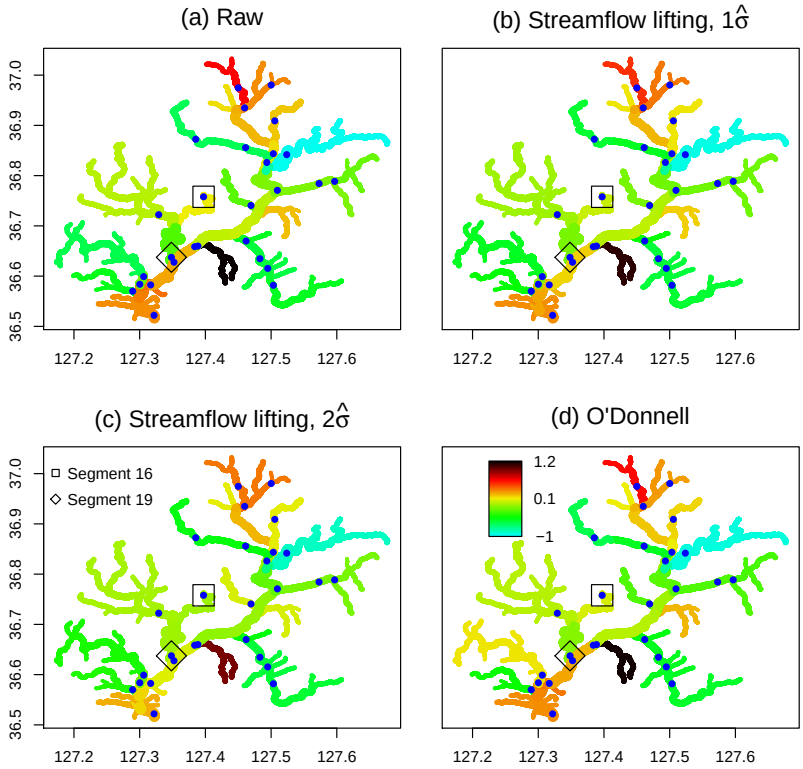

Figure 4.6: Lifting applied to Miho-cheon streamflow data residuals. (a) Spatial prediction using actual residuals. (b)∼(c) Spatial prediction using proposed streamflow lifting scheme with various standard deviation estimators. (d) Spatial smoothing result using O'Donnell's method.

4.4 Real data analysis

For real data analysis, we analyze the Miho-cheon data with a two-step approach. First, to eliminate temporal variability and seasonality, we fit B-spline smoothing using timewise and seasonal B-spline basis, following equation (4.6). After then, we compute the median of each time-series to make a spatial residual dataset. Finally, we apply the proposed streamflow lifting scheme with spatial residuals.

Figure 4.6 shows the result of streamflow lifting scheme applied to

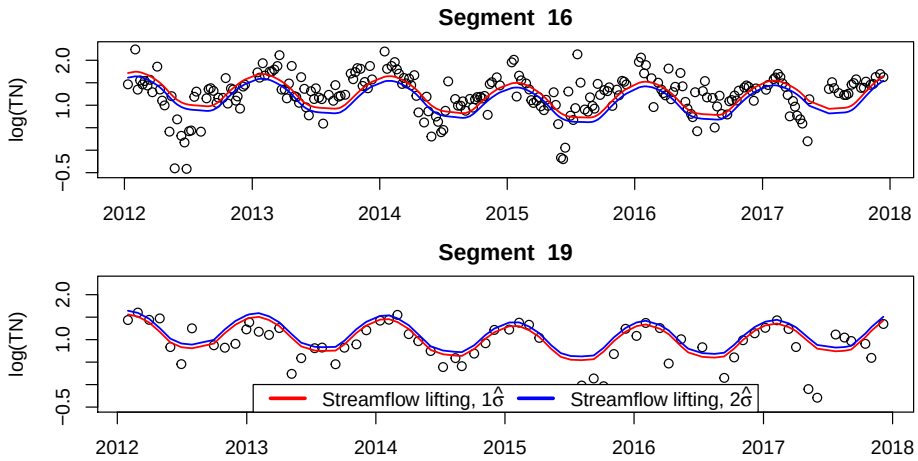

Figure 4.7: Smoothing results of stream segment 16 and 19 with different standard deviation estimators.

Miho-cheon streamflow data residuals. We find that all results are similar to raw residuals, meaning that smoothing is not conducted for the data. To check the effect of the smoothing, we consider two streamflow lifting scheme with various standard deviation estimators, $1\hat{\sigma}$, and $2\hat{\sigma}$. Note that we have more a smoothed result when the standard deviation estimator value is large. For example, segment 16, which is marked as a rectangle, has lower values while affected to nearby values when we give a strong penalty.

Figure 4.7 shows the spatio-temporal smoothing results of stream segment 16 and 19 with a two-step approach. Note that values are not residual but logarithm of original or estimated values. When we produce more smoothed output such as (c) in Figure 4.6, the corresponding smoothing fits of nearby stations are getting closer to each other. It is useful with an assumption that every station has its measurement error, i.e., the data are generated from (4.5), and the underlying structure m is sparse, which means there are only a few distinct values in m, like in (4.3).

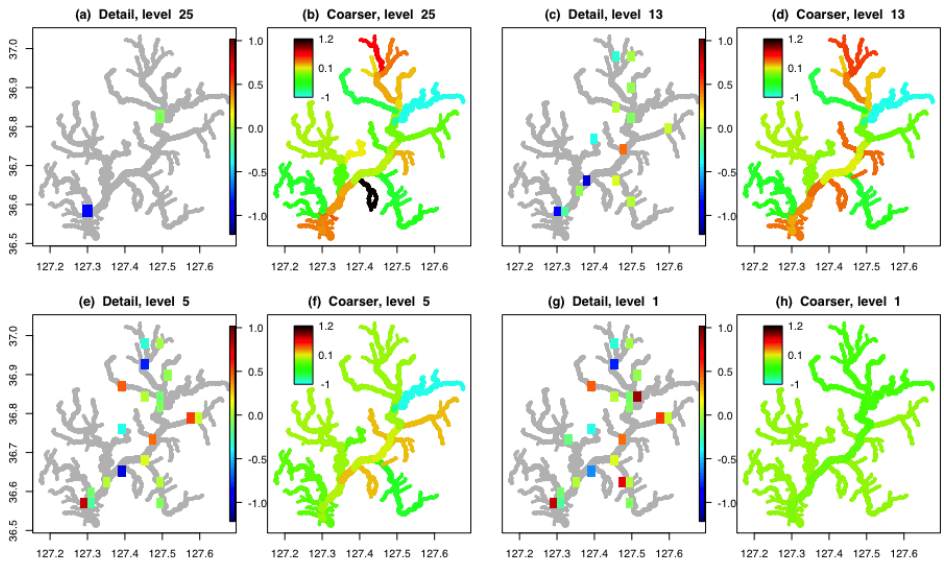

Figure 4.8: Proposed lifting scheme results at various resolution level. To represent streamflow network, (b) 26 observations, (d) 14 observations, (f) 6 observations, and (h) 2 observations are used. Corresponding detail coefficients are shown in (a), (c), (e), and (g), respectively.

Figure 4.8 shows streamflow representation results considering various resolution levels. Following notations in Chapter 2, the lower level indicates the more coarse network representation result. The Miho-cheon network can be represented well with a relatively small number of observations, shown in Figure 4.8 (a)∼(d). However, if we try to estimate the field only with six observations, shown in Figure 4.8 (e)∼(f), we fail to conserve local characteristics of the stream. When there are only two observations in the field, shown in Figure 4.8 (g)∼(h), the streamflow network is represented as a constant.

4.5 Summary and further works

In this chapter, we suggest a new lifting scheme for streamflow data. The proposed methods enable lifting scheme to streamflow data by (i) choosing a river network adaptive neighborhood selection, (ii) constructing a prediction filter with flow-adaptive weighted averages and (iii) setting a removal order by defining neighborhood flows of each observation point. The proposed method is useful not only to denoise the noisy streamflow data but also to understand the multiscale structure of the network.

However, the proposed approach has a limitation because it does not consider spatio-temporal simultaneous analysis. Since data are irregularly observed in both space and time domain, we need additional measurements to do spatio-temporal streamflow analysis. Lindström *et al.* (2014) and O'Donnell *et al.* (2014) solve the problem by computing biweekly or a monthly average of data at each station. Then by O'Donnell *et al.* (2014)'s method, one can build a space-time basis with a tensor product. On the other hand, if we find a way to construct a multiscale spatio-temporal basis without data merging, it will be more useful to capture multiscale spatio-temporal behavior of the data. This is left for future research.

Chapter 5

Multiscale analysis for PM_{10} extremes in Seoul

In this chapter, a case study of the spatio-temporal analysis of particulate matter PM_{10} data in Seoul, which is a novel combination of multiscale analysis and scaling equation theory used in hydrology is introduced.[1]

5.1 Data description

The data used in this chapter are hourly observed PM_{10} data at 25 air quality observation stations in Seoul for 2010-2014 years (43,824 hours). There are three types of air quality stations in Seoul: city, road, and suburb type stations. In this thesis, only city type stations are considered. Figure 5.1 shows the distribution of 25 city type air quality observation stations in Seoul. Most of them are located at the top of a particular building. All stations have sea surface altitude between 10m and 55m, except Dobong-gu

[1]The work presented in this chapter has been published in Stochastic Environmental Research and Risk Assessment with the title "Spatio-temporal analysis of particulate matter extremes in Seoul: use of multiscale approach" (Park and Oh, 2017).

(75.1m) and Seodaemun-gu (97m) stations.

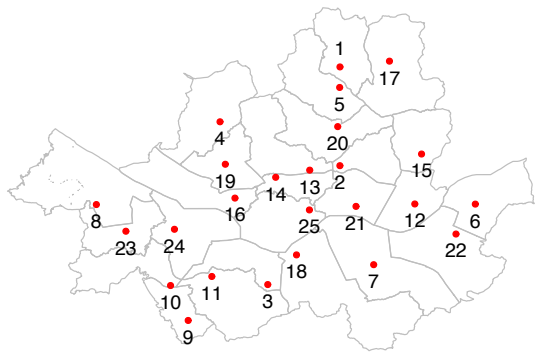

Figure 5.1: The distribution of 25 city type air quality observation stations in Seoul.

Station list					
Id.	Name	Id.	Name	Id.	Name
1	Dobong-gu	10	Guro-gu	19	Seodaemun-gu
2	Dongdaemun-gu	11	Gwanak-gu	20	Seongbuk-gu
3	Dongjak-gu	12	Gwangjin-gu	21	Seongdong-gu
4	Eunpyeong-gu	13	Jongno-gu	22	Songpa-gu
5	Gangbuk-gu	14	Jung-gu	23	Yangcheon-gu
6	Gangdong-gu	15	Jungrang-gu	24	Yeongdeungpo-gu
7	Gangnam-gu	16	Mapo-gu	25	Yongsan-gu
8	Gangseo-gu	17	Nowon-gu		
9	Geumcheon-gu	18	Seocho-gu		

Table 5.1: Air quality observation station list.

Before data analysis, missing value imputation and detrending of the data should be done. Due to the maintenance issue, there are some missing values in data. For imputation of missing values, under the assumption that missing PM_{10} would be positively correlated to those of nearby stations, we use a weighted average of all observed values with inverse distance weights.

While observing annual and seasonal patterns of the data, it seems that PM_{10} density has been decreased for last few years because of the effort of Seoul government to reduce PM_{10} and it has been increased in spring and autumn due to yellow dust and dry weather ((Kim and Kim, 2011) and (Kim et al., 2014)). We eliminate these deterministic cycles or/and trends for further analysis. Pure sine and cosine functions are employed to estimate deterministic trends. Let $x_{t,i}$ denotes original PM_{10} data at time t and station i. In this study, we have detected annual periodicity as $43824/5 = 8764.8$, and removed the corresponding annual cycle from each time series as follows,

$$Y(x_{t,i}) = \beta_0 + \beta_1 t + \beta_2 \cos(\frac{2\pi}{8764.8} x_{t,i}) + \beta_3 \sin(\frac{2\pi}{8764.8} x_{t,i}) + \text{others}, \quad (5.1)$$

where $i = 1, \ldots, 25$. We compute the deterministic trend $\hat{Y}(x_{t,i})$ using linear regression and remove it from original time series.

After detrending, we adjust each PM_{10} series by adding a constant that can hold the same average as that of the original series. Also, to avoid negative PM_{10} values, all negative PM_{10} values are changed to zero. It does not be affected to the main results because only small amounts of the data have negative values. We define

$$\tilde{Y}(x_{t,i}) = \max(x_{t,i} - \hat{Y}(x_{t,i}) + \text{constant}_i, 0), \qquad i = 1, \ldots, 25 \qquad (5.2)$$

as the stationary PM_{10} data at station i for data analysis. For simplicity of notation, we omit t and i.

5.2 Temporal analysis of Seoul extreme PM_{10} data

5.2.1 Temporal aggregation and conventional scale property

In this subsection, temporal aggregation is used to bind nearby data to make n-hour average data from the original data, which might be useful to represent the data with different temporal scales. It can be easily identified that hourly PM_{10} and daily PM_{10} data would have several temporal structures even though they are observed at the same place. We consider data with various temporal scales such as 2, 3, 4, 6, 8, 12, 16, 24, 32, 48, 72, 96 and 128-hour PM_{10} averages at each station. To preserve the length of time series regardless of the aggregation time, we compute d-hour aggregation using previous d-hour PM_{10} data. For analysis of PM_{10} maxima, we use a block-maxima setting with one month as a block.

Scaling property is a useful method to investigate the relationship between two nonnegative data, $Y_d(x)$ and $Y_{d_{ref}}(x)$ over several time scales. It has been widely used for multiscale analysis in various fields such as hydrology, and this concept is an important ingredient of our analysis. A review of the scaling property is presented in Appendix B. In this thesis, we consider $d_{ref} = 1$ hour as the reference time scale. Let $Z_d(x)$ and $Z_{d_{ref}}(x)$ be the monthly maximum of $Y_d(x)$ and $Y_{d_{ref}}(x)$, respectively. That is, in the case of $d = 1$ hour, $Z_d(x)$ is defined as

$$Z_{d=1}(x) = \max_{1 \le j \le 24, 1 \le k \le length(month)} Y_{\text{hour}=j,\text{date}=k}(x).$$

Suppose that monthly PM_{10} maxima at station x with duration time d_{ref} follow a GEV distribution, that is,

$$Z_{d_{ref}} \sim \text{GEV}(\mu_{d_{ref}}(x), \sigma_{d_{ref}}(x), \xi(x)),$$

where $\mu_{d_{ref}}(x), \sigma_{d_{ref}}(x), \xi(x)$ are the location, scale and shape parameters of GEV at location x, respectively. More details about GEV distribution

are presented in Appendix A. Suppose that there is a simple scaling property between $Z_d(x)$ and $Z_{d_{ref}}(x)$. It implies that the GEV distribution at duration time d can be characterized by the GEV parameters at the reference time d_{ref},

$$Z_d(x) \sim \text{GEV}(\mu_d(x), \sigma_d(x), \xi(x)),$$

where

$$\begin{aligned} \mu_d(x) &= \mu_{d_{ref}}(x) \times \Big(\frac{d}{d_{ref}}\Big)^{-\nu(x)}, \\ \sigma_d(x) &= \sigma_{d_{ref}}(x) \times \Big(\frac{d}{d_{ref}}\Big)^{-\nu(x)}, \text{ and} \\ \xi_d(x) &= \xi(x). \end{aligned} \tag{5.3}$$

Thus, one can obtain $\mu_d(x)$ and $\sigma_d(x)$ using the estimation of $\mu_{d_{ref}}(x)$, $\sigma_{d_{ref}}(x)$ and $\nu(x)$ and the scaling property. The above GEV parameters can be estimated by maximum likelihood (ML) method. For estimation of the scale exponent $\nu(x)$, we consider a concept of intensity-duration-frequency (IDF). First, we define a p-month return level associated with return level $1/p$ as

$$q_p = \begin{cases} \mu - \frac{\sigma}{\xi}\big[1 - \log(1-p)^{-\xi}\big] & \text{if } \xi \neq 0 \\ \mu - \sigma \log\big\{ -\log(1-p)\big\} & \text{if } \xi = 0. \end{cases} \tag{5.4}$$

The q_p can be considered as an expected value that GEV observation would be exceeded every $1/p$ month. $p = 1/12$ indicates that the amount of extreme PM_{10} event occurs once a year. Let p-month GEV return level function at duration time d denotes $q_d(x;p)$. Under the simple scaling assumption, the following equations hold,

$$q_d(x;p) = (\frac{d}{d_{ref}})^{-\nu} q_{d_{ref}}(x;p) \text{ and } \log q_d(x;p) \propto -\nu(x) \log\Big(\frac{d}{d_{ref}}\Big). \tag{5.5}$$

In practice, to investigate a scaling relation, we use a wide-sense IDF plot which is defined as boxplots of log(extreme PM_{10} density) across

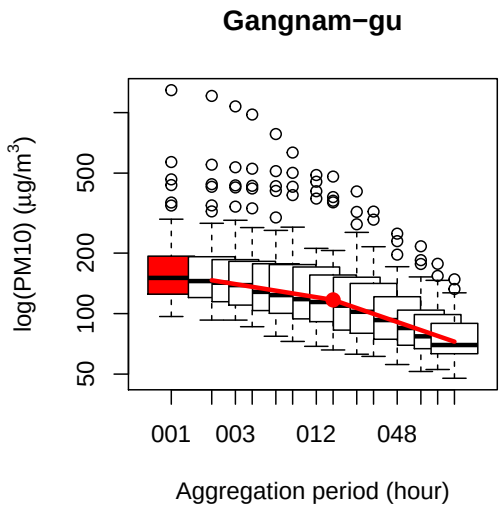

Figure 5.2: IDF plot. Red line is a piecewise linear trend approximation using a knot at $d = 16$hour.

log(duration time ratio). The slope estimate of IDF plot is usually used as an estimate of $\nu(x)$. Figure 5.2 shows the IDF plot at Gangnam-gu station. We compute boxplot and display them at each duration time. As shown, we observe that there is a decreasing tendency of the relation between log(duration time ratio) and log(extreme PM_{10} density).

5.2.2 Temporal multiscale modeling and modified scaling property

From a careful look of Figure 5.2, we observe that the relation between log(duration time ratio) and log(extreme PM_{10} density) is not exactly linear. Besides, one can observe that lengths of the tail in each boxplot are different. It means that the degree of GEV skewness ξ varies across duration time. Note that the simple scaling property assumes that the parameter ξ is constant over duration time. Thus, the conventional scale

property cannot be directly applied to extreme PM_{10} data.

To overcome the above problems, we modify the conventional scale property as follows: (i) From the fact that IDF slope is not entirely linear but close to linear, we consider a piecewise linear approximation with a single knot, which clarifies the relation between log(duration time ratio) and log(extreme PM_{10} density) with conserving advantages of the conventional scale property piecewisely. (ii) We further consider a scale property of shape parameter to reflect the behavior of GEV skewness and tail length. Hence, we propose new scaling equations as

$$\begin{aligned}
\mu_d(x) &= \mu_{d_{ref}}(x) \times \left(d/d_{ref}\right)^{-\left(\nu_{\mu_1}(x)I(d\le\tau)+\nu_{\mu_2}(x)I(d\ge\tau)\right)}, \\
\sigma_d(x) &= \sigma_{d_{ref}}(x) \times \left(d/d_{ref}\right)^{-\left(\nu_{\sigma_1}(x)I(d\le\tau)+\nu_{\sigma_2}(x)I(d\ge\tau)\right)}, \\
\xi_d(x) &= \xi_{d_{ref}}(x) \times \left(d/d_{ref}\right)^{-\left(\nu_{\xi_1}(x)I(d\le\tau)+\nu_{\xi_2}(x)I(d\ge\tau)\right)},
\end{aligned} \tag{5.6}$$

where τ denotes a knot for piecewise linear approximation. As one can see, the new scaling equations require six scaling exponents. Therefore ν_{μ_1}, ν_{μ_2}, ν_{σ_1}, ν_{σ_2}, ν_{ξ_1}, and ν_{ξ_2} should be estimated. To that end, for each duration time d and each station x, we obtain monthly PM_{10} maxima and estimate GEV parameters by ML method. We finally estimate the six scaling exponent between $\log(d/d_{ref})$ and $\log(\mu_d(x)/\mu_{d_{ref}}(x))$, $\log(d/d_{ref})$ and $\log(\sigma_d(x)/\sigma_{d_{ref}}(x))$, and $\log(d/d_{ref})$ and $\log(\xi_d(x)/\xi_{d_{ref}}(x))$ using multivariate adaptive regression spline (MARS) of Friedman (1991) with data-adaptive τ selection. In this thesis, we note that $d_{ref} = 1$ hour.

5.2.3 Result 1: GEV parameter estimation via piecewise linear approximation

Figure 5.3 shows two histograms of sixty monthly maxima of Gwanak-gu station according to two different aggregation hours, 1 and 24 hours. The overlaid red line denotes the fitted GEV density based on ML estimation.

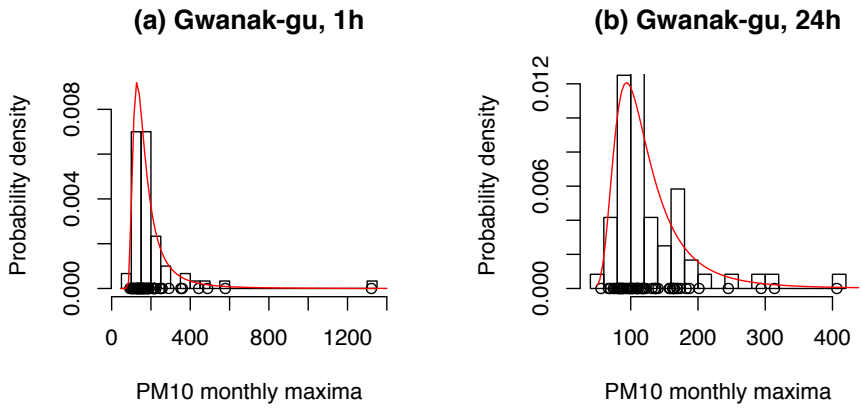

Figure 5.3: (a) Histograms of 1-hour PM_{10} aggregation of Gwanak-gu station and (b) 24-hour PM_{10} aggregation.

Results	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\xi}$
Gwanak-gu, 1h	143.3462(6.3548)	43.4916(5.8377)	0.4407(0.1180)
Gwanak-gu, 24h	100.6678(4.5413)	31.3843(3.7102)	0.2534(0.1013)

Table 5.2: GEV parameter estimate used in Figure 5.3 and its standard error.

It turns out that the monthly maximum PM_{10} follows a Fréchet distribution with positive shape parameter regardless of aggregation time. GEV parameter estimates used in Figure 5.3 and their standard errors are listed in Table 5.2. Thus, the corresponding estimated CDFs are

$$F(x;\hat{\mu}_{1\text{h}},\hat{\sigma}_{1\text{h}},\hat{\xi}_{1\text{h}}) = \exp\left(-\Big(1+0.4407\big(\frac{x-143.3462}{43.4916}\big)\Big)^{-1/0.4407}\right),$$

and

$$F(x;\hat{\mu}_{24\text{h}},\hat{\sigma}_{24\text{h}},\hat{\xi}_{24\text{h}}) = \exp\left(-\Big(1+0.2534\big(\frac{x-100.6678}{31.3843}\big)\Big)^{-1/0.2534}\right).$$

Duration time	1h	2h	3h	4h	6h	8h	12h
Ratio	0.99	1	1	1	1	1	1
Duration time	16h	24h	32h	48h	72h	96h	128h
Ratio	0.97	0.97	0.97	0.95	0.95	0.95	0.95

Table 5.3: The ratio following GEV distribution after the Anderson-Darling test over various duration time.

Before applying GEV distribution, it is important to check whether the density of monthly maximum PM_{10} properly fits a GEV distribution or not. For this purpose, the Anderson-Darling test is performed. In this study, we use the approach of Laio (2004), which increases test accuracy by bootstrap. We perform Anderson-Darling test for each station and duration time with a significant level of $\alpha = 0.05$. If one rejects the null hypothesis, it has strong evidence of following GEV distribution. The results of the goodness-of-fit test are listed in Table 5.3. "Ratio" means the ratio of the number of Seoul extreme PM_{10} data that follow GEV distribution after the Anderson-Darling test. From the result, we have the following observations: (i) Most of the monthly PM_{10} maxima follow a GEV distribution. (ii) As the duration time is getting longer, the ratio is getting lower because the tail of a GEV distribution becomes short. We note that if the duration time is longer than 128 hour, then the maxima do not follow a Fréchet distribution anymore.

We now estimate six scaling exponents of extreme PM_{10} based on the proposed modified scaling property. Figure 5.4 shows estimation results of data observed at Dongdaemun-gu station. Open circles in three panels denote log ratios of three parameters as function of duration time, i.e., $(\log(d/d_{ref})$ vs. $\log(\hat{\mu}/\hat{\mu}_{ref}))$, $(\log(d/d_{ref})$ vs. $\log(\hat{\sigma}/\hat{\sigma}_{ref}))$ and $(\log(d/d_{ref})$ vs. $\log(\hat{\xi}/\hat{\xi}_{ref}))$ at d =2, 3, 4, 6, 8, 12, 16, 24, 32, 48, 72, 96, 128 hours,

respectively. The red line denotes a piecewise linear fit for each parameter. As one can see, by dividing the data into two parts, we obtain much more accurate parameter ratio approximations. We note that the slope estimates in each panel represent $-\hat{\nu}$ of location, scale and shape parameters of GEV distribution. For all 25 stations, we estimate $-\hat{\nu}$ of location, scale and shape parameters by the modified scaling property, and then obtain boxplots of estimated scaling exponents $\hat{\nu}_{\mu_1}(x)$, $\hat{\nu}_{\mu_2}(x)$, $\hat{\nu}_{\sigma_1}(x)$, $\hat{\nu}_{\sigma_2}(x)$, $\hat{\nu}_{\xi_1}(x)$ and $\hat{\nu}_{\xi_2}(x)$, which are shown in Figure 5.5. From the results of Figures 5.4 and 5.5, we observe that (i) when $d > 16$ hour, all three scaling exponents of GEV parameters are larger than those of $d < 16$ hour, (ii) the scaling exponent of the scale parameter shows most significant difference between two duration regions ($d < 16$ and $d > 16$), and (iii) it seems that the scaling exponent of the shape parameter has a large variability, compared to others.

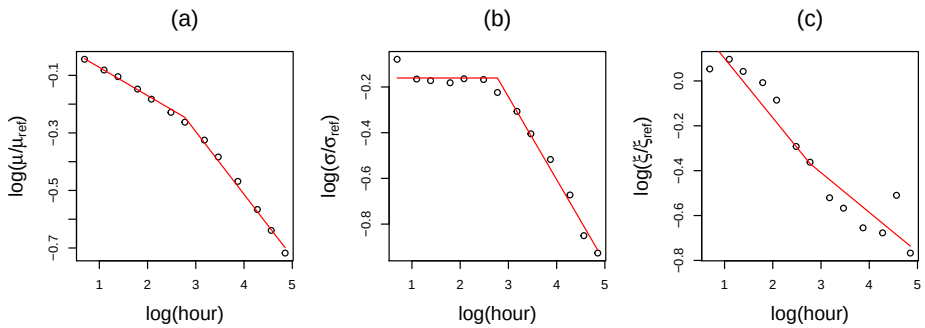

Figure 5.4: The ratios of (a) GEV location, (b) scale, and (c) shape parameters as a function of time duration d and their projections by the modified scaling property. The x-axis is logarithm of a duration time, and open circles indicate the ratio between parameter estimates at d and d_{ref}.

Figure 5.5 shows that histograms of shape < 16 and shape > 16 are similar. It is a piece of indirect evidence that the extremal behavior of

Station name	$\hat{\mu}$ (se($\hat{\mu}$))	$\hat{\sigma}$ (se($\hat{\sigma}$))	$\hat{\xi}$ (se($\hat{\xi}$))
Dobong-gu	123.2164 (5.7480)	39.3819 (5.6250)	0.5270 (0.1243)
Dongdaemun-gu	136.2017 (5.8789)	40.0155 (5.7240)	0.5233 (0.1277)
Dongjak-gu	128.7860 (5.1722)	35.8812 (4.5401)	0.3686 (0.1049)
Eunpyeong-gu	132.0092 (6.1290)	41.6974 (5.7925)	0.4839 (0.1248)
Gangbuk-gu	124.9847 (5.8425)	40.3171 (4.9634)	0.3229 (0.1032)
Gangdong-gu	120.9016 (5.1895)	34.5499 (5.2552)	0.5921 (0.1451)
Gangnam-gu	133.9267 (5.7236)	37.9227 (5.6944)	0.5687 (0.1459)
Gangseo-gu	136.8681 (5.3665)	36.5063 (5.2095)	0.5187 (0.1275)
Geumcheon-gu	128.3423 (5.5184)	38.2306 (4.7305)	0.3368 (0.1030)
Guro-gu	140.1121 (6.5387)	45.9020 (5.7589)	0.3739 (0.0976)
Gwanak-gu	143.3462 (6.3548)	43.4917 (5.8377)	0.4407 (0.1180)
Gwangjin-gu	131.2846 (5.9748)	40.6665 (5.7588)	0.5071 (0.1263)
Jongno-gu	135.2207 (5.9948)	41.0919 (5.3864)	0.4043 (0.1148)
Jung-gu	129.8264 (6.2782)	42.9755 (5.7496)	0.4361 (0.1173)
Jungrang-gu	133.0972 (5.4624)	36.8371 (5.5643)	0.5942 (0.1385)
Mapo-gu	142.5363 (6.3004)	43.4325 (5.8918)	0.4637 (0.1156)
Nowon-gu	124.0556 (6.2201)	42.3364 (5.2029)	0.2960 (0.1106)
Seocho-gu	138.0121 (6.4080)	42.2380 (6.1260)	0.5113 (0.1438)
Seodaemun-gu	130.3773 (6.4251)	43.6680 (5.8304)	0.4184 (0.1211)
Seongbuk-gu	120.3172 (5.2870)	35.4828 (5.2151)	0.5472 (0.1371)
Seongdong-gu	133.0790 (5.6153)	37.3465 (5.5710)	0.5636 (0.1434)
Songpa-gu	120.2376 (5.8832)	38.8925 (5.7911)	0.5533 (0.1457)
Yangcheon-gu	133.9718 (6.2614)	43.4168 (5.1549)	0.2726 (0.0989)
Yeongdeungpo-gu	135.9925 (5.1807)	33.4709 (5.4145)	0.6582 (0.1660)
Yongsan-gu	133.8583 (6.1208)	42.0606 (5.7330)	0.4664 (0.1177)

Table 5.4: GEV parameter estimation (standard error) of each station at 1-hour scale point data.

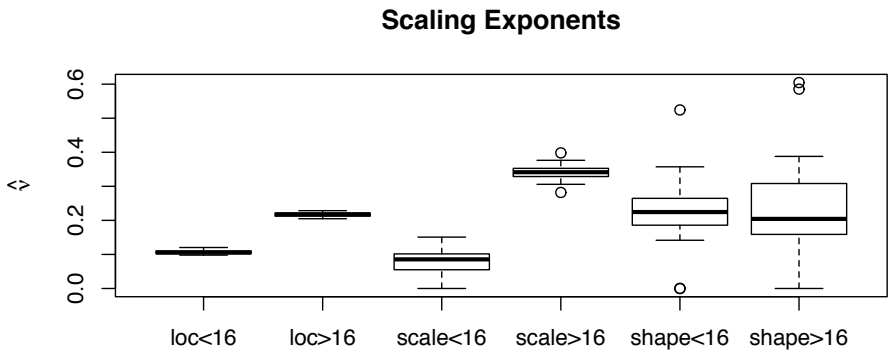

Figure 5.5: Boxplots of scaling exponents of GEV location, scale, and shape parameters.

PM_{10} does not depend on the aggregation time.

5.2.4 Result 2: Return level map by the proposed modified scaling approach

In this subsection, we want to construct a return level map of PM_{10} based on GEV distribution and the proposed modified scaling approach. In this thesis, we consider 12-month return level map in Seoul, which implies that an extreme PM_{10} intensity occurs once a year with q_p and $p = 1/12$. For example, if 12-month return level at $d = 1$h is $300\mu g/m^3$, that means $300\mu g/m^3$ high-density PM_{10} event with 1-hour duration occurs once a year. For this purpose, we first compute 12-month PM_{10} return levels based on GEV parameters for all 25 stations according to different duration time. Then we construct a 12-month PM_{10} return level map in Seoul by applying kriging to the 12-month return level values. Figure 5.6 shows the 12-month return level maps according to different duration time $d = 1, 4, 24$, and 96. As shown, overall patterns are similar that the region with relatively high return levels forms a peanut shape. However, as the duration time

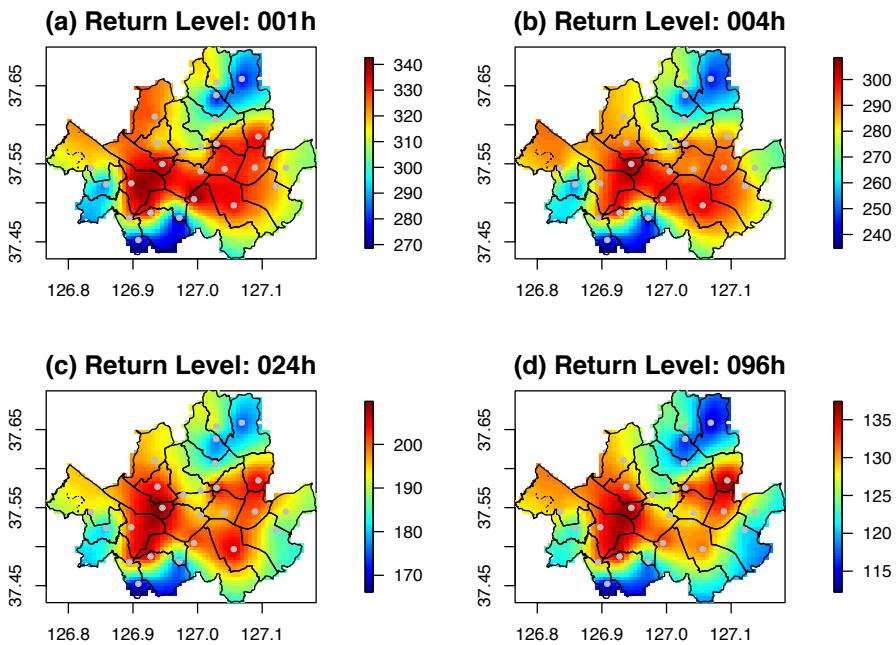

Figure 5.6: 12-month PM_{10} return level maps in Seoul over various duration times.

increases, the area of the peanut shape decreases. Especially, the return levels near the south-east area in Seoul are getting weaker as the duration time increases.

We now discuss temporal multiscale modeling based on the modified scaling property. To be specific, we would like to construct a projected return level map of arbitrary d-hour aggregated PM extremes in Seoul. In other words, given return level values, we want to obtain projections of return level values (or maps) at any arbitrary duration time d, where d can be any continuous value. Of course, when the data size is small, it is feasible to compute return levels at lots of d values. However, when data are massive since the number of stations is huge and the length of data is large, constructing return level maps at numerous d's is computationally

massive; hence, it cannot be able to analyze PM extremes over various temporal scales.

To perform a flexible temporal multiscale analysis of PM extremes, we utilize the proposed modified scaling property of (5.6), which provides GEV parameters of any duration time d, $\mu_d(x)$, $\sigma_d(x)$, $\xi_d(x)$ without computing the corresponding aggregated PM_{10} data and estimating the parameters by ML estimation. Hence, it is easy to construct the 12-month return level maps according to various time scales. Generating maps with various duration times might be useful for making environmental policy by the local government. From Figure 5.6, we observe that return level of Jungrang-gu station is very high when $d = 96$ hour, which implies that Jungrang-gu should carefully consider long-term period PM_{10} extremes, while return levels of Yongsan-gu and Seongdong-gu are relatively low at $d = 96$ and are high at $d = 0$ or 6, so they should focus on the short-period extremes.

To evaluate the performance of the proposed scale property, we compare it with a direct estimation used for Figure 5.6 and the conventional scale property of (5.3). Figure 5.7 shows the 12-month return level maps by three methods. As one can see, the conventional scale property gives higher return levels than those of the direct estimation. These results are almost identical to those computed from the direct estimation. Note that we control the color scale for a fair comparison. For further comparison, we compute the root mean square error (RMSE) as a function of duration time d, defined as

$$\mathrm{RMSE}_d = \sqrt{\frac{\sum_{i=1}^{n}(\hat{R}_{d,i} - R_{d,i})^2}{n}},$$

where $R_{d,i}$ is the return level value at station i and duration time d by the direct estimation and $\hat{R}_{d,i}$ is the return level value by using the conventional or proposed scale property. In the case that the value of RMSE is near to zero, the corresponding scaling method is a good approximation

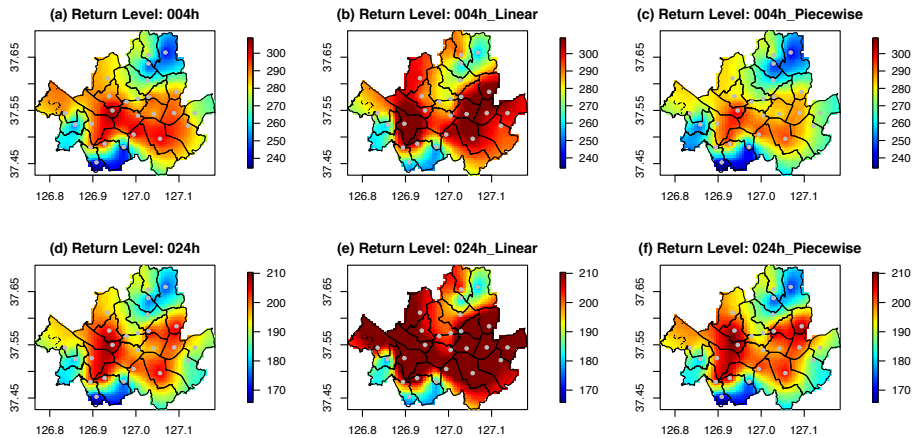

Figure 5.7: Projected 12-month PM_{10} return level maps in Seoul with our proposed modified approach (right) compared with direct estimation without scale invariance (left) and simple linear scaling (center).

of the direct estimation. Figure 5.8 shows the RMSE values by two scaling methods. As shown in Figure 5.8, the proposed method outperforms the conventional simple linear scaling approach over the entire range of d.

5.3 Spatio-temporal multiscale analysis of Seoul extreme PM_{10} data

In this section, we extend the temporal multiscale approach in Section 5.2 to spatio-temporal multiscale analysis of extreme PM_{10} data in Seoul.

5.3.1 Spatio-temporal aggregation of Seoul extreme PM_{10} data

Each extreme PM_{10} event has its event area. For a better understanding of the area variability, we construct areal PM_{10} data through pointwise PM_{10} data. Our final goal is to obtain the 12-month PM_{10} return level intensity

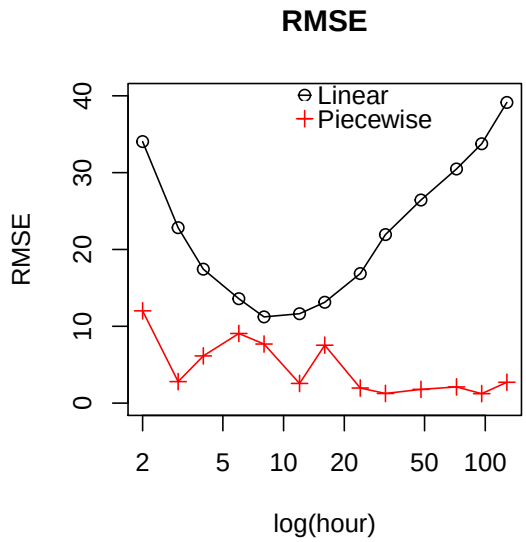

Figure 5.8: Root mean square errors of the simple linear scaling (black) and the modified scaling approach (red) with various duration times (x-axis).

over various areal scales and temporal aggregations simultaneously.

For constructing areal data, we use kriging to make areal data. To fill the area, we choose 5000 grid points on a regular grid. We then obtain spatial interpolated 5000 data on the designed grid points by kriging. Finally, we compute $Y_{d,a}(x)$ that is the average of sample points inside a circle with radius a. Figure 5.9 shows an example of constructing areal data.

For data analysis, we consider multiple radius, from r=0.25km, to r=5km, by increasing r=0.25km. In the case that the radius is larger than 5km, the corresponding circle contains too much area outside Seoul, which brings severe distortion of areal PM_{10} intensity. Once hourly areal PM_{10} at every station are obtained, we compute n-hour aggregated PM_{10} data by simply following the same steps of the temporal aggregation in Section 5.2. Let $Z_{d,a}(x)$ be the monthly maximum of $Y_{d,a}(x)$, which is the weighted average of kriging samples at station x. To check whether the

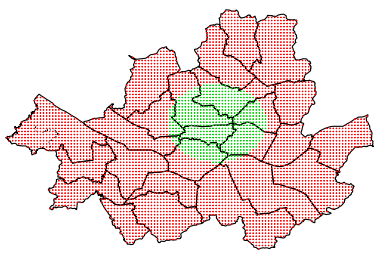

Figure 5.9: An example of areal aggregation using kriging with radius 5km. Red points are 5000 grid samples inside Seoul and green points are an example of samples inside an area with radius 5km.

Radius	Point	0.5km	1km	1.5km	2km	2.5km	3km	3.5km	4km	5km
Ratio	0.97	0.96	0.96	0.96	0.95	0.96	0.96	0.95	0.95	0.96

Table 5.5: The ratio following GEV distribution after the Anderson-Darling test over various area radii.

constructed monthly maximum PM_{10} data across various areas follow a GEV distribution, we perform Anderson-Darling test. As listed in Table 5.5, all monthly PM_{10} maxima follow a GEV distribution regardless of aggregation area radius.

For modeling areal aggregated PM_{10} with the assumption that there is a scaling invariance across area and time, we consider the scale relation of DeMichele et al. (2001) that derived when data have simple scaling and areal reduction as

$$q_{d,a}(x;p) = q_{d_{ref},a_{ref}}(x;p) \times \Big(\frac{d}{d_{ref}}\Big)^{-\nu(x)} \times \Big(1+\omega(x)\big(\frac{a^{z(x)}}{d}\big)^{b(x)}\Big)^{-\nu(x)/b(x)}, \tag{5.7}$$

where $\omega(x), z(x)$ and $b(x)$ are coefficients and a_{ref} implies the point data

$(a_{ref} = 0)$. Note that $\left(1+\omega(x)\left(\frac{a^{z(x)}}{d}\right)^{b(x)}\right)^{-\nu(x)/b(x)}$ is the reduction effect of areal return level. In the case of $a = 0$, the equation represents a point data case in (5.5), and the parameter $\nu(x)$ is equivalent to the scaling exponent in the scale property which is fitted by the IDF curve.

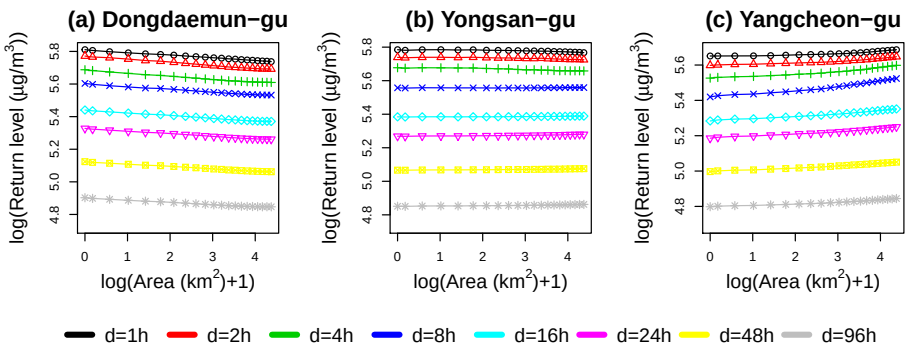

Figure 5.10: GEV return level values of Dongdaemun-gu (left), Yongsan-gu (center), and Yangcheon-gu (right) stations along with various area radius excluding point area data. The y-axis denotes 12-month PM_{10} GEV return level.

Figure 5.10 shows 12-month PM_{10} return levels as a function of log (area) in three different stations, which are obtained by direct calculation with actual data. Different color represents different duration times. As shown, all three stations have parallel trends over varying duration time. It implies that log (area) and log (duration time) may affect GEV return level independently. In other words, not all air pollution stations have decreasing return level at all as area radius increases, which implies that Seoul PM_{10} data are not fit by the equation (5.7). Nevertheless, an investigation of the spatio-temporal variation is still important. The result shows that for a certain short duration time and small area, the spatio-temporal variation of return level can be approximated by a linear or piecewise linear function. By allowing a simple interaction term, the log-linear approximation of (5.7)

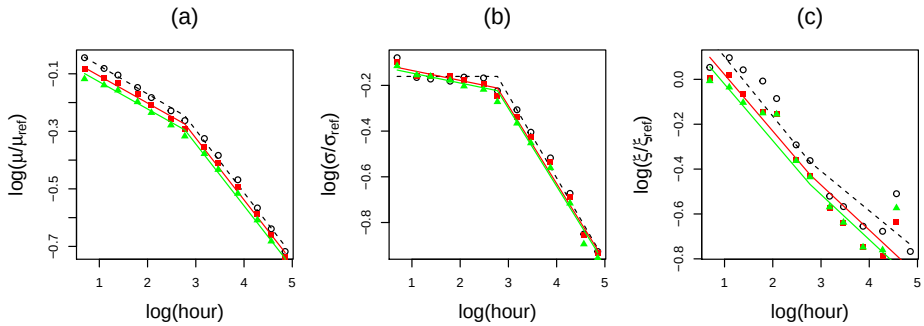

Figure 5.11: Ratios of GEV (a) location, (b) scale and (c) shape parameter as function of d and a at Dongdaemun-gu station. The x-axis denotes duration time, and different color means different area coverage, black points and dotted lines are point data, same in Figure 5.4, red is r=2km and green is r=5km case. Solid lines are estimated values using (5.9).

for the short duration time and small area can be expressed as

$$q_{d,a}(x;p) \approx q_{d_{ref},a_{ref}}(x;p) \times \left(\frac{d}{d_{ref}}\right)^{-\nu(x)} \times (1+a)^{c(x)} \times e^{-b(x)\log\left(\frac{d}{d_{ref}}\right)\log(1+a)} \tag{5.8}$$

which is a simple expression of the behavior of Seoul extreme PM_{10} data, shown in Figure 5.10. By taking the logarithm on both sides in (5.8), we generate simple linear modeling. Here $c(x)$ and $b(x)$ measure the amount of influence of $\log(1+a)$ and $\log(1+a)\log\left(\frac{d}{d_{ref}}\right)$ to $\log(q_{d,a}(x;p)/q_{d_{ref},a_{ref}}(x;p))$, respectively. Note that when $a = 0$, the equation (5.8) is equal to the simple scaling property.

Figure 5.11 shows three ratios $\mu_{d,a}/\mu_{d_{ref},a_{ref}}$, $\sigma_{d,a}/\sigma_{d_{ref},a_{ref}}$, $\xi_{d,a}/\xi_{d_{ref},a_{ref}}$ of GEV parameters (location, scale, shape) as function of duration time d and area a at Dongdaemun-gu station. Colors imply different area coverage: black points and dotted lines are exactly same as in Figure 5.4, and red and green lines are GEV parameter ratios with areal aggregation radius

2km and 5km, respectively. The GEV parameter ratio varies across area. Especially, the ratio of the shape parameters is dynamically changed over area. To reflect this feature, we consider new piecewise log-linear scaling equations for GEV parameters by combining the approximation (5.6) and (5.8),

$$\begin{aligned}
\mu_{d,a} \quad (x) &= \mu_{d_{ref},a_{ref}}(x) \times \Big(\frac{d}{d_{ref}}\Big)^{-\left(\nu_{\mu_1}(x)I(d\le\tau)+\nu_{\mu_2}(x)I(d\ge\tau)\right)} \\
\times \quad (1+a)^{c_\mu(x)} &\times e^{-\{b_{\mu_1}(x)I(d\le\tau)+b_{\mu_2}(x)I(d\ge\tau)\}\log\left(\frac{d}{d_{ref}}\right)\log(1+a)}, \\
\sigma_{d,a} \quad (x) &= \sigma_{d_{ref},a_{ref}}(x) \times \Big(\frac{d}{d_{ref}}\Big)^{-\left(\nu_{\sigma_1}(x)I(d\le\tau)+\nu_{\sigma_2}(x)I(d\ge\tau)\right)} \\
\times \quad (1+a)^{c_\sigma(x)} &\times e^{-\{b_{\sigma_1}(x)I(d\le\tau)+b_{\sigma_2}(x)I(d\ge\tau)\}\log\left(\frac{d}{d_{ref}}\right)\log(1+a)}, \\
\xi_{d,a} \quad (x) &= \xi_{d_{ref},a_{ref}}(x) \times \Big(\frac{d}{d_{ref}}\Big)^{-\left(\nu_{\xi_1}(x)I(d\le\tau)+\nu_{\xi_2}(x)I(d\ge\tau)\right)} \\
\times \quad (1+a)^{c_\xi(x)} &\times e^{-\{b_{\xi_1}(x)I(d\le\tau)+b_{\xi_2}(x)I(d\ge\tau)\}\log\left(\frac{d}{d_{ref}}\right)\log(1+a)},
\end{aligned} \tag{5.9}$$

where $c_\mu(x)$, $c_\sigma(x)$, and $c_\xi(x)$ denote new scaling exponents for areal aggregation that can be negative values and $b_{\mu_1}(x)$, $b_{\mu_2}(x)$, $b_{\sigma_1}(x)$, $b_{\sigma_2}(x)$, $b_{\xi_1}(x)$, and $b_{\xi_2}(x)$ reflect the amount of interaction between a and d. When $a = 0$, (5.9) is same as (5.6). As one can see, we consider an interaction term between area coverage a and time duration d. Parameters in (5.9) can be easily estimated by MARS after log-transformation.

5.3.2 Result: Areal aggregation of Seoul extreme PM_{10} data

Figure 5.12 shows surface projections of 12-month PM_{10} return level over ranges of duration time and area, which are obtained by (5.8) and (5.9). For all estimations, τ is data-adaptively chosen, 12 or 16 hours. We note that the red open circles denote the actual return levels obtained by direct calculation with the data. In the case of Yongsan-gu, estimates give higher return levels compared with direct computation. However, PM_{10} return levels surface well describe overall spatial and temporal Seoul PM_{10}

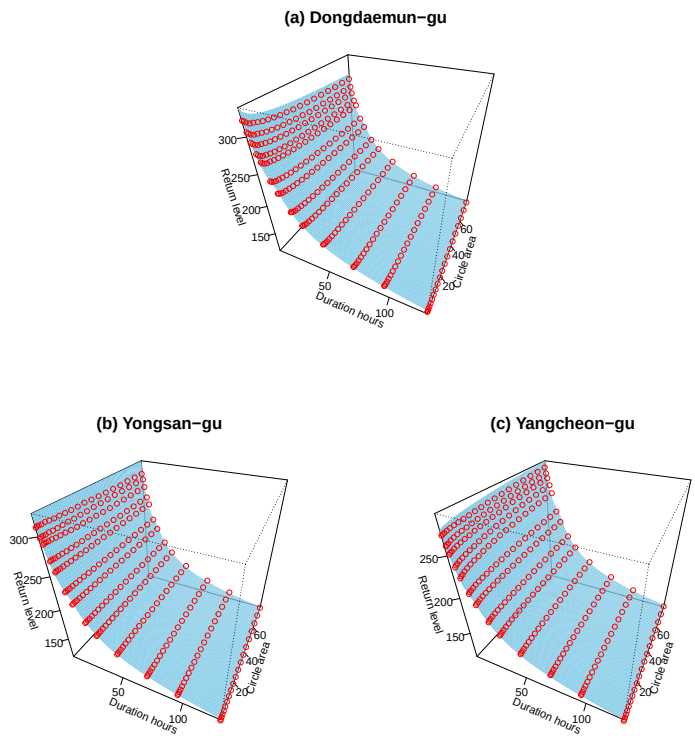

Figure 5.12: Projected 12-month PM_{10} return levels surface (sky blue) across various duration times and area at (a) Dongdaemun-gu, (b) Yongsan-gu and (c) Yangcheon-gu stations. Red points are actual return levels obtained from the direct calculation with the data.

extreme trends. As shown, return levels of all three stations decrease as duration time increases, and the values seem to be independent of area coverage. One interesting observation is that the tendency of return levels as a function of the area varies according to the station; as area coverage increases, 12-month PM_{10} return level at Dongdaemun-gu station is linearly decreasing, that at Yongsan-gu is a constant and that at Yangcheon-gu is linearly increasing.

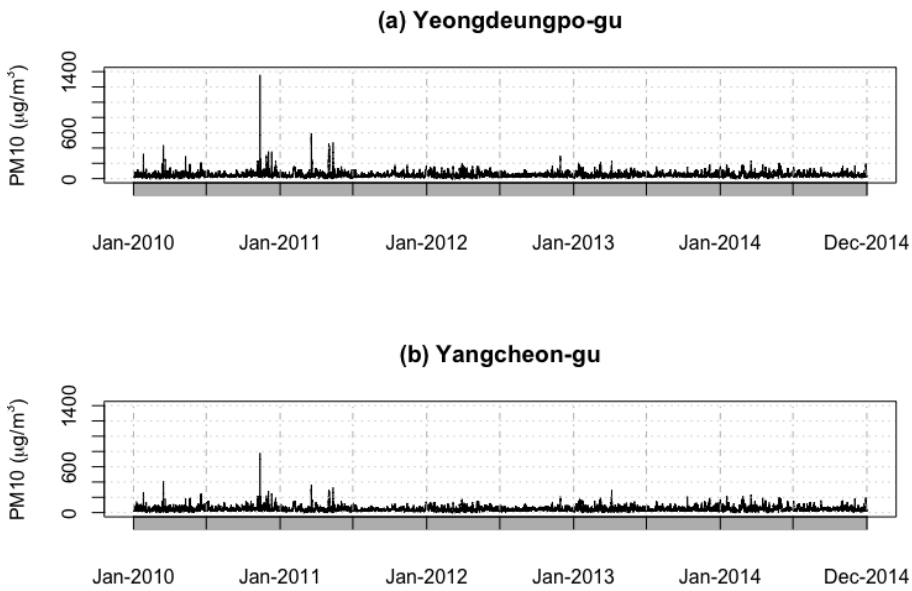

Figure 5.13: Hourly PM_{10} time series from (a) Yeongdeungpo-gu and (b) Yangcheon-gu.

To analyze these tendencies, we observe spatial variations of hourly PM_{10} time series from Yeongdeungpo-gu and Yangcheon-gu where two stations are closely located, which are shown in Figure 5.13. In Yeongdeungpo-gu station, the number of PM_{10} intensity over 400 $\mu g/m^3$ is 39, whereas, in Yangcheon-gu station, it is only 4. In addition, we observe that, when PM_{10} density is high, the density of Yeongdeungpo-gu station is much

stronger than that of Yangcheon-gu. As area radius increases, areal aggregated data of Yangcheon-gu are affected by those of Yeongdeungpo-gu. Therefore, it turns out that areal monthly PM_{10} maximum density of Yangcheon-gu increases along with area radius containing high-level PM_{10} data of Yeongdeungpo-gu.

5.4 Summary and discussion

In this chapter, we have considered multiscale analysis and modeling of monthly PM_{10} maxima in Seoul to understand their spatio-temporal variations over various spatial and temporal scales.

For multiscale temporal analysis of PM_{10} extremes, including temporal projections of them, we have adopted the scale property that has widely used in hydrology. We have found that there is a scaling property of PM_{10} data in Seoul, but the scale relation is not the same as the conventional one. To solve the disagreement, we have proposed two modifications. One is to take a piecewise linear approximation to reflect a nonlinear scale relation and hold advantages of the conventional scale property. The other is to introduce an additional scaling exponent for the shape parameter of GEV distribution and use different estimation for three GEV parameters. The proposed modified scaling approach has provided accurate projection results for Seoul PM_{10}, compared to the conventional scaling approach. This result implies that the behavior of extreme PM_{10} is similar to rainfall data, but slightly different, which means that the underlying physics of extreme PM_{10} related in duration time might also be different. We remark that the parameter ν of the modified scaling property in (5.6) cannot be directly estimated by the slope of IDF plot, but it has shown some benefit when data do not follow the exact simple scaling property. The modified scaling property can be practically implemented by MARS.

Furthermore, we have studied the spatio-temporal structure of high-level PM_{10} across various temporal and spatial scales. For the construction of spatially aggregated data, we have proposed a new approach based on grid sampling and kriging. To investigate temporal and spatial variation simultaneously, we show spatio-temporal surface GEV plot according to duration time and area coverage and found different types of spatio-temporal variation, which can be used for spatial-temporal projection at any arbitrary duration time and area coverage.

We have remarked that there is no integrated areal reduction property to explain the spatial behavior in extreme Seoul PM_{10} data. It can be conjectured for two reasons. First, it seems that the temporal and spatial scale in this thesis is too short to observe the areal reduction effect. Second, although grid sampling and kriging provide a good approximation result, it cannot reflect the complex spatial behavior of PM_{10} data.

There are some possible approaches to understand the multiscale structure of PM_{10} extremes. For GEV distribution, L-moment estimation of Hosking (1990) has been well used, which is known that it provides robust and accurate results in a small finite sample. We have applied the L-moment estimation to obtain the return level map. The results are almost identical to those by MLE in this chapter so that we have omitted the results. A threshold-based approach, such as peak-over-threshold (POT) method is another option to analyze extreme PM_{10} at a single scale. Threshold selection is an important issue for the implementation of the POT method. However, the PM_{10} intensity is varying across the spatial and temporal scale. Therefore, for multiscale analysis of PM_{10} data, it should be required to modulate thresholds according to spatial and temporal scale. Using a spatial extreme model such as max-stable model is also a possible attempt. It might be interested in the comparison between the spatial extreme model and our proposed approach. These topics are left

for future research.

Chapter 6

Concluding remarks

In this thesis, we have proposed the analysis of spatio-temporal data; enhancement of lifting scheme on graph signal data via clustering-based network design, lifting scheme for streamflow data and multiscale analysis for PM_{10} extremes in Seoul. In the enhancement of lifting scheme on graph signal data, we reduce mean of the total prediction error under the piecewise generalized moving average graph signal data setting by using cluster-based neighborhood selection. In lifting scheme for streamflow data, we suggest a new method, by combining flow-adaptive neighborhood selection, prediction filter construction, and removal order selection, while reflecting characteristics of streamflow data. In the multiscale analysis for PM_{10} extremes, we introduce a new concept called "piecewise scaling property", which is a variation of simple scaling property, to construct extreme PM_{10} return map at arbitrary event duration time and area coverage. Throughout the numerical experiments and real data analysis, we show that all proposed methods are useful compared to other existing methods.

Appendix A

Generalized extreme value distribution

Generalized extreme value (GEV) distribution is known that it is suitable for fitting distribution of blockwise extreme value data, such as a distribution of monthly PM_{10} maxima. The probability density function (PDF) of GEV distribution is

$$f(s;\mu,\sigma,\xi) = \frac{1}{\sigma}t(x)^{\xi+1}e^{-t(x)},$$

where

$$t(x) = \begin{cases} \left(1+(\frac{x-\mu}{\sigma})\xi\right)^{-1/\xi} & \text{if } \xi \neq 0 \\ e^{-(x-\mu)/\sigma} & \text{if } \xi = 0. \end{cases}$$

μ, σ, ξ are called the location, scale and shape parameters, respectively. The corresponding cumulative distribution function (CDF) is

$$F(x;\mu,\sigma,\xi) = \exp\left(-\left(1+\xi\left(\frac{x-\mu}{\sigma}\right)\right)^{-1/\xi}\right).$$

GEV location and scale parameters have similar roles with those in a normal distribution. Sign of the shape parameter decides GEV family types. Figure A.1 shows the three types of GEV class by changing the sign of the

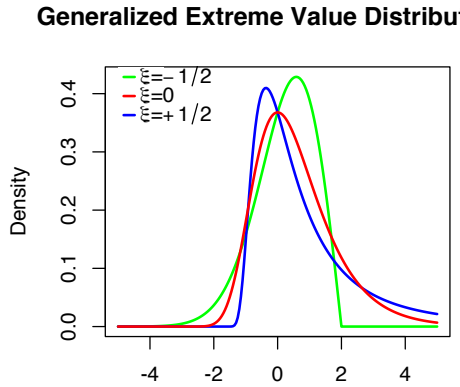

Figure A.1: GEV distribution at $\mu = 0$, $\sigma = 1$ with different shape parameters, $\xi = -0.5$ (green), $\xi = 0$ (red), and $\xi = 0.5$ (blue).

shape parameter: when the sign is zero, it is called Gumbel family, when the sign is positive, it is called Fréchet family, and when the sign is negative, it is called Weibull family. Note that for the computation of GEV, we use the maximum likelihood fitting for GEV in R package `ismev`.

Appendix B

Scaling property theory

Scaling property is widely used in multiscale rainfall analysis. It explains the relationship between two random variables $Y_d(x)$ and $Y_{d_{ref}}(x)$. For simplicity, I only consider nonnegative random variable Y. Every evenly-timewise-spaced climate data, including extreme PM_{10}, have their own reference observation time such as an hour, a month, etc. In this thesis, the reference time is an hour since the PM_{10} data are observed at every hour. Let $Z_d(x)$ be a maximum of $Y_d(x)$, d hour duration PM_{10} density at location x. We are interested in the distribution of $Z_d(x)$. If we could know the information about the whole distribution, we also know the information about quantiles of $Z_d(x)$; hence, we could compute the extreme PM_{10} intensity at any observation location and/or duration time.

On the other hand, a wide sense simple scaling property between $Y_{d_{ref}}(x)$ and $Y_d(x)$ is expressed by the following equation (Gupta *et al.*, 1990), (Menabde *et al.*, 1999)

$$Y_d(x) \stackrel{d}{=} \left(\frac{d}{d_{ref}}\right)^{-\nu} Y_{d_{ref}}(x)$$

where $\frac{d}{d_{ref}}$ is a scaling factor and ν is a scaling exponent. The meaning of this simple scaling property is $(\frac{d}{d_{ref}})^{-\nu} Y_{d_{ref}}(x)$ and $Y_d(x)$ are equal in

distribution. Thus, if we know the distribution of $Y_{d_{ref}}(x)$, the scaling factor and the scaling exponent, it is able to obtain the distribution of $Y_d(x)$ as well. This property also holds for $Z_d(x)$, that is,

$$Z_d(x) \stackrel{d}{=} \left(\frac{d}{d_{ref}}\right)^{-\nu} Z_{d_{ref}}(x). \tag{B.1}$$

The role of the scaling exponent can be interpreted as follows. When the scaling exponent is less than zero, then $d > d_{ref}$ means $Z_d(x) > Z_{d_{ref}}(x)$. Conversely, when the scaling exponent is bigger than zero, then $d < d_{ref}$ implies $Z_d(x) < Z_{d_{ref}}(x)$.

On the other hand, strict sense simple scaling property is defined by cumulative distribution function (CDF) of Y, F,

$$F_d(x) = F_{d_{ref}}\Big(\Big(\frac{d}{d_{ref}}\Big)^{\nu} x\Big). \tag{B.2}$$

If we assume that F_{Y_d} and $F_{Y_{d_{ref}}}$ are CDF of GEV(μ_d, σ_d, ξ_d) and GEV$(\mu_{d_{ref}}, \sigma_{d_{ref}}, \xi_{d_{ref}})$ distribution. With assumption $\xi_d = \xi_{d_{ref}} = \xi$, we derive scaling equation between GEV parameters from (B.2)

$$\mu_d = \mu_{d_{ref}} \times (d/d_{ref})^{-\nu} \text{ and } \sigma_d = \sigma_{d_{ref}} \times (d/d_{ref})^{-\nu}. \tag{B.3}$$

Equation (B.3) is used for estimating μ_d and σ_d from $\mu_{d_{ref}}$, $\sigma_{d_{ref}}$ and ν.

Bibliography

Artiola, J, Pepper, I. L. and Brusseau, M. L. (2004). *Environmental Monitoring and Characterization*. Elsevier Science and Technology Books.

Bernard, E., Naveau, P. and Vrac, M. (2013). Clustering of Maxima: Spatial Dependencies among Heavy Rainfall in France. *J. Clim.*, **26**, 7929–7937.

Cressie, N., Frey, J., Harch, B. and Smith, M. (2006). Spatial Prediction on a River Network. *J. Agric. Biol. Environ. Stat.*, **11**, 127–150.

Daubechies, I. (1992). *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics.

De Michele, C., Kottegoda, N. T. and Rosso, R. (2001). The derivation of areal reduction factor of storm rainfall from its scaling properties. *Water Resour. Res.*, **37**, 3247–3252.

Donoho, D. L. and Johnstone, I. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, **81**, 425–455.

Ferreira, M. A. R. and Lee, H. K. H. (2007) *Multiscale Modeling: A Bayesian Perspective*, Springer Science and Business Media.

Friedman, J. H. (1991) Multivariate Adaptive Regression Splines. *Ann. Stat.*, **19**, 1–67.

Gallacher, K., Miller, C., Scott, E. M., Willows, R., Pope, L. and Douglass, J. (2017). Flow-directed PCA for monitoring networks. *Environmetrics*, **28**, e2434.

Goswami, J. C. and Chan, A. K. (2005). *Fundamentals of Wavelets; Theory, Algorithms, and Applications*. John Wiley and Sons.

Gupta, V. K. and Waymire, E. (1990). Multiscaling properties of spatial rainfall and river flow distribution. *J. Geophys. Res. Atmos.*, **95**, 1999–2009.

Hamilton, J., Nunes, M. A., Knight, M. A. and Fryzlewicz, P. (2018). Complex-Valued Wavelet Lifting and Applications. *IEEE Trans. Signal Process. Lett.*, **60**, 48–60.

Haris, A., Shojaie, A. and Simon, N. (2018). Wavelet regression and additive models for irregularly spaced data. *Adv. Neural Inf. Process Syst.*, **NIPS2018**.

Hosking, J. R. M. (1990). L-Moments: analysis and estimation of distributions using linear combinations of order statistics, *J. Roy. Stat. Soc. B.*, **52**, 105–124.

Jansen, M. H. and Oonincx, P. (2005). *Second Generation Wavelets and Applications*. Springer Science and Business Media.

Jansen, M., Nason, G. P. and Silverman, B. W. (2009). Multiscale methods for data on graphs and irregular multidimensional situations. *J. Roy. Stat. Soc. B.*, **71**, 97–125.

Johnstone, I. M. and Silverman, B. W. (2004). Needles and straw in haystacks: Empirical Bayes estimates of possibly sparse sequences. *Ann. Stat.*, **32**, 1594–1649.

Kim, U. and Kim, J. (2011). A Study of Building Customized Management Strategies Based on Local PM_{10} Emission Inventory in Seoul (written in Korean). Seoul Development Institute.

Kim, U., Kim, J. and Hong, J. (2014). A Study on Establishing $PM_{2.5}$ Advisory Criteria with Emission Source Management System in Seoul (written in Korean). The Seoul Institute.

Knight, M. I. (2006). A Second Generation Wavelet Construction and Applications to Regression and Time Series Analysis. Ph.D. thesis, University of Bristol.

Knight, M. I. and Nason, G. P. (2009). A 'nondecimated' lifting transform. *Stat Comput.*, **19**, 1–16.

Knight, M. I., Nunes, M. A. and Nason, G. P. (2012). Spectral estimation for locally stationary time series with missing observations. *Stat Comput.*, **22**, 877–895.

Knight, M. I., Nason, G. P. and Nunes, M. A. (2017). A wavelet lifting approach to long-memory estimation. *Stat Comput.*, **27**, 1453–1471.

Knight, M. I. and Nunes, M. A. (2019). Long memory estimation for complex-valued time series. *Stat Comput.*, **29**, 517–536.

Laio, F. (2004). Cramer–von Mises and Anderson-Darling Goodness of Fit Tests for Extreme Value Distributions with Unknown Parameters. *Water Resour. Res.*, **40**, W09308.

Li, F., Osher, S., Qin, J. and Yan, M. (2016). A Multiphase Image Segmentation Based on Fuzzy Membership Functions and L1-Norm Fidelity. *J. Sci. Comput.*, **69**, 82–106.

Lindström, J., Szpiro, A. A., Sampson, P. D., Oron, A. P., Richards, M., Larson, T. V. and Sheppard, L. (2014). A Flexible Spatio-Temporal Model for Air Pollution with Spatial and Spatio-Temporal Covariates. *Environ. Ecol. Stat.*, **21**, 411–433.

Mallat, S. G. (1989). Multiresolution Approximations and Wavelet Orthonormal Bases of $L^2(\mathbb{R})$. *Trans. Am. Math Soc.*, **315**, 69–87.

Mallat, S. (1999). *A Wavelet Tour of Signal Processing*. Academic Press.

Martínez-Enríquez, E., Cid-Sueiro, J., Díaz-de-María, F. and Ortega, A. (2018a). Directional Transforms for Video Coding Based on Lifting on Graphs. *IEEE Trans. Circuits Syst. Video Technol.*, **28**, 933–946.

Martínez-Enríquez, E., Cid-Sueiro, J., Díaz-de-María, F. and Ortega, A. (2018b). Optimized Update/Prediction Assignment for Lifting Transforms on Graphs. *IEEE Trans. Signal. Process.*, **66**, 2098–2111.

Menabde, M., Seed, A. and Pegram, G. (1999). A simple scaling model for extreme rainfall. *Water Resour. Res.*, **35**, 335–339.

Nason, G. P. (1993). *Wavelet Methods in Statistics with R*. Springer Science and Business Media.

Nock, R. and Nielsen, F. (2004). Statistical Region Merging *IEEE Trans. Pattern. Anal. Mach. Intell.*, **26**, 1–7.

Nunes, M. A., Knight, M. I. and Nason, G. P. (2006). Adaptive lifting for nonparametric regression. *Stat. Comput.*, **16**, 143–159.

O'Donnell, D., Rushworth, A., Bowman, A. W. and Scott, E. M. (2014). Flexible regression models over river networks. *J. Roy. Stat. Soc. C.*, **63**, 47–63.

Park, S. and Oh, H.-S. (2017). Spatio-temporal analysis of particulate matter extremes in Seoul: Use of multiscale approach. *Stoch. Environ. Res. Risk Assess.*, **31**, 2401—2414.

Sweldens, W. (1996). The lifting scheme: A custom-design construction of biorthogonal wavelets. *Appl. Comput. Harmon. Anal.*, **3**, 186–200.

Sweldens, W. (1998). The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, **29**, 511–546.

Ver Hoef, J. M. and Peterson, E. E. and Theobald, D. (2006). Spatial statistical models that use flow and stream distance. *Environ. Ecol. Stat.*, **13**, 449–464.

Ver Hoef, J. M. and Peterson, E. E. (2010). A moving average approach for spatial statistical models of stream networks. *J. Am. Stat. Assoc.*, **105**, 6–18.

국문초록

이 논문은 다중 척도 분석을 시공간 자료에 응용한 방법들을 제시한다. 첫째, 그래프 신호 자료에서의 다중 척도 분석 방법 중 하나인 리프팅 스킴을 군집에 기반한 이웃 재설정을 통해 기대 예측 오차를 줄일 수 있음을 보이고 이를 통해 리프팅 스킴의 성능을 향상하였다. 둘째, 공간적으로 복잡하고 방향성이 있는 구조에서 생성된 유량 네트워크 자료에 알맞는 리프팅 스킴을 구성하기 위해 네트워크의 특성을 반영한 이웃 선택, 예측 필터 구성 및 영역 설정으로 유량 네트워크 자료에 대한 리프팅 스킴을 구성하고 시공간 자료에의 확장 가능성을 살펴보았다. 마지막으로 서울특별시 고농도 미세먼지 자료를 다양한 시간, 공간 및 시공간 집적을 통해 변환한 후 얻어진 일반화 극단값 모형의 모수들의 관계를 수문학에서 사용하는 강도-지속시간-발생빈도 곡선에 매듭을 추가한 변형된 형태의 강도-지속시간-발생빈도 곡선을 따라 모델링하였고 사례 연구를 통해 원 자료의 복귀 수준 지도를 좀 더 정확히 묘사할 수 있음을 보였다.

주요어 : 다중 척도 방법론, 리프팅 스킴, 유량 자료, 일반화 극단값 분포

학　번 : 2013 – 20215