이학석사 학위논문

A review of recent methods improving performance of lifting scheme

리프팅 스킴의 성능개선 방법에 대한 연구

2021년 2월

서울대학교 대학원

통계학과

조 준 일

이학석사 학위논문

A review of recent methods improving performance of lifting scheme

리프팅 스킴의 성능개선 방법에 대한 연구

2021년 2월

서울대학교 대학원

통계학과

조 준 일

A review of recent methods improving performance of lifting scheme

지도교수 오 희 석

이 논문을 이학석사 학위논문으로 제출함

2020년 12월

서울대학교 대학원

통계학과

조 준 일

조준일의 이학석사 학위논문을 인준함

2020년 12월

위 원 장	박 태 성	(인)
부위원장	오 희 석	(인)
위 원	임 채 영	(인)

Abstract

The main purpose of this paper is to compare fourier transform, wavelet, and lifting scheme, and to look at the recent studies of improving performances of lifting scheme. Fourier transform, which started from spectral analysis, has been studied and has several application. In recent times, studies on wavelet, which using wavelet as basis function and multiresolution analysis, are highly progressing. Among them, the lifting scheme, which called the second-generation wavelet, has the characteristic of generating a basis function without using a fourier transform, unlike first-generation wavelets. In this paper, we compare fourier transform, wavelet, and lifting scheme, and look at two papers that have improved performance of lifting scheme by focusing on selective removal and neighborhood selection. And we discuss further methods on improving performances of lifting scheme.

Keywords: Wavelet, Lifting Scheme, Fourier Transform

Student Number: 2017-20642

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In signal processing, fourier transform is major method which uses frequency information for analysis. However, fourier transform only consider frequency resolution, cannot get any information for time. There are several methods attempt to overcome this limitations, such as short time Fourier transform. Wavelet was also proposed to overcome this limitations of fourier transform. One major difference of wavelet and fourier transform is basis function, fourier transform uses sine wave as basis function while Wavelet uses small function called mother wavelet.

However, wavelet has also several limitations. First, data for wavelet must have dyadic size. If data size is not dyadic, we need to set data size dyadic by some process, for example, cutting out some data. Also data should be identically distributed. Sweldens (1996) proposed lifting scheme, which is called second-generation wavelet, to overcome this limitations.

Lifting scheme works by repeating three steps, split, prediction and update. Lifting scheme does not depends on dyadic data size, not require conditions such as data should be identically distributed. In this paper, we discuss about methods that improving performances of lifting scheme, and look at recent studies aiming improvement of performances of lifting scheme. In chapter 2, we introduce wavelet and lifting scheme. In chapter 3, we discuss several methods improving performances of lifting scheme. In chapter 4, we review two recent studies.

Chapter 2

Overview

2.1 Fourier Transform and Wavelets

Fourier transform was proposed to analysis data on frequency domain. With fourier transform, we can transform data from the time domain to the frequency domain, and the fourier coefficient represents the degree of contribution of the sine/cosine function for each frequency.

$$\widehat{f}(\lambda) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\lambda t} dt$$
$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \widehat{f}(\lambda) e^{-i\lambda t} dt$$

Fourier transform has some limitations. With fourier transform, we have perfect frequency resolution but cannot have any time information. Also fourier transform cannot represent properties of non-stationary waves, since fourier transform uses sine wave as basis function. One example is seismic wave, which is non-stationary wave and has sharp corner when seismic wave has arrived. To overcome this limitations, several methods have been proposed. One is short time fourier transform (STFT), which analyze function by split up the signal in time domain, and take a fourier transform for each section. Short time fourier transform can get some time resolution, but since short time frequency transform constructs result by combining the resulting transforms for each split, it has uniform time and frequency resolution for high and low frequency.

Wavelet transform uses wavelets, an orthonormal basis functions in $L^2(\mathbb{R})$ generated by translations and dilations of functions which called the father wavelet and mother wavelet. Two functions play a primary role in wavelet analysis : the scaling function (ϕ), and the wavelet (ψ). They are also called the father wavelet and mother wavelet, respectively. Significant difference between Wavelet and Fourier transform is basis function. Fourier transform uses sine wave as basis function, while wavelet uses function called mother wavelet, which have limited duration, as basis function.

2.2 Multiresoultion analysis

Multiresolution analysis is a method for L^2-approximation of functions with arbitrary precision. With multiresolution analysis, wavelet can be defined as MRA of $L^2(\mathbb{R})$. The following definition is Mallat's general notation of multiresolution analysis.

Definition 2.2.1 (Multiresolution analysis)**.** Let V_j, $j = \cdots, -2, -1, 0, 1, 2, \cdots$ be a sequence of subspaces of functions in $L^2(\mathbb{R})$. The collection of spaces $\{V_j, j \in Z\}$ is called a *multiresolution analysis with scaling function* ϕ if the following conditions hold.

1. (Nested) $V_j \subset V_{j+1}$.

2. (Density) $\overline{\cup V_j} = L^2(R)$.

3. (Separation) $\cap V_j = \{0\}$.

4. (Scaling) The function $f(x)$ belongs to V_j if and only if the function $f(2^{-j}x)$ belongs to V_0.

5. (Orthonormal basis) The function ϕ belongs to V_0 and the set $\{\phi(x - k), k \in Z\}$ is an orthonormal basis (using the L^2 inner product) for V_0.

The V_j's are called *approximation spaces* and different choice of ϕ results different multiresolution analysis.

2.3 Lifting scheme

Lifting scheme, proposed by Sweldens (1996), is called second-generation wavelet. Traditional wavelet has some limitations on data, such as size of data must be dyadic, and data should be identically distributed. Lifting scheme was proposed to overcome these limitations of wavelet. Lifting scheme consists 3 types of operation : split, predict, and update.

1. Split : input data at each specific level of decomposition j is split into prediction ($\mathcal{P}_j$) and update ($\mathcal{U}_j$) disjoint sets.

2. Predict : $\mathcal{P}_j$ set is predicted from the data of the $\mathcal{U}_j$ set using the prediction $\mathbf{p}_j$ filters, yielding the detail coefficients.

3. Update : data of the $\mathcal{U}_j$ set is filtered with detail coefficients of the $\mathcal{P}_j$ set using the update $\mathbf{u}_j$ filters, giving rise to the smooth coefficients.

4. Repeat : repeat above operations until we reach the expected resolution level.

This process is called forward lifting transform.

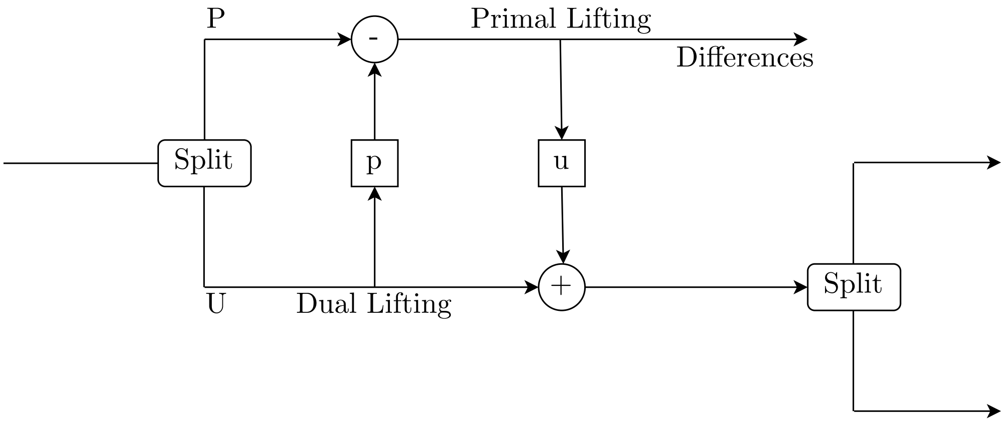

Figure 2.1 Forward lifting transform

In split stage, we first divide data into two sets, prediction set and update set. Next, in predict stage, we predict value of data in predict set by data in update set. In update stage, we update data in update set by predicted value with update filter. Then we repeat these stages in next level of data.

One characteristics of lifting scheme is once forward lifting transform is constructed, then its reverse transform is also available. In wavelet case, we cannot guarantee that inverse wavelet transform is that inverse of forward wavelet transform. In lifting scheme, inverse lifting transform can be easily constructed, by reverse order of processes and changing positive and negative sign.

1. undo update

2. undo predict

3. undo split

4. repeat above operations until we reach the expected resolution level

Following figure shows the flow chart of the inverse lifting scheme.

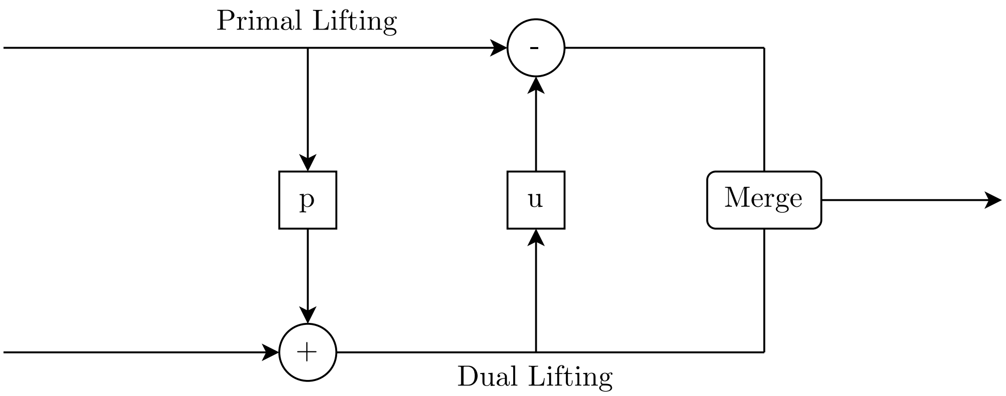

Figure 2.2 Inverse lifting transform

Chapter 3

Important factors improving lifting scheme on graph data

In this chapter, we first discuss the important factors improving lifting scheme, and then briefly introduce notations and models used in paper of Martínez-Enríquez *et al.* (2018), which will be introduced in next chapter.

3.1 Important factors improving lifting scheme

While constructing lifting scheme, there are some important factors determines performance of lifting scheme.

1. Size of prediction set : number of removal points at once

   Before beginning the lifting transform, we need to predetermine some conditions. Size of prediction set at one lifting step is one of major settings that influences performance of lifting scheme. Most studies use similar proportions as in wavelets. The LOCAAT method, proposed by Jansen *et al.* (2009), lifts one coefficient at one time.

2. Neighborhood selection

   It is also important to determine the neighborhoods of each point. Depending on this neighborhood selection, the lifting results will vary a lot. Too large neighborhoods will not represent the local nature of signals, and too narrow neighborhoods could cause some bias to prediction.

3. Removal point selection

   After determining the number of points to be removed at a time, we need to select that number of points to remove. Removal order can be said to be similar to the question that asking the priority of points in data. We can predetermine specific measure for comparing priority of points to solve this problem. In LOCAAT method, proposed by Jansen *et al.* (2009), they used the integral of scaling function for selecting the removal point.

4. Prediction filter

   Choosing prediction filter is essential part in lifting transform. In several studies, Haar (local constant), local linear methods are frequently used.

These factors greatly influences the performance of lifting scheme. In next chapter, we will discuss about this factors and review some papers related to these factors, especially neighborhood selection and removal point selection.

3.2 Notations on graph data

In this section, we follow some notations of Martínez-Enríquez(2018).

- Undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ where $\mathcal{V} = \{1, \cdots, N\}$ is a set of nodes between nodes, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is a set of edges between nodes, $\mathbf{W} = [w_{mn}]$ is the weighted adjacency matrix, with w_{mn} is a non-negative weight of the edge, when $mn \in \mathcal{E}$.
- The order of graph is the number of nodes, $N = |\mathcal{V}|$.
- $\mathcal{N}_m = \{n \in \mathcal{V} : mn \in \mathcal{E}\}$ is the set of neighbors of node m, and $\mathcal{N}_{[m]}$ is its closed neighborhood set.
- For any partition of $\mathcal{V}$ into two disjoint sets, $\mathcal{U}$ refers the update set and $\mathcal{P}$ refers the prediction set.
- The degree of a node m, $D_m = \sum_{n \in \mathcal{N}_m} w_{mn}$.
- A graph signal $\mathbf{x} = [x_1, x_2, \cdots, x_m, \cdots, x_N]$ is a signal defined on $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$, x_m is the value associated to node $m \in V$.

3.3 Lifting scheme on graph data

Lifting scheme can be extended to graph, which means can provide general solution for arbitrary undirected graph data. Lifting scheme on graph has also split, predict, update stage. In split stage, we split input graph into two disjoint set of nodes. In prediction stage, nodes in prediction set are predicted by nodes in update set with appropriate prediction filter. In update stage, nodes in update set are updated by details.

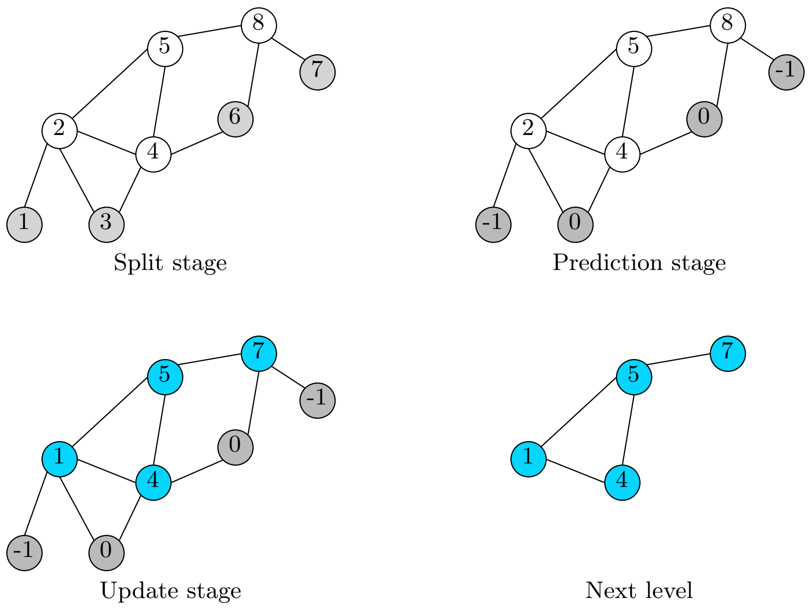

Figure 3.1 Lifting scheme on graph-signal data

In next chapter, we will introduce greedy UPA (update-predict assignment) algorithm proposed by Martínez-Enríquez *et al.* (2018). Before moving on to the next chapter, we will discuss about moving average model.

Definition 3.3.1 (General moving average model (MA model))**.** The moving average model for observed signal $\mathbf{x}$ defined on $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$ and for general node $m \in \mathcal{V}$ is

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

for some coefficients $q_{m,n}$, constant c, zero mean independent random variables ϵ_n and η_m, with respect to variances σ_ϵ^2 and σ_η^2.

The vector form of this equation is,

$$\mathbf{x} = c\mathbf{1} + \mathbf{Q}\boldsymbol{\epsilon} + \boldsymbol{\eta},$$

where $\mathbf{Q}_{m,n}$ is $q_{m,n}$ for $n \in \mathcal{N}_{[m]}$ and 0 otherwise.

Chapter 4

Recent approaches improving performances of lifting scheme

4.1 Previous approach

There are some traditional methods of lifting scheme, for example, max-lifting scheme and median-lifting scheme using nonlinear filter. Max-lifting scheme uses two coefficients for lifting so that makes the result of lifting which is sensitive to update/predict set selection more robust. Median-lifting scheme is also used for similar purpose.

LOCAAT algorithm is also well-known method for lifting scheme. LOCAAT (lifting one coefficient at a time) algorithm was proposed by Jansen *et al.* (2009), which removes one coefficients at a time with constructed removal order. At first step, n, algorithm defines set of indices $\mathcal{U}_n = \{1, 2, \cdots, n\}$, $\mathcal{P}_n = \{\emptyset\}$. At next step, $n-1$, algorithm choose the index which will be removed from the current set. To choosing the index to be removed, Jansen *et al.* (2009) used the minimal integral of scaling function for some appropriate measure. After we choose index, say i_n, then new update set and prediction set for $n-1$ level is $\mathcal{U}_{n-1} = \mathcal{U}_n \setminus \{i_n\}$ and $\mathcal{P}_{n-1} = \mathcal{P}_n \cup \{i_n\} = \{i_n\}$.

Weighted maximum cut is one example of lifting scheme methods for graph data, which provides split of graph which maximized the specified weighted maximum sum of graph. However this method only reflects structure of graph, does not reflect the signal of each nodes in graph. To overcome this limitation, Martínez-Enríquez *et al.* (2018) proposed greedy UPA algorithm using generalized moving average model which minimized the predefined prediction error. In addition, Park (2019) extended the idea, using piecewise generalized moving average model instead the generalized moving average model and applied clustering method for neighborhood selection. In this chapter, we will review these two papers.

4.2 Greedy UPA algorithm

In this section, we will introduce about greedy UPA algorithm proposed by Martínez-Enríquez *et al.* (2018), and review his paper. In this paper, we will only deal with a minimum preliminaries necessary for reviewing this paper. For details, you may refer the original paper, Martínez-Enríquez *et al.* (2018). This paper focuses on the UPA problem, which is split step of lifting scheme on graph-signal data. The main purpose of this paper is to analyze the UPA problem and derive optimal UPA for given assumptions on signal model and filters. Before introducing greedy UPA algorithm, we begin with some preliminaries.

We will use notations introduced in chapter 3. They define E_{tot} is total prediction error they defined as the sum of the expected value of squared prediction error. The main goal of this paper is solving following problem.

Problem 1. *UPA Problem: Find the UPA that minimizes the total prediction error*

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

where $|\mathcal{P}|$ is given.

The model which we will assume affects solution for this UPA problem. In this paper, he used general moving average model introduced in chapter 3. Also they defined linear predictor as follows.

Definition 4.2.1. *General linear predictor* Under condition $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{V})$, $\mathbf{x}$, and UPA,

$$\hat{x}_i = \sum_{k \in \mathcal{N}_i \cap \mathcal{U}} p_{i,k} x_k.$$

in vector form,

$$\hat{\mathbf{x}} = \mathbf{P}\mathbf{x}$$

where $\mathbf{P}_{i,j} = p_{i,j}$ for $j \in \mathcal{N}_i \cap \mathcal{U}$, and zero otherwise.

They claimed that correlation between nodes of graph through $\mathbf{M} = \sigma_\epsilon^2 \mathbf{Q}\mathbf{Q}^T$ affects the UPA that minimizes E_{tot}.

Back to the problem, it is not suitable to find the brute-force solution for UPA problem. They proposed greedy UPA algorithm, which locally minimize E_{tot} in each iteration. Algorithm starts with the state that every nodes of graphs are in $\mathcal{P}$ set, and for every iteration, chosen node c^* that minimizes E_{tot} is moved to $\mathcal{U}$ set. Define $\Theta^{\{t,c\}} = E_{tot}^{\{t-1\}} - E_{tot}^{\{t,c\}}$, the difference of E_{tot} of iteration t and $t-1$ when node c is moved. Algorithm 1 shows summary of the greedy UPA algorithm.

Algorithm 1 UPA Greedy Algorithm proposed by Martinez-Enriquez *et al.* (2018)

Require: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with predefined conditions, given $|\mathcal{P}|, \mathcal{U} = \{\varnothing\}$, $\mathcal{P} = \{\mathcal{E}\}$

Calculate $\mathbf{M} = \sigma_\epsilon^2 \mathbf{Q}\mathbf{Q}^T$ and get two-hop neighbors for every node in $\mathcal{V}$

while $|\mathcal{P}|^{\{t\}} > |\mathcal{P}|$ **do** for every $c = 1$ to $c = |\mathcal{P}|^{\{t\}}$, calculate $\Theta^{\{t,c\}}$ in the neighborhood of c

  Calculate $\boldsymbol{\Theta}^{\{t,c\}}$ and select the node c^* with maximum $\Theta^{\{t,c\}}$, $c^* = \text{argmax}_{c \in \mathcal{P}} \boldsymbol{\Theta}^{\{t,c\}}$

  Let $\mathcal{U} \leftarrow \mathcal{U} \cup \{c^*\}$, $\mathcal{P} \leftarrow \mathcal{P} \setminus \{c^*\}$

end while

return UPA solution

In his paper, several experimental results are introduced. The main contribution of this paper is that they prove the optimal depends on the correlation between nodes on graph, and it is not equivalent to weight maximum-cut or minimizing the discarded edges. Also proposed greedy UPA algorithm, and experimentally validated his conclusion. For further study, they suggested to using different model instead of generalized moving average model. Also setting different $|\mathcal{P}|$ nodes are also suggested.

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

In Park (2019)'s paper, in chapter 3, he introduced extended version of greedy UPA algorithm of Martinez-Enriquez *et al.* (2018), using piecewise generalized moving average model instead generalized moving average model and adjusting neighborhood selection by clustering based network design.

He expand the idea of Martinez-Enriquez *et al.* (2018) on more general settings. First, we need to introduce piecewise generalized moving average model, linear predictor, and prediction error. For details, you may refer Park (2019)'s paper, chapter 3.

Definition 4.3.1. *PGMA model (piecewise generalized moving average model)* Let c_m be signal of node m, and assume that the value of signals can be clustered into some values.

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

in vector form,

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

where $\mathbf{Q}_{m,n} = q_{m,n}$ for $n \in \mathcal{N}_{[m]}$ and zero otherwise.

Definition 4.3.2. *Linear predictors for the clustered version of piecewise generalized moving average model* For given $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{V})$, $\mathbf{x}$, and UPA, linear predictors for the clustered version of piecewise generalized moving average model are defined as follows,

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

in vector form,

$$\hat{\mathbf{x}} = \mathbf{P}^* \mathbf{x}$$

where $\mathbf{P}^*_{i,j} = p^*_{i,j}$ for $j \in \mathcal{N}_i^* \cap \mathcal{U}$, and zero otherwise.

Definition 4.3.3. *Prediction error for piecewise generalized moving average model* Total prediction error is defined by

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

where

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

where $\mathcal{U}_i = \mathcal{N}_i \cap \mathcal{U}$, $\mathbf{K} = \mathbb{E}\{\mathbf{x}\mathbf{x}^T\}$, $\mathbf{p}_i$ is column vector, $p_{i,k}$ for $k \in \mathcal{N}_i \cap \mathcal{U}$, and $\mathbf{M} = \sigma_\epsilon^2\mathbf{Q}\mathbf{Q}^T$ with $\mu_{m,n} = \sigma_\epsilon^2 \sum_{l\in\mathcal{N}_{[m]}\cap\mathcal{N}_{[n]}} q_{m,l}q_{n,l}$.

Park (2019) expanded idea of Martinez-Enriquez *et al.* (2018), and extend greedy UPA algorithm with different model assumption and clustering network-design. Greedy UPA algorithm with clustering based network design is summerized in algorithm 2. There are some updates in Park (2019)'s algorithm from Martinez-Enriquez *et al.* (2018)'s algorithm, is that additional steps are added such that computes clustering result and changes $\mathbf{W}$ matrix. Park (2019) also introduced UPA problem under the piecewise homogeneous model, but we will omit this part in this paper.

Algorithm 2 UPA Greedy Algorithm with clustering based network design proposed by Park (2019)

Require: $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, clustering sets $\mathcal{B} = (B_1, \cdots, B_R)$, weights $\hat{w}_s, \hat{w}_t$, $\mathcal{U} = \{\emptyset\}$, $\mathcal{P} = \{\mathcal{V}\}$ and optimal number $|\mathcal{P}_{opt}|$.

1: Define a new neighborhood of node $i \in B_r$ by clustering sets $\mathcal{B}$, $\mathcal{N}_i^* = \mathcal{N}_i \cap B_r$, $r = 1, \cdots, R$, where i is in B_r. It is equivalent to changing $\mathbf{W}$ into $\mathbf{W}^*$, where $\mathbf{W}^*$ is the neighborhood corrected version.

2: **while** $|\mathcal{P}| \leq |\mathcal{P}_{opt}|$ **do**

3: Set $\mathcal{U} \leftarrow \mathcal{U} \cup \{v\}$, $\mathcal{P} \leftarrow \mathcal{P} \setminus \{v\}$ and estimate $\hat{y}_i = \sum_{k \in \mathcal{N}_i^*} \mathbf{p}_i(y_k)$.

4: Compute empirical version of the total prediction error $\hat{E}_{tot}$, $\hat{E}_{tot} = \sum_{i \in \mathcal{P}} (y_i - \hat{y}_i)^2$.

5: Find v^* that minimizes $\hat{E}_{tot}$.

6: Set $\mathcal{U} \leftarrow \mathcal{U} \cup \{v^*\}$ and $\mathcal{P} \leftarrow \mathcal{P} \setminus \{v^*\}$

7: **end while**

8: Return UPA

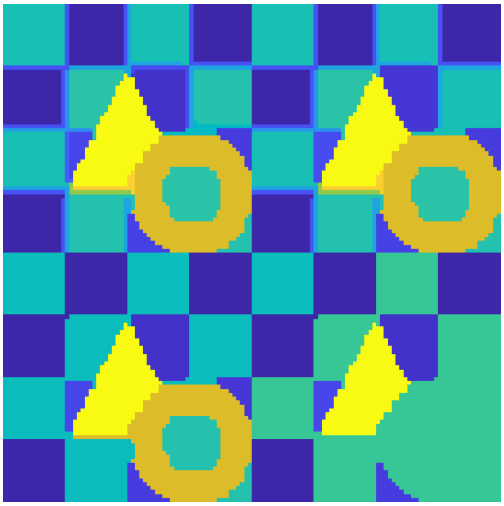

Figure 4.1 Segmentation-based edge addition called statistical region merging (SRM) image. Number of segmentation are 34,30,14,9 respectively.

In Park (2019), there are several simulation study and real data analysis. In this paper, we will review one of results, image data analysis. Image data analysis used test image data from Li *et al.* (2016) and described in Figure 4.1. S.Park simulated this data by his proposed method (Proposed) and 2-hop edges proposed by Martinez-Enriquez *et al.* (Enriquez) and compared result. Park (2019) claims that proposed method gives better construction from Figure 4.2 and table 4.1. S.Park claims that in his proposed method, choosing several nodes near edges that belongs into $\mathcal{U}$ set in the next level, has advantage in finding edges.

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)
$\sqrt{\hat{E}_{tot}}$ Enriquez (std.error)	87.81 (2.03)	31.75 (0.99)	27.83 (1.16)

Table 4.1 Blockwise image data simulation results of $\sqrt{E_{tot}}$ from Park (2019)

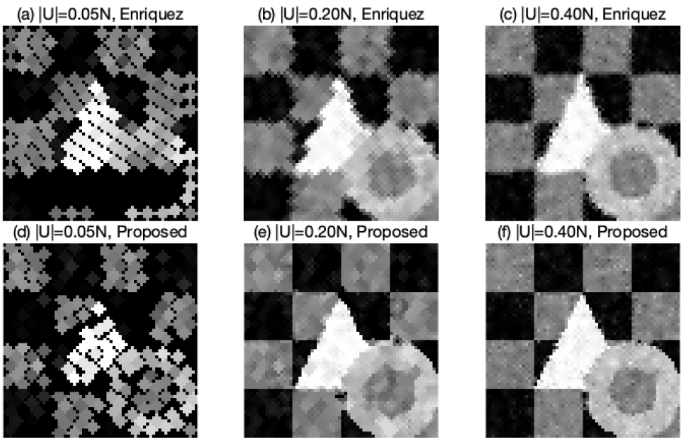

Figure 4.2 Graph image signal reconstruction results from Park (2019). (a),(b),(c) uses 2-hop edges (Enriquez) and (d),(e),(f) uses 2-hop intercluster edges (Proposed).

The main contribution of Park (2019)'s chapter 3, is that extend the concept of UPA algorithm with new models. As result, clustering method can be easily applied. For further research, he emphasized that we need to focus on how to constructing coarser graph-signal network.

He also focused on spatio-temporal data. Piecewise generalized moving average model can be reconsidered as space-time version, called piecewise constant spatio-temporal model. However, refer to spatio-temporal distances, we cannot apply this concept. For further study, the next step will be finding a way to overcome this problem.

Chapter 5

Conculsion

In this paper, we introduced the second-generation wavelet lifting scheme, reviewed the latest papers aiming at improving performance of lifting scheme, and discussed how to improve the performance of the lifting scheme in the future.

Unlike conventional wavelets, the lifting scheme consists three types of operation, split, predict, and update. Lifting scheme overcomes the conditions, dyadic data size and identically distributed data, which are the limitations of wavelets. Related studies on lifting schemes are ongoing, and among them, researches aiming at improving the performance of the lifting scheme are actively being continued. In this paper, we discussed the points can improve the performance of the lifting scheme, and introduced two recent studies about improving performance of lifting scheme by focusing on this points, removal point selection and neighborhood selection.

References

[1] E. Martínez-Enríquez, J. Cid-Sueiro, F. Díaz-de-María and A. Ortega. (2018), "Optimized Update/Prediction Assignment for Lifting Transforms on Graphs," *IEEE Transactions on Signal Processing*, vol. 66, no. 8, pp. 2098-2111.

[2] S. Park. (2019) "Multiscale Analysis of Spatio-Temporal Data," Ph. D. Dissertation, Department of Statistics, Seoul National University.

[3] S.G. Mallat. (1989) "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, pp.674-693

[4] Sweldens, W. (1996). "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Applied and Computational Harmonic Analysis*, Vol. 3, pp.186-200.

[5] Jansen, M., Nason, G. P. and Silverman, B. W. (2009). "Multiscale methods for data on graphs and irregular multidimensional situations". *Journal of the Royal Statistical Society Series B*, Vol. 71, pp.97-125.

[6] Li, F., Osher, S., Qin, J., Yan, M. (2016). "A Multiphase Image Segmentation Based on Fuzzy Membership Functions and L1-Norm Fidelity". *Journal of Scientific Computing*. Vol. 69.

국문초록

본 논문에서는 푸리에 변환과 웨이블릿, 그리고 리프팅 스킴을 비교하고, 리프팅의 성능을 개선하는 최신 연구 동향들을 살펴보는 것을 주 목적으로 한다. 파동분석에서 시작된 푸리에 변환은 현대에 이르러 다양한 방향으로 강화되어 왔고 그 중 웨이블릿을 이용한 기저와 다중해상도분석을 사용한 웨이블릿 변환에 관한 연구가 활발히 진행되고 있다. 그 중 제 2세대 웨이블릿으로 불리는 리프팅 스킴은 1세대 웨이블릿과 다르게 푸리에 변환을 사용하지 않고 기저함수를 생성한다는 특징을 가지고 있다. 푸리에 변환과 웨이블릿, 그리고 리프팅 스킴을 비교하고, 리프팅 스킴의 성능 향상에 관련한 최근 논문들 중 선택적 제거와 이웃 설정에 초점을 두어 성능 개선을 이루어낸 논문들을 살펴보고, 앞으로 이를 더 개선할 수 있을 거라 예상되는 방법을 논의한다.

주요어: 웨이블릿, 리프팅 스킴, 푸리에 변환

학번: 2017-20642