Signal Processing on Graphs: Recent Results, Challenges and Applications [1]

Antonio Ortega

Signal and Image Processing Institute
Department of Electrical Engineering
University of Southern California

Sept. 2013

[1] Supported in part by NSF (CCF-1018977) and NASA (AIST-05-0081).

Acknowledgements

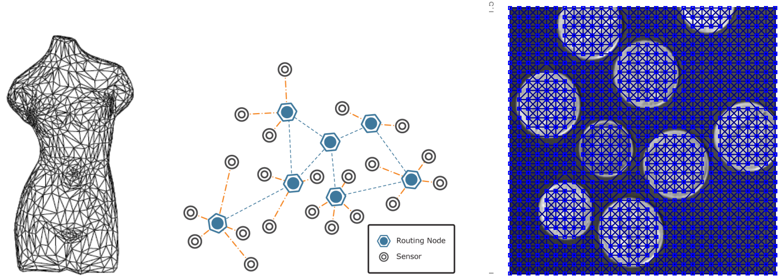
- Collaborators
  - **Dr. Sunil Narang** (Microsoft)
  - **Dr. Godwin Shen** (Northrop-Grumman)
  - **Eduardo Martínez Enríquez** (Univ. Carlos III, Madrid)
  - **Akshay Gadde, Jessie Chao, Yongzhe Wang** (USC)
  - Prof. Marco Levorato (UCI), Prof. Urbashi Mitra (USC)
  - Prof. Fernando Díaz de María (Univ. Carlos III, Madrid)
  - Prof. Gene Cheung (NII)
  - Prof. Pierre Vandergheynst (EPFL), Prof. Pascal Frossard (EPFL), Dr. David Shuman (EPFL).
- Funding
  - NASA AIST-05-0081
  - NSF CCF-1018977

Next Section

Motivation

Graphs provide a flexible model to represent many datasets:

- Examples in Euclidean domains



(a) (b) (c)

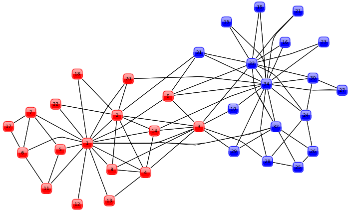
(a) Computer graphics[2] (b) Wireless sensor networks [3] (c) image - graphs

[2] From [Sweldens, 1999]

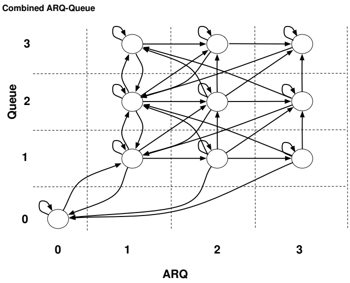
[3] From http://www.purelink.ca

Motivation

- Examples in non-Euclidean settings



(a)



(b)

(a) Social Networks [4], (b) Finite State Machines(FSM)

Graphs can capture complex relational characteristics (e.g., spatial, topological).

[4]Zacharay Karate Club [Zacahary, 1977]

Graph Signal Processing?

- Assume fixed graph structure: different graph signals on a given graph
- Define linear transforms for graph signals
- Use these for compression, denoising, interpolation, etc

What do we know about transforms for graph signals?

What do we know about transforms for graph signals?

- More than you think

What do we know about transforms for graph signals?

- More than you think

$$H = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

What do we know about transforms for graph signals?

- More than you think

$$H = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

  - Interpretation
    - Circulant matrix – Circular convolution
    - Eigenvectors: DFT
    - High pass filter: each row adds to 0
- Where is the graph?

What do we know about transformations on Graphs?

- Alternative representation

$$\mathbf{H} = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

What do we know about transformations on Graphs?

- Alternative representation

$$\mathbf{H} = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

$$\mathbf{H} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix} - \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

What do we know about transformations on Graphs?

- Alternative representation

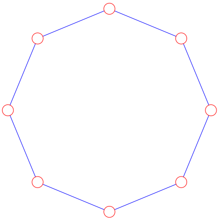
$$\mathbf{H} = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

$$\mathbf{H} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix} - \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{H} = \mathbf{D} - \mathbf{A}$$

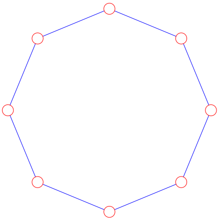
- Interpretation?

What do we know about transformations on Graphs?



$$\mathbf{H} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix} - \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

What do we know about transformations on Graphs?

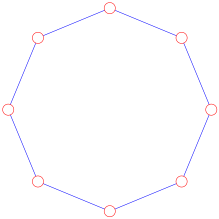


$$\mathbf{H} = \begin{pmatrix} 2&0&0&0&0&0&0&0\\ 0&2&0&0&0&0&0&0\\ 0&0&2&0&0&0&0&0\\ 0&0&0&2&0&0&0&0\\ 0&0&0&0&2&0&0&0\\ 0&0&0&0&0&2&0&0\\ 0&0&0&0&0&0&2&0\\ 0&0&0&0&0&0&0&2 \end{pmatrix} - \begin{pmatrix} 0&1&0&0&0&0&0&1\\ 1&0&1&0&0&0&0&0\\ 0&1&0&1&0&0&0&0\\ 0&0&1&0&1&0&0&0\\ 0&0&0&1&0&1&0&0\\ 0&0&0&0&1&0&1&0\\ 0&0&0&0&0&1&0&1\\ 1&0&0&0&0&0&1&0 \end{pmatrix}$$

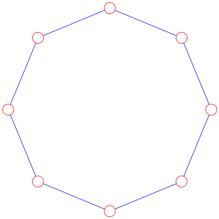
$$\mathbf{H} = \mathbf{D} - \mathbf{A}$$

- $\mathbf{A}$ and $\mathbf{D}$: adjacency and degree matrices
- $\mathbf{H} = \mathbf{L}$: graph Laplacian
- $\mathbf{H}$ can be interpreted as a local operation on this graph

Graphs



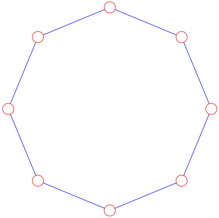
Graphs

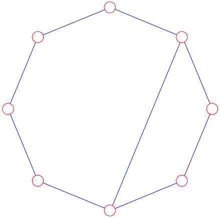


$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

- $\mathbf{H}$ is a simple polynomial of $\mathbf{L} = \mathbf{D} - \mathbf{A}$ on the cycle graph
- Can we do similar things on more complex graphs?

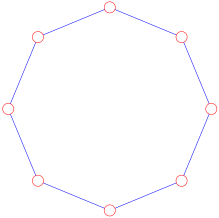
Graphs

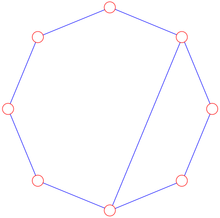




- $\mathbf{H}$ is a simple polynomial of $\mathbf{L} = \mathbf{D} - \mathbf{A}$ on the cycle graph
- Can we do similar things on more complex graphs?

Graphs

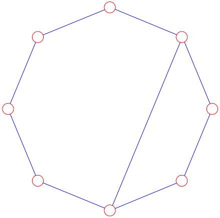




- $\mathbf{H}$ is a simple polynomial of $\mathbf{L} = \mathbf{D} - \mathbf{A}$ on the cycle graph
- Can we do similar things on more complex graphs?
- Yes! But things get more complicated

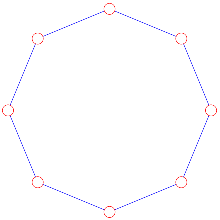
Graphs

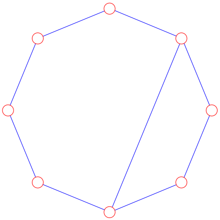
$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



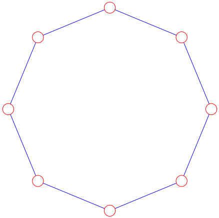
- $\mathbf{A}$ is no longer circulant – no DFT in general, but...
- Polynomials of $\mathbf{L} = \mathbf{D} - \mathbf{A}$ or $\mathbf{A}$ are local operators
- There will be a frequency interpretation

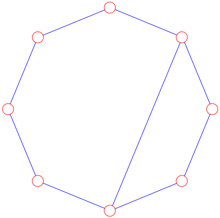
What makes these "graph transforms"?





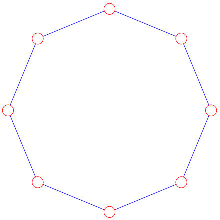
What makes these "graph transforms"?

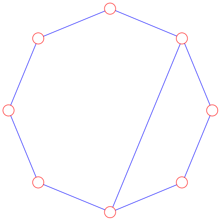




- Shift invariance: same filter at every sample

What makes these “graph transforms”?





- Shift invariance: same filter at every sample
- Graph-based shift invariance – Operator is the same, local variations captured by **A** or **L**.

$$\mathbf{H} = \mathbf{L} = \mathbf{D} - \mathbf{A}$$

What makes these "graph transforms"?

- Shift invariance: same filter at every sample
- Graph-based shift invariance – Operator is the same, local variations captured by $\mathbf{A}$ or $\mathbf{L}$.

$$\mathbf{H} = \mathbf{L} = \mathbf{D} - \mathbf{A}$$

- This can be generalized:

$$\mathbf{H} = \sum_{k=0}^{L-1} \alpha_k \mathbf{L}^k \quad \text{or} \quad \mathbf{H} = \sum_{k=0}^{L-1} \alpha_k \mathbf{A}^k$$

- Or alternatively, because based on Graph Fourier Transform

Summary

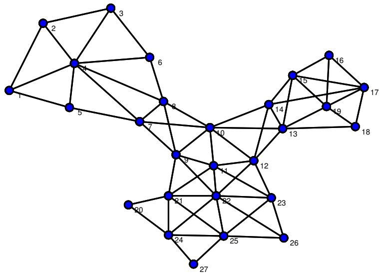
- Localized linear operations on graphs using polynomials of **A** or **L**.
- Frequency interpretation is possible for eigenvectors of **A** or **L**.
- A great deal depends on the topology of the graph

- In what follows we consider mostly undirected graphs without self loops and use **L**.
  [Shuman, Narang, Frossard, Ortega, Vandergheysnt, SPM'2013]
- Other approaches are possible based on **A**
  [Sandryhaila and Moura 2013]

Research Goals

- Extend signal processing methods to arbitrary graphs
  - Downsampling, graph-frequency localization, multiresolution, wavelets, interpolation
- Outcomes
  - Work with massive graph-datasets: localized "frequency" analysis
  - Novel insights about traditional applications (image/video processing)
  - New applications
- This talk
  - Graph Signal Processing
    - Graph Filterbank design
  - Applications
    - Edge Aware Image Filtering
    - Depth image coding
    - Wireless network optimization
    - Recommendation System Example

Graphs 101



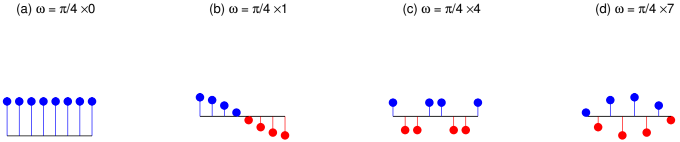
- Graph $G = (\mathcal{V}, E, w)$.
- Adjacency matrix $\mathbf{A}$
- Degree matrix $\mathbf{D} = diag\{d_i\}$
- Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$.
- Normalized Laplacian matrix $\mathcal{L} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$
- Graph Signal $\mathbf{f} = \{f(1), f(2), ..., f(N)\}$

- Assumptions:
  1. Undirected graphs without self loops.
  2. Scalar sample values

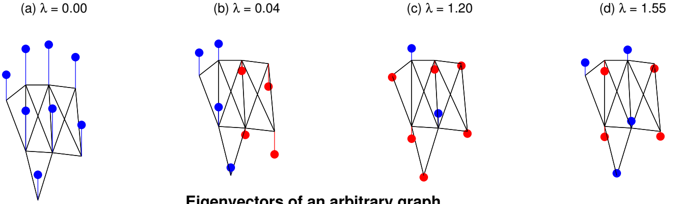
Spectrum of Graphs

- Graph Laplacian Matrix $\mathbf{L} = \mathbf{D} - \mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}'$
- Eigen-vectors of $\mathbf{L}$: $\mathbf{U} = \{\mathbf{u}_k\}_{k=1:N}$
- Eigen-values of $\mathbf{L}$: $diag\{\mathbf{\Lambda}\} = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_N$
- **Eigen-pair system $\{(\lambda_k, \mathbf{u}_k)\}$ provides Fourier-like interpretation — Graph Fourier Transform (GFT)**

Graph Frequencies

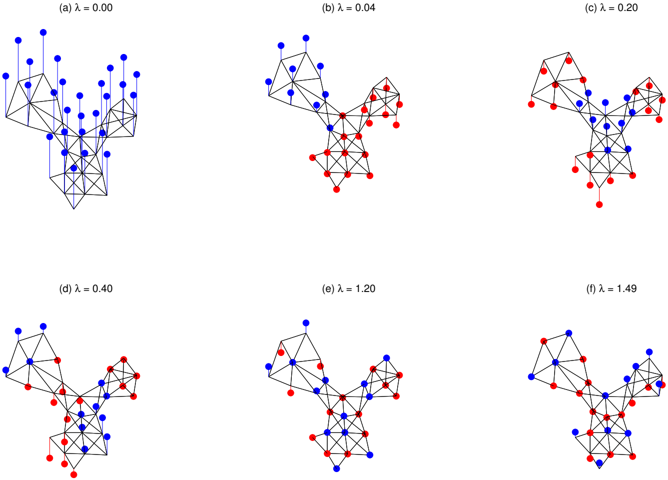


DCT basis for regular signals

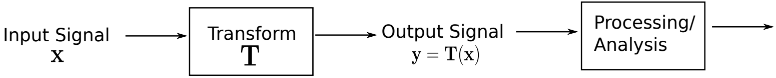


Eigenvectors of an arbitrary graph

Eigenvectors of graph Laplacian



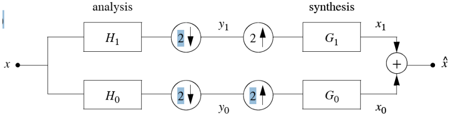
Graph Transforms



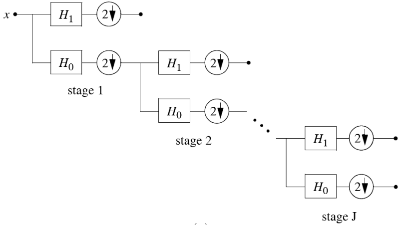
- Desirable properties
  - Invertible
  - Critically sampled
  - Orthogonal
  - Localized in graph (space) and graph spectrum (frequency)
- Local Linear Transform
- Can we define Graph Wavelets?

Next Section

Discrete Wavelet Transforms in 2 slides – 1



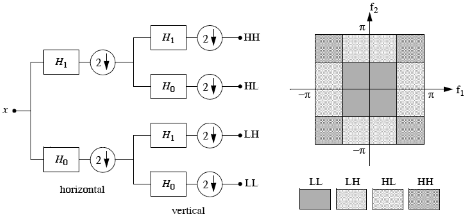
(a) 2 Channel Filterbank



(b) Tree-structured Filterbank

From Vetterli and Kovacevic, Wavelets and Subband Coding, '95

Discrete Wavelet Transforms in 2 slides – 2



(a) Separable Transform

(b) Example Image

Note: Filters have some frequency and space localization

From Vetterli and Kovacevic, [Ding'07]

Prior work – Spatial Graph Transforms

- Designed in the vertex domain of the graph. Examples:
  - Graph wavelets [Crovella'03]
  - Approaches for WSN [Wang'06], [Wagner'05] [Shen-ICASSP08]

- 1-hop averaging transform

$$y[n] = \frac{1}{d_n} \sum_{m=1}^{N} A[n,m] x[m] \quad \Rightarrow \quad \mathbf{y} = \mathbf{D}^{-1}\mathbf{A}\mathbf{x} = \mathbf{P}_{rw}\mathbf{x}$$

- 1-hop difference transform

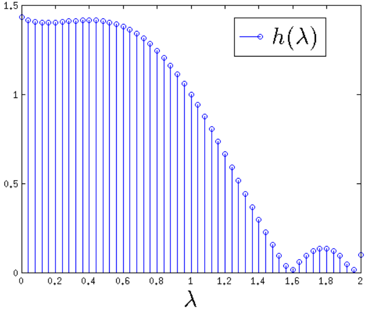
$$y[n] = \frac{1}{d_n} \sum_{m=1}^{N} A[n,m](x[n] - x[m]) \quad \Rightarrow \quad \mathbf{y} = \mathcal{L}_{\mathbf{rw}}\mathbf{x} = \mathbf{x} - \mathbf{P}_{\mathbf{rw}}\mathbf{x}$$

Prior Work – Spectral Graph Transforms

- Designed in the spectral domain of the graph. Examples:
  - Diffusion Wavelets [Coifman and Maggioni 2006]
  - Spectral Wavelets on Graphs [Hammond et al. 2011]

- Spectral Wavelet transforms [Hammond et al. 2011]:

  Design spectral kernels: $h(\lambda) : \sigma(G) \to \mathbb{R}$.



$$\mathbf{T}_h = h(\mathcal{L}) = \mathbf{U}h(\mathbf{\Lambda})\mathbf{U}^t$$

where

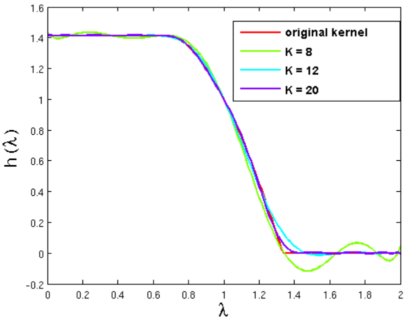
$$h(\mathbf{\Lambda}) = diag\{h(\lambda_i)\}$$

Spectral Graph Transforms Cont'd

- Output Coefficients:

$$\mathbf{w}_f = \mathbf{T}_h \mathbf{f} = \sum_{\lambda \in \sigma(G)} h(\lambda).\bar{f}(\lambda)\mathbf{u}_\lambda$$

- Polynomial kernel approximation:



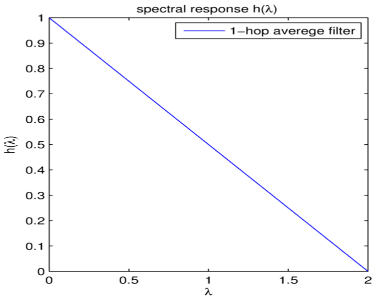
$$h(\lambda) \approx \sum_{k=0}^{K} a_k \lambda^k$$

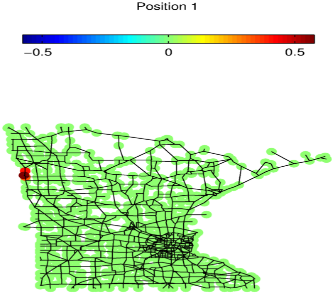
$$\mathbf{T}_h \approx \sum_{k=0}^{K} a_k \mathcal{L}^k$$

K-hop localized: no spectral decomposition required.

Vertex-Frequency Localization on Graphs

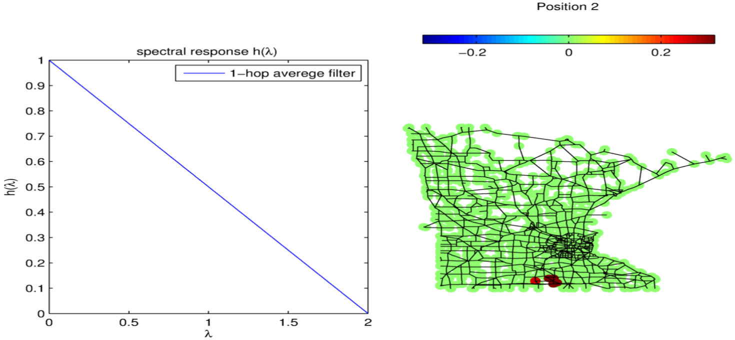
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:





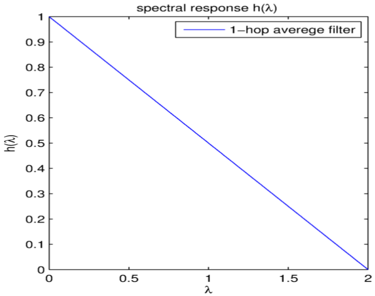
Vertex-Frequency Localization on Graphs

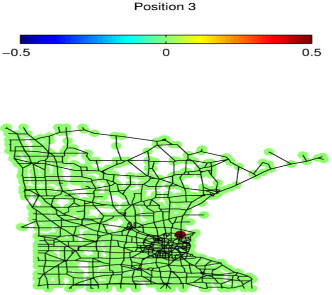
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:



Vertex-Frequency Localization on Graphs

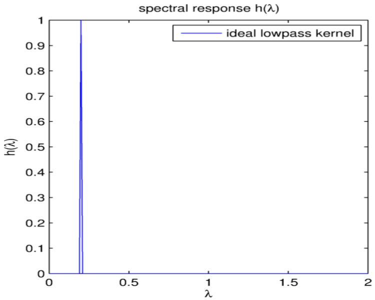
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:

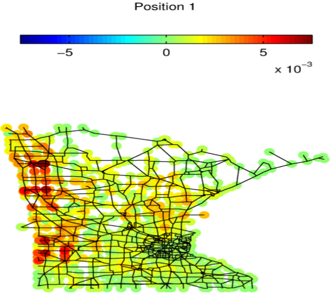




Vertex-Frequency Localization on Graphs

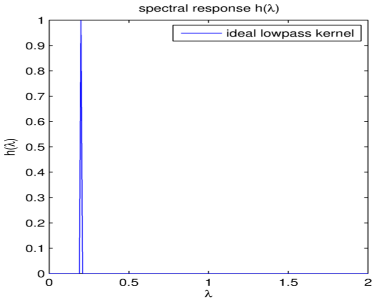
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:

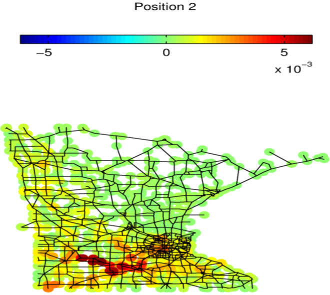




Vertex-Frequency Localization on Graphs

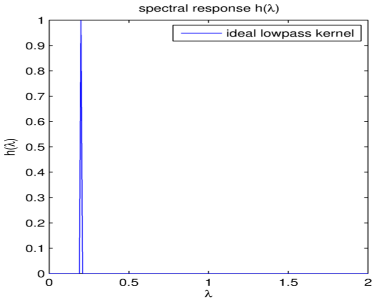
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:

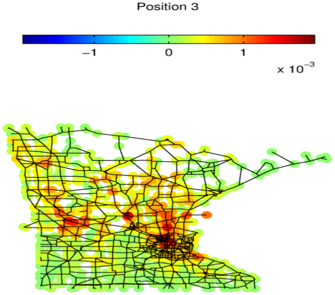




Vertex-Frequency Localization on Graphs

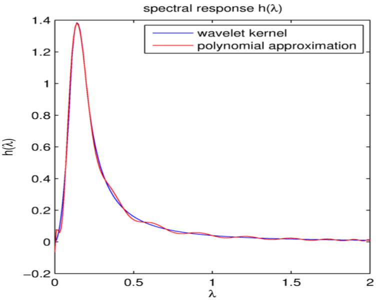
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:

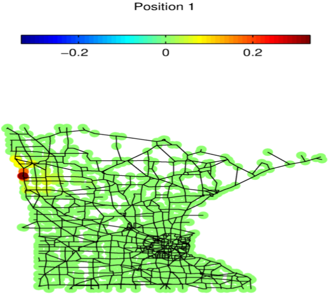




Vertex-Frequency Localization on Graphs

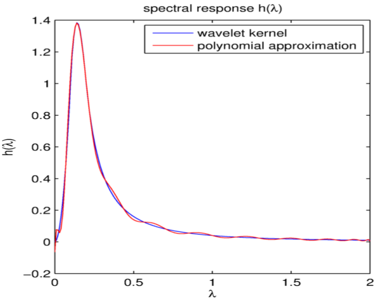
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:

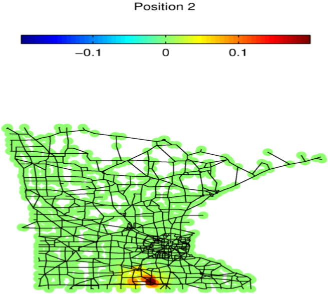




Vertex-Frequency Localization on Graphs

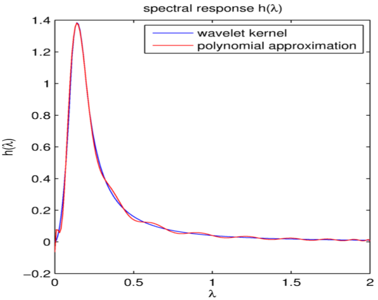
- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:

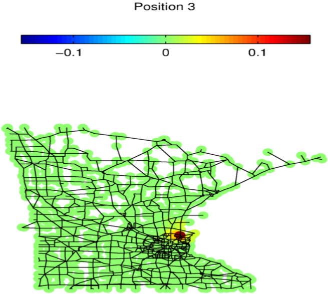




Vertex-Frequency Localization on Graphs

- Wavelet Filters: provide simultaneous localization in spatial and spectral domain:

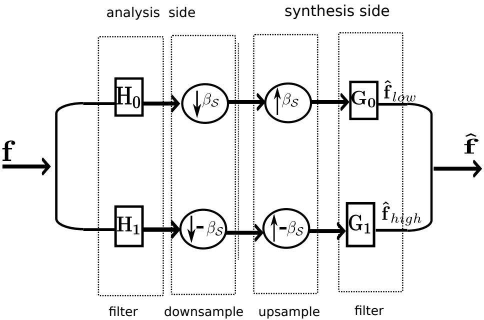




- Advantages:
  - Possible benefits of “localized” frequency analysis.
  - Fast approximate solutions to global optimization problems.

Graph Filterbank Designs

- Formulation of critically sampled graph filterbank design problem
- Design filters using spectral techniques [Hammond et al. 2009].
- Orthogonal (not compactly supported) [IEEE TSP June 2012]
- Bi-Orthogonal (compactly supported) [IEEE TSP Oct 2013]



Downsampling/Upsampling in Graphs

Downsampling-upsampling operation:

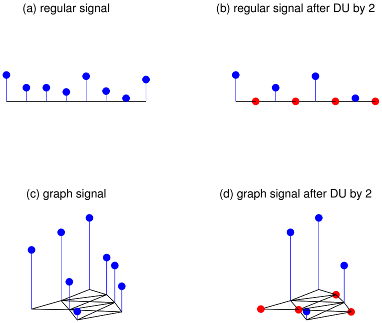
- Regular Signals:

$$f_{du}(n) = \begin{cases} f(n) & \text{if } n = 2m \\ 0 & \text{if } n = 2m+1 \end{cases}$$

- Graph signals:

$$f_{du}(n) = \begin{cases} f(n) & \text{if } n \in \mathcal{S} \\ 0 & \text{if } n \notin \mathcal{S} \end{cases}$$

for some set $\mathcal{S}$.



- For regular signals DU by 2 operation is equivalent to $F_{du}(e^{j\omega}) = 1/2(F(e^{j\omega}) + F(e^{-j\omega}))$ in the DFT domain.
- What is the DU by 2 for graph signals in GFT domain?

Downsampling in Graphs

- Downsampling function : define $\beta_H : \mathcal{V} \to \{\pm 1\}$ s.t.

$$\beta_H(n) = \begin{cases} 1 & \text{if } n \in H \\ -1 & \text{if } n \notin H \end{cases} \quad (1)$$

- Downsample-upsample (DU) operation given β_H:

$$f_{du}(n) = \frac{1}{2}[f(n) + \beta_H(n) f(n)] \quad (2)$$

Downsampling in Graphs

- Define $\mathbf{J}_\beta = \mathbf{J}_{\beta_H} = diag\{\beta_H(n)\}$.
- In vector form:

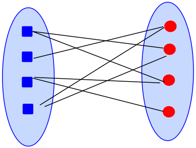
$$\begin{aligned} \mathbf{f}_{du} &= \frac{1}{2}(\mathbf{f} + \mathbf{J}_\beta \mathbf{f}) \\ &= \frac{1}{2}(\mathbf{f} + \tilde{\mathbf{f}}) \end{aligned}$$

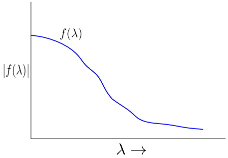
Downsampling in Graphs

- Define $\mathbf{J}_\beta = \mathbf{J}_{\beta_H} = diag\{\beta_H(n)\}$.
- In vector form:

$$\begin{aligned} \mathbf{f}_{du} &= \frac{1}{2}(\mathbf{f} + \mathbf{J}_\beta \mathbf{f}) \\ &= \frac{1}{2}(\mathbf{f} + \tilde{\mathbf{f}}) \end{aligned}$$

- Spectral Folding [4]: For a bipartite graph $\tilde{f}(\lambda) = f(2 - \lambda)$.



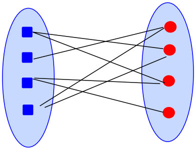


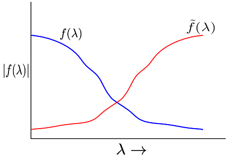
Downsampling in Graphs

- Define $\mathbf{J}_\beta = \mathbf{J}_{\beta_H} = diag\{\beta_H(n)\}$.
- In vector form:

$$\begin{aligned} \mathbf{f}_{du} &= \frac{1}{2}(\mathbf{f} + \mathbf{J}_\beta \mathbf{f}) \\ &= \frac{1}{2}(\mathbf{f} + \tilde{\mathbf{f}}) \end{aligned}$$

- Spectral Folding [4]: For a bipartite graph $\tilde{f}(\lambda) = f(2 - \lambda)$.



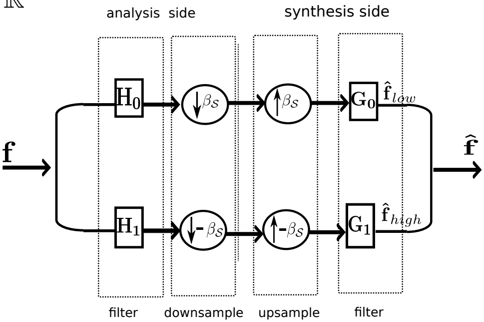


DFT aliasing vs GFT aliasing

Property	DU by 2 regular signal	DU by 2 bipartite graph signal
frequency	$\omega_k = \frac{2\pi k}{N}$; uniformly spaced in $[0\ 2\pi]$,	eigenvalues λ of $\mathcal{L}$; irregularly spaced in $[0\ 2]$
Fourier basis	$W_N^k = exp\{j\omega_k n\}$; complex	eigenvectors $\mathbf{u}_\lambda$ of $\mathcal{L}$; real
frequency folding	$F_{du}(e^{j\omega}) = 1/2(F(e^{j\omega}) + F(e^{-j\omega}))$	$\bar{f}_{du}(\lambda) = 1/2(\bar{f}(\lambda) + \bar{f}(2-\lambda))$

Graph filterbanks

- Filters designed in spectral domain (as [Hammond et al, 2009])
- Analysis:
  - $h_i(\lambda) : \mathbb{R} \to \mathbb{R}$ for $i = 0, 1$
  - $\mathbf{H}_i = h_i(\mathcal{L}) = \mathbf{U} h_i(\mathbf{\Lambda}) \mathbf{U}^t$
- Synthesis:
  - $g_i(\lambda) : \mathbb{R} \to \mathbb{R}$
  - $\mathbf{G}_i = g_i(\mathcal{L})$



Graph filterbanks

- Downsampling functions $\beta_H = \beta$ and $\beta_L = -\beta$ in two channels. $\Rightarrow$ nodes in H (or L) store the output of $\mathbf{H}_1$ (or $\mathbf{H}_0$) $\Rightarrow$ *critically sampled output.*

- Equivalent transform $\hat{\mathbf{f}} = \mathbf{T}_{eq}\mathbf{f}$, s.t.,

$$\begin{aligned} \mathbf{T}_{eq} &= \frac{1}{2}\mathbf{G}_1(\mathbf{I}+\mathbf{J}_\beta)\mathbf{H}_1 + \frac{1}{2}\mathbf{G}_0(\mathbf{I}-\mathbf{J}_\beta)\mathbf{H}_0 \\ &= \frac{1}{2}\underbrace{(\mathbf{G}_1\mathbf{H}_1 + \mathbf{G}_0\mathbf{H}_0)}_{A} + \frac{1}{2}\underbrace{(\mathbf{G}_1\mathbf{J}_\beta\mathbf{H}_1 - \mathbf{G}_0\mathbf{J}_\beta\mathbf{H}_0)}_{B} \end{aligned} \quad (3)$$

- B term is due to downsampling. For *perfect reconstruction* $A = c\mathbf{I}$ and $B = 0$.

Graph filterbanks

- Downsampling functions $\beta_H = \beta$ and $\beta_L = -\beta$ in two channels. $\Rightarrow$ nodes in H (or L) store the output of $\mathbf{H}_1$ (or $\mathbf{H}_0$) $\Rightarrow$ *critically sampled output.*

- Equivalent transform $\hat{\mathbf{f}} = \mathbf{T}_{eq}\mathbf{f}$, s.t.,

$$\begin{aligned} \mathbf{T}_{eq} &= \frac{1}{2}\mathbf{G}_1(\mathbf{I} + \mathbf{J}_\beta)\mathbf{H}_1 + \frac{1}{2}\mathbf{G}_0(\mathbf{I} - \mathbf{J}_\beta)\mathbf{H}_0 \\ &= \frac{1}{2}\underbrace{(\mathbf{G}_1\mathbf{H}_1 + \mathbf{G}_0\mathbf{H}_0)}_{A} + \frac{1}{2}\underbrace{(\mathbf{G}_1\mathbf{J}_\beta\mathbf{H}_1 - \mathbf{G}_0\mathbf{J}_\beta\mathbf{H}_0)}_{B} \end{aligned} \tag{3}$$

- B term is due to downsampling. For *perfect reconstruction* $A = c\mathbf{I}$ and $B = 0$.
- Since we use spectral filtering: choosing $\mathbf{H}_i$ is equivalent to choosing $h_i(\lambda)$

Wavelet filterbanks on bipartite graphs

- Aliasing Cancellation $\Rightarrow$ $\mathbf{B} = 0$ if for all $\lambda \in \sigma(G)$:

$$B(\lambda) \;=\; g_1(\lambda)h_1(2-\lambda) - g_0(\lambda)h_0(2-\lambda) = 0$$

Wavelet filterbanks on bipartite graphs

- Aliasing Cancellation $\Rightarrow \mathbf{B} = 0$ if for all $\lambda \in \sigma(G)$:

$$B(\lambda) = g_1(\lambda)h_1(2-\lambda) - g_0(\lambda)h_0(2-\lambda) = 0$$

- Perfect Reconstruction $\Rightarrow \mathbf{A} = c\mathbf{I}$ if for all $\lambda \in \sigma(G)$:

$$A(\lambda) = g_1(\lambda)h_1(\lambda) + g_0(\lambda)h_0(\lambda) = c$$

Graph-QMF design –1

- Solution analogous to Quadrature Mirror Filters (QMF), choose:

$$
\begin{aligned}
h_1(\lambda) &= h_0(2-\lambda)\\
g_0(\lambda) &= h_0(\lambda)\\
g_1(\lambda) &= h_1(\lambda)
\end{aligned}
$$

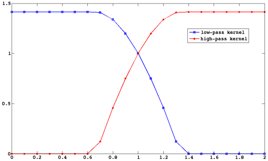
Graph-QMF design –1

- Solution analogous to Quadrature Mirror Filters (QMF), choose:

$$\begin{aligned} h_1(\lambda) &= h_0(2-\lambda) \\ g_0(\lambda) &= h_0(\lambda) \\ g_1(\lambda) &= h_1(\lambda) \end{aligned}$$

- Design $h_0(\lambda)$ s.t. for all λ

$$h_0^2(\lambda) + h_0^2(2-\lambda) = c$$



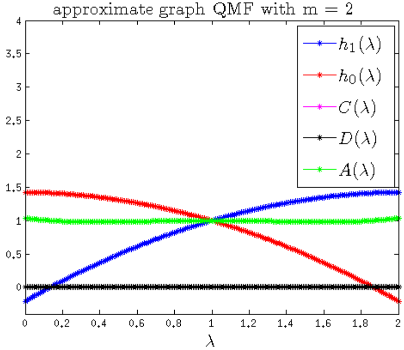
- no exact polynomial solutions, good polynomial approximations

Graph-QMF design –2

- Polynomial kernel approximation:
  - Approximate Meyer kernels as *m* degree polynomial.
  - trade off between accuracy and complexity .

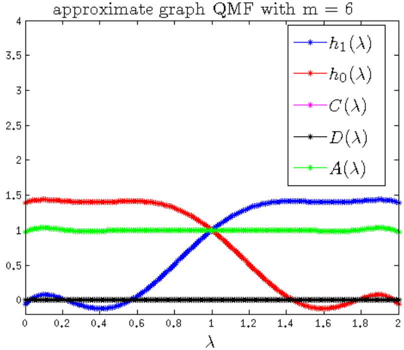
Graph-QMF design –2

- Polynomial kernel approximation:
  - Approximate Meyer kernels as *m* degree polynomial.
  - trade off between accuracy and complexity .



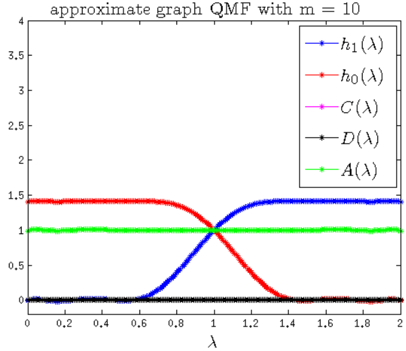
Graph-QMF design –2

- Polynomial kernel approximation:
  - Approximate Meyer kernels as *m* degree polynomial.
  - trade off between accuracy and complexity .



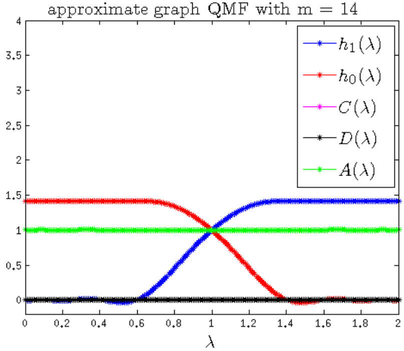
Graph-QMF design –2

- Polynomial kernel approximation:
  - Approximate Meyer kernels as *m* degree polynomial.
  - trade off between accuracy and complexity .



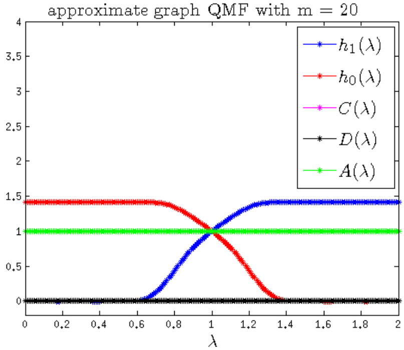
Graph-QMF design –2

- Polynomial kernel approximation:
  - Approximate Meyer kernels as *m* degree polynomial.
  - trade off between accuracy and complexity .



Graph-QMF design –2

- Polynomial kernel approximation:
  - Approximate Meyer kernels as *m* degree polynomial.
  - trade off between accuracy and complexity .



GraphBior design –1

- Analogous to CDF wavelet Filters [Narang and Ortega, IEEE TSP, 2013]

GraphBior design –1

- Analogous to CDF wavelet Filters [Narang and Ortega, IEEE TSP, 2013]
- Choose kernels, s.t.,

$$\begin{aligned} h_0(\lambda) &= g_1(2-\lambda) \\ g_0(\lambda) &= h_1(2-\lambda), \end{aligned}$$

for aliasing cancellation ($\mathbf{B} = 0$).

GraphBior design –1

- Analogous to CDF wavelet Filters [Narang and Ortega, IEEE TSP, 2013]
- Choose kernels, s.t.,

$$\begin{aligned} h_0(\lambda) &= g_1(2-\lambda) \\ g_0(\lambda) &= h_1(2-\lambda), \end{aligned}$$

  for aliasing cancellation ($\mathbf{B} = 0$).
- The PR condition ($\mathbf{A} = 0$) becomes:

$$\underbrace{h_1(\lambda)g_1(\lambda)}_{p(\lambda)} + \underbrace{h_1(2-\lambda)g_1(2-\lambda)}_{p(2-\lambda)} = c$$

GraphBior design –1

- Analogous to CDF wavelet Filters [Narang and Ortega, IEEE TSP, 2013]
- Choose kernels, s.t.,

$$\begin{aligned} h_0(\lambda) &= g_1(2-\lambda) \\ g_0(\lambda) &= h_1(2-\lambda), \end{aligned}$$

  for aliasing cancellation ($\mathbf{B} = 0$).
- The PR condition ($\mathbf{A} = 0$) becomes:

$$\underbrace{h_1(\lambda)g_1(\lambda)}_{p(\lambda)} + \underbrace{h_1(2-\lambda)g_1(2-\lambda)}_{p(2-\lambda)} = c$$

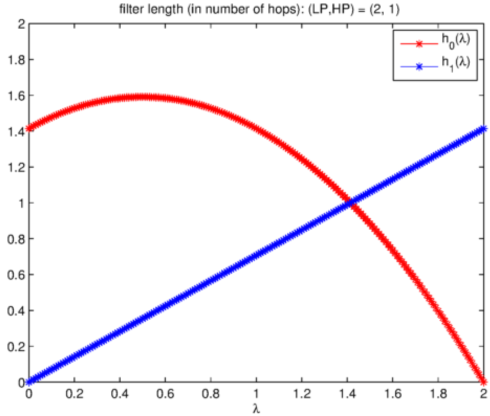
- Design $p(\lambda)$ as a "maximally flat" polynomial and factorize into $h_1(\lambda)$, $g_1(\lambda)$ terms. Exact reconstruction with polynomial filter (compact support).

GraphBior design –3

- Trade-off between spatial and spectral localization:
  - All solutions satisfy perfect reconstruction.
  - Spectral localization increases with longer filters.

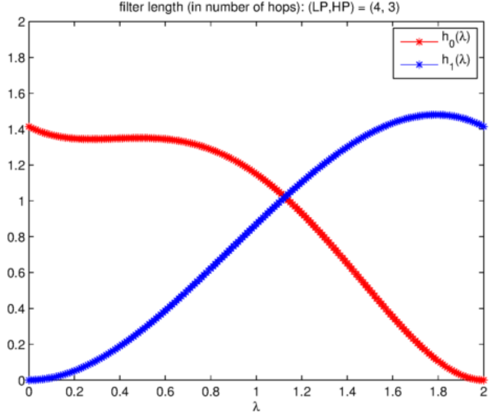
GraphBior design –3

- Trade-off between spatial and spectral localization:
  - All solutions satisfy perfect reconstruction.
  - Spectral localization increases with longer filters.



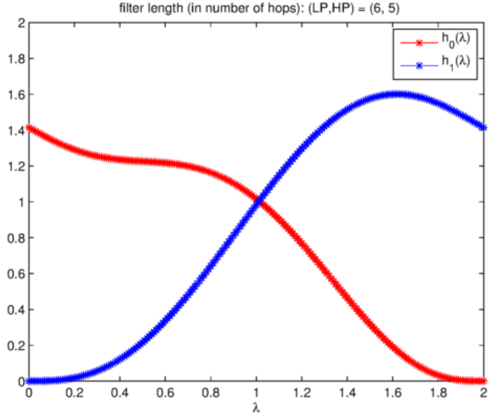
GraphBior design –3

- Trade-off between spatial and spectral localization:
  - All solutions satisfy perfect reconstruction.
  - Spectral localization increases with longer filters.



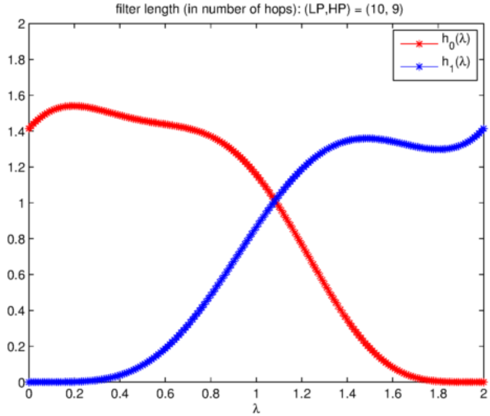
GraphBior design –3

- Trade-off between spatial and spectral localization:
  - All solutions satisfy perfect reconstruction.
  - Spectral localization increases with longer filters.



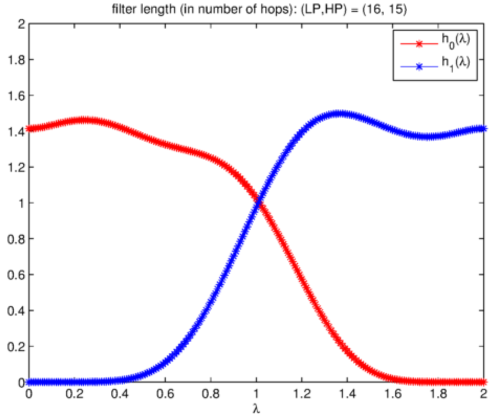
GraphBior design –3

- Trade-off between spatial and spectral localization:
  - All solutions satisfy perfect reconstruction.
  - Spectral localization increases with longer filters.



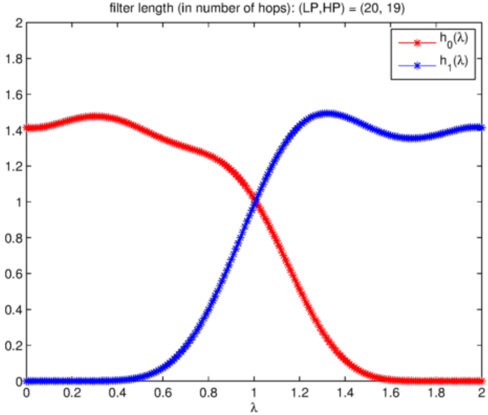
GraphBior design –3

- Trade-off between spatial and spectral localization:
  - All solutions satisfy perfect reconstruction.
  - Spectral localization increases with longer filters.



GraphBior design –3

- Trade-off between spatial and spectral localization:
  - All solutions satisfy perfect reconstruction.
  - Spectral localization increases with longer filters.



Bipartite Subgraph Decomposition

- But not all graphs are bipartite...

Bipartite Subgraph Decomposition

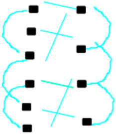
- But not all graphs are bipartite...
- Solution: "Iteratively" decompose non-bipartite graph G into K bipartite subgraphs:
  - each subgraph covers the same vertex set.
  - each edge in G belongs to exactly one bipartite graph.

Bipartite Subgraph Decomposition

- But not all graphs are bipartite...
- Solution: “Iteratively” decompose non-bipartite graph G into K bipartite subgraphs:
  - each subgraph covers the same vertex set.
  - each edge in G belongs to exactly one bipartite graph.
- apply wavelet filterbanks in K stages (dimensions).
- in the k^{th} stage restrict filtering downsampling operations on k^{th} bipartite graph.

Bipartite Subgraph Decomposition

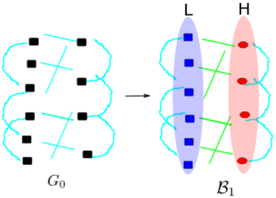
- Example of a 2-dimensional ($K = 2$) decomposition:



G_0

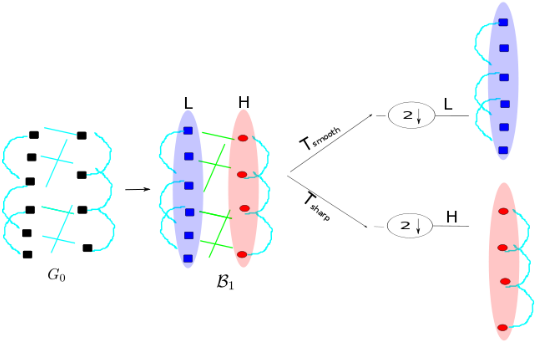
Bipartite Subgraph Decomposition

- Example of a 2-dimensional ($K = 2$) decomposition:



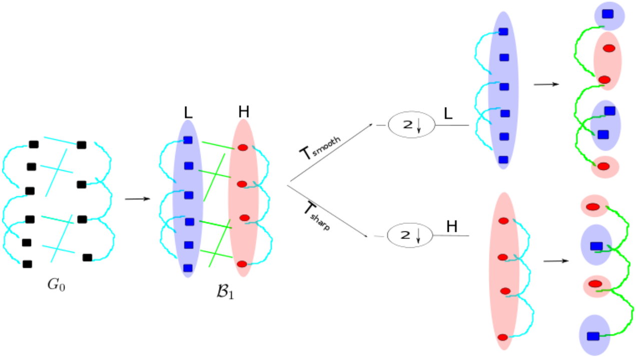
Bipartite Subgraph Decomposition

- Example of a 2-dimensional ($K = 2$) decomposition:



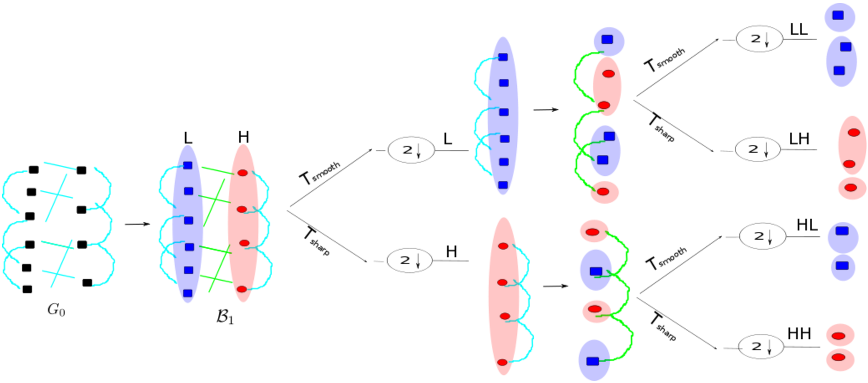
Bipartite Subgraph Decomposition

- Example of a 2-dimensional ($K = 2$) decomposition:



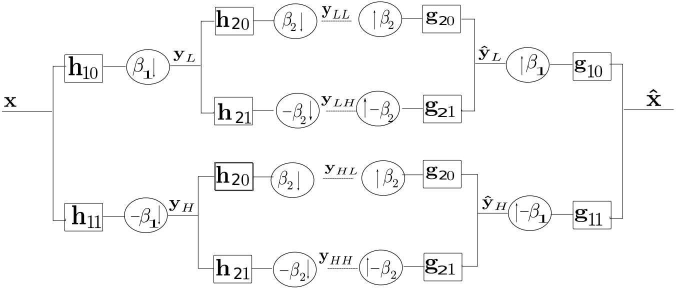
Bipartite Subgraph Decomposition

- Example of a 2-dimensional ($K = 2$) decomposition:



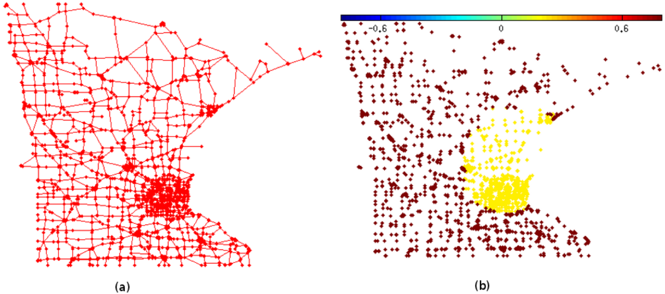
"Multi-dimensional" Filterbanks on graphs

Two-dimensional two-channel filterbank on graphs:



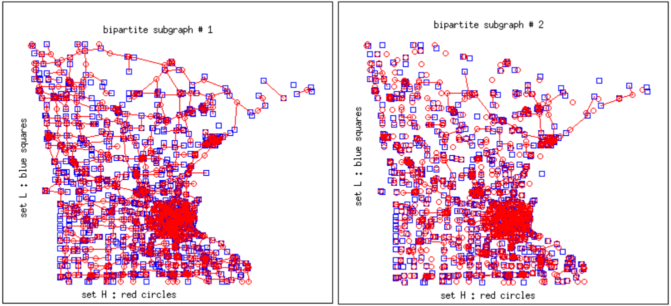
- **Advantages:**
  - Perfect reconstruction and orthogonal for *any* graph and *any* bpt decomposition.
  - defined metrics to find "good" bipartite decompositions.

Example



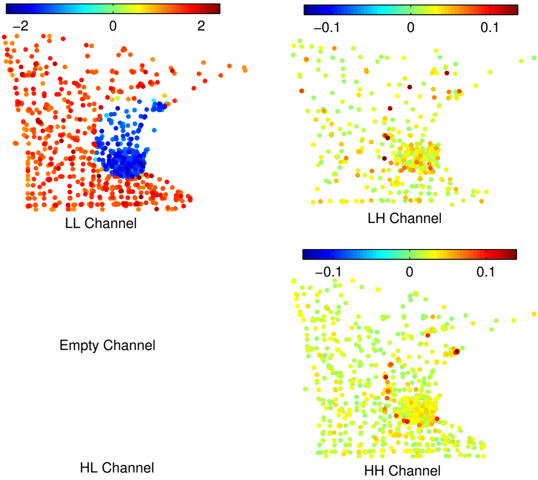
Minnesota traffic graph and graph signal

Example



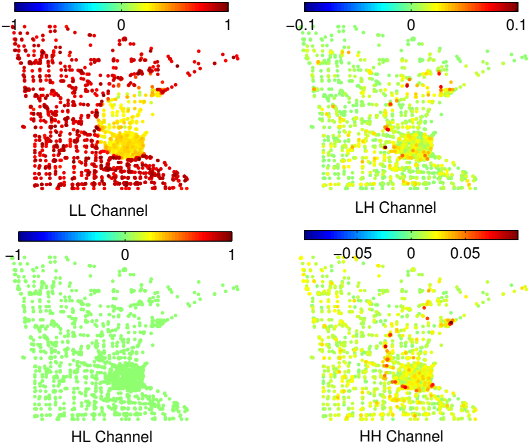
Bipartite decomposition

Example



Output coefficients of the proposed filterbanks with parameter $m = 24$.

Example

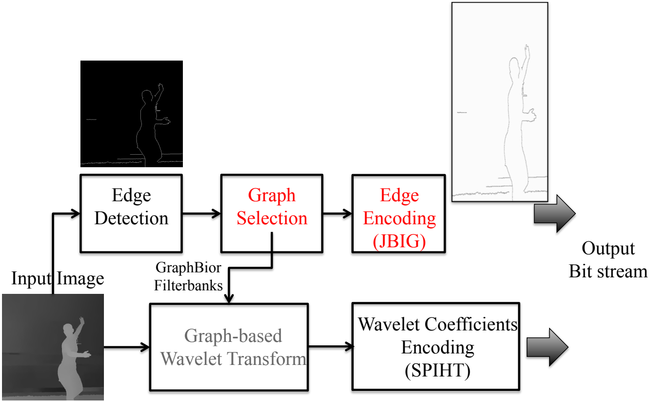


Reconstructed graph-signals for each channel.

Next Section

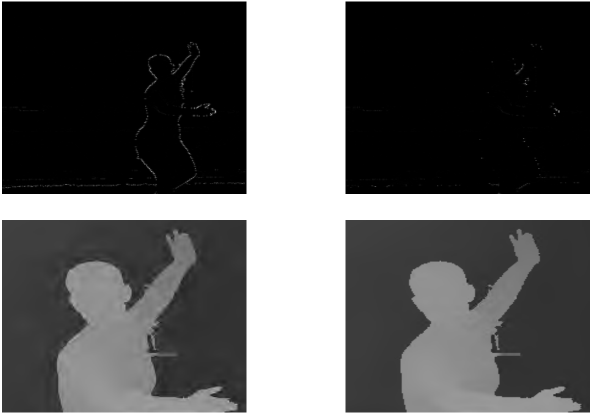
Depth Image Coding [Narang, Chao and Ortega, 2013]

- Block Diagram



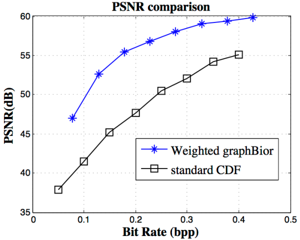
Depth Image Coding [Narang, Chao and Ortega, 2013]

CDF 9/7 | Graph 9/7



Depth Image Coding [Narang, Chao and Ortega, 2013]

- Edge detection: Prewitt
- Laplacian Normalization: Random Walk Laplacian
- Filterbanks: GraphBior 4/3 and CDF 9/7
- Unreliable Link Weight: 0.01
- Transform level: 5
- Encoder: SPIHT



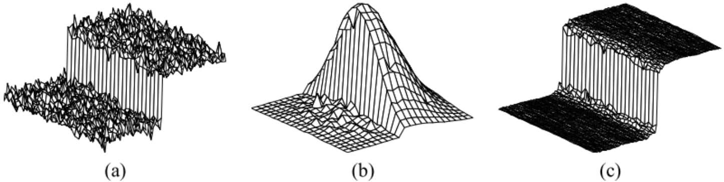
Bilateral Filtering (BF) [Tomasi and Manduchi, '98]

Weighted average of nearby similar pixels

$$\mathbf{x}_{out}[j] = \sum_i \frac{w_{ij}}{\sum_i w_{ij}} \mathbf{x}_{in}[i] \quad (4)$$

with weights given by

$$w_{ij} = \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_s^2}\right) . \exp\left(-\frac{(\mathbf{x}_{in}[i] - \mathbf{x}_{in}[j])^2}{2\sigma_x^2}\right) \quad (5)$$

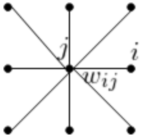


(a) Noisy data (b) Similarity weights (c) Filtered output (From Tomasi and Manduchi, 1998)

BF as a Graph Based Transform

Graph $G = (\mathcal{V}, E)$ with

- pixels as nodes
  $\mathcal{V} = \{1, 2, \ldots, n\}$
- edges $E = \{(i, j, w_{ij})\}$
- image $\mathbf{x}_{in}$ as graph signal



Bilateral Filter Graph

We can write bilateral filtering in (4) as

$$\mathbf{x}_{out} = \mathbf{D}^{-1}\mathbf{W}\mathbf{x}_{in} \tag{6}$$

Spectral Interpretation

Using the definition of graph Laplacian $\mathcal{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$

$$\mathbf{D}^{1/2}\mathbf{x}_{out} = (\mathbf{I} - \mathcal{L})\mathbf{D}^{1/2}\mathbf{x}_{in} \tag{7}$$

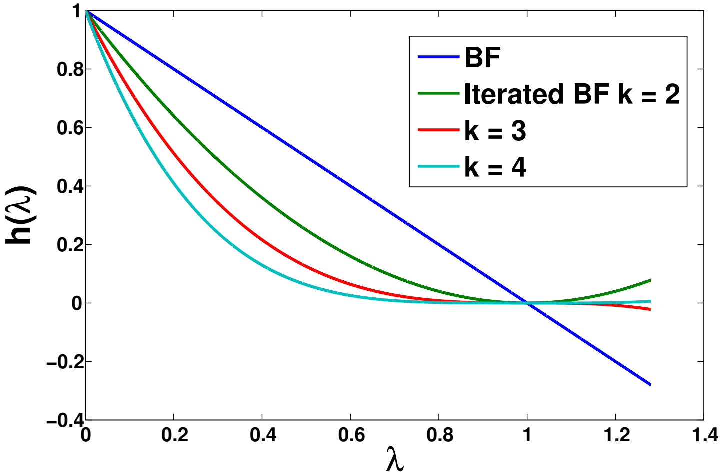
Using $\mathcal{L} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^t$ and $\hat{\mathbf{x}} = \mathbf{D}^{1/2}\mathbf{x}$

$$\hat{\mathbf{x}}_{out} = \mathbf{U}(\mathbf{I} - \boldsymbol{\Lambda})\mathbf{U}^t\hat{\mathbf{x}}_{in} \tag{8}$$

Iterated bilateral filter

$$\hat{\mathbf{x}}_{out} = \mathbf{U}(\mathbf{I} - \boldsymbol{\Lambda})^k\mathbf{U}^t\hat{\mathbf{x}}_{in} \tag{9}$$

Spectral Response of the BF



Spectral responses of the BF and iterated BF. The graph is formed using the *lena* image which has maximum eigenvalue equal to 1.28.

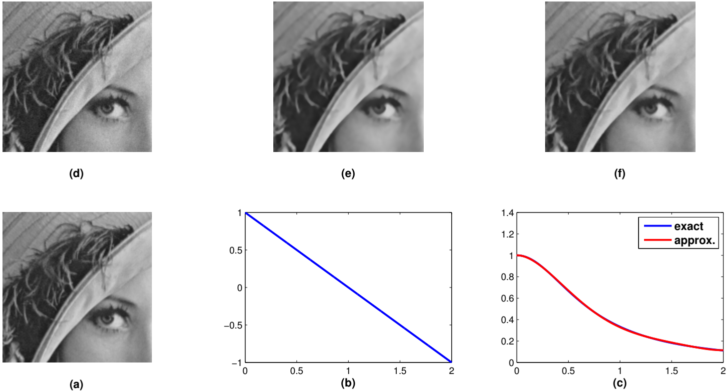
Flexible Spectral Design [Gadde, Narang and Ortega, 2013]

Key idea: use graph derived from bilateral filter

$$\mathbf{x}_{out} = \underbrace{\mathbf{U}}_{\substack{\text{Inverse} \\ \text{GFT}}} \underbrace{h(\boldsymbol{\Lambda})}_{\substack{\text{Spectral} \\ \text{response}}} \underbrace{\mathbf{U}^t \mathbf{x}_{in}}_{\text{GFT}} = h(\mathcal{L})\mathbf{x}_{in} \tag{10}$$

- Design polynomial $h(\lambda)$ to have local implementation.

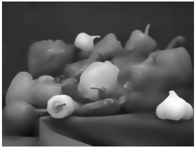
Examples: Smoothing a noisy image



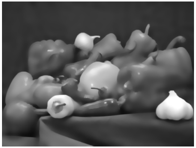
(a) Original (d) Noisy SNR = 20 dB (b) Spectral response of the BF (c) Spectral response obtained by the regularization (e) Output of the BF, SNR = 20.65 dB (f) Output of $h(\lambda)$ filter, SNR = 22.64 dB

Examples: Edge preserving coarsening

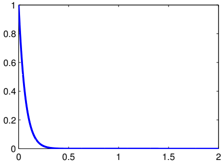
(a)



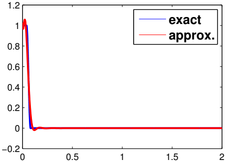
(b)



(c)



(d)



(e)

(a) Original image (b) 20 iterations of BF (d) Spectral response of the iterated BF (c) output of the proposed spectral filter (e) Corresponding Spectral response and its polynomial approximation

Graph Filtering of Cost-to-Go Functions [Levorato, Narang, Mitra, Ortega 2012]

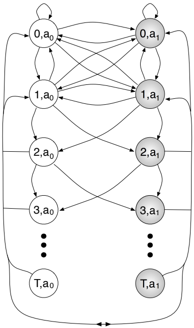
Markov decision process:

$\mathbf{S} = \{S(0), S(1), \ldots\}$ sequence of states

- $S(t) \in \mathcal{S}$ state at time t
- $\mathcal{S}$ state space

$\mathbf{A} = \{A(1), A(2), \ldots\}$ sequence of actions

- $A(t) \in \mathcal{A}_{S(t)}$ action at time t
- $\mathcal{A}$ action space.



Example of a FSM with T states and 2 actions

Graph Filtering of Cost-to-Go Functions

Graph Formulation:

- Nodes set : $\mathcal{V} = \mathcal{S} \times \mathcal{A} = \{(s, a)\}_{s \in \mathcal{S}, a \in \mathcal{A}}$.
- Graph signal: expected long term discounted cost $v(s, a)$ from state s given action a conditioned upon the policy μ:

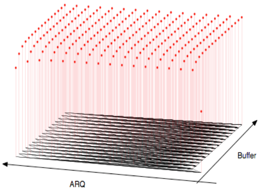
$$V_\mu(s, a) = c(s, a) + \sum_{\tau=1}^{\infty} \sum_{s_2 \in \mathcal{S}} \sum_{a_2 \in \mathcal{A}} \gamma^\tau p_\mu^\tau(s, a, s_2) \mu(s_2, a_2) c(s_2, a_2)$$

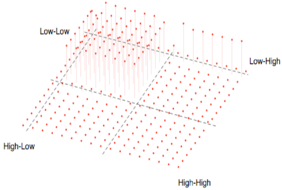
- An optimal policy exists in the set of randomized policies past independent policies $\mu(s, a) : \mathcal{S} \times \mathcal{A} \to [0,\ 1]$ maps state s to the probability that action a is selected.

Problem: Computation, compression and optimization of discounted cost function $v(s, a)$.

Graph Filtering of Cost-to-Go Functions

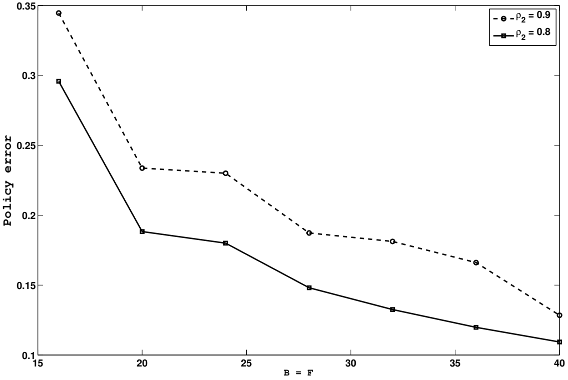
- Very large state space
- Wavelet based approach:
  - Reduce the size of the problem by downsampling and filtering.
  - Operate upon the smooth approximation of cost function on downsampled graph.
- Example: Expected cost for secondary transmitter observing state (which depends on unobserved primary transmitter)





Graph Filtering of Cost-to-Go Functions

Results [Globecom, 2012]

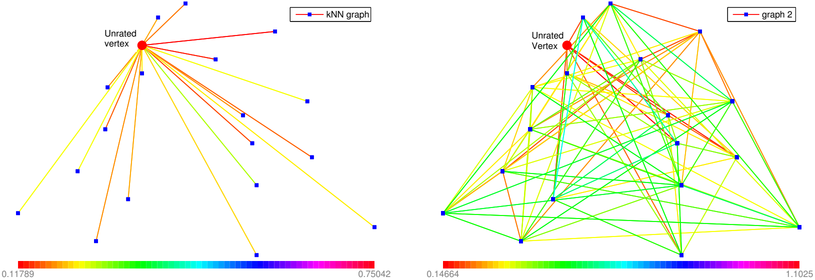


Error between policies computed on original and downsampled graph (as a function of graph size.) ρ_1 and ρ_2: transmission failure probabilities.

Graph based Prediction in Recommendation Systems [Gadde, Narang, Ortega 2013]

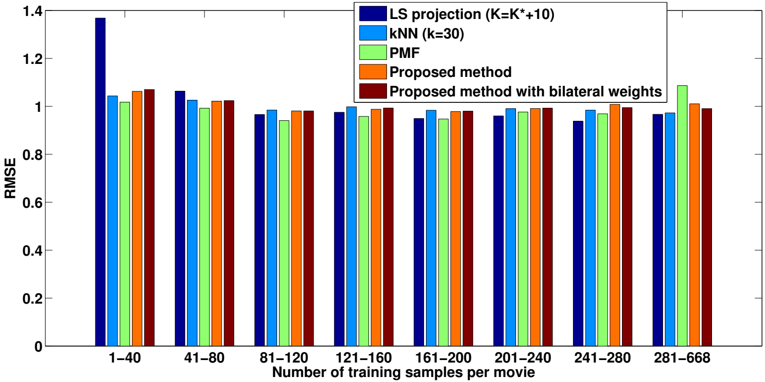
- Collaborative filtering problem: given known movie ratings for a large set of users, identify recommendations for a specific user.
- Graph representation of recommender systems:
  - movies (or users) as vertices and
  - edge-weights reflecting similarity between them.
- Interpolation based methods for rating prediction:
  - find all movies that the specific user has rated and are neighbors in weighted graph.
  - interpolate ratings of these movies to unknown movie.

Graph based Prediction in Recommendation Systems



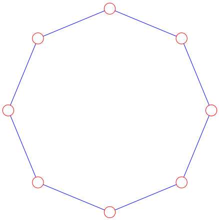
A typical instance of interpolation in MovieLens 100k dataset: (a) kNN method ($err = 2.81$ in this example). (b) Interpolation based on local sub-graph ($err = 0.78$ in this case).

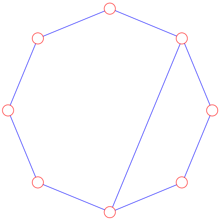
Preliminary results [ICASSP 2013]



Next Section

What makes these “graph transforms”?





- Graph-based shift invariance:

$$\mathbf{H} = \sum_{k=0}^{L-1} \alpha_k \mathbf{L}^k \quad \text{or} \quad \mathbf{H} = \sum_{k=0}^{L-1} \alpha_k \mathbf{A}^k$$

- Graph Fourier Transform

$$\mathbf{H} = h(\mathcal{L}) = \mathbf{U} h(\boldsymbol{\Lambda}) \mathbf{U}$$

Conclusions

- Extending signal processing methods to arbitrary graphs: Downsampling, Space-frequency, Multiresolution, Wavelets
- Many open questions: very diverse types of graphs, results may apply to special classes only
- Outcomes
  - Work with massive graph-datasets: potential benefits of localized “frequency” analysis
  - Novel insights about traditional applications (image/video processing)
- To get started: [Shuman, Narang, Frossard, Ortega, Vandergheysnt, SPM'2013]
- GlobalSIP Symposium on Graph Signal Processing

References I

S.K. Narang and A. Ortega.
Lifting based wavelet transforms on graphs.
In *Proc. of Asia Pacific Signal and Information Processing Association (APSIPA)*, October 2009.

S. Narang, G. Shen, and A. Ortega.
Unidirectional graph-based wavelet transforms for efficient data gathering in sensor networks.
In *In Proc. of ICASSP'10*.

S. Narang and A. Ortega.
Downsampling Graphs using Spectral Theory
In *In Proc. of ICASSP'11*.

G. Shen and A. Ortega.
Transform-based Distributed Data Gathering.
IEEE Transactions on Signal Processing.

G. Shen, S. Pattem, and A. Ortega.
Energy-efficient graph-based wavelets for distributed coding in wireless sensor networks.
In *Proc. of ICASSP'09*, April 2009.

G. Shen, S. Narang, and A. Ortega.
Adaptive distributed transforms for irregularly sampled wireless sensor networks.
In *Proc. of ICASSP'09*, April 2009.

G. Shen and A. Ortega.
Tree-based wavelets for image coding: Orthogonalization and tree selection.
In *Proc. of PCS'09*, May 2009.

References II

I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci.
A survey on sensor networks.
IEEE Communication Magazine, 40(8):102–114, August 2002.

R. Baraniuk, A. Cohen, and R. Wagner.
Approximation and compression of scattered data by meshless multiscale decompositions.
Applied Computational Harmonic Analysis, 25(2):133–147, September 2008.

C.L. Chang and B. Girod.
Direction-adaptive discrete wavelet transform for image compression.
IEEE Transactions on Image Processing, 16(5):1289–1302, May 2007.

C. Chong and S. P. Kumar.
Sensor networks: Evolution, opportunities, and challenges.
Proceedings of the IEEE, 91(8):1247–1256, August 2003.

R.R. Coifman and M. Maggioni.
Diffusion wavelets.
Applied Computational Harmonic Analysis, 21(1):53–94, 2006.

R. Cristescu, B. Beferull-Lozaon, and M. Vetterli.
Networked Slepian-Wolf: Theory, algorithms, and scaling laws.
IEEE Transactions on Information Theory, 51(12):4057–4073, December 2005.

M. Crovella and E. Kolaczyk.
Graph wavelets for spatial traffic analysis.
In *IEEE INFOCOMM*, 2003.

References III

TinyOS-2.
Collection tree protocol.
http://www.tinyos.net/tinyos-2.x/doc/.

I. Daubechies, I. Guskov, P. Schroder, and W. Sweldens.
Wavelets on irregular point sets.
Phil. Trans. R. Soc. Lond. A, 357(1760):2397–2413, September 1999.

W. Ding, F. Wu, X. Wu, S. Li, and H. Li.
Adaptive directional lifting-based wavelet transform for image coding.
IEEE Transactions on Image Processing, 16(2):416–427, February 2007.

M. Gastpar, P. Dragotti, and M. Vetterli.
The distributed Karhunen-Loève transform.
IEEE Transactions on Information Theory, 52(12):5177–5196, December 2006.

B. Girod and S. Han.
Optimum update for motion-compensated lifting.
IEEE Signal Processing Letters, 12(2):150–153, February 2005.

V.K. Goyal.
Theoretical foundations of transform coding.
IEEE Signal Processing Magazine, 18(5):9–21, September 2001.

S. Haykin.
Adaptive Filter Theory.
Prentice Hall, 4th edition, 2004.

References IV

W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan.
Energy-efficient routing protocols for wireless microsensor networks.
In *Proc. of Hawaii Intl. Conf. on Sys. Sciences*, January 2000.

M. Jansen, G. Nason, and B. Silverman.
Scattered data smoothing by empirical Bayesian shrinkage of second generation wavelet coefficients.
In *Wavelets: Applications in Signal and Image Processing IX, Proc. of SPIE*, 2001.

D. Jungnickel.
Graphs, Networks and Algorithms.
Springer-Verlag Press, 2nd edition, 2004.

M. Maitre and M. N. Do,
"Shape-adaptive wavelet encoding of depth maps,"
In *Proc. of PCS'09*, 2009.

K. Mechitov, W. Kim, G. Agha, and T. Nagayama.
High-frequency distributed sensing for structure monitoring.
In *In Proc. First Intl. Workshop on Networked Sensing Systems (INSS)*, 2004.

Y. Morvan, P.H.N. de With, and D. Farin,
"Platelet-based coding of depth maps for the transmission of multiview images,"
2006, vol. 6055, SPIE.

S. Pattem, B. Krishnamachari, and R. Govindan.
The impact of spatial correlation on routing with compression in wireless sensor networks.
ACM Transactions on Sensor Networks, 4(4):60–66, August 2008.

References V

E. Le Pennec and S. Mallat.
Sparse geometric image representations with bandelets.
IEEE Transactions on Image Processing, 14(4):423– 438, April 2005.

J.G. Proakis, E.M. Sozer, J.A. Rice, and M. Stojanovic.
Shallow water acoustic networks.
IEEE Communications Magazine, 39(11):114–119, 2001.

P. Rickenbach and R. Wattenhofer.
Gathering correlated data in sensor networks.
In *Proceedings of the 2004 Joint Workshop on Foundations of Mobile Computing*, October 2004.

S. Lafon and A. B. Lee, "Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1393–1403, Sep. 2006.

D. Ron, I. Safro, and A. Brandt, "Relaxation-based coarsening and multiscale graph organization," *Multiscale Model. Simul.*, vol. 9, no. 1, pp. 407–423, Sep. 2011.

G. Karypis, and V. Kumar, "Multilevel k-way Partitioning Scheme for Irregular Graphs", *J. Parallel Distrib. Comput.* vol. 48(1), pp. 96-129, 1998.

A. Said and W.A. Pearlman.
A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees.
IEEE Transactions on Circuits and Systems for Video Technology, 6(3):243– 250, June 1996.

References VI

A. Sanchez, G. Shen, and A. Ortega,
"Edge-preserving depth-map coding using graph-based wavelets,"
In *Proc. of Asilomar'09*, 2009.

G. Shen and A. Ortega.
Optimized distributed 2D transforms for irregularly sampled sensor network grids using wavelet lifting.
In *Proc. of ICASSP'08*, April 2008.

G. Shen and A. Ortega.
Joint routing and 2D transform optimization for irregular sensor network grids using wavelet lifting.
In *IPSN '08*, April 2008.

G. Shen and A. Ortega.
Compact image representation using wavelet lifting along arbitrary trees.
In *Proc. of ICIP'08*, October 2008.

G. Shen, S. Pattem, and A. Ortega.
Energy-efficient graph-based wavelets for distributed coding in wireless sensor networks.
In *Proc. of ICASSP'09*, April 2009.

G. Shen, S. Narang, and A. Ortega.
Adaptive distributed transforms for irregularly sampled wireless sensor networks.
In *Proc. of ICASSP'09*, April 2009.

G. Shen and A. Ortega.
Tree-based wavelets for image coding: Orthogonalization and tree selection.
In *Proc. of PCS'09*, May 2009.

References VII

G. Shen, W.-S. Kim, A. Ortega, J. Lee and H.C. Wey.
Edge-aware Intra Prediction for Depth Map Coding.
Submitted to *Proc. of ICIP'10*.

G. Shen and A. Ortega.
Transform-based Distributed Data Gathering.
To Appear in *IEEE Transactions on Signal Processing*.

G. Strang.
Linear Algebra and its Applications.
Thomson Learning, 3rd edition, 1988.

W. Sweldens.
The lifting scheme: A construction of second generation wavelets.
Tech. report 1995:6, Industrial Mathematics Initiative, Department of Mathematics, University of South Carolina, 1995.

M. Tanimoto, T. Fujii, and K. Suzuki,
"View synthesis algorithm in view synthesis reference software 2.0(VSRS2.0)," ISO/IEC JTC1/SC29/WG11, Feb. 2009.

G. Valiente.
Algorithms on Trees and Graphs.
Springer, 1st edition, 2002.

V. Velisavljevic, B. Beferull-Lozano, M. Vetterli, and P.L. Dragotti.
Directionlets: Anisotropic multidirectional representation with separable filtering.
IEEE Transactions on Image Processing, 15(7):1916– 1933, July 2006.

References VIII

R. Wagner, H. Choi, R. Baraniuk, and V. Delouille.
Distributed wavelet transform for irregular sensor network grids.
In *IEEE Stat. Sig. Proc. Workshop (SSP)*, July 2005.

R. Wagner, R. Baraniuk, S. Du, D.B. Johnson, and A. Cohen.
An architecture for distributed wavelet analysis and processing in sensor networks.
In *IPSN '06*, April 2006.

A. Wang and A. Chandraksan.
Energy-efficient DSPs for wireless sensor networks.
IEEE Signal Processing Magazine, 19(4):68–78, July 2002.

Y. Zhu, K. Sundaresan, and R. Sivakumar.
Practical limits on achievable energy improvements and useable delay tolerance in correlation aware data gathering in wireless sensor networks.
In *IEEE SECON'05*, September 2005.

S.K. Narang, G. Shen and A. Ortega,
"Unidirectional Graph-based Wavelet Transforms for Efficient Data Gathering in Sensor Networks".
pp.2902-2905, ICASSP'10, Dallas, April 2010.

S.K. Narang and A. Ortega,
"Local Two-Channel Critically Sampled Filter-Banks On Graphs",
Intl. Conf. on Image Proc. (2010),

R. R. Coifman and M. Maggioni,
"Diffusion Wavelets,"
Appl. Comp. Harm. Anal., vol. 21 no. 1 (2006), pp. 53–94

References IX

A. Sandryhaila and J. Moura,
"Discrete Signal Processing on Graphs"
IEEE Transactions on Signal Processing, 2013

D. K. Hammond, P. Vandergheynst, and R. Gribonval,
"Wavelets on graphs via spectral graph theory,"
Applied and Computational Harmonic Analysis, March 2011.

D. Shuman, S. K. Narang, P. Frossard, A. Ortega, P. Vandergheynst,
"Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Data Domains"
Signal Processing Magazine, May 2013

M. Crovella and E. Kolaczyk,
"Graph wavelets for spatial traffic analysis,"
in *INFOCOM 2003*, Mar 2003, vol. 3, pp. 1848–1857.

G. Shen and A. Ortega,
"Optimized distributed 2D transforms for irregularly sampled sensor network grids using wavelet lifting,"
in *ICASSP'08*, April 2008, pp. 2513–2516.

W. Wang and K. Ramchandran,
"Random multiresolution representations for arbitrary sensor network graphs,"
in *ICASSP*, May 2006, vol. 4, pp. IV–IV.

R. Wagner, H. Choi, R. Baraniuk, and V. Delouille.
Distributed wavelet transform for irregular sensor network grids.
In *IEEE Stat. Sig. Proc. Workshop (SSP)*, July 2005.

References X

S. K. Narang and A. Ortega,
"Lifting based wavelet transforms on graphs,"
(APSIPA ASC' 09), 2009.

B. Zeng and J. Fu,
"Directional discrete cosine transforms for image coding,"
in *Proc. of ICME 2006*, 2006.

E. Le Pennec and S. Mallat,
"Sparse geometric image representations with bandelets,"
IEEE Trans. Image Proc., vol. 14, no. 4, pp. 423–438, Apr. 2005.

M. Vetterli V. Velisavljevic, B. Beferull-Lozano and P.L. Dragotti,
"Directionlets: Anisotropic multidirectional representation with separable filtering,"
IEEE Trans. Image Proc., vol. 15, no. 7, pp. 1916–1933, Jul. 2006.

P.H.N. de With Y. Morvan and D. Farin,
"Platelet-based coding of depth maps for the transmission of multiview images,"
in *In Proceedings of SPIE: Stereoscopic Displays and Applications*, 2006, vol. 6055.

M. Tanimoto, T. Fujii, and K. Suzuki,
"View synthesis algorithm in view synthesis reference software 2.0 (VSRS2.0),"
Tech. Rep. Document M16090, ISO/IEC JTC1/SC29/WG11, Feb. 2009.

S. K. Narang and A. Ortega,
"Local two-channel critically-sampled filter-banks on graphs,"
In *ICIP'10*, Sep. 2010.

References XI

S.K. Narang and Ortega A.,
"Perfect reconstruction two-channel wavelet filter-banks for graph structured data,"
IEEE trans. on Sig. Proc., vol. 60, no. 6, June 2012.

J. P-Trufero, S.K. Narang and A. Ortega,
"Distributed Transforms for Efficient Data Gathering in Arbitrary Networks,"
In *ICIP'11.*, Sep 2011.

E. M-Enrquez, F. Daz-de-Mara and A. Ortega,
"Video Encoder Based On Lifting Transforms On Graphs,"
In *ICIP'11.*, Sep 2011.

Gilbert Strang,
"The discrete cosine transform,"
SIAM Review, vol. 41, no. 1, pp. 135–147, 1999.

S.K. Narang and A. Ortega,
"Downsampling graphs using spectral theory,"
in *ICASSP '11.*, May 2011.

Fattal, R. (2009).
Edge-avoiding wavelets and their applications.
In *SIGGRAPH '09: ACM SIGGRAPH 2009 papers*, pages 1–10, New York, NY, USA. ACM.

Hsu, C.-P. (1983).
Minimum-via topological routing.
Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, 2(4):235 – 246.

References XII

Kim, B.-J. and Pearlman, W. A. (1997).
An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees (SPIHT).
In *Data Compression Conference, 1997. DCC '97. Proceedings*, pages 251 –260.

Le Pennec, E. and Mallat, S. (2005).
Sparse geometric image representations with bandelets.
Image Processing, IEEE Transactions on, 14(4):423 –438.

Martínez Enríquez, E. and Ortega, A. (2011).
Lifting transforms on graphs for video coding.
In *Data Compression Conference (DCC), 2011*, pages 73 –82.

Pesquet-Popescu, B. and Bottreau, V. (2001).
Three-dimensional lifting schemes for motion compensated video compression.
In *ICASSP '01: Proceedings of the Acoustics, Speech, and Signal Processing, 2001. on IEEE International Conference*, pages 1793–1796, Washington, DC, USA.

I. Pesenson,
"Sampling in Paley-Wiener spaces on combinatorial graphs,"
Trans. Amer. Math. Soc, vol. 360, no. 10, pp. 5603–5627, 2008.

Secker, A. and Taubman, D. (2003).
Lifting-based invertible motion adaptive transform (limat) framework for highly scalable video compression.
Image Processing, IEEE Transactions on, 12(12):1530 – 1542.

References XIII

Shapiro, J. M. (1992).
An embedded wavelet hierarchical image coder.
In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 4, pages 657 –660 vol.4.

Shen, G. and Ortega, A. (2008).
Compact image representation using wavelet lifting along arbitrary trees.
In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 2808 –2811.

Velisavljevic, V., Beferull-Lozano, B., Vetterli, M., and Dragotti, P. L. (2006).
Directionlets: anisotropic multidirectional representation with separable filtering.
Image Processing, IEEE Transactions on, 15(7):1916 –1933.

M. Levorato, S.K. Narang, U. Mitra, A. Ortega.
Reduced Dimension Policy Iteration for Wireless Network Control via Multiscale Analysis.
Globecom, 2012.

A. Gjika, M. Levorato, A. Ortega, U. Mitra.
Online learning in wireless networks via directed graph lifting transform.
Allerton, 2012.

I. Daubechies and I. Guskov and P. Schrder and W. Sweldens.
Wavelets on Irregular Point Sets.
Phil. Trans. R. Soc. Lond. A, 1991.

W. W. Zachary.
An information flow model for conflict and fission in small groups.
Journal of Anthropological Research, 33, 452-473 (1977).

References XIV

S. K. Narang, Y. H. Chao, and A. Ortega,
"Graph-wavelet filterbanks for edge-aware image processing,"
IEEE SSP Workshop, pp. 141–144, Aug. 2012.

A. Sandryhaila and J.M.F. Moura (2013).
Discrete Signal Processing on Graphs.
Signal Processing, IEEE Transactions on, 61(7):1644-1656

S.K. Narang, A. Ortgea (2013).
Compact Support Biorthogonal Wavelet Filterbanks for Arbitrary Undirected Graphs.
Signal Processing, IEEE Transactions on

V. Ekambaram, G. Fanti, B. Ayazifar, K. Ramchandran.
Critically-Sampled Perfect-Reconstruction Spline-Wavelet Filterbanks for Graph Signals.
IEEE GlobalSIP 2013