Generalized Graph Signal Processing

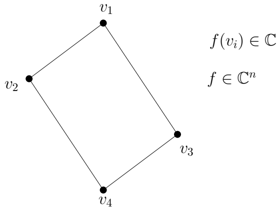
Feng Ji Wee Peng Tay

Nanyang Technological University

November 2019

Graph signal processing

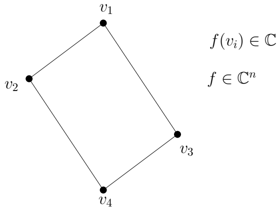
- Signal f on a graph $G = (V, E)$: $f : V \mapsto \mathbb{C}$



- Examples: sensor networks, social networks, transportation networks, ...
- Exploits the underlying graph structure (correlations between nodes) to perform signal processing and inference.

Graph signal processing

- Signal f on a graph $G = (V, E)$: $f : V \mapsto \mathbb{C}$

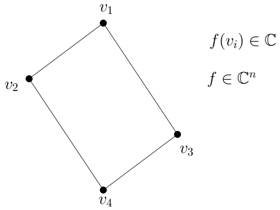


- Examples: sensor networks, social networks, transportation networks, ...
- Exploits the underlying graph structure (correlations between nodes) to perform signal processing and inference.
- Main idea: represent f using basis Φ associated with graph shift operator $A_G = \Phi\Lambda\Phi^*$ (adjacency, Laplacian, etc. that captures the local graph structure).

GFT: $\hat{f} = \Phi^* f$

Graph signal processing

- Signal f on a graph $G = (V, E)$: $f : V \mapsto \mathbb{C}$



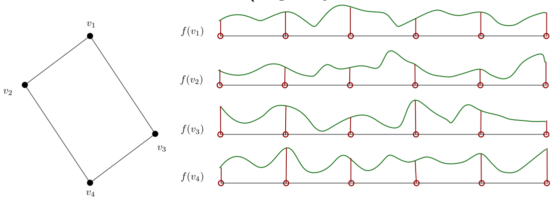
- Examples: sensor networks, social networks, transportation networks, ...
- Exploits the underlying graph structure (correlations between nodes) to perform signal processing and inference.
- Main idea: represent f using basis Φ associated with graph shift operator $A_G = \Phi \Lambda \Phi^*$ (adjacency, Laplacian, etc. that captures the local graph structure).

$$\text{GFT: } \hat{f} = \Phi^* f \overset{[1]}{=} (\langle f,\, \phi \rangle_{\mathbb{C}^n})_{\phi \in \Phi}$$

[1] B. Girault, A. Ortega, and S. S. Narayanan, "Irregularity-aware graph Fourier transforms," *IEEE Transactions on Signal Processing*, vol. 66, no. 21, pp. 5746–5761, Nov. 2018.

Time-vertex GSP

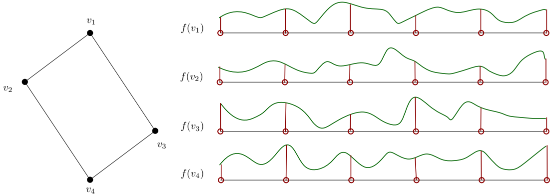
- [2]: $f(v,\cdot) \in \mathbb{C}^T$ for each $v \in V$, $T < \infty$, is a discrete time series with common time indices (e.g., synchronous uniform sampling at every vertex).



[2] F. Grassi, A. Loukas, N. Perraudin, and B. Ricaud, "A time-vertex signal processing framework: Scalable processing and meaningful representations for time-series on graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 817–829, Feb. 2018.

Time-vertex GSP

- [2]: $f(v, \cdot) \in \mathbb{C}^T$ for each $v \in V$, $T < \infty$, is a discrete time series with common time indices (e.g., synchronous uniform sampling at every vertex).



- For each $v \in V$, can apply DFT on $f(v, \cdot)$:

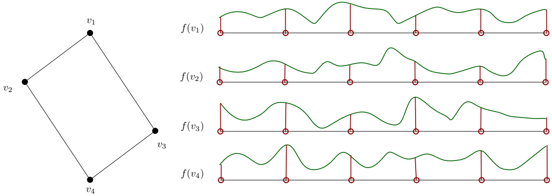
$$\mathsf{DFT}(f(v, \cdot)) = \Xi^* \operatorname{vec}(f(v, \cdot)),$$

where Ξ^* is the DFT matrix.

[2] F. Grassi, A. Loukas, N. Perraudin, and B. Ricaud, "A time-vertex signal processing framework: Scalable processing and meaningful representations for time-series on graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 817–829, Feb. 2018.

Time-vertex GSP

- [2]: $f(v, \cdot) \in \mathbb{C}^T$ for each $v \in V$, $T < \infty$, is a discrete time series with common time indices (e.g., synchronous uniform sampling at every vertex).



- For each $v \in V$, can apply DFT on $f(v, \cdot)$:

$$\mathrm{DFT}(f(v, \cdot)) = \Xi^* \operatorname{vec}(f(v, \cdot)),$$

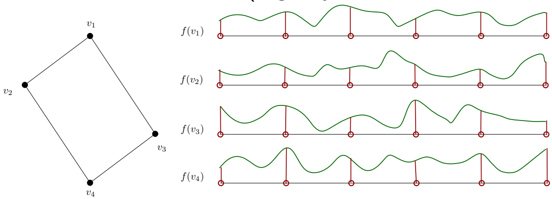
  where Ξ^* is the DFT matrix.
- Joint time-vertex Fourier transform: view $f = (f(v, t))$ as a matrix,

$$\text{TV-transform: } \hat{f} = \Phi^* f \overline{\Xi}.$$

[2] F. Grassi, A. Loukas, N. Perraudin, and B. Ricaud, "A time-vertex signal processing framework: Scalable processing and meaningful representations for time-series on graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 817–829, Feb. 2018.

Time-vertex GSP

- [2]: $f(v,\cdot) \in \mathbb{C}^T$ for each $v \in V$, $T < \infty$, is a discrete time series with common time indices (e.g., synchronous uniform sampling at every vertex).



- For each $v \in V$, can apply DFT on $f(v,\cdot)$:

$$\mathsf{DFT}(f(v,\cdot)) = \Xi^* \operatorname{vec}(f(v,\cdot)),$$

where Ξ^* is the DFT matrix.

- Joint time-vertex Fourier transform: view $f = (f(v,t))$ as a matrix,

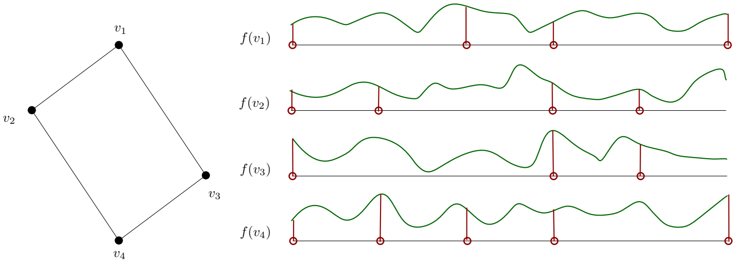
$$\text{TV-transform: } \hat{f} = \Phi^* f \overline{\Xi}.$$

- A representation in the basis $\Phi \otimes \Xi$.

[2] F. Grassi, A. Loukas, N. Perraudin, and B. Ricaud, "A time-vertex signal processing framework: Scalable processing and meaningful representations for time-series on graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 817–829, Feb. 2018.

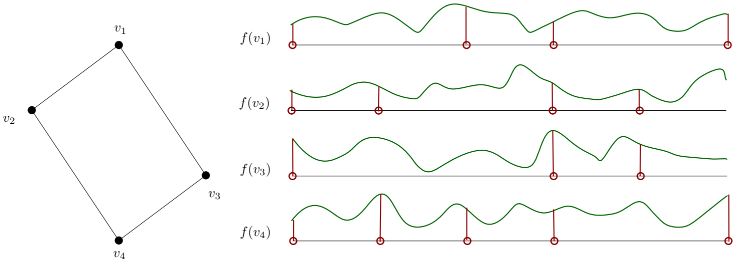
Time-vertex GSP

- Missing samples.



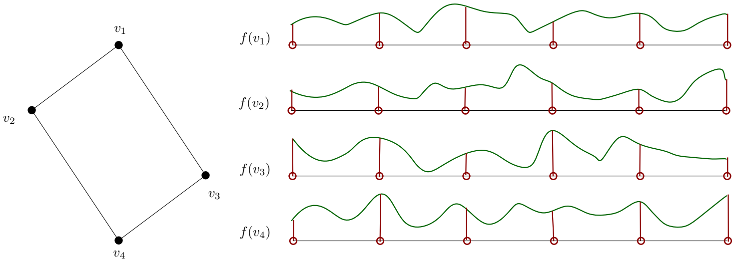
Time-vertex GSP

- Missing samples.
- If vertex domain dimension $= 2$, can reconstruct.



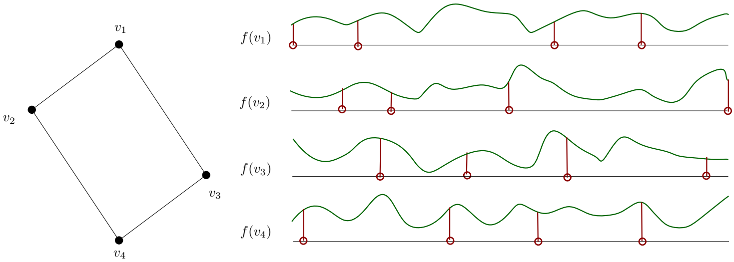
Time-vertex GSP

- Missing samples.
- If vertex domain dimension $= 2$, can reconstruct.



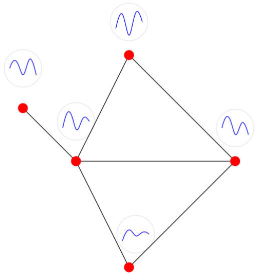
Time-vertex GSP

- Missing samples.
- If vertex domain dimension $= 2$, can reconstruct.
- However, asynchronous sampling (e.g., sensor networks) ... now impossible to reconstruct the signal.



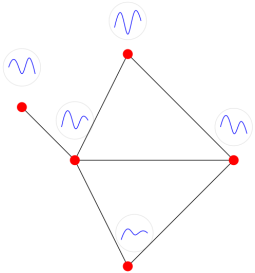
Generalized GSP

- Signal at each vertex is from an infinite dimensional separable Hilbert space $\mathcal{H} = L^2(\Omega, \mu)$.



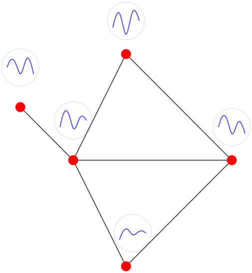
Generalized GSP

- Signal at each vertex is from an infinite dimensional separable Hilbert space $\mathcal{H} = L^2(\Omega, \mu)$.
- Maybe non-bandlimited in Ω direction (Shannon-Nyquist: impossible to represent as discrete time series).



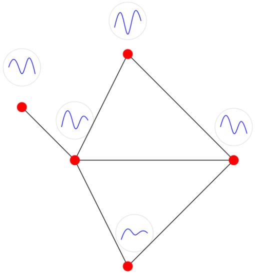
Generalized GSP

- Signal at each vertex is from an infinite dimensional separable Hilbert space $\mathcal{H} = L^2(\Omega, \mu)$.
- Maybe non-bandlimited in Ω direction (Shannon-Nyquist: impossible to represent as discrete time series).
- Non-synchronous sampling (time indices are not same for different vertices).



Generalized GSP

- Signal at each vertex is from an infinite dimensional separable Hilbert space $\mathcal{H} = L^2(\Omega, \mu)$.
- Maybe non-bandlimited in Ω direction (Shannon-Nyquist: impossible to represent as discrete time series).
- Non-synchronous sampling (time indices are not same for different vertices).
- Allows joint modeling of $f(v, x)$ over $v \in V$, $x \in \mathcal{H}$.



The rest of this talk...

1. Generalized Graph Signals and $\mathcal{F}$-Transform
2. Sampling Theorem
3. Filtering
4. Conclusion

Details in

F. Ji and W. P. Tay, "A Hilbert space theory of generalized graph signal processing," *IEEE Trans. Signal Process.*, 2019, accepted. [Online]. Available: https://arxiv.org/abs/1904.11655.

Outline

Tensor product space

- Tensor product $\mathbb{C}^n \otimes \mathcal{H} = \left\{ \sum_{i=1}^n v_i \otimes h_i \right\}$ with
  1. (a) $v_1 \otimes h + v_2 \otimes h = (v_1 + v_2) \otimes h$;
  2. (b) $v \otimes h_1 + v \otimes h_2 = v \otimes (h_1 + h_2)$;
  3. (c) $rv \otimes h = v \otimes rh$ for $r \in \mathbb{C}$.

Tensor product space

- Tensor product $\mathbb{C}^n \otimes \mathcal{H} = \left\{ \sum_{i=1}^n v_i \otimes h_i \right\}$ with
  1. (a) $v_1 \otimes h + v_2 \otimes h = (v_1 + v_2) \otimes h$;
  2. (b) $v \otimes h_1 + v \otimes h_2 = v \otimes (h_1 + h_2)$;
  3. (c) $rv \otimes h = v \otimes rh$ for $r \in \mathbb{C}$.
- Inner product $\langle \cdot, \cdot \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$ induced (linearly) by:

$$\langle v_1 \otimes h_1, v_2 \otimes h_2 \rangle_{\mathbb{C}^n \otimes \mathcal{H}} = \langle v_1, v_2 \rangle_{\mathbb{C}^n} \langle h_1, h_2 \rangle_{\mathcal{H}}.$$

$\mathbb{C}^n \otimes \mathcal{H}$ is a Hilbert space.

Tensor product space

- Tensor product $\mathbb{C}^n \otimes \mathcal{H} = \left\{ \sum_{i=1}^n v_i \otimes h_i \right\}$ with
  1. (a) $v_1 \otimes h + v_2 \otimes h = (v_1 + v_2) \otimes h$;
  2. (b) $v \otimes h_1 + v \otimes h_2 = v \otimes (h_1 + h_2)$;
  3. (c) $rv \otimes h = v \otimes rh$ for $r \in \mathbb{C}$.
- Inner product $\langle \cdot, \cdot \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$ induced (linearly) by:

$$\langle v_1 \otimes h_1, v_2 \otimes h_2 \rangle_{\mathbb{C}^n \otimes \mathcal{H}} = \langle v_1, v_2 \rangle_{\mathbb{C}^n} \langle h_1, h_2 \rangle_{\mathcal{H}}.$$

  $\mathbb{C}^n \otimes \mathcal{H}$ is a Hilbert space.
- Generalized graph signal $f : V \mapsto \mathcal{H}$. $S(G, \mathcal{H})$ - space of graph signals in $\mathcal{H}$.

Tensor product space

- Tensor product $\mathbb{C}^n \otimes \mathcal{H} = \left\{ \sum_{i=1}^n v_i \otimes h_i \right\}$ with
  1. (a) $v_1 \otimes h + v_2 \otimes h = (v_1 + v_2) \otimes h$;
  2. (b) $v \otimes h_1 + v \otimes h_2 = v \otimes (h_1 + h_2)$;
  3. (c) $rv \otimes h = v \otimes rh$ for $r \in \mathbb{C}$.
- Inner product $\langle \cdot, \cdot \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$ induced (linearly) by:

$$\langle v_1 \otimes h_1, v_2 \otimes h_2 \rangle_{\mathbb{C}^n \otimes \mathcal{H}} = \langle v_1, v_2 \rangle_{\mathbb{C}^n} \langle h_1, h_2 \rangle_{\mathcal{H}}.$$

  $\mathbb{C}^n \otimes \mathcal{H}$ is a Hilbert space.
- Generalized graph signal $f : V \mapsto \mathcal{H}$. $S(G, \mathcal{H})$ - space of graph signals in $\mathcal{H}$.

Lemma

$S(G, \mathcal{H})$ is a Hilbert space isomorphic to $\mathbb{C}^n \otimes \mathcal{H}$, $|V| = n$.

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$
- A : compact, self-adjoint operator on $\mathcal{H}$, eigenvectors Ξ of A form orthonormal basis of $\mathcal{H}$ (Hilbert-Schmidt Theorem)

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$
- A : compact, self-adjoint operator on $\mathcal{H}$, eigenvectors Ξ of A form orthonormal basis of $\mathcal{H}$ (Hilbert-Schmidt Theorem)
- $\Phi \otimes \Xi$ is a basis for $\mathbb{C}^n \otimes \mathcal{H}$.
- Joint $\mathcal{F}$-transform:

$$\mathcal{F}_f(\phi \otimes \xi) = \langle f, \phi \otimes \xi \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$$

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$
- A : compact, self-adjoint operator on $\mathcal{H}$, eigenvectors Ξ of A form orthonormal basis of $\mathcal{H}$ (Hilbert-Schmidt Theorem)
- $\Phi \otimes \Xi$ is a basis for $\mathbb{C}^n \otimes \mathcal{H}$.
- Joint $\mathcal{F}$-transform:

$$\mathcal{F}_f(\phi \otimes \xi) = \langle f, \phi \otimes \xi \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$$

$$= \left\langle \left(\langle f(v, \cdot), \xi \rangle_{\mathcal{H}} \right)_{v \in V}, \phi \right\rangle_{\mathbb{C}^n}$$

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$
- A : compact, self-adjoint operator on $\mathcal{H}$, eigenvectors Ξ of A form orthonormal basis of $\mathcal{H}$ (Hilbert-Schmidt Theorem)
- $\Phi \otimes \Xi$ is a basis for $\mathbb{C}^n \otimes \mathcal{H}$.
- Joint $\mathcal{F}$-transform:

$$\mathcal{F}_f(\phi \otimes \xi) = \langle f, \phi \otimes \xi \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$$

$$= \left\langle (\overbrace{\langle f(v, \cdot), \xi \rangle_{\mathcal{H}}}^{\substack{\text{Fourier series,} \\ \text{Fourier transform,} \\ \text{wavelet transform,} \\ \text{Chebyshev poly.,} \\ \dots}})_{v \in V}, \phi \right\rangle_{\mathbb{C}^n}$$

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$
- A : compact, self-adjoint operator on $\mathcal{H}$, eigenvectors Ξ of A form orthonormal basis of $\mathcal{H}$ (Hilbert-Schmidt Theorem)
- $\Phi \otimes \Xi$ is a basis for $\mathbb{C}^n \otimes \mathcal{H}$.
- Joint $\mathcal{F}$-transform:

$$\mathcal{F}_f(\phi \otimes \xi) = \langle f, \phi \otimes \xi \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$$

$$= \underbrace{\left\langle \Big(\overbrace{\langle f(v, \cdot), \xi \rangle_{\mathcal{H}}}^{\text{Fourier series, Fourier transform, wavelet transform, Chebyshev poly., ...}} \Big)_{v \in V}, \phi \right\rangle_{\mathbb{C}^n}}_{\text{GFT}}$$

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$
- A : compact, self-adjoint operator on $\mathcal{H}$, eigenvectors Ξ of A form orthonormal basis of $\mathcal{H}$ (Hilbert-Schmidt Theorem)
- $\Phi \otimes \Xi$ is a basis for $\mathbb{C}^n \otimes \mathcal{H}$.
- Joint $\mathcal{F}$-transform:

$$
\begin{aligned}
\mathcal{F}_f(\phi \otimes \xi) &= \langle f, \phi \otimes \xi \rangle_{\mathbb{C}^n \otimes \mathcal{H}} \\
&= \underbrace{\left\langle (\overbrace{\langle f(v,\cdot), \xi \rangle_{\mathcal{H}}}^{\substack{\text{Fourier series,} \\ \text{Fourier transform,} \\ \text{wavelet transform,} \\ \text{Chebyshev poly.,} \\ \dots}})_{v \in V}, \phi \right\rangle_{\mathbb{C}^n}}_{\text{GFT}} \\
&= \langle (\langle f(\cdot, x), \phi \rangle_{\mathbb{C}^n})_{x \in \Omega}, \xi \rangle_{\mathcal{H}}
\end{aligned}
$$

$\mathcal{F}$-transform

- A_G : self-adjoint graph shift operator of G, eigenvectors Φ form orthonormal basis of $\mathbb{C}^n$
- A : compact, self-adjoint operator on $\mathcal{H}$, eigenvectors Ξ of A form orthonormal basis of $\mathcal{H}$ (Hilbert-Schmidt Theorem)
- $\Phi \otimes \Xi$ is a basis for $\mathbb{C}^n \otimes \mathcal{H}$.
- Joint $\mathcal{F}$-transform:

$$\mathcal{F}_f(\phi \otimes \xi) = \langle f, \phi \otimes \xi \rangle_{\mathbb{C}^n \otimes \mathcal{H}}$$

$$= \underbrace{\left\langle (\overbrace{\langle f(v, \cdot), \xi \rangle_{\mathcal{H}}}^{\substack{\text{Fourier series,} \\ \text{Fourier transform,} \\ \text{wavelet transform,} \\ \text{Chebyshev poly.,} \\ \dots}})_{v \in V}, \phi \right\rangle_{\mathbb{C}^n}}_{\text{GFT}}$$

$$= \langle (\langle f(\cdot, x), \phi \rangle_{\mathbb{C}^n})_{x \in \Omega}, \xi \rangle_{\mathcal{H}}$$

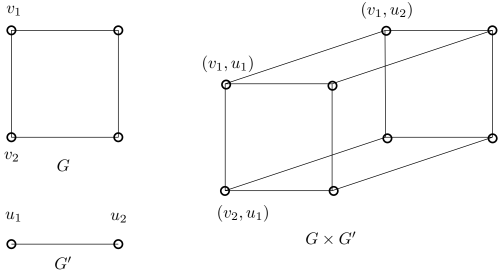
- $f = \sum_{\phi \otimes \xi} \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi$

Relation to traditional GSP

- $\mathcal{H} = \mathbb{C}$, can take $\Xi = \{1\}$ and $\mathcal{F}$-transform = GFT.

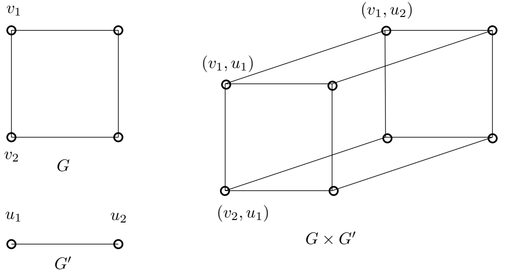
Relation to traditional GSP

- $\mathcal{H} = \mathbb{C}$, can take $\Xi = \{1\}$ and $\mathcal{F}$-transform = GFT.
- $\mathcal{H} = L^2(G')$ with discrete measure, where $G' = (V', E')$ is a finite graph.



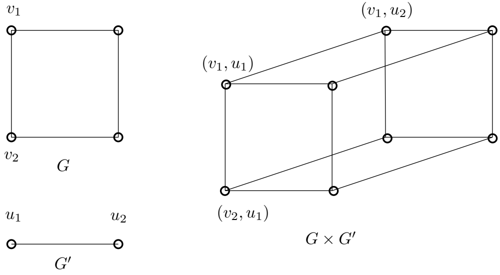
Relation to traditional GSP

- $\mathcal{H} = \mathbb{C}$, can take $\Xi = \{1\}$ and $\mathcal{F}$-transform = GFT.
- $\mathcal{H} = L^2(G')$ with discrete measure, where $G' = (V', E')$ is a finite graph.
  - $S(G, \mathcal{H})$ - signals on product graph $G \times G'$. Suitable A: $\mathcal{F}$-transform = GFT



Relation to traditional GSP

- $\mathcal{H} = \mathbb{C}$, can take $\Xi = \{1\}$ and $\mathcal{F}$-transform = GFT.
- $\mathcal{H} = L^2(G')$ with discrete measure, where $G' = (V', E')$ is a finite graph.
  - $S(G, \mathcal{H})$ - signals on product graph $G \times G'$. Suitable A: $\mathcal{F}$-transform = GFT
  - G' = path graph: $\mathcal{F}$-transform = TV-transform



Example of infinite dimensional $\mathcal{H}$

- $\mathcal{H}$ = subspace of $L^2([-\pi, \pi], \lambda_{\text{Leb.}})$ consisting of f such that $f(-\pi) = f(\pi)$.

Example of infinite dimensional $\mathcal{H}$

- $\mathcal{H}$ = subspace of $L^2([-\pi,\pi],\lambda_{\text{Leb.}})$ consisting of f such that $f(-\pi) = f(\pi)$.
- Choose

$$Af(x) = \frac{\mathrm{i}}{2}\left(\int_{-\pi}^{x} f(y)\mathrm{d}y - \int_{x}^{\pi} f(y)\mathrm{d}y\right).$$

Example of infinite dimensional $\mathcal{H}$

- $\mathcal{H}$ = subspace of $L^2([-\pi,\pi],\lambda_{\text{Leb.}})$ consisting of f such that $f(-\pi)=f(\pi)$.
- Choose

$$Af(x)=\frac{\mathrm{i}}{2}\left(\int_{-\pi}^{x} f(y)\mathrm{d}y-\int_{x}^{\pi} f(y)\mathrm{d}y\right).$$

- A is compact with eigenvectors $\Xi=\left\{\frac{\exp(\mathrm{i}mx)}{\sqrt{2\pi}} : m\in\mathbb{Z}\right\}$ (Fourier series basis):

$$f(x)=\sum_{m\in\mathbb{Z}}\frac{a_m}{\sqrt{2\pi}}e^{\mathrm{i}mx},\ a_m\in\mathbb{C} \text{ for all } m\in\mathbb{Z}.$$

Example of infinite dimensional $\mathcal{H}$

- $\mathcal{H}$ = subspace of $L^2([-\pi,\pi],\lambda_{\text{Leb.}})$ consisting of f such that $f(-\pi)=f(\pi)$.
- Choose

$$Af(x)=\frac{\mathrm{i}}{2}\left(\int_{-\pi}^{x} f(y)\mathrm{d}y-\int_{x}^{\pi} f(y)\mathrm{d}y\right).$$

- A is compact with eigenvectors $\Xi=\left\{\frac{\exp(\mathrm{i}mx)}{\sqrt{2\pi}}: m\in\mathbb{Z}\right\}$ (Fourier series basis):

$$f(x)=\sum_{m\in\mathbb{Z}}\frac{a_m}{\sqrt{2\pi}}e^{\mathrm{i}mx},\ a_m\in\mathbb{C} \text{ for all } m\in\mathbb{Z}.$$

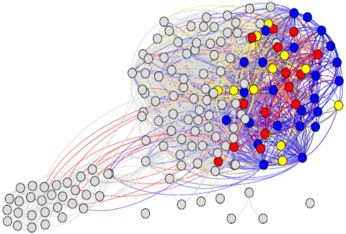
- Fredholm operator

$$Af(x)=\int_{\Omega}K(x,y)f(y)\mathrm{d}\mu(y),$$

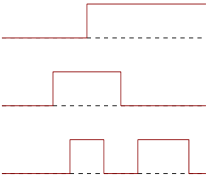
Hermitian $K\in L^2(\Omega\times\Omega)\implies A$ compact, self-adjoint.
Choose different kernels for different applications.

Spectral analysis example

- Information propagation over a network: SI, SIR, SIRI

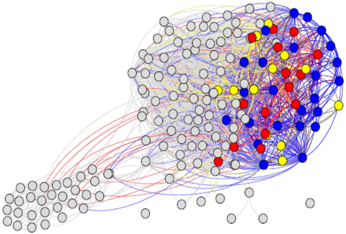


Source: J. McAuley and J. Leskovec. Learning to Discover Social Circles in Ego Networks. NIPS, 2012

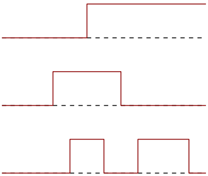


Spectral analysis example

- Information propagation over a network: SI, SIR, SIRI



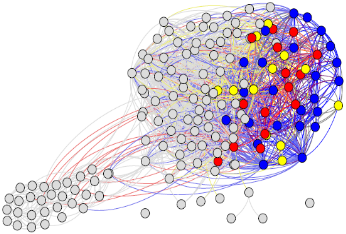
Source: J. McAuley and J. Leskovec. Learning to Discover Social Circles in Ego Networks. NIPS, 2012

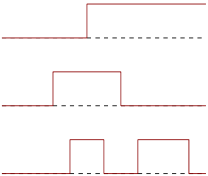


- Signal at each node v: $f(v) \in L^2([0, T])$, step function unbandlimited in time domain.

Spectral analysis example

- Information propagation over a network: SI, SIR, SIRI



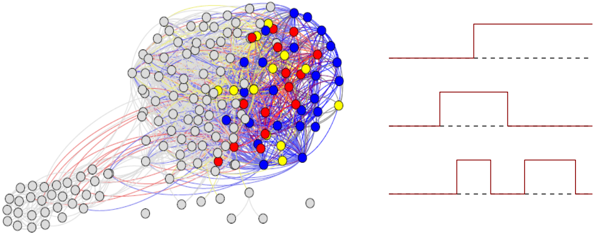


Source: J. McAuley and J. Leskovec. Learning to Discover Social Circles in Ego Networks. NIPS, 2012

- Signal at each node v: $f(v) \in L^2([0,T])$, step function unbandlimited in time domain.
- Suppose infection rate $\lambda_I = 1$, what is the recovery rate λ_R?

Spectral analysis example

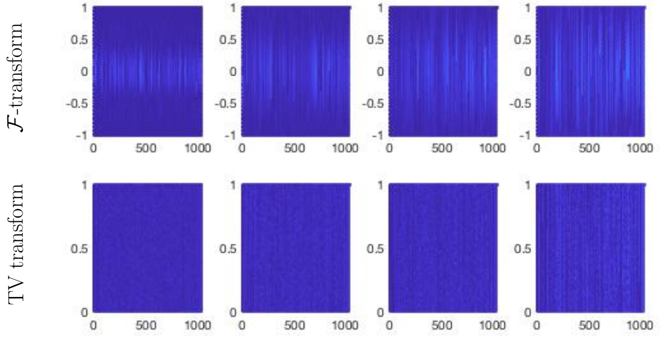
- Information propagation over a network: SI, SIR, SIRI



Source: J. McAuley and J. Leskovec. Learning to Discover Social Circles in Ego Networks. NIPS, 2012

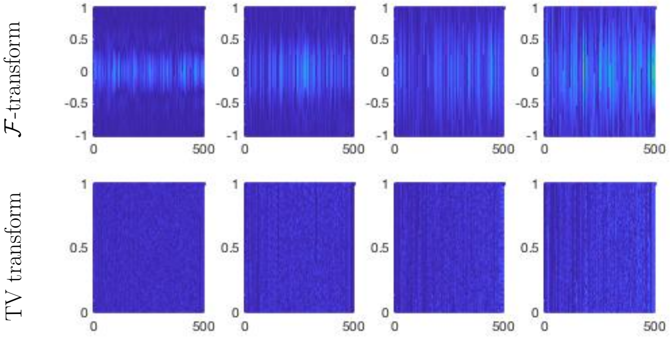
- Signal at each node v: $f(v) \in L^2([0,T])$, step function unbandlimited in time domain.
- Suppose infection rate $\lambda_I = 1$, what is the recovery rate λ_R?
- Loss of information in using
  - GSP (aggregated statistics over time) or
  - TV-GSP (uniform sampling over time).

Spectral analysis example



Facebook network, $\lambda_I = 1$: $\lambda_R = 0$, $\lambda_R = 1/5$, $\lambda_R = 1/2$ and $\lambda_R = 1$.

Spectral analysis example



Enron email network, $\lambda_I = 1$: $\lambda_R = 0$, $\lambda_R = 1/5$, $\lambda_R = 1/2$ and $\lambda_R = 1$.

Outline

Asynchronous joint sampling

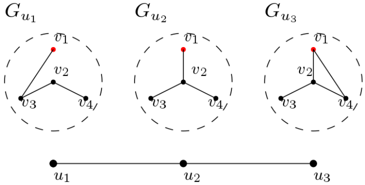
- Joint sampling over vertex and Hilbert space domains.

Asynchronous joint sampling

- Joint sampling over vertex and Hilbert space domains.
- Synchronous sampling is not always easy: sensor networks, social networks, ...

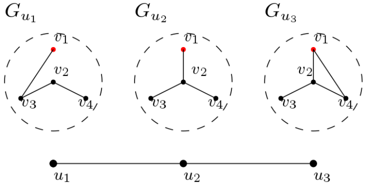
Asynchronous joint sampling

- Joint sampling over vertex and Hilbert space domains.
- Synchronous sampling is not always easy: sensor networks, social networks, ...
- May need different graph sampling at different time vertices:



Asynchronous joint sampling

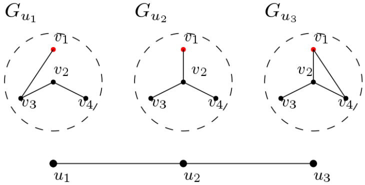
- Joint sampling over vertex and Hilbert space domains.
- Synchronous sampling is not always easy: sensor networks, social networks, ...
- May need different graph sampling at different time vertices:



- Suppose $f \in \mathrm{span}(\Phi' \otimes \Xi')$ - bandlimited.

Asynchronous joint sampling

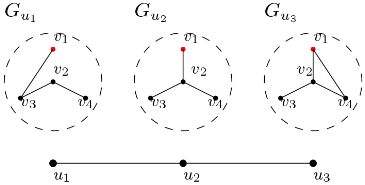
- Joint sampling over vertex and Hilbert space domains.
- Synchronous sampling is not always easy: sensor networks, social networks, ...
- May need different graph sampling at different time vertices:



- Suppose $f \in \mathrm{span}(\Phi' \otimes \Xi')$ - bandlimited.
- Sampling: choose $W \subset V \times \Omega$ such that f is uniquely determined by W.

Asynchronous joint sampling

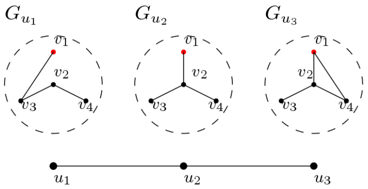
- Joint sampling over vertex and Hilbert space domains.
- Synchronous sampling is not always easy: sensor networks, social networks, ...
- May need different graph sampling at different time vertices:



- Suppose $f \in \text{span}(\Phi' \otimes \Xi')$ - bandlimited.
- Sampling: choose $W \subset V \times \Omega$ such that f is uniquely determined by W.
- $f \in \text{span}(\Phi' \otimes \Xi') \implies |W| \geq |\Phi'| \cdot |\Xi'|$. But not all sampling schemes work.

Asynchronous joint sampling

- Joint sampling over vertex and Hilbert space domains.
- Synchronous sampling is not always easy: sensor networks, social networks, ...
- May need different graph sampling at different time vertices:



- Suppose $f \in \operatorname{span}(\Phi' \otimes \Xi')$ - bandlimited.
- Sampling: choose $W \subset V \times \Omega$ such that f is uniquely determined by W.
- $f \in \operatorname{span}(\Phi' \otimes \Xi') \implies |W| \geq |\Phi'| \cdot |\Xi'|$. But not all sampling schemes work.
- "Reconstructible vertex set": can reconstruct whole graph signal at each instant $x \in \Omega$ from signals in this set. $\implies$ linearly independent rows of matrix Φ'.

Asynchronous joint sampling

Theorem

Two ***asynchronous*** *ways (not exhaustive) to form sample set W:*

Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

1. - *Choose V' to be any reconstructible vertex set.*

Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

1.
   - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*

Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

1. - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
2. - *Choose $V' = V$.*

Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

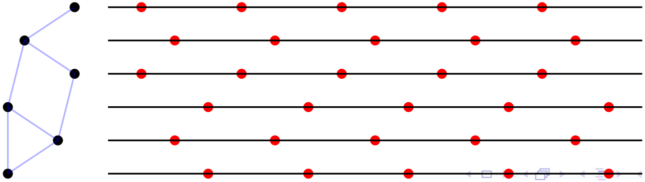
1. - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
2. - *Choose $V' = V$.*
   - *Choose $|\Xi'|$ points Ω' such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*

Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

1. - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
2. - *Choose $V' = V$.*
   - *Choose $|\Xi'|$ points Ω' such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
   - *$\delta(\Phi')$ = size of maximal partition of V into disjoint reconstructible vertex sets. Partition $\Omega = \bigcup_{j=1}^{\delta(\Phi')} \Omega_j$ with $|\Omega_j| < |\Xi'|/\delta(\Phi') + 1$ and Ω_j are the sample points for all $v \in I_j$.*



Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

1. - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
2. - *Choose $V' = V$.*
   - *Choose $|\Xi'|$ points Ω' such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
   - *$\delta(\Phi') =$ size of maximal partition of V into disjoint reconstructible vertex sets. Partition $\Omega = \bigcup_{j=1}^{\delta(\Phi')} \Omega_j$ with $|\Omega_j| < |\Xi'|/\delta(\Phi') + 1$ and Ω_j are the sample points for all $v \in I_j$.*

Furthermore, if $\mathcal{H}$ is spanned by analytic functions, then any random ($\prec$ Lebesgue measure) perturbation of W still determines f a.s.

Asynchronous joint sampling

Theorem

Two ***asynchronous*** *ways (not exhaustive) to form sample set W:*

1.
   - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
2.
   - *Choose $V' = V$.*
   - *Choose $|\Xi'|$ points Ω' such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
   - *$\delta(\Phi')$ = size of maximal partition of V into disjoint reconstructible vertex sets. Partition $\Omega = \bigcup_{j=1}^{\delta(\Phi')} \Omega_j$ with $|\Omega_j| < |\Xi'|/\delta(\Phi') + 1$ and Ω_j are the sample points for all $v \in I_j$.*

Furthermore, if $\mathcal{H}$ is spanned by analytic functions, then any random ($\prec$ Lebesgue measure) perturbation of W still determines f a.s.

- Suppose $f(v, \cdot) \in L^2([-\pi, \pi])$ is bandlimited to a frequency band $[-B, B]$ in the classical Fourier series sense.

Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

1. - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
2. - *Choose $V' = V$.*
   - *Choose $|\Xi'|$ points Ω' such that $f(v, \cdot) \in \operatorname{span} \Xi'$.*
   - *$\delta(\Phi')$ = size of maximal partition of V into disjoint reconstructible vertex sets. Partition $\Omega = \bigcup_{j=1}^{\delta(\Phi')} \Omega_j$ with $|\Omega_j| < |\Xi'|/\delta(\Phi') + 1$ and Ω_j are the sample points for all $v \in I_j$.*

Furthermore, if $\mathcal{H}$ is spanned by analytic functions, then any random ($\prec$ Lebesgue measure) perturbation of W still determines f a.s.

- Suppose $f(v, \cdot) \in L^2([-\pi, \pi])$ is bandlimited to a frequency band $[-B, B]$ in the classical Fourier series sense.
- Shannon-Nyquist Theorem: at least $2B$ samples to recover $f(v, \cdot)$ for each v individually.

Asynchronous joint sampling

Theorem

*Two **asynchronous** ways (not exhaustive) to form sample set W:*

1. - *Choose V' to be any reconstructible vertex set.*
   - *For each $v \in V'$, choose $|\Xi'|$ points in Ω such that $f(v,\cdot) \in \operatorname{span}\Xi'$.*
2. - *Choose $V' = V$.*
   - *Choose $|\Xi'|$ points Ω' such that $f(v,\cdot) \in \operatorname{span}\Xi'$.*
   - *$\delta(\Phi')$ = size of maximal partition of V into disjoint reconstructible vertex sets. Partition $\Omega = \bigcup_{j=1}^{\delta(\Phi')} \Omega_j$ with $|\Omega_j| < |\Xi'|/\delta(\Phi') + 1$ and Ω_j are the sample points for all $v \in I_j$.*

Furthermore, if $\mathcal{H}$ is spanned by analytic functions, then any random ($\prec$ Lebesgue measure) perturbation of W still determines f a.s.

- Suppose $f(v,\cdot) \in L^2([-\pi,\pi])$ is bandlimited to a frequency band $[-B,B]$ in the classical Fourier series sense.
- Shannon-Nyquist Theorem: at least $2B$ samples to recover $f(v,\cdot)$ for each v individually.
- But if $\{f(v,\cdot) : v \in V\}$ is bandlimited in graph vertex domain, then only need $\approx 2B/\delta(\Phi')$ for each vertex to recover all signals.

Sampling example

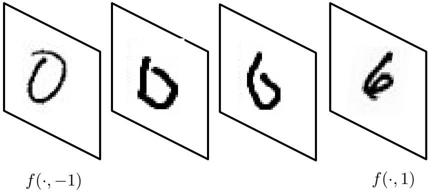
- G - pixels of an image, $|V| = 784$, Φ : eigenvectors of graph Laplacian matrix A_G.

Sampling example

- G - pixels of an image, $|V| = 784$, $\Phi:$ eigenvectors of graph Laplacian matrix A_G.
- $\mathcal{H} = L^2([-1,1])$, $\Xi:$ Chebyshev polynomials of the first kind $\{P_j\}_{j\geq 0}$.

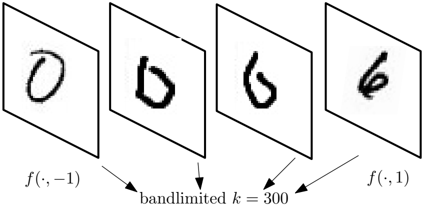
Sampling example

- G - pixels of an image, $|V| = 784$, Φ : eigenvectors of graph Laplacian matrix A_G.
- $\mathcal{H} = L^2([-1,1])$, Ξ : Chebyshev polynomials of the first kind $\{P_j\}_{j\geq 0}$.
- $f(\cdot,1)$ and $f(\cdot,-1)$ correspond to images of handwritten digits '0' and '6'.



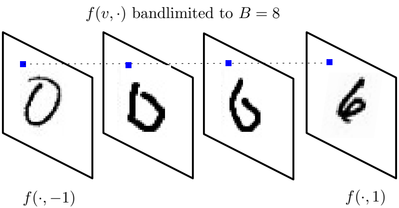
Sampling example

- G - pixels of an image, $|V| = 784$, Φ : eigenvectors of graph Laplacian matrix A_G.
- $\mathcal{H} = L^2([-1,1])$, Ξ : Chebyshev polynomials of the first kind $\{P_j\}_{j\geq 0}$.
- $f(\cdot,1)$ and $f(\cdot,-1)$ correspond to images of handwritten digits '0' and '6'.
- For each $x \in [-1,1]$, $f(\cdot,x)$ is graph bandlimited to the first $k = 300$ eigenvalues of A_G.



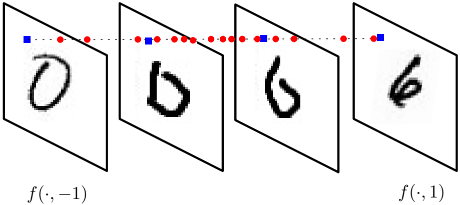
Sampling example

- G - pixels of an image, $|V| = 784$, Φ : eigenvectors of graph Laplacian matrix A_G.
- $\mathcal{H} = L^2([-1,1])$, Ξ : Chebyshev polynomials of the first kind $\{P_j\}_{j\geq 0}$.
- $f(\cdot,1)$ and $f(\cdot,-1)$ correspond to images of handwritten digits '0' and '6'.
- For each $x \in [-1,1]$, $f(\cdot,x)$ is graph bandlimited to the first $k = 300$ eigenvalues of A_G.
- For each node v, $f(v,\cdot)$ is in the span of the first $B = 8$ Chebyshev polynomials.



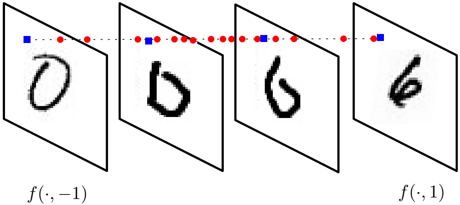
Sampling example

- Asynchronous Sampling Theorem:



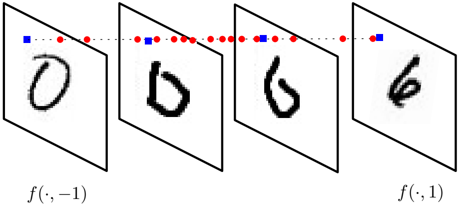
Sampling example

- Asynchronous Sampling Theorem:
  - sample $2k$ nodes (pixels)



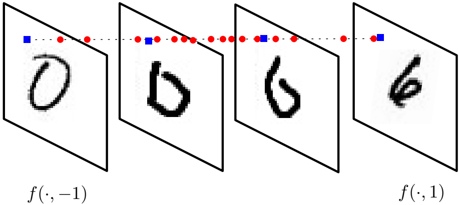
Sampling example

- Asynchronous Sampling Theorem:
  - sample $2k$ nodes (pixels)
  - each node: sample $B/2$ random positions in $[-1, 1]$ following $\mathcal{N}(0,\ 0.5)$



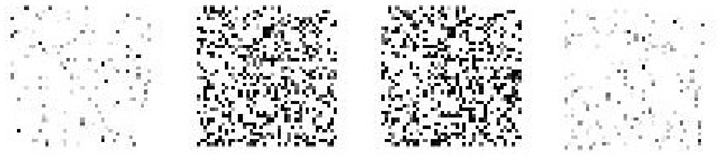
Sampling example

- Asynchronous Sampling Theorem:
  - sample $2k$ nodes (pixels)
  - each node: sample $B/2$ random positions in $[-1, 1]$ following $\mathcal{N}(0,\ 0.5)$
- With probability 1, impossible to reconstruct uniform samples.



Sampling example

- Asynchronous Sampling Theorem:
  - sample $2k$ nodes (pixels)
  - each node: sample $B/2$ random positions in $[-1, 1]$ following $\mathcal{N}(0,\ 0.5)$
- With probability 1, impossible to reconstruct uniform samples.



$f(\cdot, [-1, -0.5])$ $f(\cdot, [-0.5, 0])$ $f(\cdot, [0, 0.5])$ $f(\cdot, [0.5, 1])$

Sampling example

- Suppose we sample $W = \{(v_m, x_l) : m = 1, \ldots, 2k,\ l = 1, \ldots, B/2\}$.

Sampling example

- Suppose we sample $W = \{(v_m, x_l) : m = 1, \ldots, 2k,\ l = 1, \ldots, B/2\}$.
- Add white Gaussian noise to obtain $\tilde{f}$.

$$\tilde{f}(v_m, x_l) = \sum_{i \le k} \sum_{0 \le j < B} y_{i,j} \phi_i(v_m) P_j(x_l) + \text{ noise}$$

Sampling example

- Suppose we sample $W = \{(v_m, x_l) : m = 1, \ldots, 2k,\ l = 1, \ldots, B/2\}$.
- Add white Gaussian noise to obtain $\tilde{f}$.

$$\tilde{f}(v_m, x_l) = \sum_{i \leq k} \sum_{0 \leq j < B} y_{i,j} \phi_i(v_m) P_j(x_l) + \text{ noise}$$

- Let M be the corresponding transformation matrix with entries $\phi_i(v_m) P_j(x_l)$.

Sampling example

- Suppose we sample $W = \{(v_m, x_l) : m = 1, \ldots, 2k,\ l = 1, \ldots, B/2\}$.
- Add white Gaussian noise to obtain $\tilde{f}$.

$$\tilde{f}(v_m, x_l) = \sum_{i \leq k} \sum_{0 \leq j < B} y_{i,j} \phi_i(v_m) P_j(x_l) + \text{ noise}$$

- Let M be the corresponding transformation matrix with entries $\phi_i(v_m) P_j(x_l)$.
- Recover $y = (y_{i,j})$ by solving the optimization:

$$\arg\min_y \left\| My - \tilde{f}(W) \right\|_2^2$$

Sampling example

- Suppose we sample $W = \{(v_m, x_l) : m = 1, \ldots, 2k,\ l = 1, \ldots, B/2\}$.
- Add white Gaussian noise to obtain $\tilde{f}$.

$$\tilde{f}(v_m, x_l) = \sum_{i \le k} \sum_{0 \le j < B} y_{i,j} \phi_i(v_m) P_j(x_l) + \text{ noise}$$

- Let M be the corresponding transformation matrix with entries $\phi_i(v_m) P_j(x_l)$.
- Recover $y = (y_{i,j})$ by solving the optimization:

$$\arg\min_y \left\| My - \tilde{f}(W) \right\|_2^2$$

Result:

Outline

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G,\mathcal{H}) \to S(G,\mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.
- L is weakly shift invariant if $(A_G \otimes A) \circ L = L \circ (A_G \otimes A)$.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.
- L is weakly shift invariant if $(A_G \otimes A) \circ L = L \circ (A_G \otimes A)$.
- Shift invariant: commutes with $A_G^p \otimes A^q$ for all $p, q \geq 0$.
- Weakly shift invariant: commutes with $(A_G \otimes A)^p = A_G^p \otimes A^p$ for all $p \geq 0$.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.
- L is weakly shift invariant if $(A_G \otimes A) \circ L = L \circ (A_G \otimes A)$.
- Shift invariant: commutes with $A_G^p \otimes A^q$ for all $p, q \geq 0$.
- Weakly shift invariant: commutes with $(A_G \otimes A)^p = A_G^p \otimes A^p$ for all $p \geq 0$.
- Example: polynomial filter

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.
- L is weakly shift invariant if $(A_G \otimes A) \circ L = L \circ (A_G \otimes A)$.
- Shift invariant: commutes with $A_G^p \otimes A^q$ for all $p, q \geq 0$.
- Weakly shift invariant: commutes with $(A_G \otimes A)^p = A_G^p \otimes A^p$ for all $p \geq 0$.
- Example: polynomial filter
  - Let $P(x) = a_0 + a_1 x + \ldots + a_p x^p$ be a polynomial of degree $p < \infty$.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.
- L is weakly shift invariant if $(A_G \otimes A) \circ L = L \circ (A_G \otimes A)$.
- Shift invariant: commutes with $A_G^p \otimes A^q$ for all $p, q \geq 0$.
- Weakly shift invariant: commutes with $(A_G \otimes A)^p = A_G^p \otimes A^p$ for all $p \geq 0$.
- Example: polynomial filter
  - Let $P(x) = a_0 + a_1 x + \ldots + a_p x^p$ be a polynomial of degree $p < \infty$.
  - $P(A_G \otimes A)$ commutes with both $A_G \otimes \mathrm{Id}$ and $\mathrm{Id} \otimes A$, $\therefore$ shift invariant.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.
- L is weakly shift invariant if $(A_G \otimes A) \circ L = L \circ (A_G \otimes A)$.
- Shift invariant: commutes with $A_G^p \otimes A^q$ for all $p, q \geq 0$.
- Weakly shift invariant: commutes with $(A_G \otimes A)^p = A_G^p \otimes A^p$ for all $p \geq 0$.
- Example: polynomial filter
  - Let $P(x) = a_0 + a_1 x + \ldots + a_p x^p$ be a polynomial of degree $p < \infty$.
  - $P(A_G \otimes A)$ commutes with both $A_G \otimes \mathrm{Id}$ and $\mathrm{Id} \otimes A$, $\therefore$ shift invariant.
- Example: $J = (1 - a^{-1}A)^{-1}$, $a > \rho(A)$, commutes with $A \implies$ $L = A_G \otimes J$ is shift invariant, but not polynomial if $\dim \mathcal{H} = \infty$.

Shift invariance

- A *filter* is a bounded linear transformation $L : S(G, \mathcal{H}) \to S(G, \mathcal{H})$.
- Tensor product filter $A_G \otimes A(v \otimes h) = A_G(v) \otimes A(h)$ is an example.
- L is shift invariant if both $(A_G \otimes \mathrm{Id}) \circ L = L \circ (A_G \otimes \mathrm{Id})$ and $(\mathrm{Id} \otimes A) \circ L = L \circ (\mathrm{Id} \otimes A)$.
- L is weakly shift invariant if $(A_G \otimes A) \circ L = L \circ (A_G \otimes A)$.
- Shift invariant: commutes with $A_G^p \otimes A^q$ for all $p, q \geq 0$.
- Weakly shift invariant: commutes with $(A_G \otimes A)^p = A_G^p \otimes A^p$ for all $p \geq 0$.
- Example: polynomial filter
    - Let $P(x) = a_0 + a_1 x + \ldots + a_p x^p$ be a polynomial of degree $p < \infty$.
    - $P(A_G \otimes A)$ commutes with both $A_G \otimes \mathrm{Id}$ and $\mathrm{Id} \otimes A$, $\therefore$ shift invariant.
- Example: $J = (1 - a^{-1} A)^{-1}$, $a > \rho(A)$, commutes with $A \implies$ $L = A_G \otimes J$ is shift invariant, but not polynomial if $\dim \mathcal{H} = \infty$. If finite dimensional, J is polynomial $\because$ $\exists$ minimal polynomial.

Shift invariance

Theorem

1. *Shift invariant* $\implies$ *weakly shift invariant*

Shift invariance

Theorem

1. *Shift invariant* $\Longrightarrow$ *weakly shift invariant*
2. $m_\lambda(A_G \otimes A)$ - *dimension of* λ*-eigenspace. If* $m_\lambda = 1$ *for all* λ, *weakly shift invariant* $\Longrightarrow$ *shift invariant.*

Shift invariance

Theorem

1. *Shift invariant* $\implies$ *weakly shift invariant*
2. $m_\lambda(A_G \otimes A)$ - *dimension of* λ*-eigenspace. If* $m_\lambda = 1$ *for all* λ*, weakly shift invariant* $\implies$ *shift invariant.*
3. *Self-adjoint* L*: weakly shift invariant* $\iff$ *shift invariant.*

Shift invariance

Theorem

1. *Shift invariant* $\implies$ *weakly shift invariant*
2. $m_\lambda(A_G \otimes A)$ - *dimension of* λ*-eigenspace. If* $m_\lambda = 1$ *for all* λ*, weakly shift invariant* $\implies$ *shift invariant.*
3. *Self-adjoint* L*: weakly shift invariant* $\iff$ *shift invariant.*

- If $m_\lambda(A_G) = 1$, traditional GSP: all shift invariant filters are polynomial. Not true in GGSP even if $m_\lambda(A_G \otimes A) = 1$ [e.g., $L = A_G \otimes J$].

Shift invariance

Theorem

1. *Shift invariant* $\implies$ *weakly shift invariant*
2. $m_\lambda(A_G \otimes A)$ - *dimension of* λ*-eigenspace. If* $m_\lambda = 1$ *for all* λ*, weakly shift invariant* $\implies$ *shift invariant.*
3. *Self-adjoint* L*: weakly shift invariant* $\iff$ *shift invariant.*

- If $m_\lambda(A_G) = 1$, traditional GSP: all shift invariant filters are polynomial. Not true in GGSP even if $m_\lambda(A_G \otimes A) = 1$ [e.g., $L = A_G \otimes J$].
- If $m_\lambda(A_G) = 1$ and $m_\lambda(A) = 1$, we hope $m_\lambda(A_G \otimes A) = 1$ so that all weakly shift invariant filters are shift invariant [but may have $\lambda_\phi \lambda_\xi = \lambda_{\phi'} \lambda_{\xi'}$].

Shift invariance

Theorem

1. *Shift invariant* $\implies$ *weakly shift invariant*
2. $m_\lambda(A_G \otimes A)$ - *dimension of* λ*-eigenspace. If* $m_\lambda = 1$ *for all* λ, *weakly shift invariant* $\implies$ *shift invariant.*
3. *Self-adjoint* L: *weakly shift invariant* $\iff$ *shift invariant.*

- If $m_\lambda(A_G) = 1$, traditional GSP: all shift invariant filters are polynomial. Not true in GGSP even if $m_\lambda(A_G \otimes A) = 1$ [e.g., $L = A_G \otimes J$].
- If $m_\lambda(A_G) = 1$ and $m_\lambda(A) = 1$, we hope $m_\lambda(A_G \otimes A) = 1$ so that all weakly shift invariant filters are shift invariant [but may have $\lambda_\phi \lambda_\xi = \lambda_{\phi'} \lambda_{\xi'}$].
- Not always true, but almost always in practice ...

Shift invariance

Theorem

- *G has at least 3 nodes.*
- *Each edge weight of G is chosen randomly ($\prec$ Lebesgue measure).*
- $m_\lambda(A) = 1 \ \forall \lambda$.

Shift invariance

Theorem

- *G has at least 3 nodes.*
- *Each edge weight of G is chosen randomly ($\prec$ Lebesgue measure).*
- *$m_\lambda(A) = 1\ \forall\lambda$.*

Then with probability one, we have:

1. *If A_G is the adjacency matrix of G: $m_\lambda = 1$ for all λ of $A_G \otimes A$.*

Shift invariance

Theorem

- G *has at least* 3 *nodes.*
- *Each edge weight of* G *is chosen randomly (*$\prec$ *Lebesgue measure).*
- $m_\lambda(A) = 1 \ \forall \lambda$.

Then with probability one, we have:

1. *If* A_G *is the adjacency matrix of* G*:* $m_\lambda = 1$ *for all* λ *of* $A_G \otimes A$*.*
2. *If* A_G *is the Laplacian matrix of* G*: then the 0-eigenspace of* $A_G \otimes A$ *is isomorphic to* $\mathcal{H}$*. For* $\lambda \neq 0$, $m_\lambda = 1$.

Shift invariance

Theorem

- *G has at least 3 nodes.*
- *Each edge weight of G is chosen randomly ($\prec$ Lebesgue measure).*
- *$m_\lambda(A) = 1 \ \forall \lambda$.*

Then with probability one, we have:

1. *If A_G is the adjacency matrix of G: $m_\lambda = 1$ for all λ of $A_G \otimes A$.*
2. *If A_G is the Laplacian matrix of G: then the 0-eigenspace of $A_G \otimes A$ is isomorphic to $\mathcal{H}$. For $\lambda \neq 0$, $m_\lambda = 1$.*

- If A_G is the Laplacian matrix, we can restrict to the orthogonal complement of 0-eigenspace of $A_G \otimes A$.

Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g *$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g*$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

- $\sum_{\phi \otimes \xi} |\mathcal{F}_g(\phi \otimes \xi)|^2 < \infty \implies g*$ is Hilbert-Schmidt operator $\implies$ compact.

Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g*$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

- $\sum_{\phi \otimes \xi} |\mathcal{F}_g(\phi \otimes \xi)|^2 < \infty \implies g*$ is Hilbert-Schmidt operator $\implies$ compact.
- Let $f = \phi \otimes \xi$ with $\phi \in \Phi$ and $\xi \in \Xi$. Then, $g * f = \mathcal{F}_g(\phi \otimes \xi) \cdot \phi \otimes \xi$.

Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g*$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

- $\sum_{\phi \otimes \xi} |\mathcal{F}_g(\phi \otimes \xi)|^2 < \infty \implies g*$ is Hilbert-Schmidt operator $\implies$ compact.
- Let $f = \phi \otimes \xi$ with $\phi \in \Phi$ and $\xi \in \Xi$. Then, $g * f = \mathcal{F}_g(\phi \otimes \xi) \cdot \phi \otimes \xi$.
    - $\phi \otimes \xi$ is an eigenvector of $g*$

Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g*$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

- $\sum_{\phi \otimes \xi} |\mathcal{F}_g(\phi \otimes \xi)|^2 < \infty \implies g*$ is Hilbert-Schmidt operator $\implies$ compact.
- Let $f = \phi \otimes \xi$ with $\phi \in \Phi$ and $\xi \in \Xi$. Then, $g * f = \mathcal{F}_g(\phi \otimes \xi) \cdot \phi \otimes \xi$.
  - $\phi \otimes \xi$ is an eigenvector of $g*$
  - $g*$ commutes with $A_G \otimes \mathrm{Id}$ and $\mathrm{Id} \otimes A$

Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g*$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

- $\sum_{\phi \otimes \xi} |\mathcal{F}_g(\phi \otimes \xi)|^2 < \infty \implies g*$ is Hilbert-Schmidt operator $\implies$ compact.
- Let $f = \phi \otimes \xi$ with $\phi \in \Phi$ and $\xi \in \Xi$. Then, $g * f = \mathcal{F}_g(\phi \otimes \xi) \cdot \phi \otimes \xi$.
  - $\phi \otimes \xi$ is an eigenvector of $g*$
  - $g*$ commutes with $A_G \otimes \mathrm{Id}$ and $\mathrm{Id} \otimes A$
  - $g*$ is shift invariant

Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g*$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

- $\sum_{\phi \otimes \xi} |\mathcal{F}_g(\phi \otimes \xi)|^2 < \infty \implies g*$ is Hilbert-Schmidt operator $\implies$ compact.
- Let $f = \phi \otimes \xi$ with $\phi \in \Phi$ and $\xi \in \Xi$. Then, $g * f = \mathcal{F}_g(\phi \otimes \xi) \cdot \phi \otimes \xi$.
  - $\phi \otimes \xi$ is an eigenvector of $g*$
  - $g*$ commutes with $A_G \otimes \mathrm{Id}$ and $\mathrm{Id} \otimes A$
  - $g*$ is shift invariant
- GSP: all polynomial filters are convolutions.

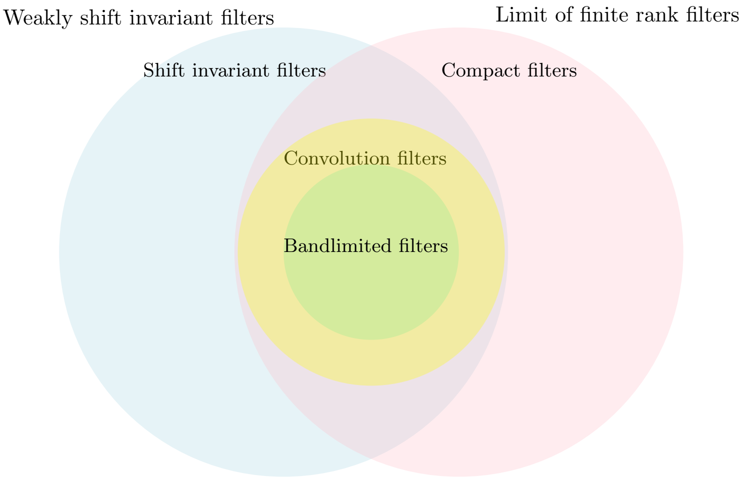
Convolution filter is shift invariant

- $g \in S(G, \mathcal{H})$, define $g*$ by $\mathcal{F}_{g*f} = \mathcal{F}_g \mathcal{F}_f$ is an element of $S(G, \mathcal{H})$, i.e.,

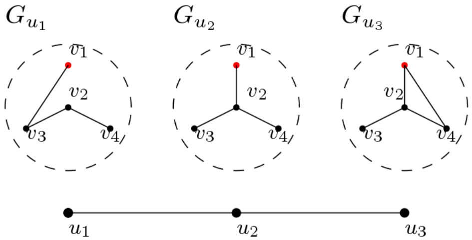
$$g * f = \sum_{\phi \otimes \xi} \mathcal{F}_g(\phi \otimes \xi) \mathcal{F}_f(\phi \otimes \xi) \cdot \phi \otimes \xi.$$

- $\sum_{\phi \otimes \xi} |\mathcal{F}_g(\phi \otimes \xi)|^2 < \infty \implies g*$ is Hilbert-Schmidt operator $\implies$ compact.
- Let $f = \phi \otimes \xi$ with $\phi \in \Phi$ and $\xi \in \Xi$. Then, $g * f = \mathcal{F}_g(\phi \otimes \xi) \cdot \phi \otimes \xi$.
  - $\phi \otimes \xi$ is an eigenvector of $g*$
  - $g*$ commutes with $A_G \otimes \mathrm{Id}$ and $\mathrm{Id} \otimes A$
  - $g*$ is shift invariant
- GSP: all polynomial filters are convolutions.
- Polynomial filter $P(A_G \otimes A)$ with $a_0 \neq 0$ is non-compact, therefore not convolution.

Different classes of filters

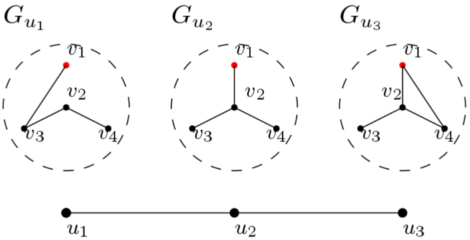


Adaptive polynomial filters



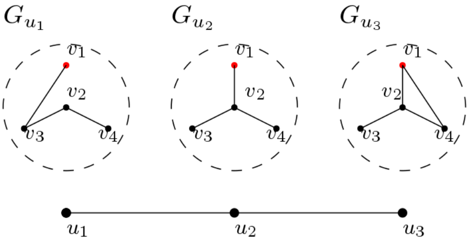
- At each vertex $u \in G$, different graph $G_u \implies$ different operator A_u (e.g., adjacency, Laplacian).

Adaptive polynomial filters



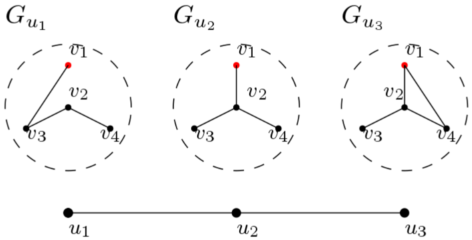
- At each vertex $u \in G$, different graph $G_u \implies$ different operator A_u (e.g., adjacency, Laplacian).
- Adaptive polynomial filter $F = \sum_u P_1(A_G)_u \otimes P_2(A_u)$

Adaptive polynomial filters



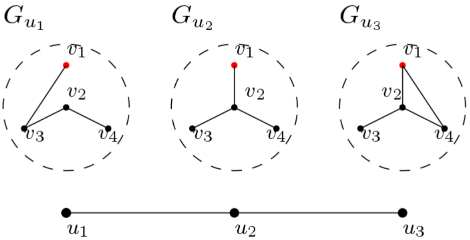
- At each vertex $u \in G$, different graph $G_u \implies$ different operator A_u (e.g., adjacency, Laplacian).
- Adaptive polynomial filter $F = \sum_u P_1(A_G)_u \otimes P_2(A_u)$
  - P_1, P_2 are polynomials

Adaptive polynomial filters



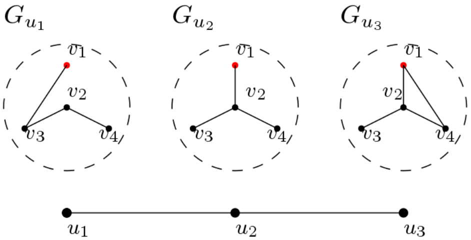
- At each vertex $u \in G$, different graph $G_u \implies$ different operator A_u (e.g., adjacency, Laplacian).
- Adaptive polynomial filter $F = \sum_u P_1(A_G)_u \otimes P_2(A_u)$
  - P_1, P_2 are polynomials
  - $P_1(A_G)_u$: matrix with u-th column of $P_1(A_G)$, 0 elsewhere.

Adaptive polynomial filters



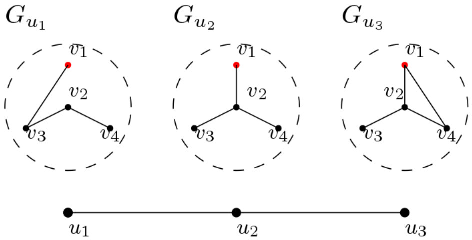
- Suppose $P_1(x) = a_1 x + b_1$, $P_2(x) = a_2 x + b_2$.

Adaptive polynomial filters



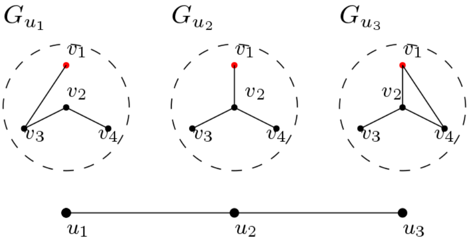
- Suppose $P_1(x) = a_1x + b_1$, $P_2(x) = a_2x + b_2$.
- $F(f)(u,i)$ - affected by $f(v,j)$ if $(u,v) \in G$ and $(i,j) \in G_v$.
- Filter F captures hidden structures in $\mathbb{C}^4$.

Adaptive polynomial filters



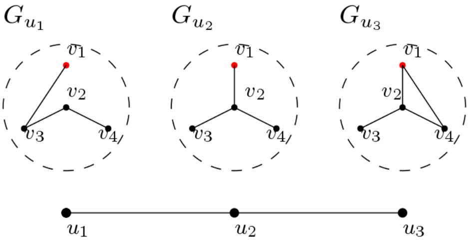
- Suppose $P_1(x) = a_1x + b_1$, $P_2(x) = a_2x + b_2$.
- $F(f)(u, i)$ - affected by $f(v, j)$ if $(u, v) \in G$ and $(i, j) \in G_v$.
- Filter F captures hidden structures in $\mathbb{C}^4$.
- G and each G_{u_i} have different physical meanings:

Adaptive polynomial filters



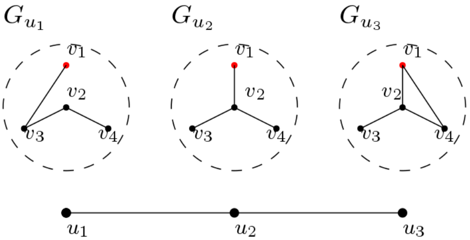
- Suppose $P_1(x) = a_1 x + b_1$, $P_2(x) = a_2 x + b_2$.
- $F(f)(u,i)$ - affected by $f(v,j)$ if $(u,v) \in G$ and $(i,j) \in G_v$.
- Filter F captures hidden structures in $\mathbb{C}^4$.
- G and each G_{u_i} have different physical meanings:
  - G - represents time

Adaptive polynomial filters



- Suppose $P_1(x) = a_1x + b_1$, $P_2(x) = a_2x + b_2$.
- $F(f)(u,i)$ - affected by $f(v,j)$ if $(u,v) \in G$ and $(i,j) \in G_v$.
- Filter F captures hidden structures in $\mathbb{C}^4$.
- G and each G_{u_i} have different physical meanings:
  - G - represents time
  - G_{u_i} correlations between node observations at time u_i

Adaptive polynomial filters



- Suppose $P_1(x) = a_1x + b_1$, $P_2(x) = a_2x + b_2$.
- $F(f)(u,i)$ - affected by $f(v,j)$ if $(u,v) \in G$ and $(i,j) \in G_v$.
- Filter F captures hidden structures in $\mathbb{C}^4$.
- G and each G_{u_i} have different physical meanings:
  - G - represents time
  - G_{u_i} correlations between node observations at time u_i
- Wrong to use GSP on big ambient graph containing all G_{u_i}s.

Adaptive polynomial filters: example

- Sensor network in dynamic environments like ocean surface. Social network topology changes over time.

Adaptive polynomial filters: example

- Sensor network in dynamic environments like ocean surface. Social network topology changes over time.
- Each time $t = 1, \ldots, n$, $f(t) \in \mathbb{C}^m$ on graph G_t with graph shift operator A_t.

Adaptive polynomial filters: example

- Sensor network in dynamic environments like ocean surface. Social network topology changes over time.
- Each time $t = 1, \dots, n$, $f(t) \in \mathbb{C}^m$ on graph G_t with graph shift operator A_t.
- A_t evolves according to a known model, starting from A_0.

Adaptive polynomial filters: example

- Sensor network in dynamic environments like ocean surface. Social network topology changes over time.
- Each time $t = 1, \ldots, n$, $f(t) \in \mathbb{C}^m$ on graph G_t with graph shift operator A_t.
- A_t evolves according to a known model, starting from A_0.
- f generated from a base signal g: $f = F(g)$ where F is an adaptive polynomial filter.

Adaptive polynomial filters: example

- Sensor network in dynamic environments like ocean surface. Social network topology changes over time.
- Each time $t = 1, \ldots, n$, $f(t) \in \mathbb{C}^m$ on graph G_t with graph shift operator A_t.
- A_t evolves according to a known model, starting from A_0.
- f generated from a base signal g: $f = F(g)$ where F is an adaptive polynomial filter.
- Given observations $\tilde{f}(t) = f(t) + N(t)$ at a subset of time indices, goal is to estimate F:

$$\min \sum_{t \in \text{Obs.}} \left\| F(g)(t) - \tilde{f}(t) \right\|_2^2$$

Adaptive polynomial filters: example

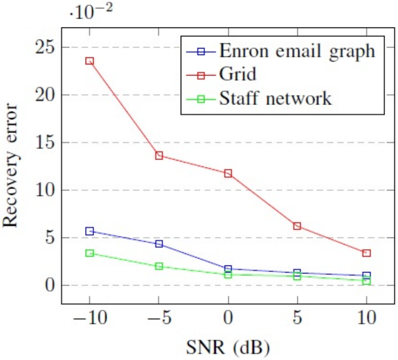
- Sensor network in dynamic environments like ocean surface. Social network topology changes over time.
- Each time $t = 1, \ldots, n$, $f(t) \in \mathbb{C}^m$ on graph G_t with graph shift operator A_t.
- A_t evolves according to a known model, starting from A_0.
- f generated from a base signal g: $f = F(g)$ where F is an adaptive polynomial filter.
- Given observations $\tilde{f}(t) = f(t) + N(t)$ at a subset of time indices, goal is to estimate F:

$$\min \sum_{t \in \text{Obs.}} \left\| F(g)(t) - \tilde{f}(t) \right\|_2^2$$

- Recovery error:

$$\sum_{t \in \text{Unobs.}} \frac{\left\| \hat{F}(g)(t) - f(t) \right\|_2}{\| f(t) \|_2}$$

Adaptive polynomial filters: example



Graphs evolve according to model in [4] (applications in social networks, biological neuron networks, etc.).

[4] J. Ito and K. Kaneko, "Spontaneous structure formation in a network of chaotic units with variable connection strengths," *Phys. Rev. Letts.*, vol. 88, no. 2, p. 028 701, 2002.

Outline

Summary

	GSP	TV-GSP	GGSP
Signal space	$\mathbb{C}^n$	$\mathbb{C}^n \otimes \mathbb{C}^m$	$\mathbb{C}^n \otimes \mathcal{H}$

Summary

	GSP	TV-GSP	GGSP
Signal space	$\mathbb{C}^n$	$\mathbb{C}^n \otimes \mathbb{C}^m$	$\mathbb{C}^n \otimes \mathcal{H}$
Fourier transform	Φ	$\Phi \otimes \Xi$ (finite dim.)	$\Phi \otimes \Xi$ (infinite dim.)

Summary

	GSP	TV-GSP	GGSP
Signal space	$\mathbb{C}^n$	$\mathbb{C}^n \otimes \mathbb{C}^m$	$\mathbb{C}^n \otimes \mathcal{H}$
Fourier transform	Φ	$\Phi \otimes \Xi$ (finite dim.)	$\Phi \otimes \Xi$ (infinite dim.)
Sampling	vertices	vertices	asynchronous joint

Summary

	GSP	TV-GSP	GGSP
Signal space	$\mathbb{C}^n$	$\mathbb{C}^n \otimes \mathbb{C}^m$	$\mathbb{C}^n \otimes \mathcal{H}$
Fourier transform	Φ	$\Phi \otimes \Xi$ (finite dim.)	$\Phi \otimes \Xi$ (infinite dim.)
Sampling	vertices	vertices	asynchronous joint
Shift invariance	A_G	same as GSP	SI and WSI

Summary

	GSP	TV-GSP	GGSP
Signal space	$\mathbb{C}^n$	$\mathbb{C}^n \otimes \mathbb{C}^m$	$\mathbb{C}^n \otimes \mathcal{H}$
Fourier transform	Φ	$\Phi \otimes \Xi$ (finite dim.)	$\Phi \otimes \Xi$ (infinite dim.)
Sampling	vertices	vertices	asynchronous joint
Shift invariance	A_G	same as GSP	SI and WSI
Convolution	polynomials are conv.	same as GSP	false

Summary

	GSP	TV-GSP	GGSP
Signal space	$\mathbb{C}^n$	$\mathbb{C}^n \otimes \mathbb{C}^m$	$\mathbb{C}^n \otimes \mathcal{H}$
Fourier transform	Φ	$\Phi \otimes \Xi$ (finite dim.)	$\Phi \otimes \Xi$ (infinite dim.)
Sampling	vertices	vertices	asynchronous joint
Shift invariance	A_G	same as GSP	SI and WSI
Convolution	polynomials are conv.	same as GSP	false
Adaptive	n.a.	n.a.	yes

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.
- Shift invariant and weakly shift invariant filters: WSI almost always SI.

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.
- Shift invariant and weakly shift invariant filters: WSI almost always SI.
- Convolution, bandpass filters, adaptive polynomial filters (dynamic networks).

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.
- Shift invariant and weakly shift invariant filters: WSI almost always SI.
- Convolution, bandpass filters, adaptive polynomial filters (dynamic networks).

Future

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.
- Shift invariant and weakly shift invariant filters: WSI almost always SI.
- Convolution, bandpass filters, adaptive polynomial filters (dynamic networks).

Future

- Framework applicable for square integrable graph stochastic processes: for each $v \in V$, $X(v, t, \omega) \in L^2([0, T] \times \Omega, \mathscr{F}, \mathbb{P})$.

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.
- Shift invariant and weakly shift invariant filters: WSI almost always SI.
- Convolution, bandpass filters, adaptive polynomial filters (dynamic networks).

Future

- Framework applicable for square integrable graph stochastic processes: for each $v \in V$, $X(v, t, \omega) \in L^2([0, T] \times \Omega, \mathscr{F}, \mathbb{P})$.
- Notions of stationarity can be defined w.r.t. the shift operators $A_G \otimes \mathrm{Id}$, $\mathrm{Id} \otimes A$ and $A_G \otimes A$ similar to [5,6].

[5] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 5911–5926, Nov. 2017. DOI: 10.1109/TSP.2017.2739099.

[6] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3462–3477, Jul. 2017. DOI: 10.1109/TSP.2017.2690388.

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.
- Shift invariant and weakly shift invariant filters: WSI almost always SI.
- Convolution, bandpass filters, adaptive polynomial filters (dynamic networks).

Future

- Framework applicable for square integrable graph stochastic processes: for each $v \in V$, $X(v,t,\omega) \in L^2([0,T] \times \Omega, \mathscr{F}, \mathbb{P})$.
- Notions of stationarity can be defined w.r.t. the shift operators $A_G \otimes \mathrm{Id}$, $\mathrm{Id} \otimes A$ and $A_G \otimes A$ similar to [5,6].
- "Strict" and "weak" strong and wide-sense stationarity.

[5] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 5911–5926, Nov. 2017. DOI: 10.1109/TSP.2017.2739099.

[6] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3462–3477, Jul. 2017. DOI: 10.1109/TSP.2017.2690388.

Conclusion

- Generalized signal processing framework for vertex signals $f \in \mathcal{H}$.
- Suitable Hilbert space and basis to define $\mathcal{F}$-transform.
- Asynchronous sampling over joint vertex and $\mathcal{H}$ domain: achieves Shannon-Nyquist rate over joint domain.
- Shift invariant and weakly shift invariant filters: WSI almost always SI.
- Convolution, bandpass filters, adaptive polynomial filters (dynamic networks).

Future

- Framework applicable for square integrable graph stochastic processes: for each $v \in V$, $X(v, t, \omega) \in L^2([0, T] \times \Omega, \mathscr{F}, \mathbb{P})$.
- Notions of stationarity can be defined w.r.t. the shift operators $A_G \otimes \mathrm{Id}$, $\mathrm{Id} \otimes A$ and $A_G \otimes A$ similar to [5,6].
- "Strict" and "weak" strong and wide-sense stationarity.
- Other high dimensional extensions: simplicial complexes [7] and hypergraphs [8].

[5] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 5911–5926, Nov. 2017. DOI: 10.1109/TSP.2017.2739099.

[6] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3462–3477, Jul. 2017. DOI: 10.1109/TSP.2017.2690388.

[7] S. Barbarossa and S. Sardellitti, "Topological signal processing over simplicial complexes," *arXiv preprint arXiv:1907.11577*, 2019.

[8] S. Zhang, Z. Ding, and S. Cui, "Introducing hypergraph signal processing: Theoretical foundation and practical applications," *arXiv preprint arXiv:1907.09203*, 2019.

Acknowledgments

Singapore Ministry of Education Academic Research Fund Tier 2 grant
MOE2018-T2-2-019

Thank you!

http://www.ntu.edu.sg/home/wptay/

References I

[1] B. Girault, A. Ortega, and S. S. Narayanan, "Irregularity-aware graph Fourier transforms," *IEEE Transactions on Signal Processing*, vol. 66, no. 21, pp. 5746–5761, Nov. 2018.

[2] F. Grassi, A. Loukas, N. Perraudin, and B. Ricaud, "A time-vertex signal processing framework: Scalable processing and meaningful representations for time-series on graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 817–829, Feb. 2018.

[3] F. Ji and W. P. Tay, "A Hilbert space theory of generalized graph signal processing," *IEEE Trans. Signal Process.*, 2019, accepted. [Online]. Available: `https://arxiv.org/abs/1904.11655`.

[4] J. Ito and K. Kaneko, "Spontaneous structure formation in a network of chaotic units with variable connection strengths," *Phys. Rev. Letts.*, vol. 88, no. 2, p. 028 701, 2002.

[5] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 5911–5926, Nov. 2017. DOI: `10.1109/TSP.2017.2739099`.

References II

[6] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3462–3477, Jul. 2017. DOI: 10.1109/TSP.2017.2690388.

[7] S. Barbarossa and S. Sardellitti, "Topological signal processing over simplicial complexes," *arXiv preprint arXiv:1907.11577*, 2019.

[8] S. Zhang, Z. Ding, and S. Cui, "Introducing hypergraph signal processing: Theoretical foundation and practical applications," *arXiv preprint arXiv:1907.09203*, 2019.