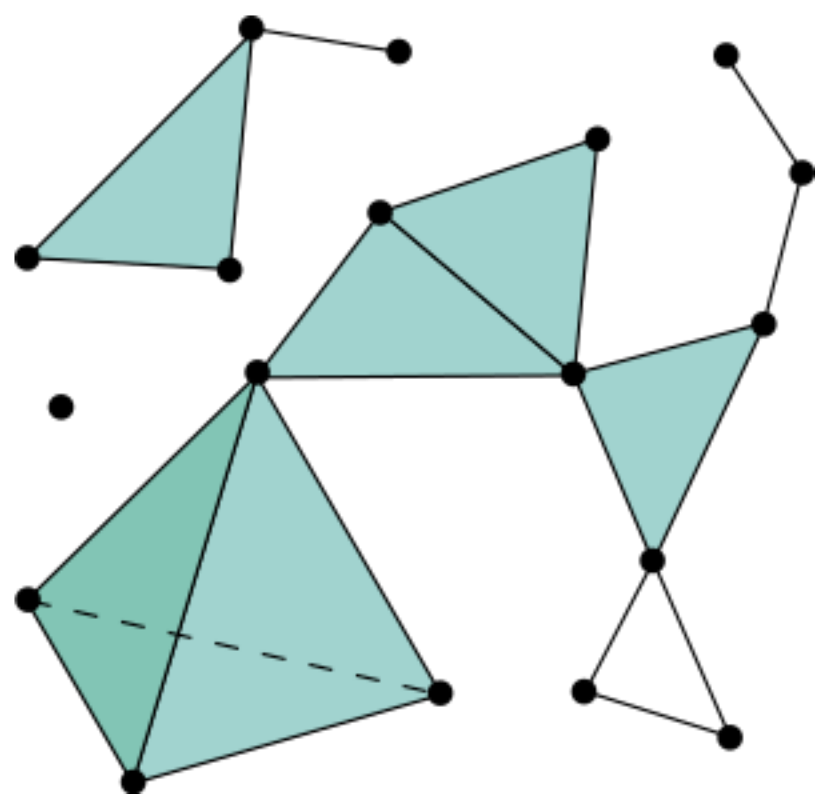
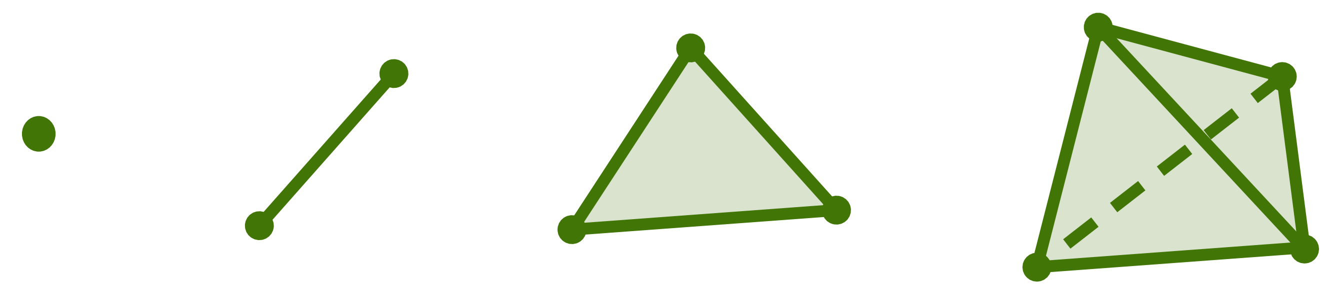


Simplicial Convolutions — an edge case

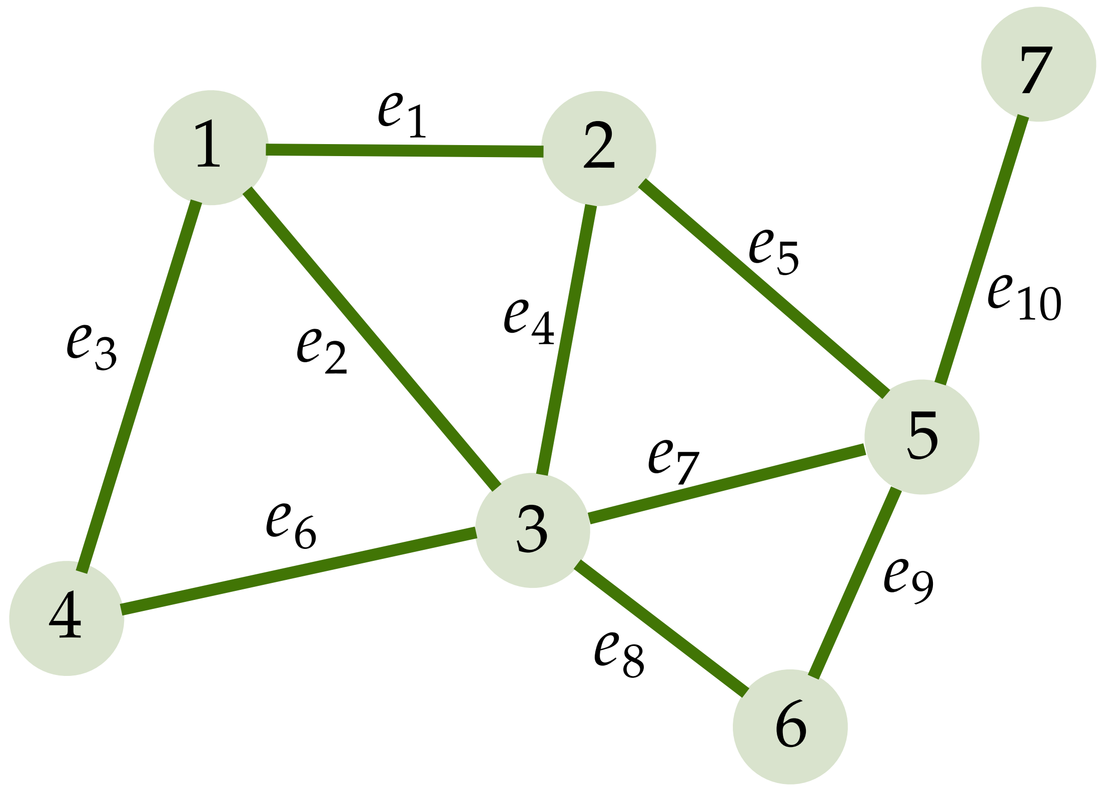
— Analysis of edge data (flows), difference vs. node data
— Convolutions on edges: Spatial; Spectral
— Processing and learning: Filters, NNs, …, GPs, …

Maosheng Yang

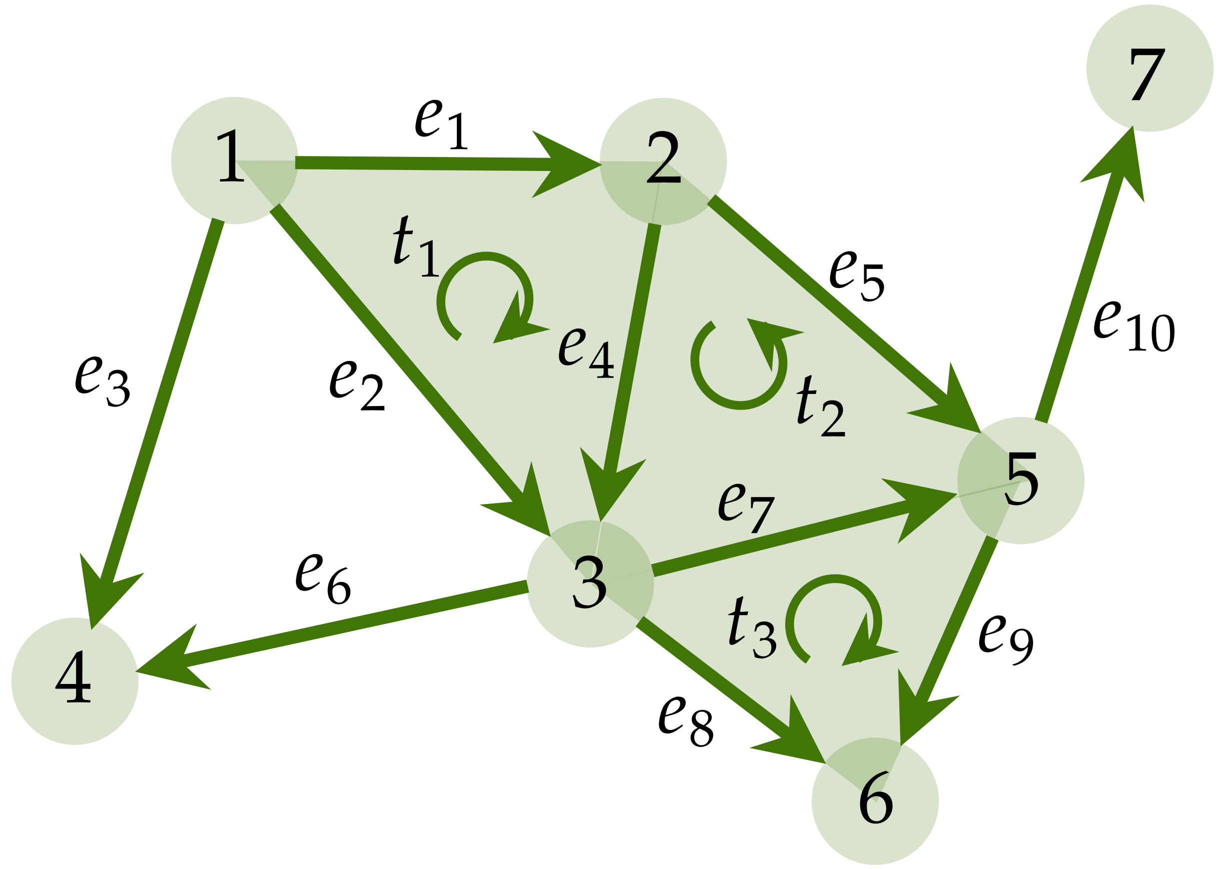
Graphs vs Simplicial 2-Complexes



0-, 1-, 2-, 3-simplices



Graph = Simplicial 1-complex

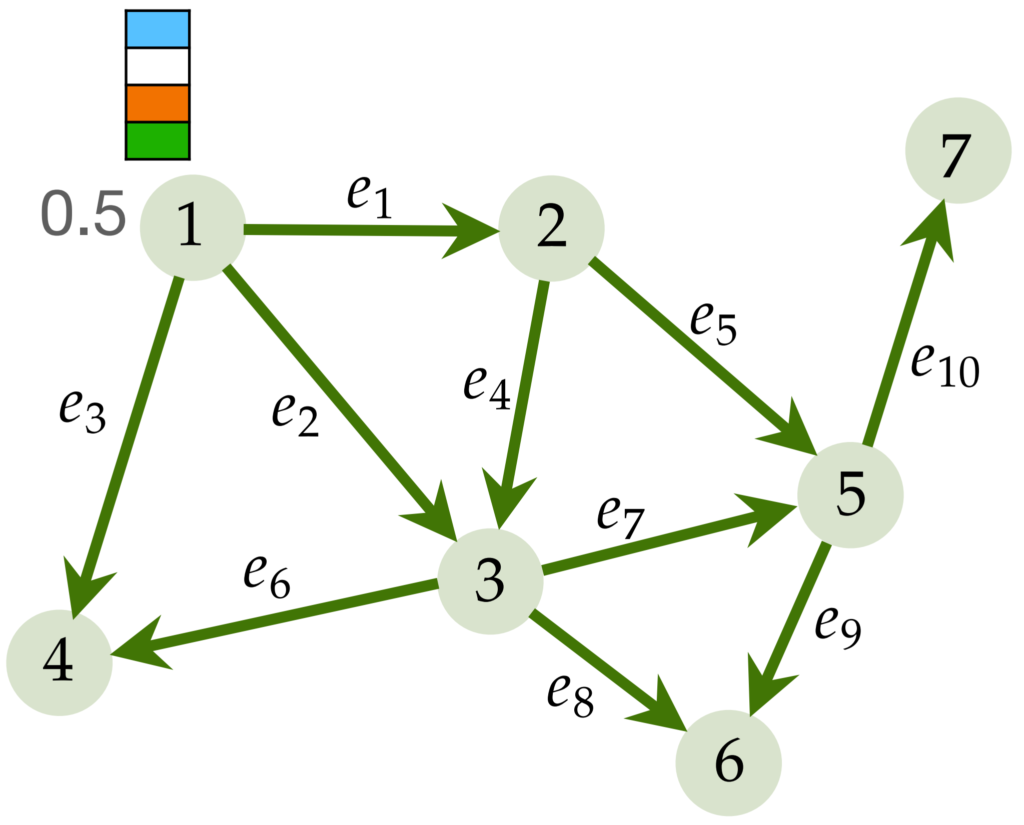


Simplicial 2-complex

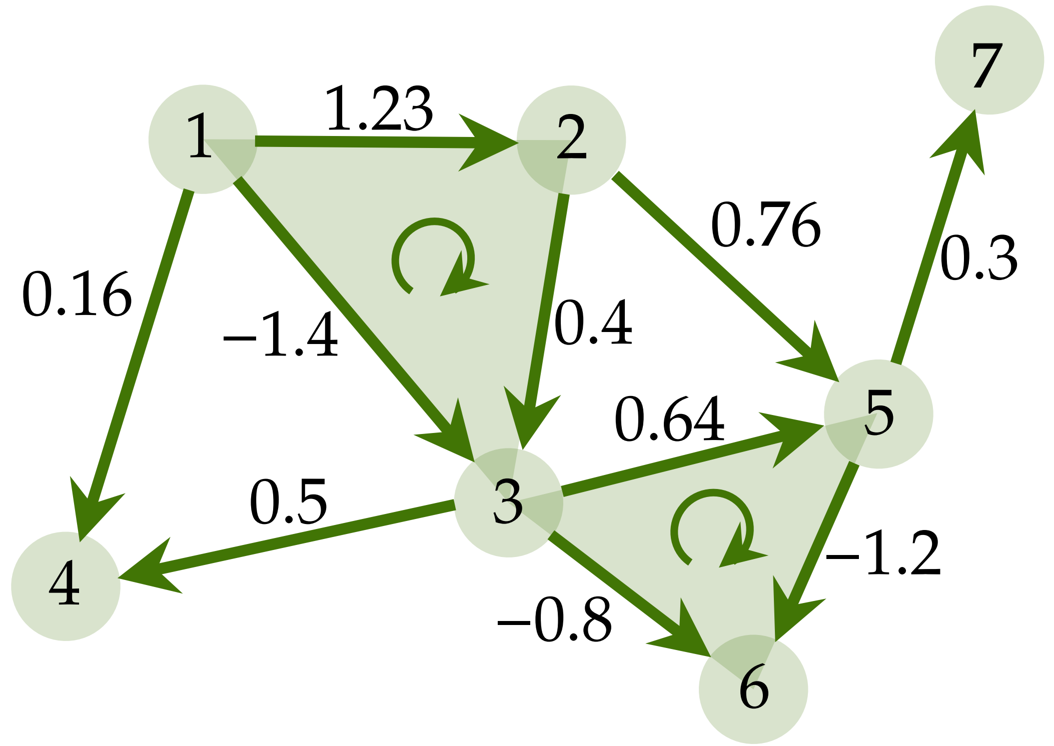
- Oriented simplices (equivalence class of permutations)

Simplicial Signals

Signals on nodes, edges, triangles, …



Node data

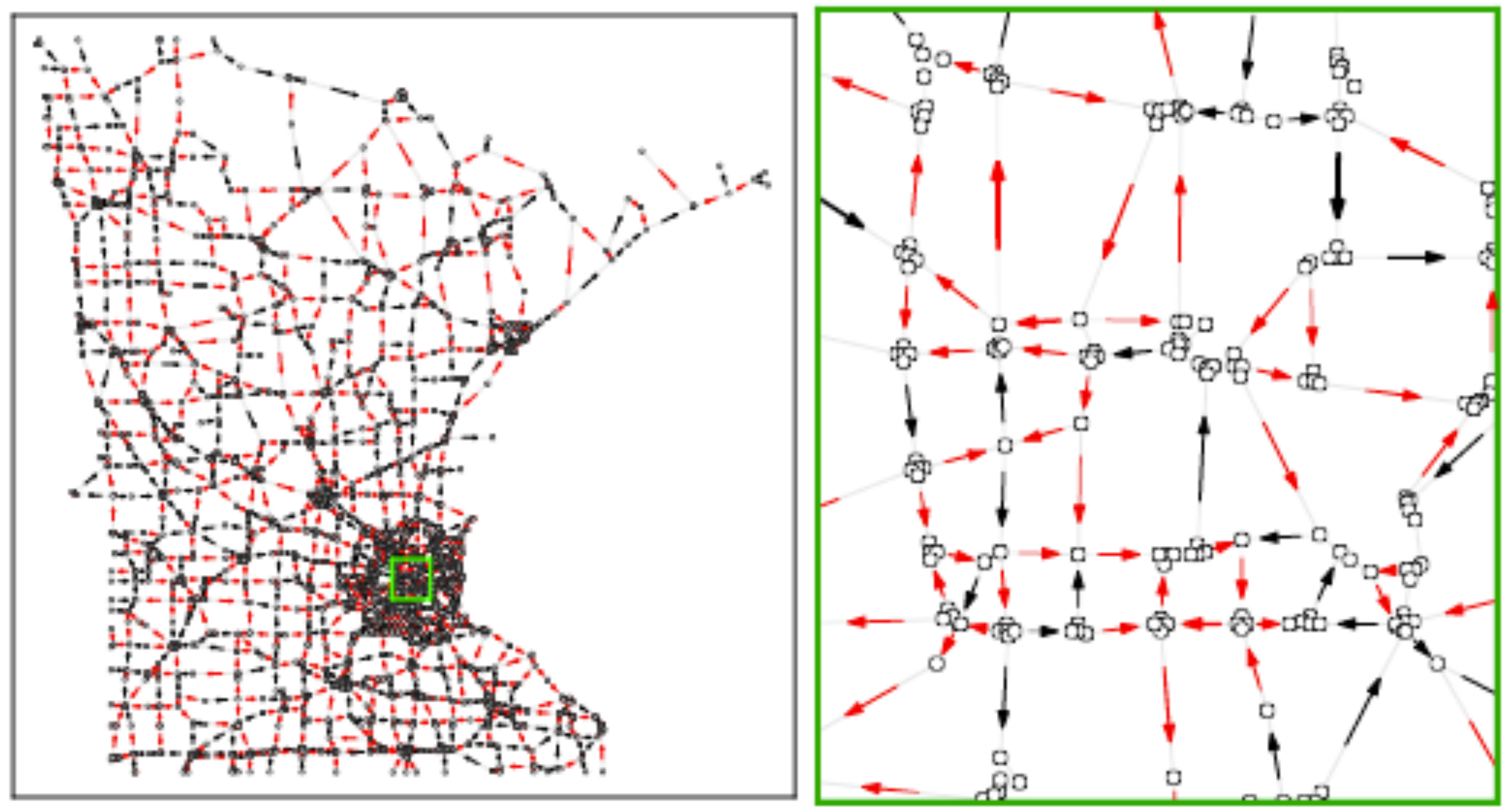


Edge flows

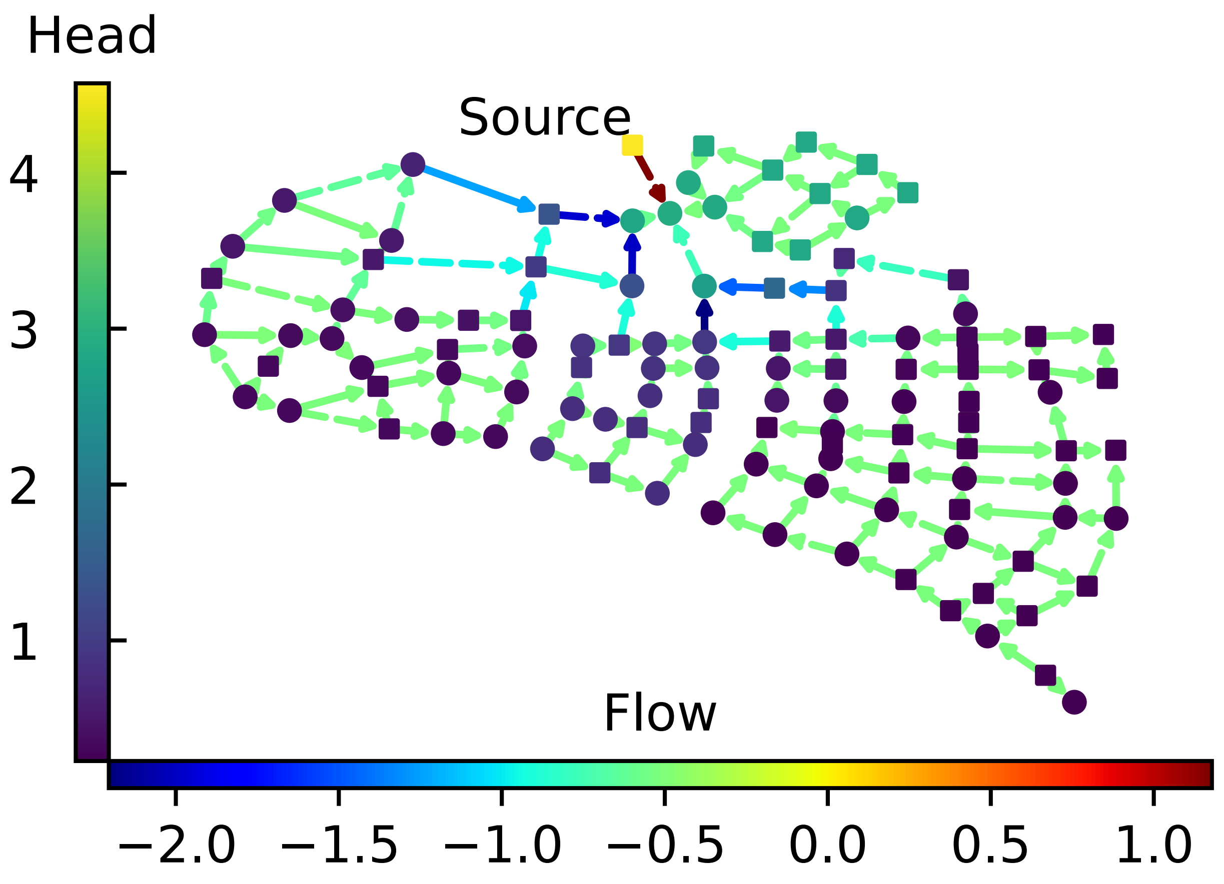
- Alternating property
- Magnitude and sign

- Flow-type data (natural)
  - Physical world: traffic flow, water flow, information flow…
  - Forex: exchange rates
  - Game theory (Candogan et al. 2011)
  - Ranking data (Jiang et al. 2011)
  - Edge-based vector field discretisation (computer graphics)
  - …
- Representation learning

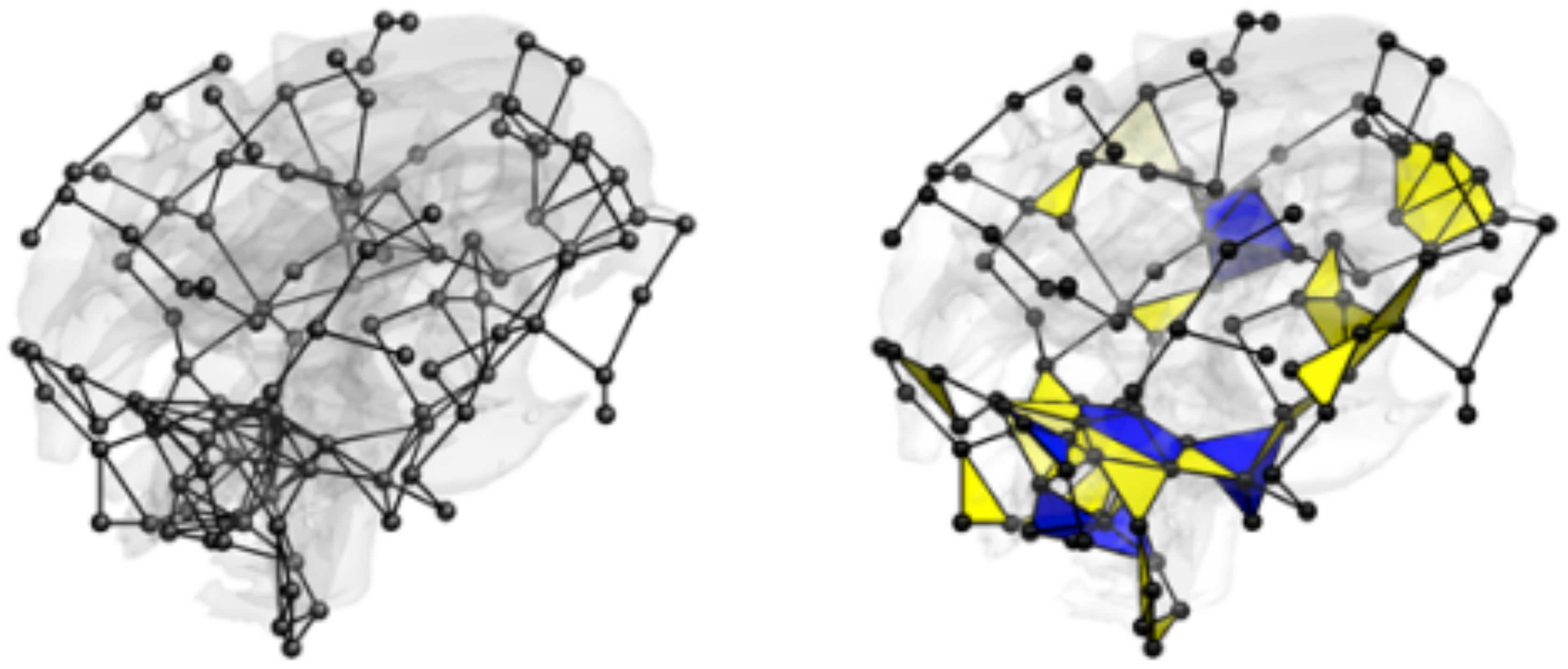
Simplicial complexes and Data in real world



Traffic flows (Jia et al. 2019)

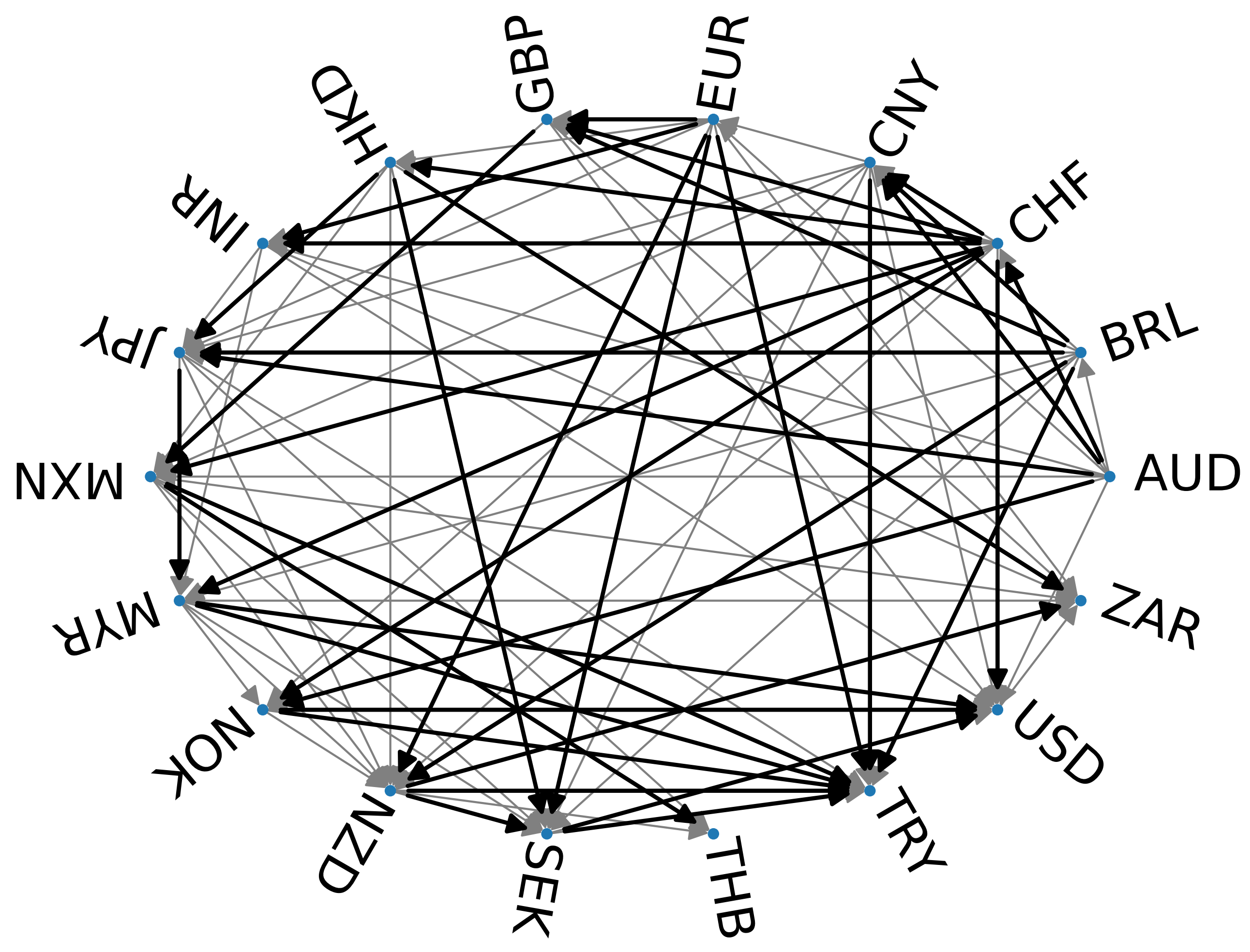


Water flows (Yang et al. 2023)



Neuroscience (Anand et al. 2023):

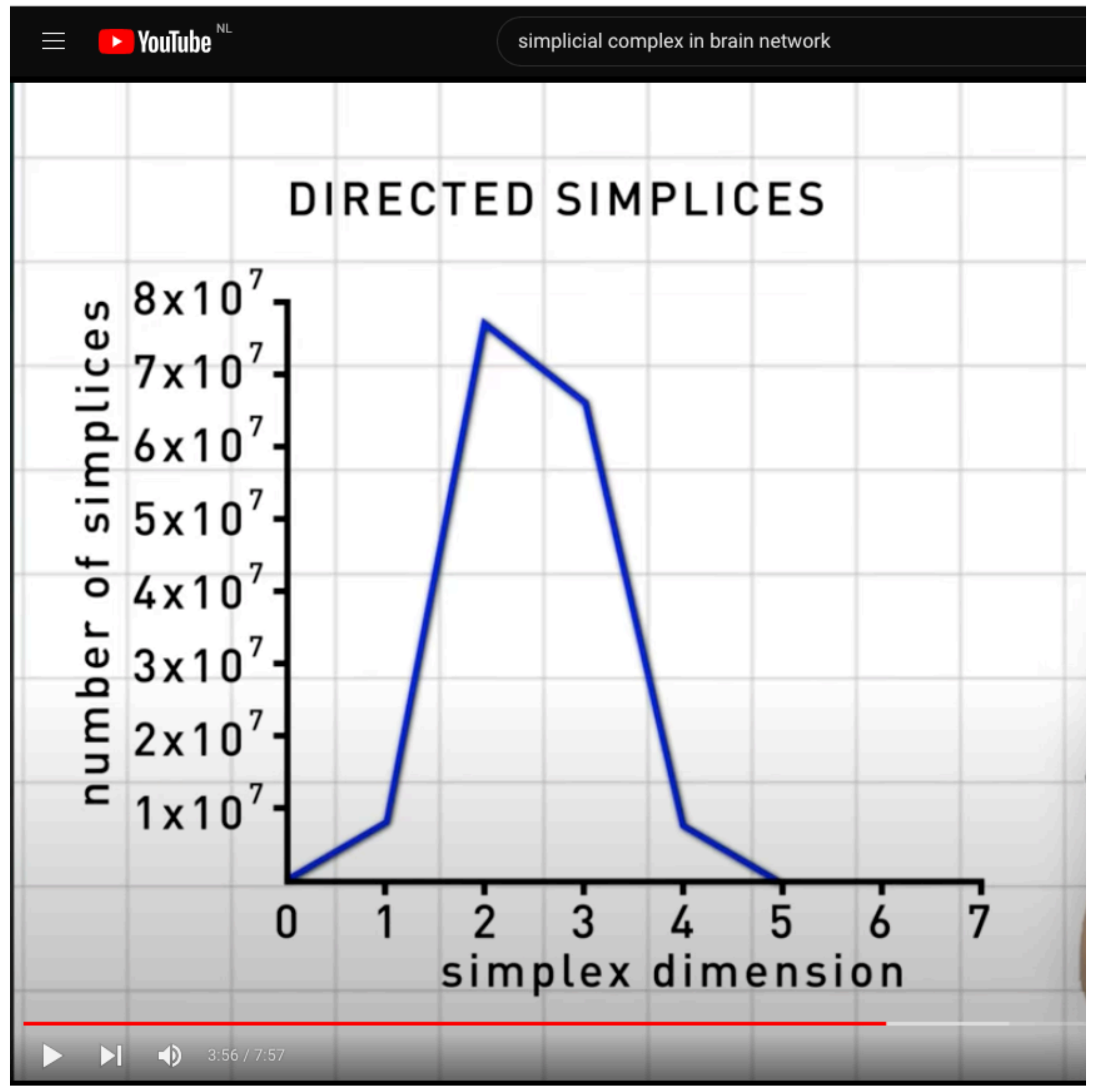
1. Firing of neurons
2. Activation of multiple brain regions



Foreign currency exchange (Jiang et al. 2011)

Others:
- Currents/Voltage in electric circuits/grid
- Game theory (Candogan et al. 2011)
- Ranking theory (Jiang et al. 2011)
- Information flows

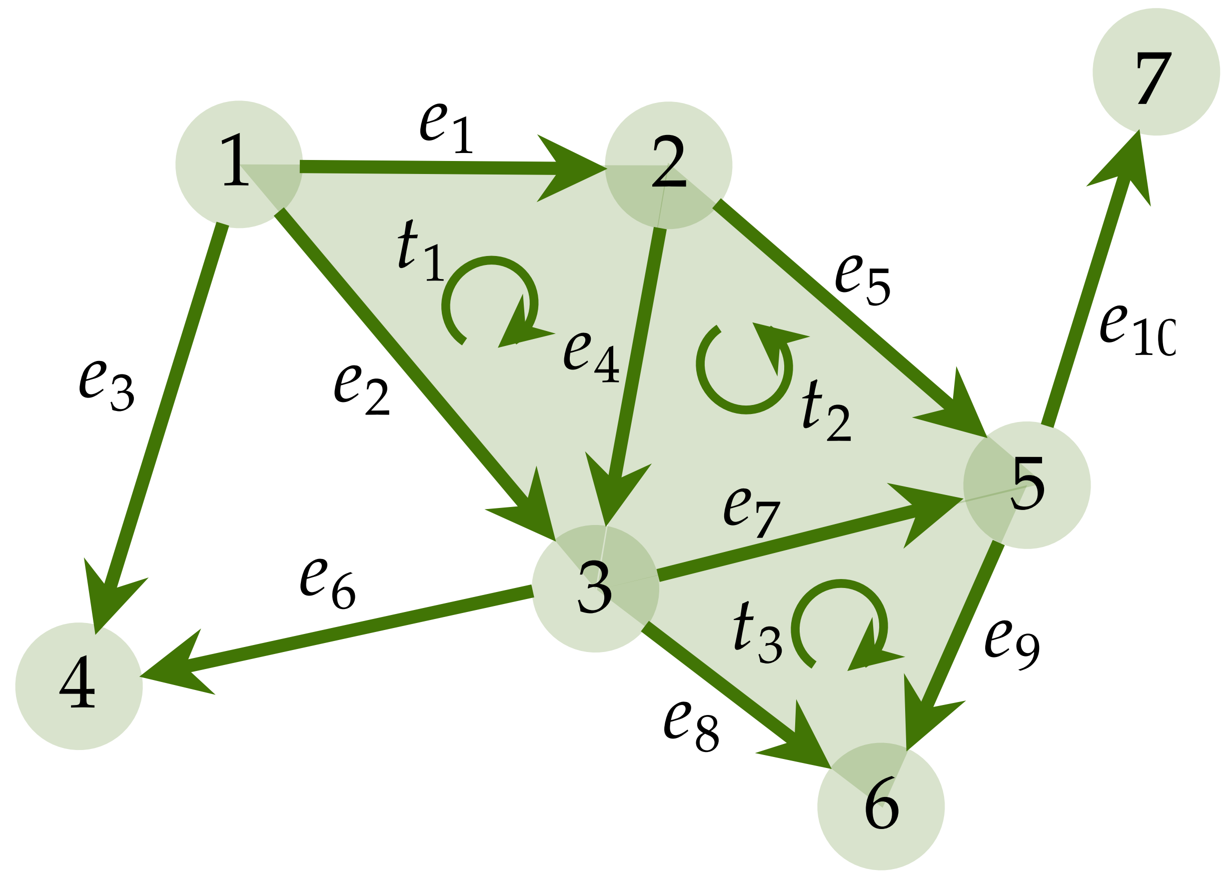
…
- Discrete vector fields



Your Mind Is Eight-Dimensional - Your Brain as Math Part 3 | Infinite Series

Algebraic reps. of simplicial 2-complex

Incidences & Laplacians



Node-to-Edge

Edge-to-Faces

$$\mathbf{B}_1 = \begin{array}{c} \\ 1\\2\\3\\4\\5\\6\\7 \end{array}\begin{array}{c} \begin{array}{cccccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 & e_9 & e_{10} \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & -1 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}, \quad \mathbf{B}_2 = \begin{array}{c} \\ e_1\\e_2\\e_3\\e_4\\e_5\\e_6\\e_7\\e_8\\e_9\\e_{10} \end{array}\begin{array}{c} \begin{array}{ccc} t_1 & t_2 & t_3 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \end{array}$$

Graph Laplacian: $\mathbf{L}_0 = \mathbf{B}_1 \mathbf{B}_1^\top$

1-Hodge Laplacian: $\mathbf{L}_1 = \mathbf{B}_1^\top \mathbf{B}_1 + \mathbf{B}_2 \mathbf{B}_2^\top := \mathbf{L}_{1,d} + \mathbf{L}_{1,u}$

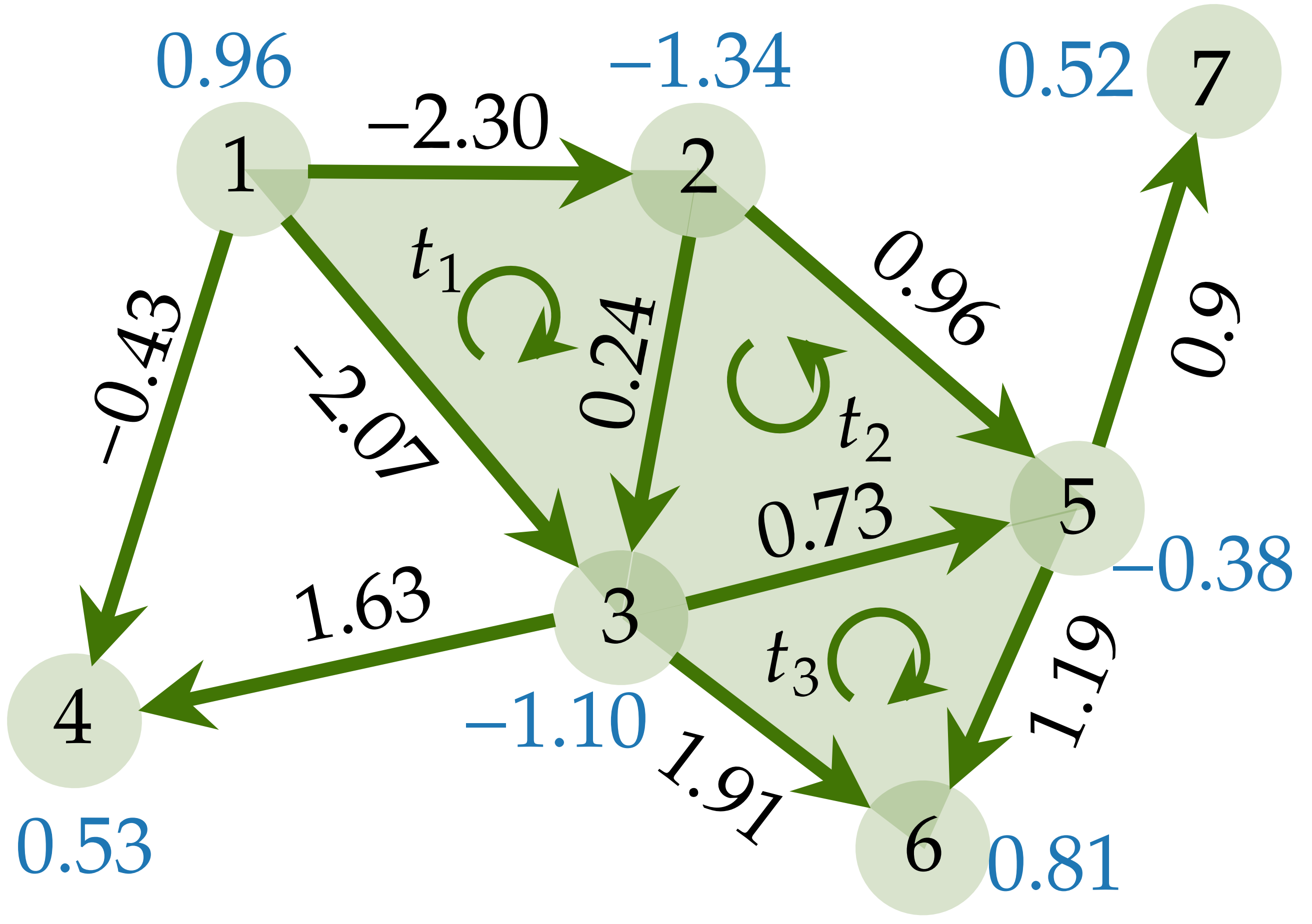
Up

Down

Incidence & Laplacians

1st and 2nd order Discrete Derivatives

- Node signal $\mathbf{v}$
- Edge flows $\mathbf{f}$



Gradient of node signal: $[\mathbf{f}_{\mathrm{G}}]_{[i,j]} = [\mathbf{B}_1^\top \mathbf{v}]_{[i,j]} = [\mathbf{v}]_j - [\mathbf{v}]_i$

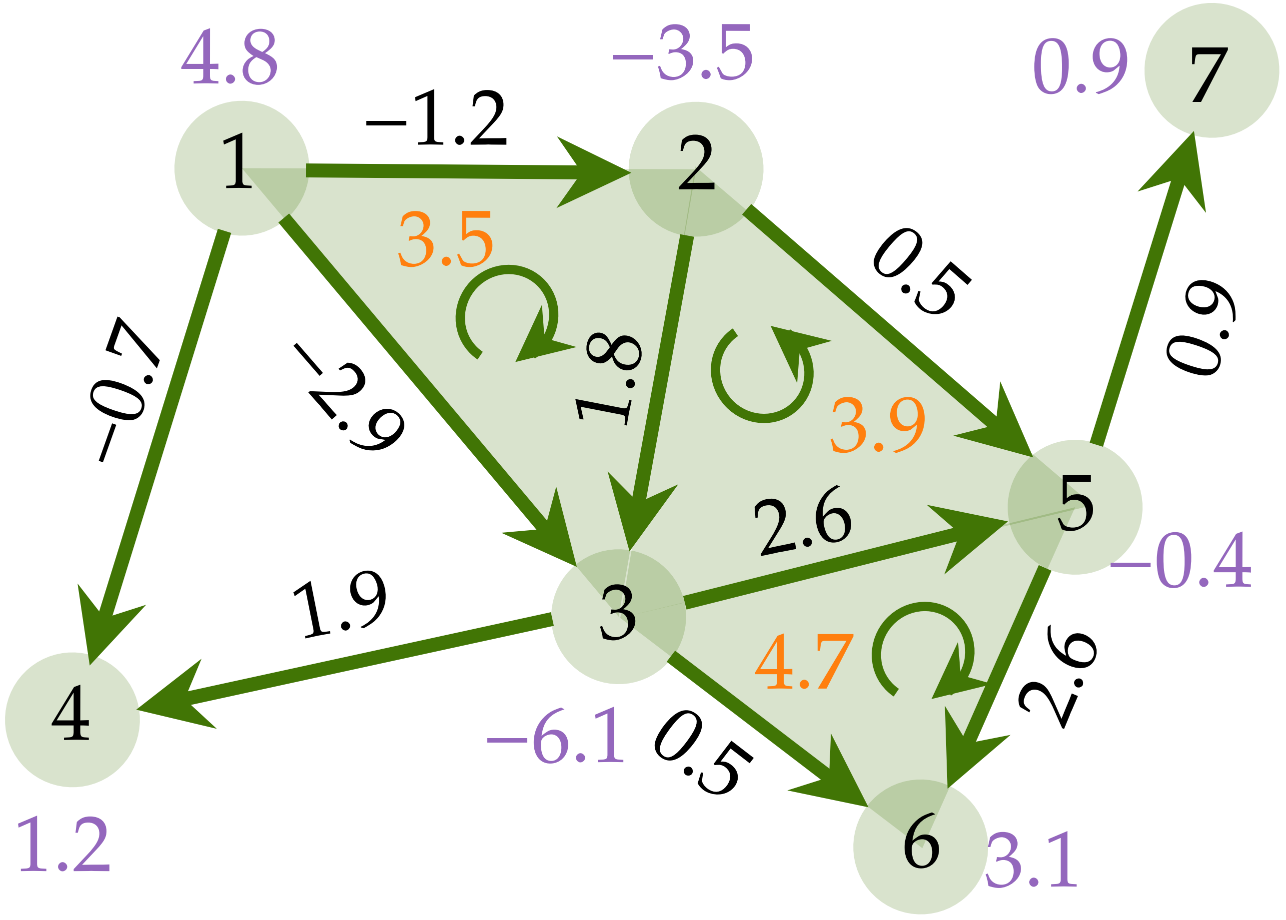
Divergence of edge flows: $[\mathbf{B}_1 \mathbf{f}]_{[i]} = \sum_{j<i} \mathbf{f}_{[j,i]} - \sum_{i<k} \mathbf{f}_{[i,k]}$

Curl of edge flows: $[\mathbf{B}_2^\top \mathbf{f}]_t = \mathbf{f}_{[i,j]} + \mathbf{f}_{[j,k]} - \mathbf{f}_{[i,k]}$, for $t = [i,j,k]$

$$[\mathbf{B}_1^\top \mathbf{v}]_{[1,2]} = -1.34 - 0.96 = -2.30$$

Incidence & Laplacians

1st and 2nd order Discrete Derivatives



Gradient of node signal: $\mathbf{B}_1^\top \mathbf{v}$ $\quad [\mathbf{f}_G]_{[i,j]} = [\mathbf{v}]_j - [\mathbf{v}]_i$

Divergence of edge flows: $[\mathbf{B}_1\mathbf{f}]_{[i]} = \sum_{j<i} \mathbf{f}_{[j,i]} - \sum_{i<k} \mathbf{f}_{[i,k]}$

Net-flow = in_flow - out_flow

Curl of edge flows: $[\mathbf{B}_2^\top \mathbf{f}]_t = \mathbf{f}_{[i,j]} + \mathbf{f}_{[j,k]} - \mathbf{f}_{[i,k]}$, for $t = [i,j,k]$

Net-circulation in triangles

$$[\mathbf{B}_1\mathbf{f}]_5 = 0.5 + 2.6 - (0.9 + 2.6) = -0.4$$

Laplacians = Grad Div + Curl* Curl

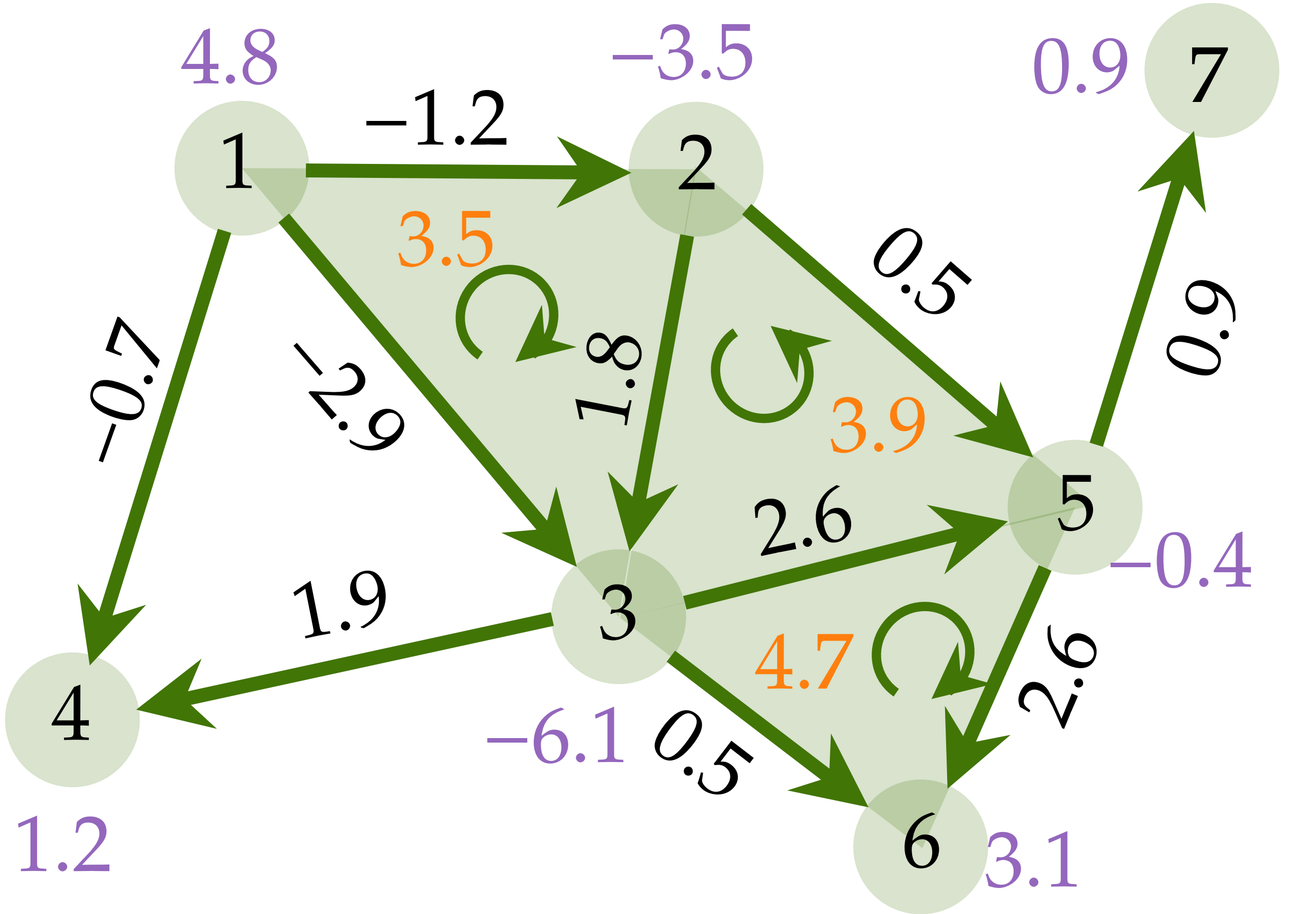
Hodge Laplacian: $\mathbf{L}_1 = \mathbf{B}_1^\top \mathbf{B}_1 + \mathbf{B}_2 \mathbf{B}_2^\top$

$$[\mathbf{B}_2^\top \mathbf{f}]_{[1,2,3]} = -1.2 + 1.8 - (-2.9) = 3.5$$

$$\Delta_1 = \nabla(\nabla\cdot) + \nabla\times(\nabla\times)$$

Incidence & Laplacians

1st and 2nd order Discrete Derivatives



Gradient of node signal: $\mathbf{B}_1^\top \mathbf{v}$ $\quad [\mathbf{f}_G]_{[i,j]} = [\mathbf{v}]_j - [\mathbf{v}]_i$

Divergence of edge flows: $[\mathbf{B}_1 \mathbf{f}]_{[i]} = \sum_{j<i} \mathbf{f}_{[j,i]} - \sum_{i<k} \mathbf{f}_{[i,k]}$

Net-flow = in_flow - out_flow

Curl of edge flows: $[\mathbf{B}_2^\top \mathbf{f}]_t = \mathbf{f}_{[i,j]} + \mathbf{f}_{[j,k]} - \mathbf{f}_{[i,k]}$, for $t = [i, j, k]$

Net-circulation in triangles

$$[\mathbf{B}_1 \mathbf{f}]_5 = 0.5 + 2.6 - (0.9 + 2.6) = -0.4$$

$$[\mathbf{B}_2^\top \mathbf{f}]_{[1,2,3]} = -1.2 + 1.8 - (-2.9) = 3.5$$

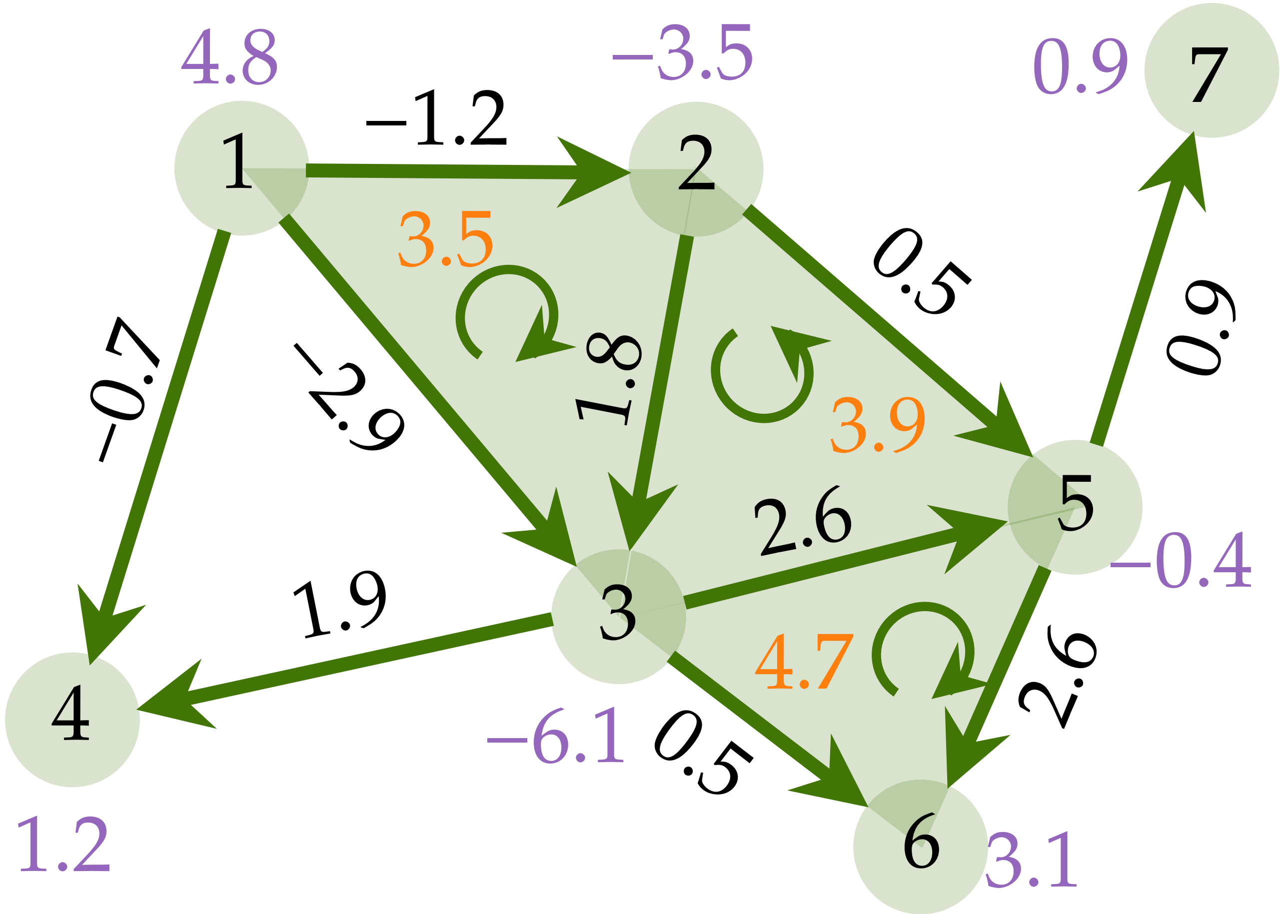
Laplacians = Grad Div + Curl* Curl

Hodge Laplacian: $\mathbf{L}_1 = \mathbf{B}_1^\top \mathbf{B}_1 + \mathbf{B}_2 \mathbf{B}_2^\top$

$$\Delta_1 = \nabla(\nabla \cdot) + \nabla \times (\nabla \times)$$

Incidence & Laplacians

1st and 2nd order Discrete Derivatives



Gradient of node signal: $\mathbf{B}_1^\top \mathbf{v}$ $\quad [\mathbf{f}_G]_{[i,j]} = [\mathbf{v}]_j - [\mathbf{v}]_i$

Divergence of edge flows: $[\mathbf{B}_1 \mathbf{f}]_{[i]} = \sum_{j<i} \mathbf{f}_{[j,i]} - \sum_{i<k} \mathbf{f}_{[i,k]}$

Net-flow = in_flow - out_flow

Curl of edge flows: $[\mathbf{B}_2^\top \mathbf{f}]_t = \mathbf{f}_{[i,j]} + \mathbf{f}_{[j,k]} - \mathbf{f}_{[i,k]}$, for $t = [i, j, k]$

Net-circulation in triangles

$$[\mathbf{B}_1 \mathbf{f}]_5 = 0.5 + 2.6 - (0.9 + 2.6) = -0.4$$

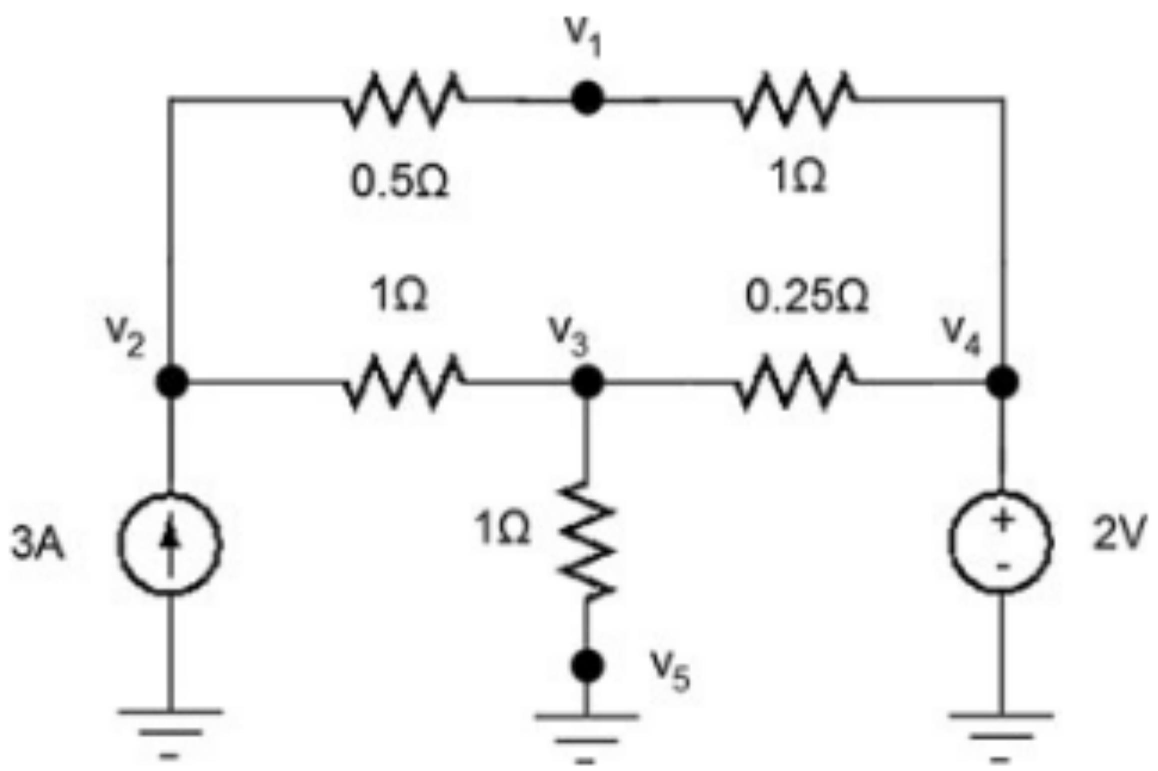
$$[\mathbf{B}_2^\top \mathbf{f}]_{[1,2,3]} = -1.2 + 1.8 - (-2.9) = 3.5$$

Hodge Laplacians = Grad Div + Curl* Curl

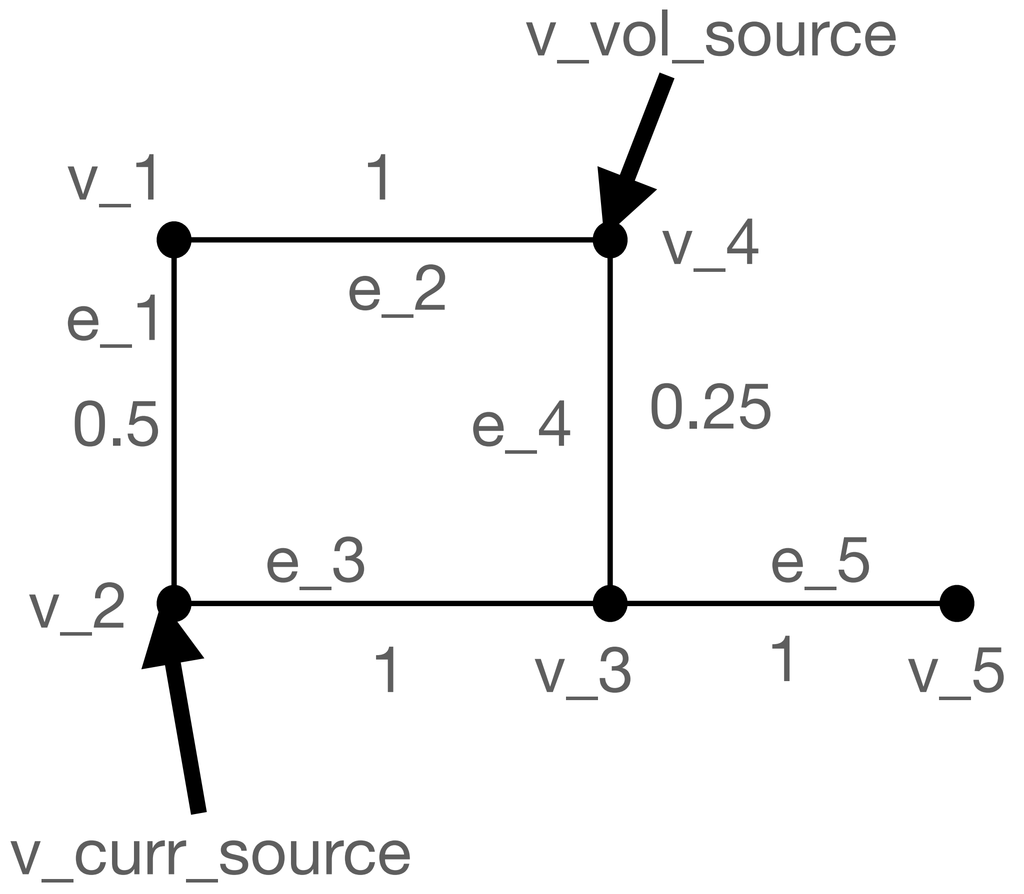
Hodge Laplacian: $\mathbf{L}_1 = \mathbf{B}_1^\top \mathbf{B}_1 + \mathbf{B}_2 \mathbf{B}_2^\top$

$\Delta_1 = \nabla(\nabla \cdot) + \nabla \times (\nabla \times)$

A Circuit toy example



(Grady et al. 2010)



$\mathbf{v} \in \mathbb{R}^{\mathcal{N}}$: Electric potential on nodes

$\mathbf{f}_{Vol} = \mathbf{B}_1^\top \mathbf{v}$: (Kirchhoff's voltage law)

$\mathbf{f}_{currents} = \mathbf{G}^{-1}\mathbf{f}_{Vol}$: currents (Ohm's law)

Diagonal resistance/conductance

Kirchhoff's current law: $\mathbf{B}_1 \mathbf{f}_{currents} = \mathbf{0}$

Or $\mathbf{B}_1 \mathbf{f}_{currents} + \mathbf{v}_{curr\,source} = \mathbf{0}$

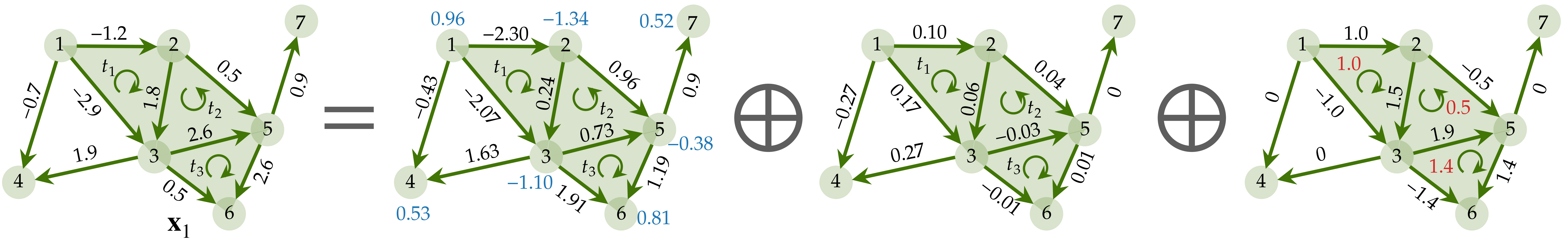
$$\mathbf{B}_1 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 1 \\ 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix} \qquad \mathbf{v}_{vol} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ 2 \\ 0 \end{pmatrix}$$

$$\mathbf{B}_1 \mathbf{G}^{-1} \mathbf{B}_1^\top \mathbf{v}_{vol} + \mathbf{v}_{curr\,source} = \mathbf{0}$$

Hodge decomposition

Lovász et al. 2004; Lim et al. 2020

$\mathbb{R}^{N_1} = \text{im}(\mathbf{B}_1^\top) \oplus \ker(\mathbf{L}_1) \oplus \text{im}(\mathbf{B}_2)$

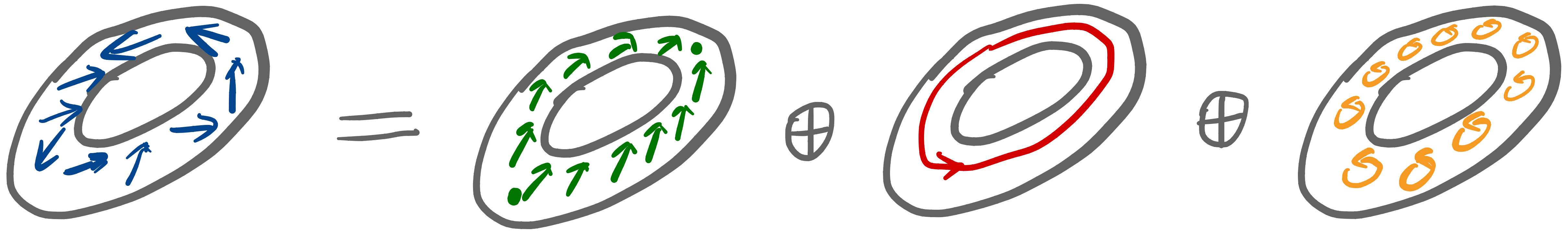


Gradient flow
Curl-free, irrotational

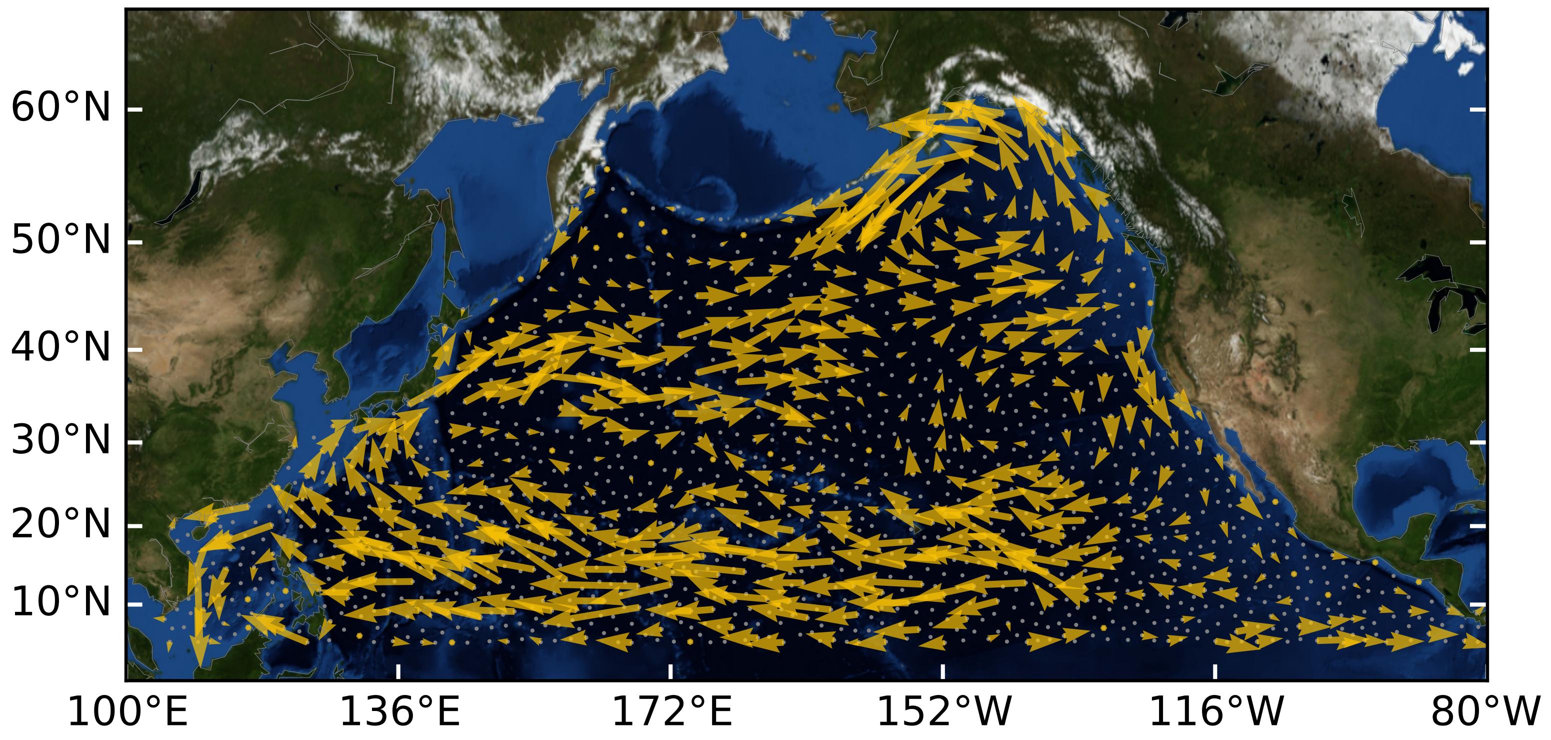
Harmonic flow
Div- and curl-free

Curl flow
Div-free, solenoidal

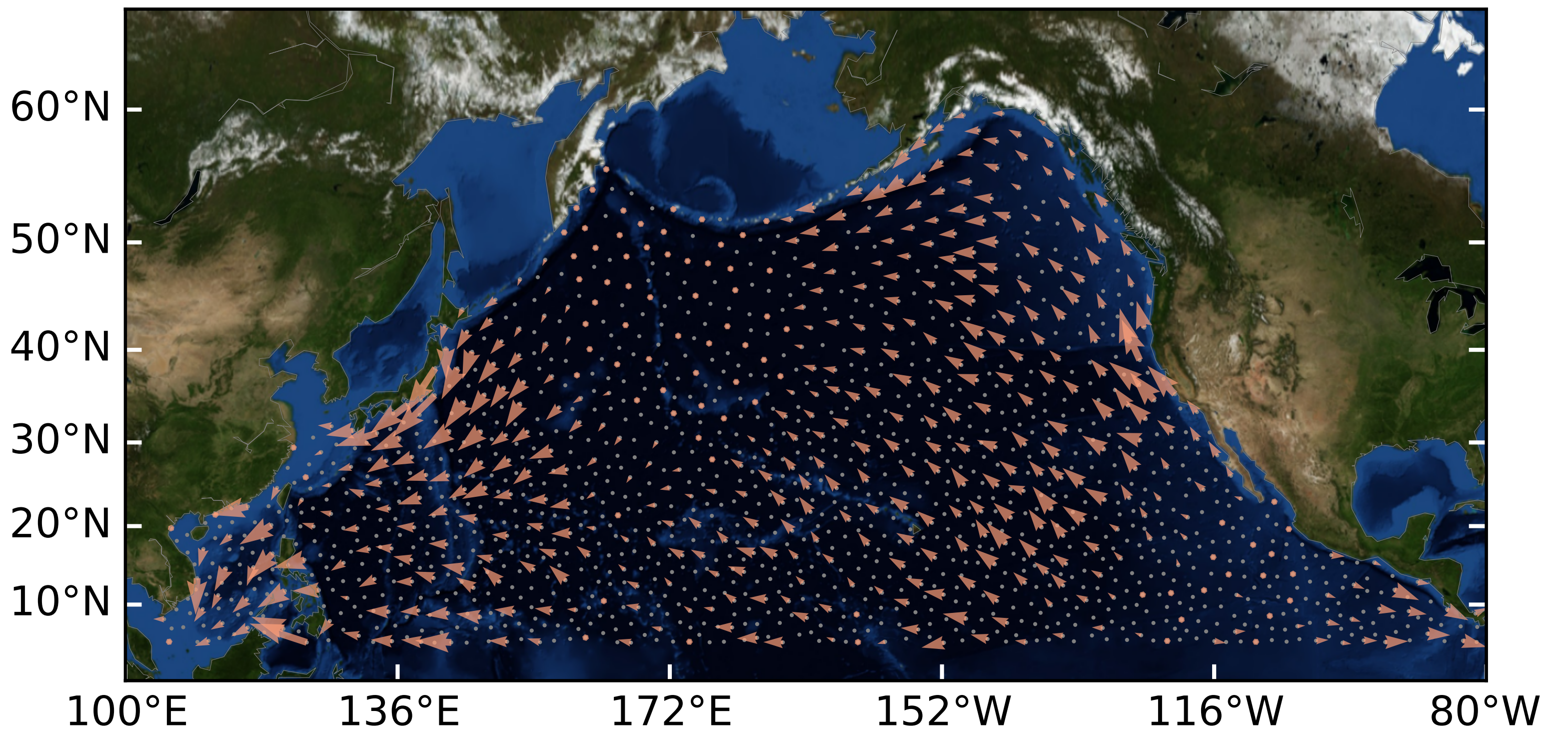
- This holds for any simplex order k
- What is the case for $k = 0$?
- Characteristic decomposition



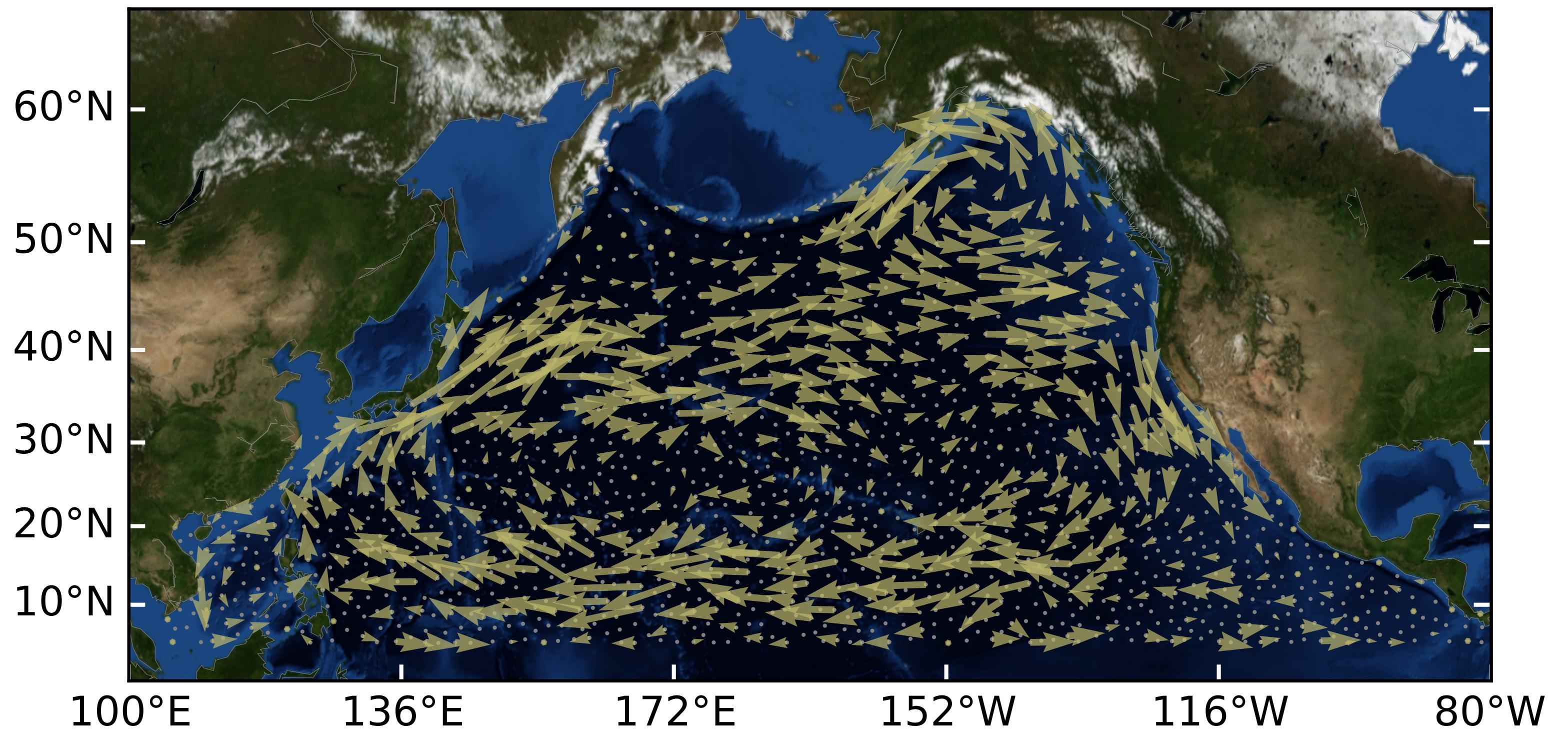
Applications of Hodge decomposition



Ocean currents



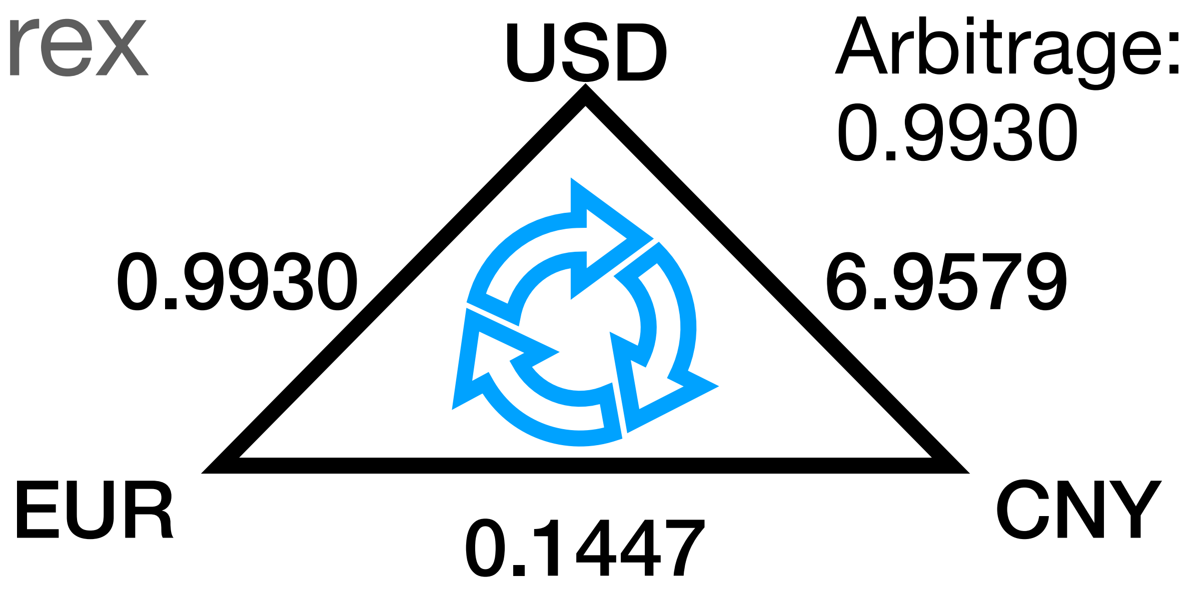
Gradient flow
Curl-free, irrotational



Curl flow
Div-free, solenoidal

Chen, Yu-Chia et al. (2021) "Helmholtzian Eigenmap."

Forex



$r^{a/b} r^{b/c} = r^{a/c}$ Arbitrage-free

$f_{[a,b]} + f_{[b,c]} - f_{[a,c]} = 0$ Curl-free

- Water flows (div-free)
- Electrical currents, voltages

- Brain networks (Anand et al. 2022)
- Game theory (Candogan et al. 2011)
- Ranking problems (Jiang et al. 2011)
- Random walks (Strang et al. 2020)
- …

Eigenspace of $\mathbf{L}_1$ spans Hodge subspaces

- **Nonzero** Eigenspace of down Laplacian spans the gradient space
- **Nonzero** Eigenspace of up Laplacian spans the curl space
- Kernel of Laplacian spans the harmonic space

Simplicial Fourier transform

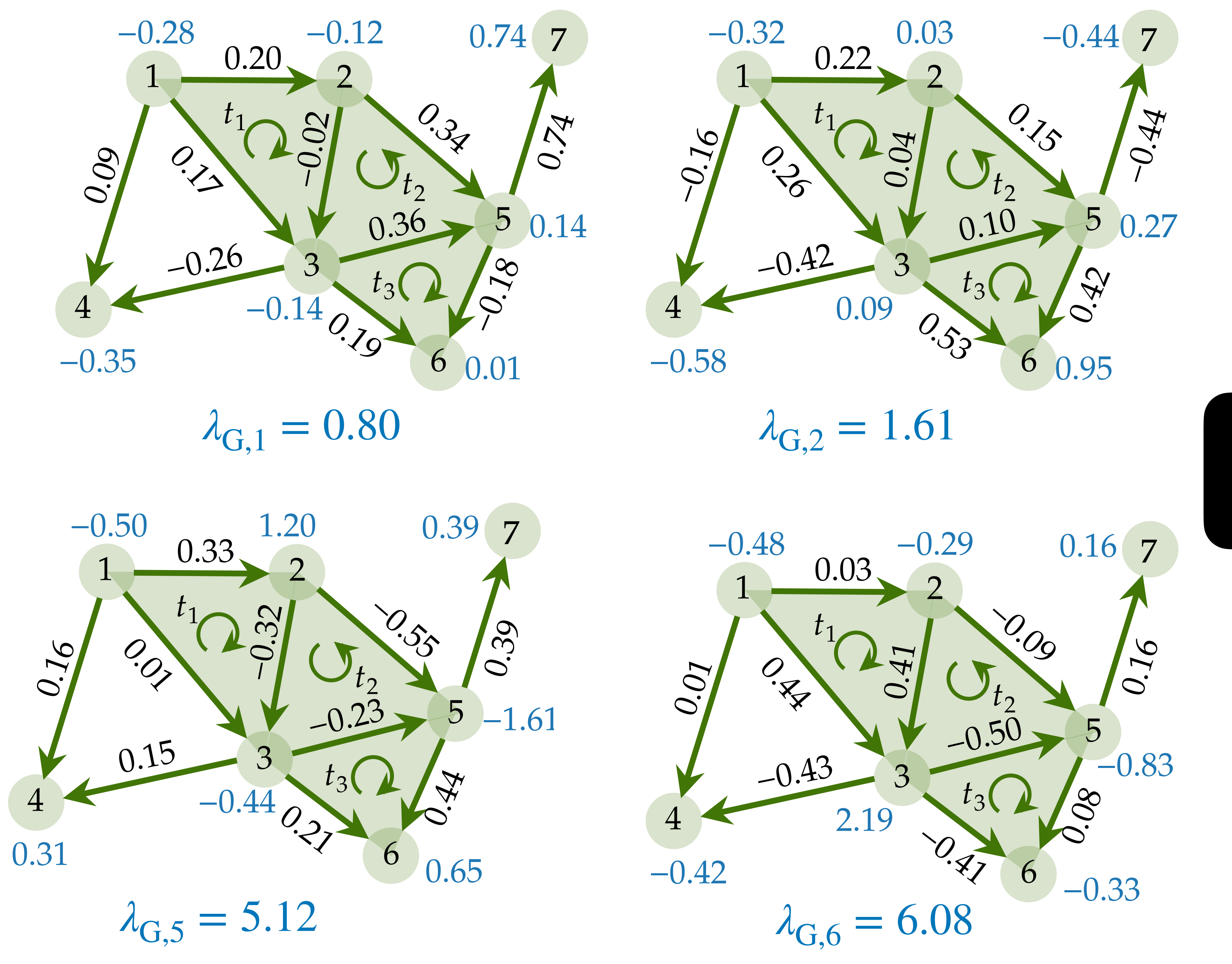
Frequency — eigenvalues

Fourier basis — eigenvectors

$\lambda_G = \|\mathbf{B}_1 \mathbf{u}_G\|_2^2$

Gradient eigenvector

Fourier basis reflecting divergent properties

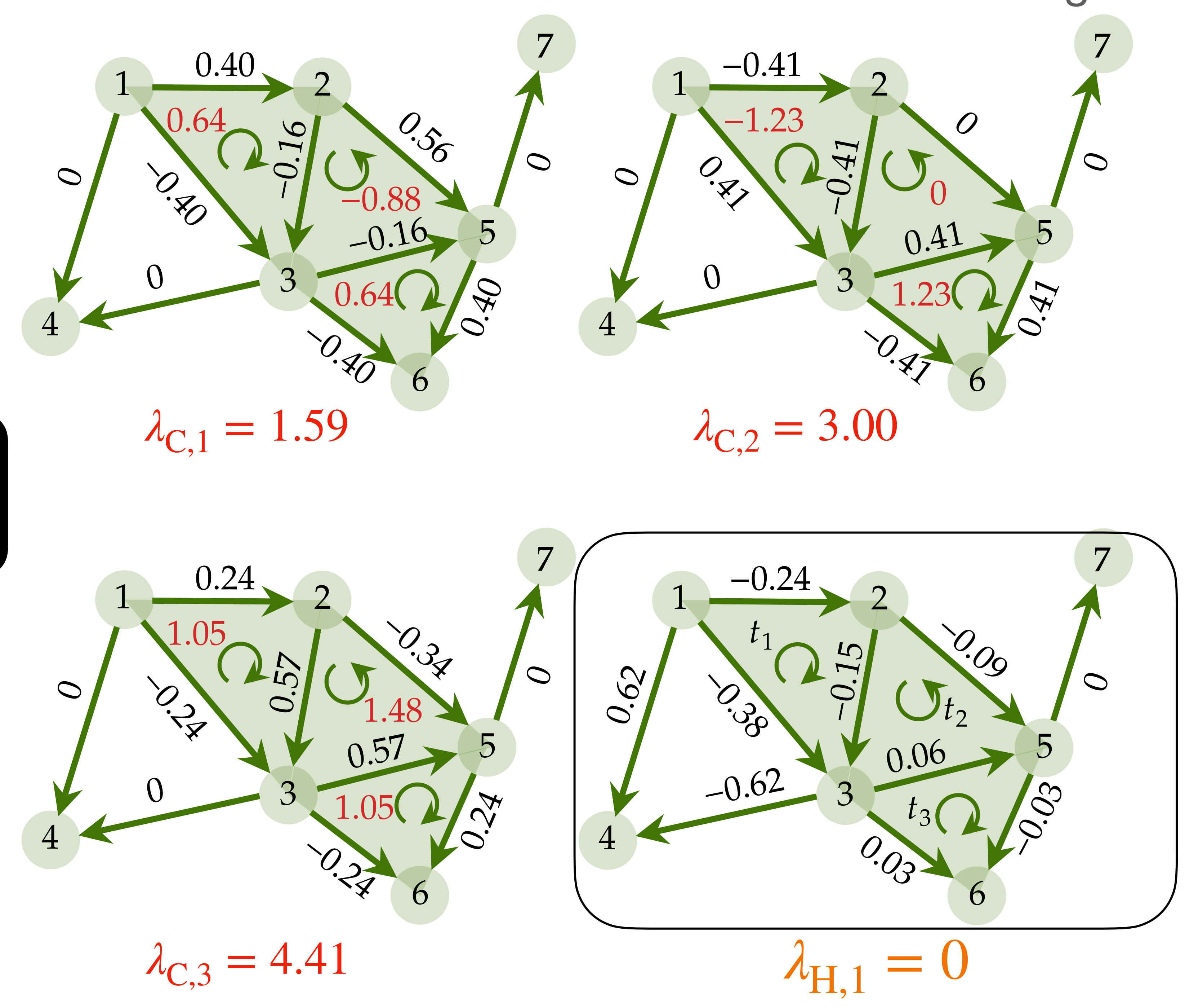


$k = 1$

$\lambda_C = \|\mathbf{B}_2^\top \mathbf{u}_C\|_2^2$

Curl eigenvector

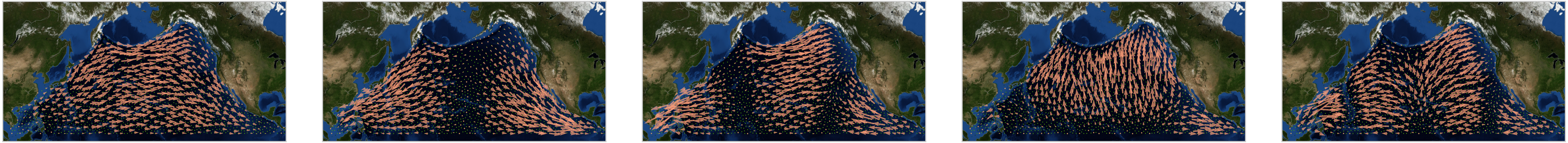
Fourier basis reflecting rotational properties



- $\tilde{\mathbf{x}}_k = \mathbf{U}_k^\top \mathbf{x}_k$
- $\tilde{\mathbf{x}}_k = [\tilde{\mathbf{x}}_{k,H}^\top, \tilde{\mathbf{x}}_{k,G}^\top, \tilde{\mathbf{x}}_{k,C}^\top]$

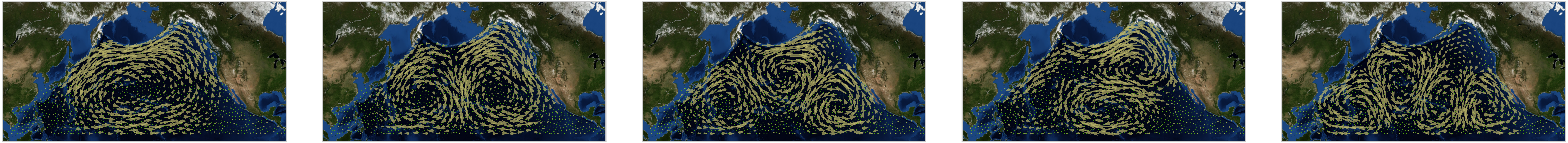
Eigenspace spans Hodge subspaces

- Down Laplacian, its nonzero eigenspace spans the gradient space



λ_G, more divergent

- Up laplacian, its nonzero eigenspace spans the curl space

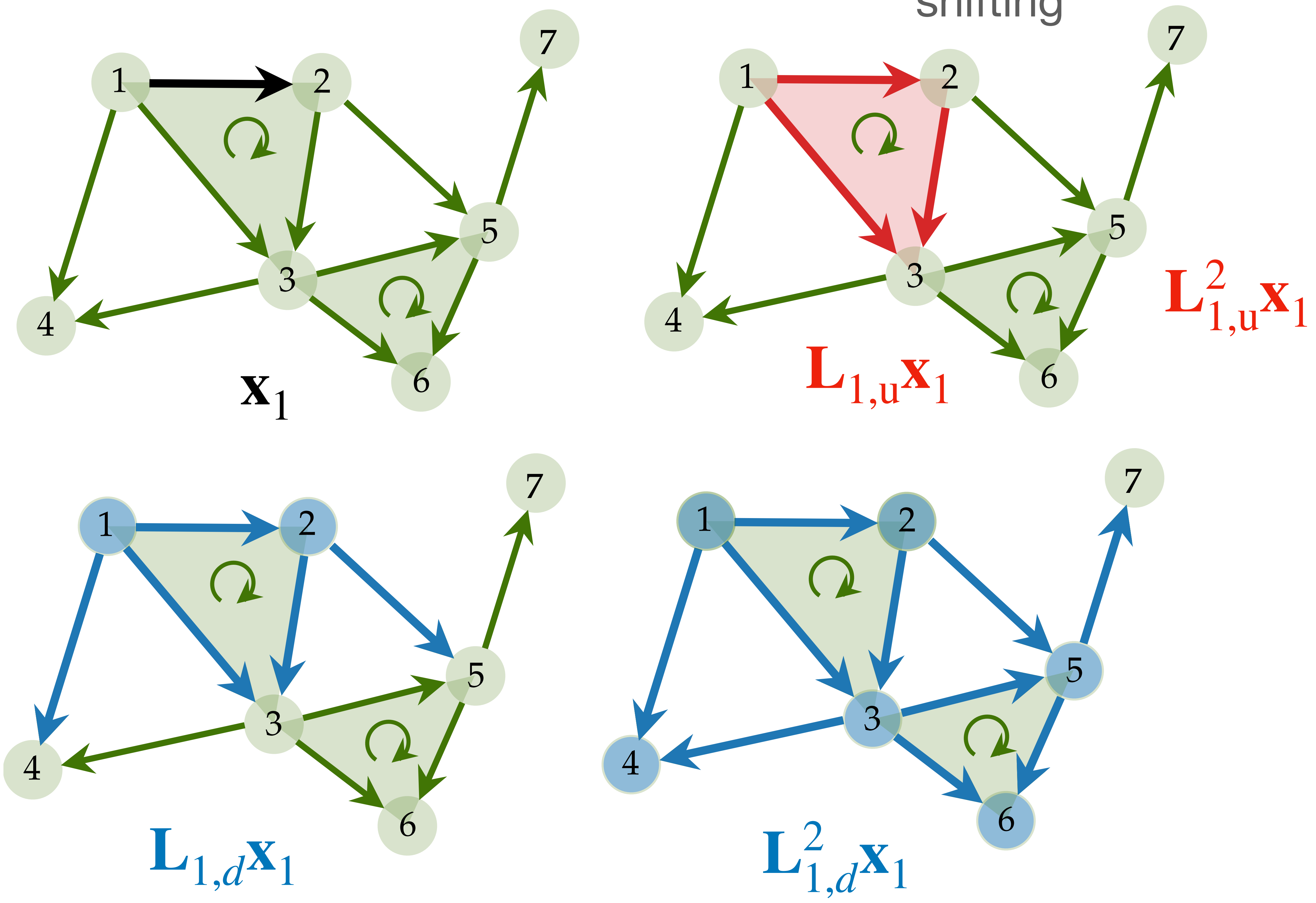


λ_C, more rotational

Edge Convolution

Shift-and-Sum

Simplicial shifting



$$[\mathbf{L}_{1,d}\mathbf{f}]_i = \sum_{j \in \{\mathcal{N}_{1,i} \cup i\}} [\mathbf{L}_{1,d}]_{ij}[\mathbf{f}]_j$$

Simplicial locality

Spatial/Topological

Convolutional filter

$$\mathbf{H} := \mathbf{H}(\mathbf{L}_{\mathrm{d}}, \mathbf{L}_{\mathrm{u}}; \boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{k=0}^{K_{\mathrm{d}}} \alpha_k \mathbf{L}_{\mathrm{d}}^k + \sum_{k=0}^{K_{\mathrm{u}}} \beta_k \mathbf{L}_{\mathrm{u}}^k$$

- Efficient, distributed
- Expressive power (Cayley-Hamilton thm)
- Hodge-invariant operator

$$\mathbf{H}_1\mathbf{x}_1 = \mathbf{H}_{1\ \mathrm{im}(\mathbf{B}_1^\top)}\mathbf{x}_{1,\mathrm{G}} + \mathbf{H}_{1\ \mathrm{im}(\mathbf{B}_2)}\mathbf{x}_{1,\mathrm{C}} + \mathbf{H}_{1\ \ker(\mathbf{L}_1)}\mathbf{x}_{1,\mathrm{H}}$$

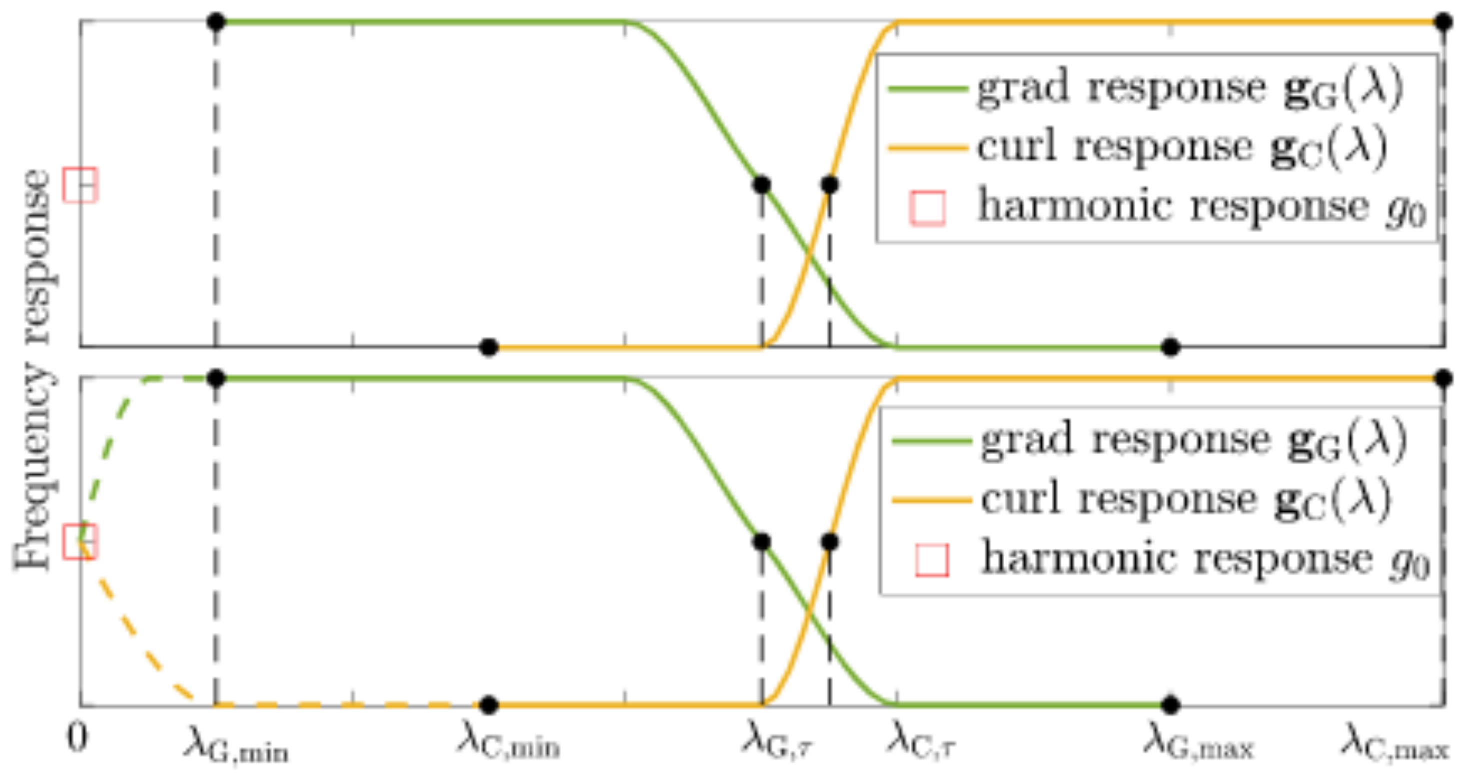
Hodge subspaces are invariant under $\mathbf{H}$

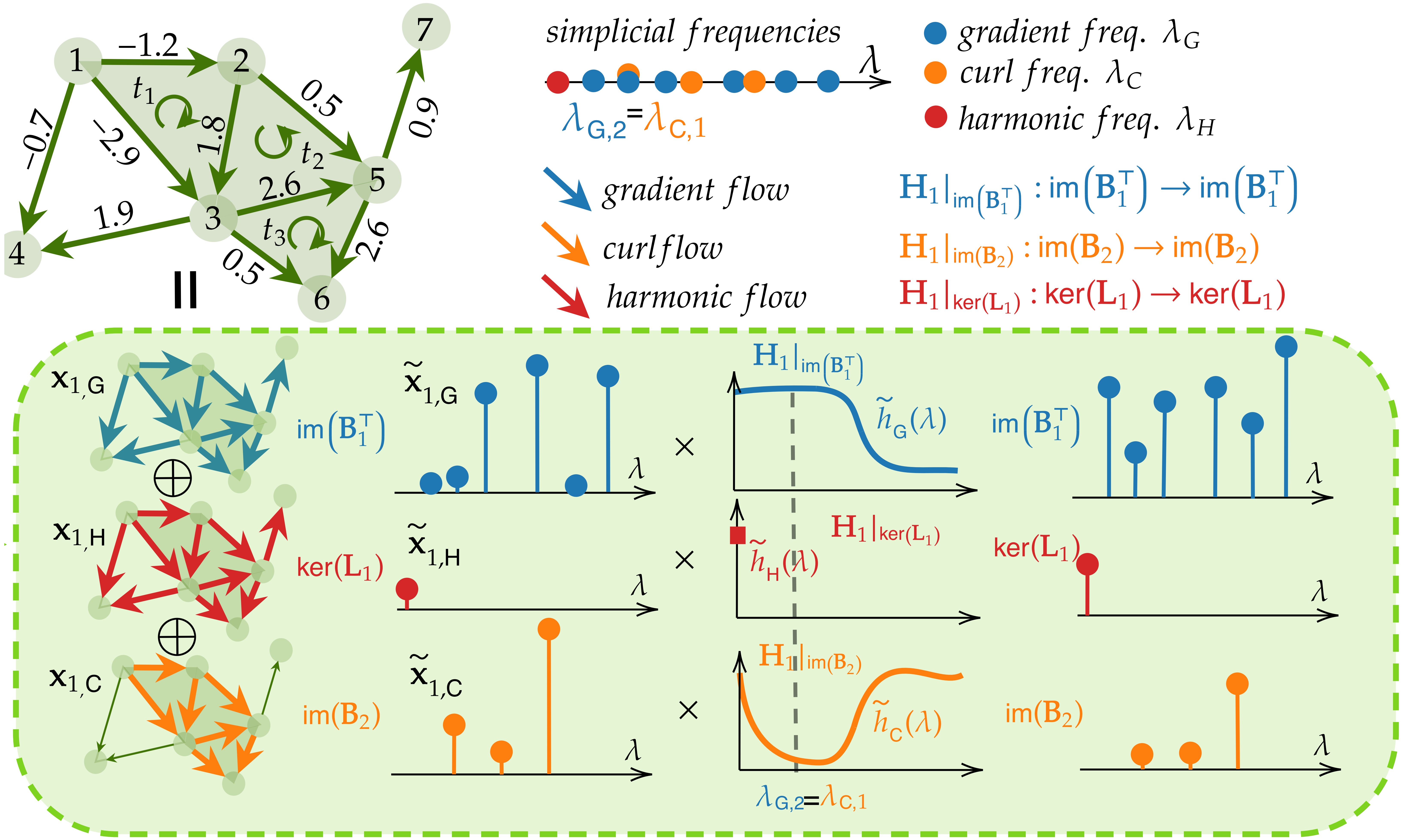
Edge Convolutions on SCs

Pointwise Multiplication at frequencies

Spectral

$$\begin{cases} \tilde{H}_{\mathrm{H}}(\lambda) = h_0, & \text{for } \lambda \in \mathcal{Q}_{\mathrm{H}}, \\ \tilde{H}_{\mathrm{G}}(\lambda) = h_0 + \sum_{k=1}^{K_d} \alpha_k \lambda^k, & \text{for } \lambda \in \mathcal{Q}_{\mathrm{G}}, \\ \tilde{H}_{\mathrm{C}}(\lambda) = h_0 + \sum_{k=1}^{K_u} \beta_k \lambda^k, & \text{for } \lambda \in \mathcal{Q}_{\mathrm{C}} \end{cases}$$

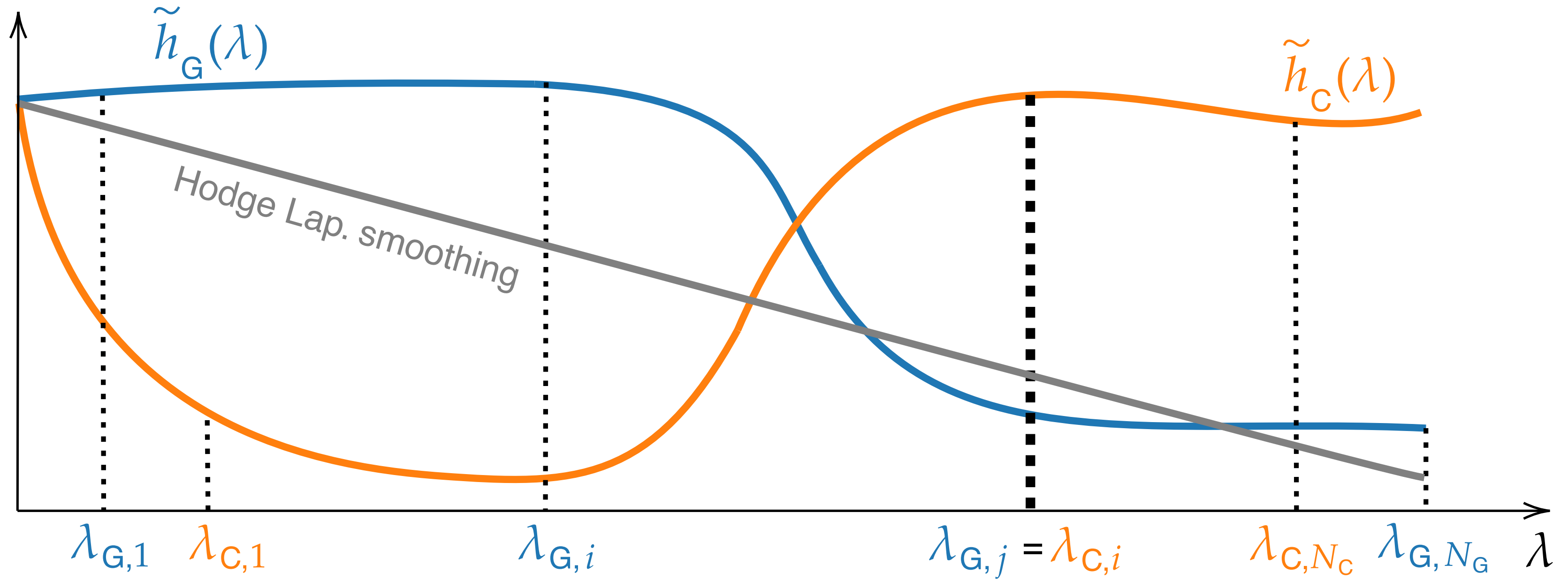




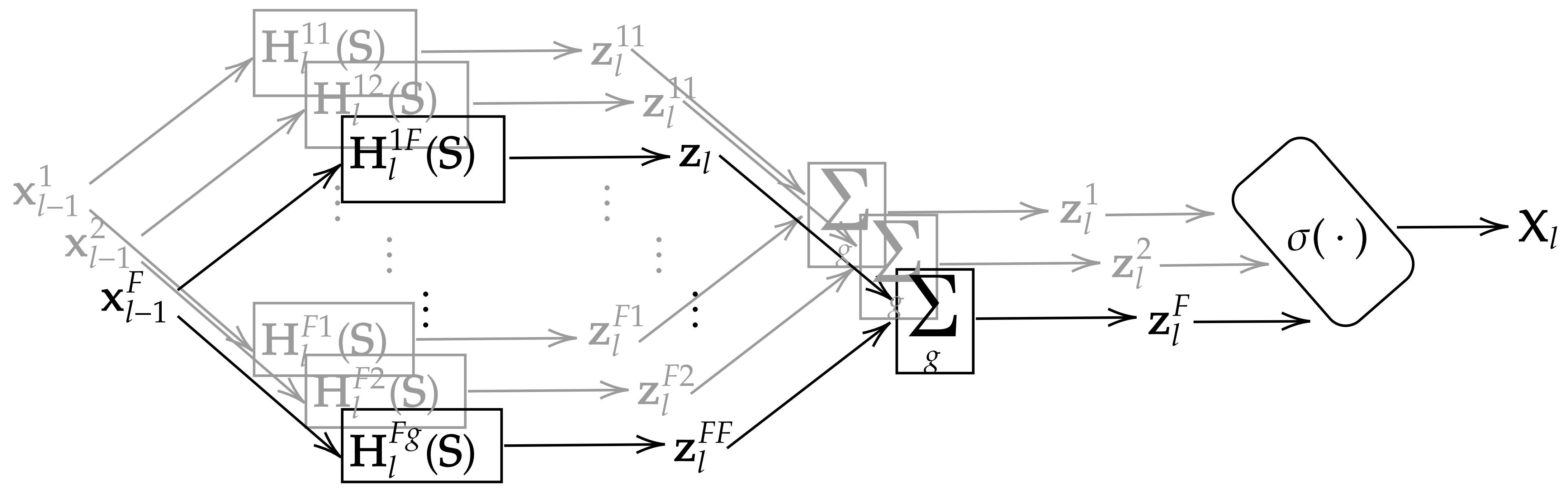
Convolutional Learning on SCs

Linear

$$\mathbf{H} := \mathbf{H}(\mathbf{L}_{\mathrm{d}}, \mathbf{L}_{\mathrm{u}}; \boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{k=0}^{K_{\mathrm{d}}} \alpha_k \mathbf{L}_{\mathrm{d}}^k + \sum_{k=0}^{K_{\mathrm{u}}} \beta_k \mathbf{L}_{\mathrm{u}}^k$$

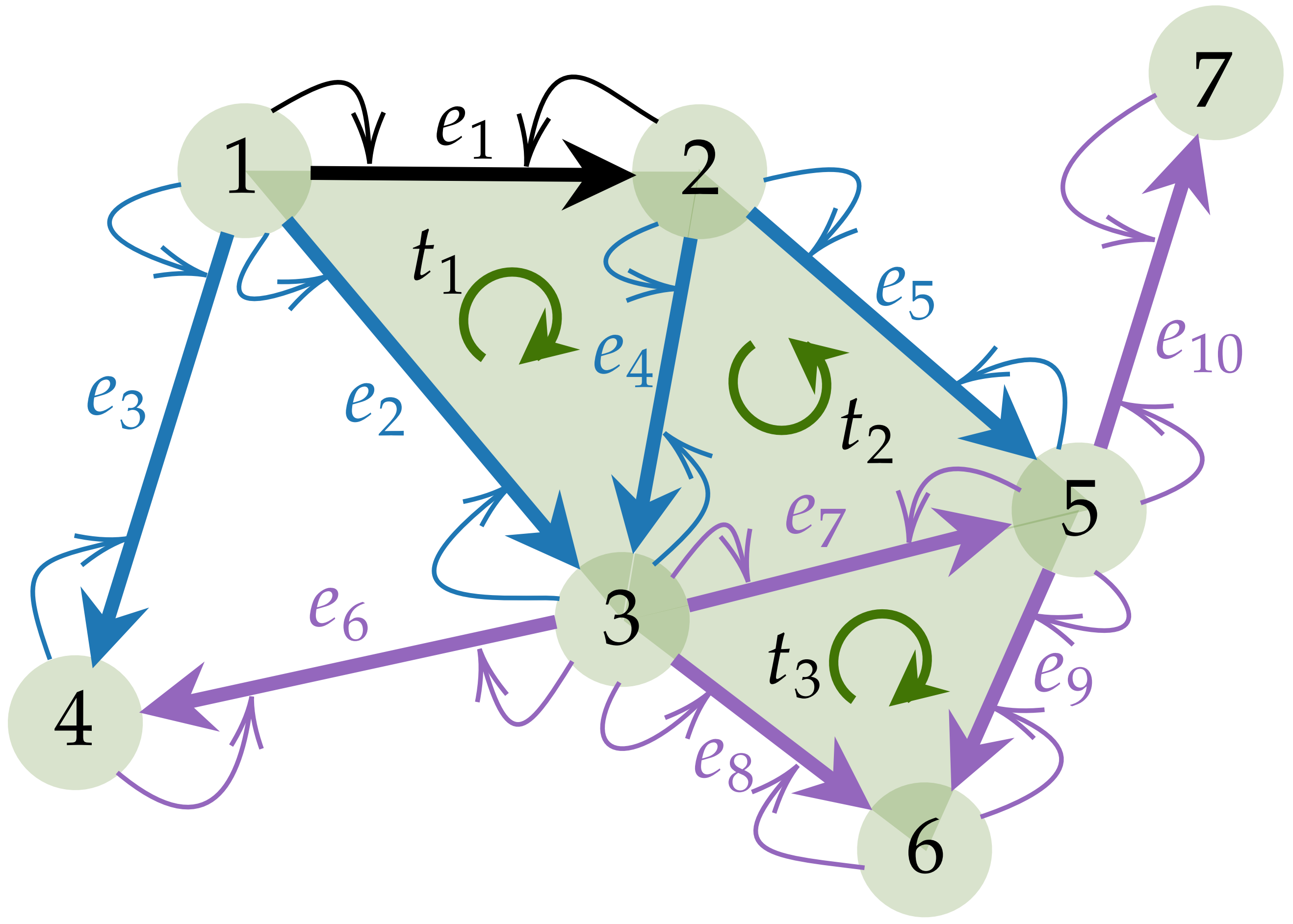


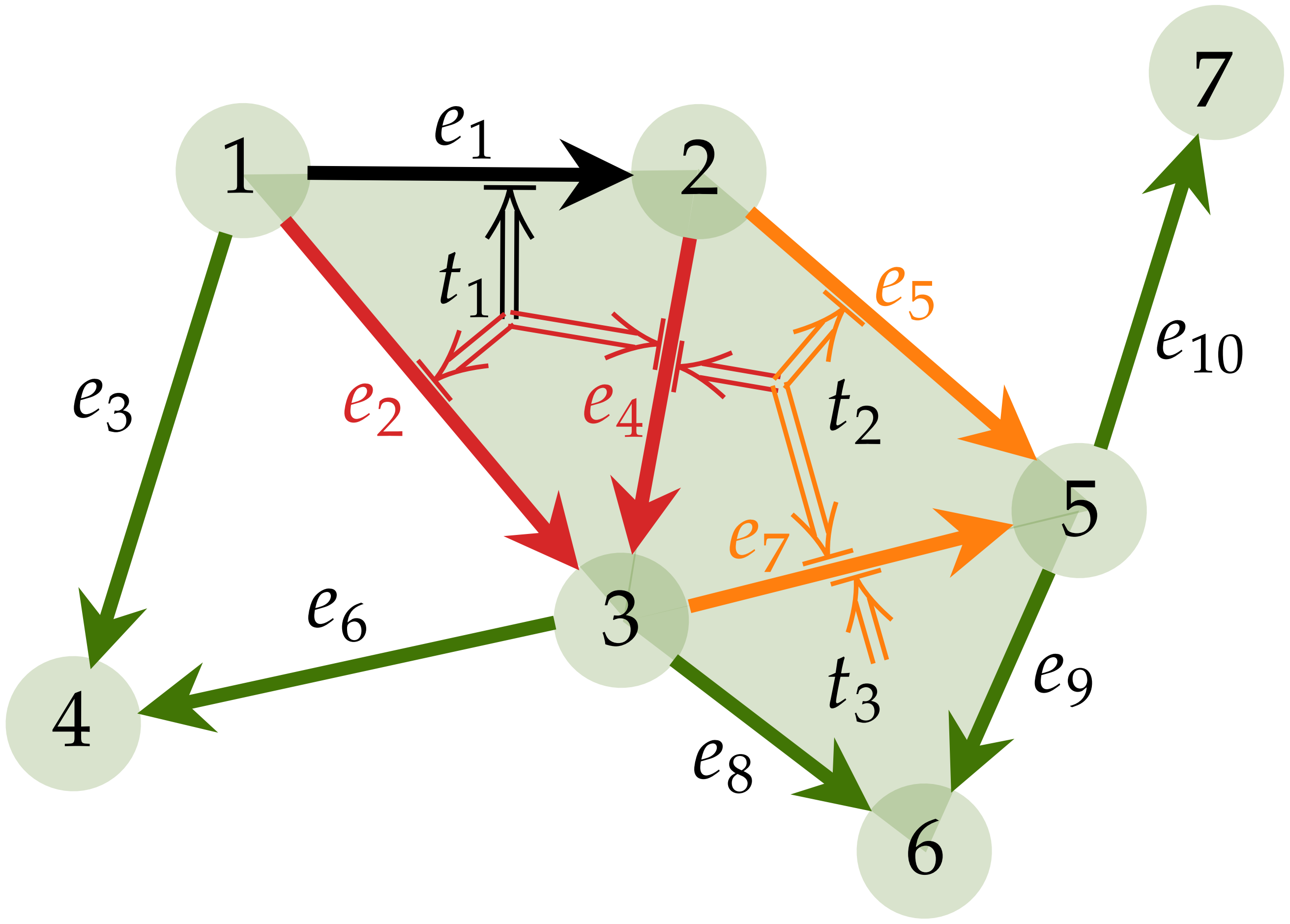
Non-Linear



Convolutional Learning on SCs

Node-edge-triangle interactions





Convolution based (Ebli et al. 2020; Roddenberry et al. 2021; Yang et al. 2022, 2023)
Message passing (Bodnar et al. 2021)

Edge Gaussian Processes

Matérn GP family on SCs

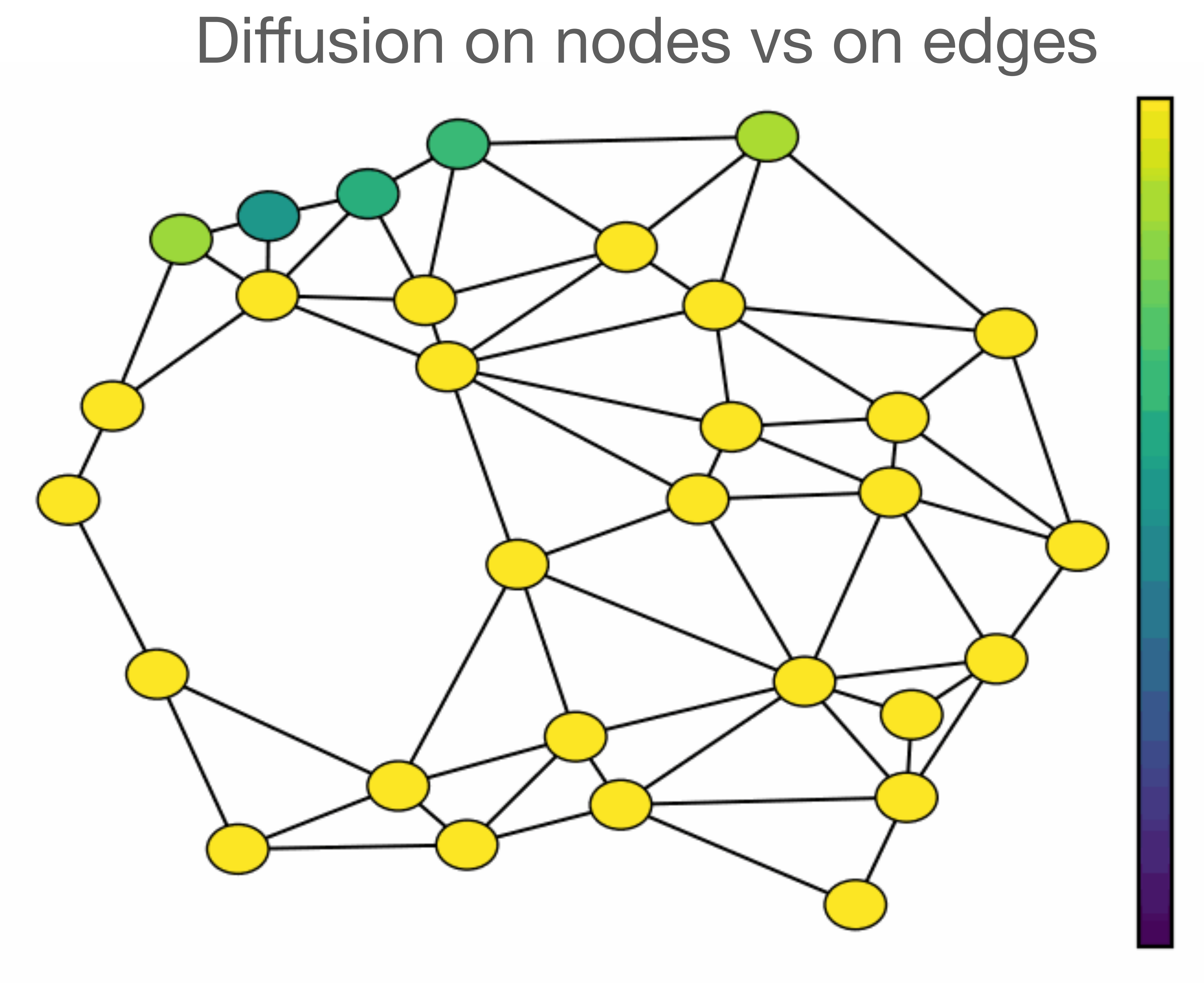
- SPDEs on edge space of SCs using Hodge Laplacians

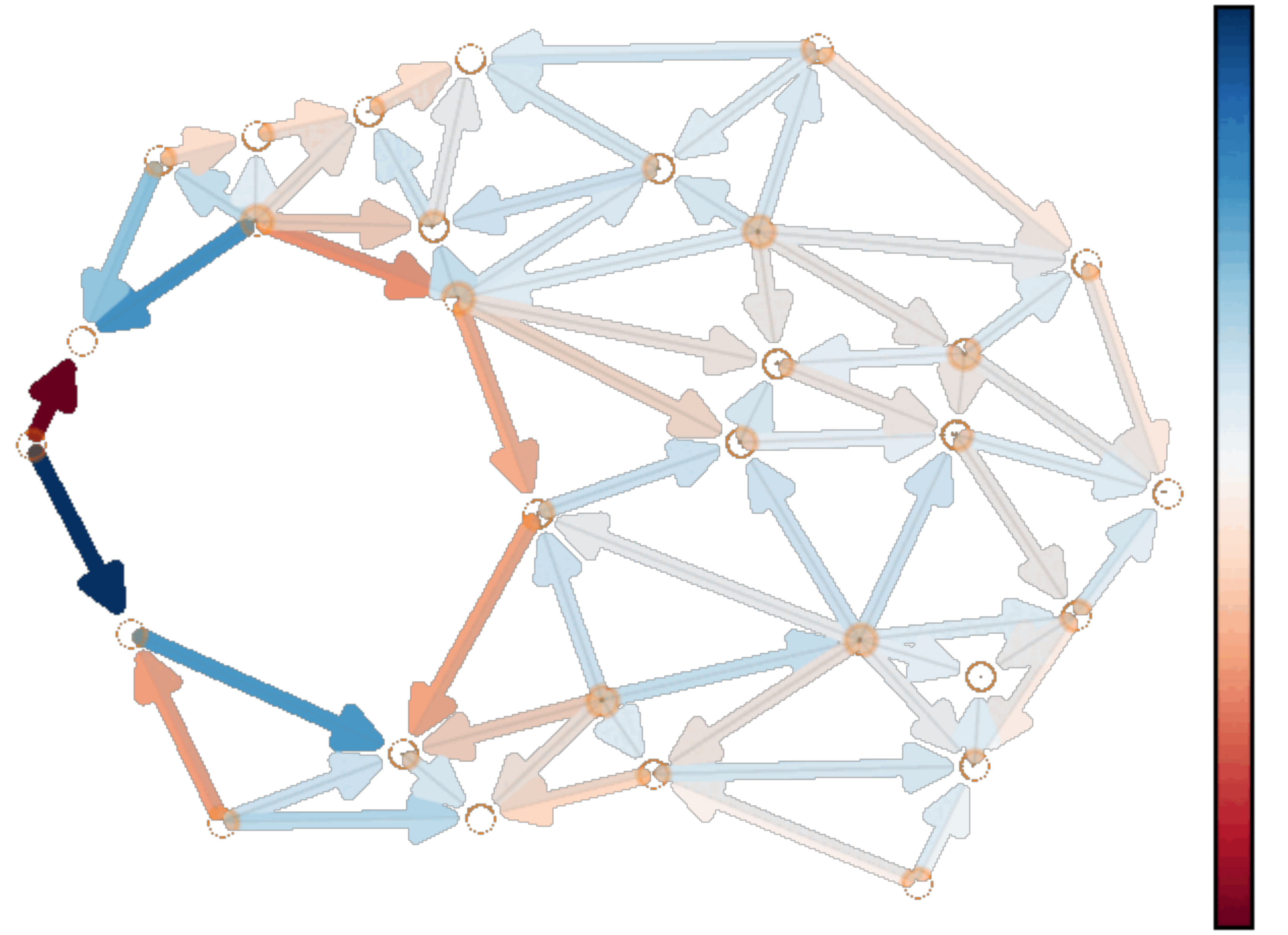
$$\Phi(L_1) f_1 = w_1, \quad w_1 \sim \mathcal{N}(0, I)$$

$$f_1 \sim \mathrm{GP}\left(0, \left(\frac{2\nu}{\kappa^2} + L_1\right)^{-\nu}\right) \quad f_1 \sim \mathrm{GP}\left(0, e^{-\frac{\kappa^2}{2} L_1}\right)$$

$$Lf = (\mathrm{grad} \circ \mathrm{div} + \mathrm{curl}^* \circ \mathrm{curl})\ \mathrm{f} = 0$$

f is a 1-form (vector field)





Hodge-compositional Edge GPs

Div-free and curl-free edge GPs

$$\text{(Spatial)}\left(\frac{2\nu}{\kappa^2}+L_1\right)^{-\nu} \rightarrow \left(\frac{2\nu}{\kappa^2}+\Lambda_1\right)^{-\nu}\text{(spectral)}$$

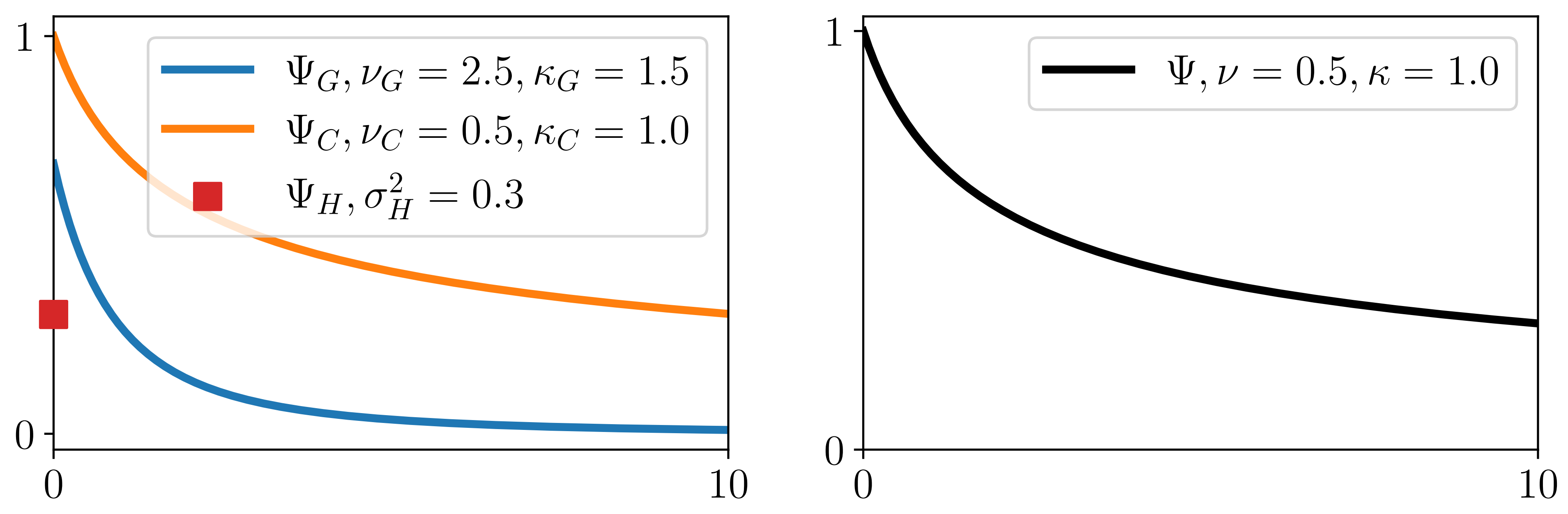
$$\Psi_{\square}(\Lambda_{\square}) = \sigma_{\square}^2\left(\frac{2\nu_{\square}}{\kappa_{\square}^2}+\Lambda_{\square}\right)^{-\nu_{\square}}, \quad \square = \mathrm{H}, \mathrm{G}, \mathrm{C}$$

- They can be obtained from SPDEs on edges as well

$$\Phi(L_{1,\mathrm{u}})f_1 = w_1, \quad w_1 \sim \mathcal{N}(0,\sigma_C^2 U_C U_C^\top)$$

- Composition of three GPs on the Hodge subspaces

- Kernel: $K_1 = K_G + K_H + K_C$



$$\mathrm{curl}^* \circ \mathrm{curl}\ \mathrm{f}_1 = \mathrm{w}_1$$

Maxwell equations for H, incompressible flows

Conclusion

- Variation (smoothness) measures of edge flows: discrete div and curl
- Smoothness of node data $\mathbf{v}^\top \mathbf{L}_0 \mathbf{v}$
- General simplicial data: variations w.r.t. faces and cofaces
- Hodge subspaces spanned by eigenbasis of Hodge Laplacians
- Principled processing, filtering, learning, modelings

Other applications of Hodge decomp.

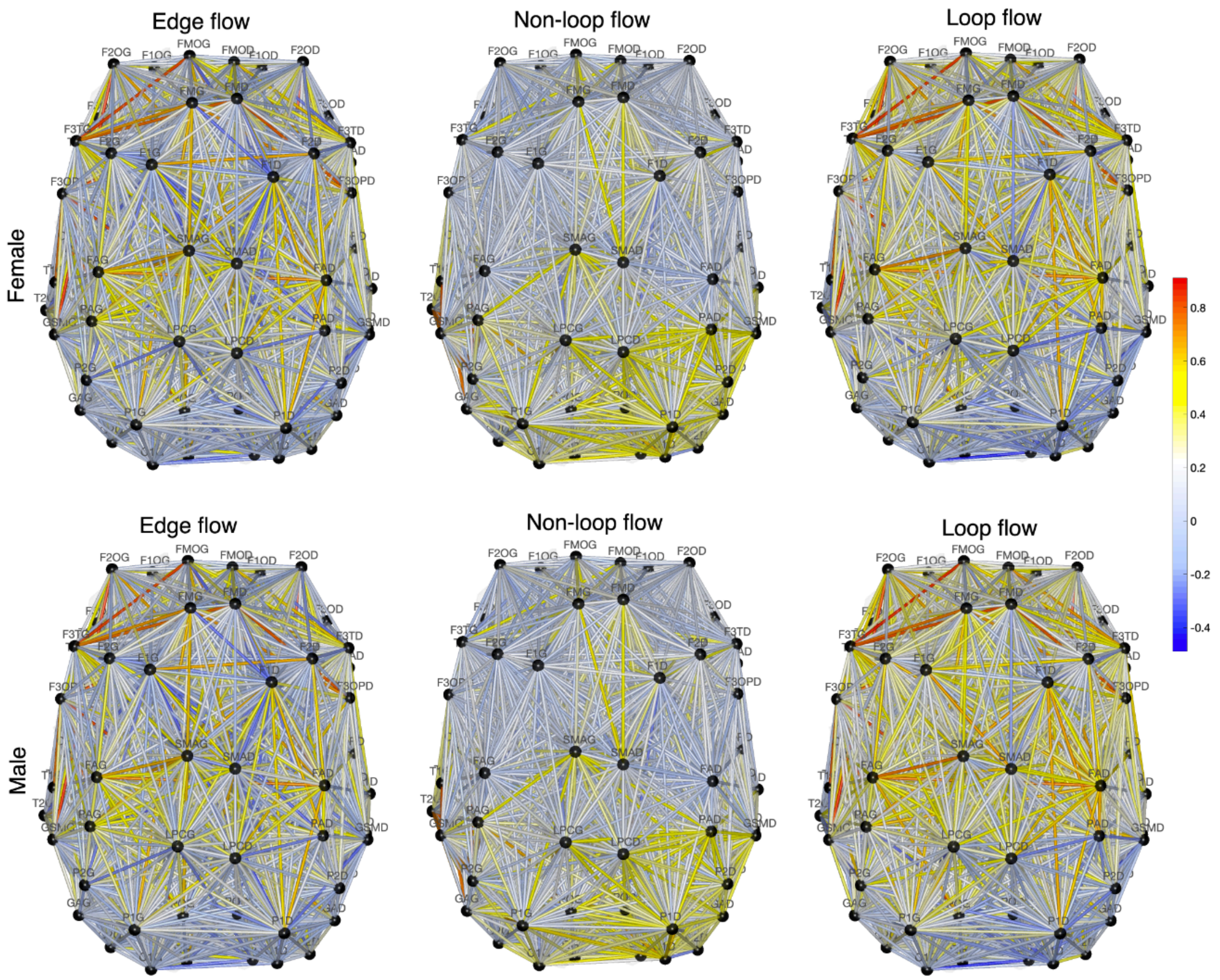


Fig. 14: Top: The average connectivity (edge flow), non-loop flow (middle) and the loop flow (right) of the female (top) and male networks (bottom).

- Brain networks (Anand et al. 2022)

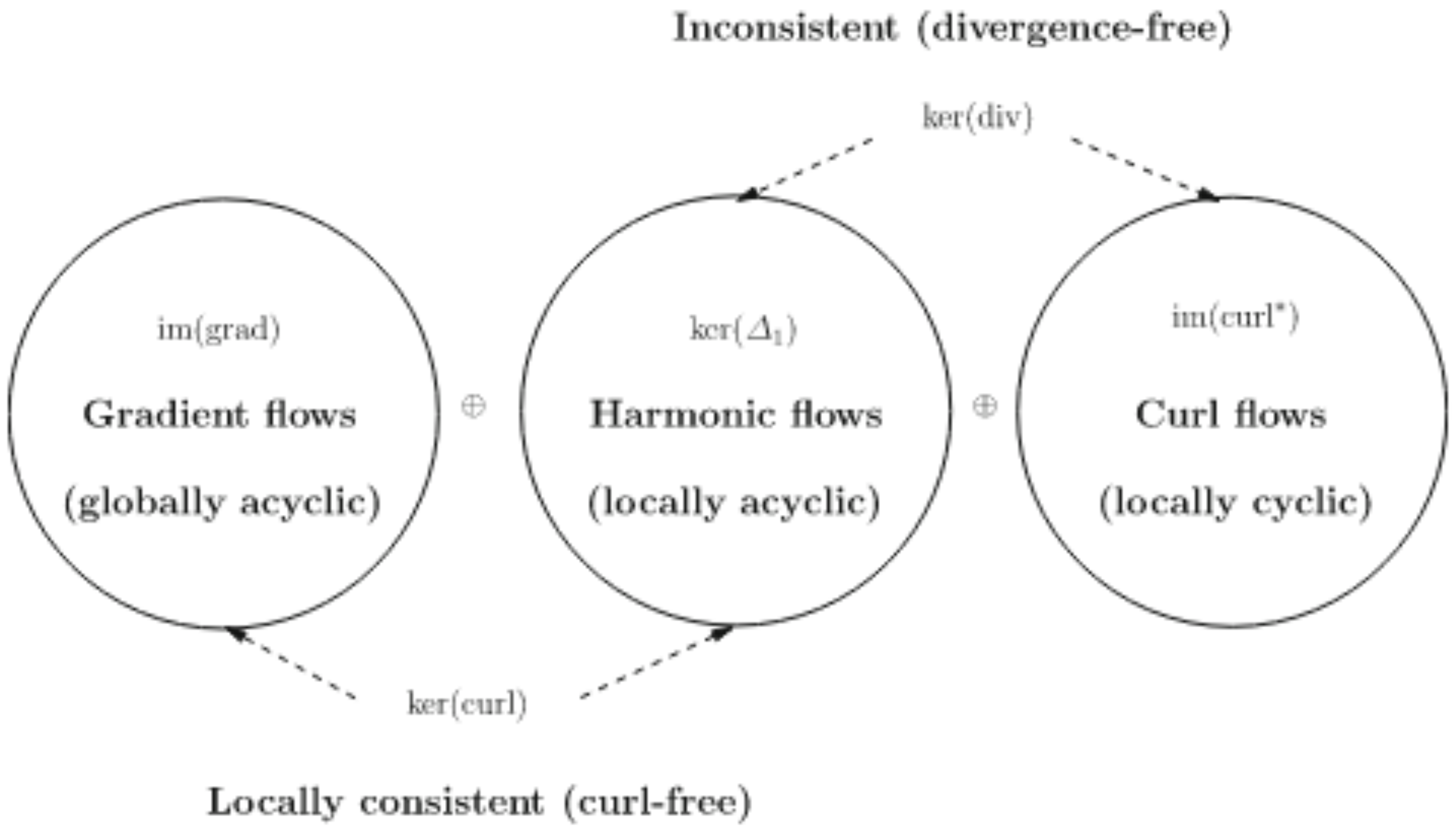
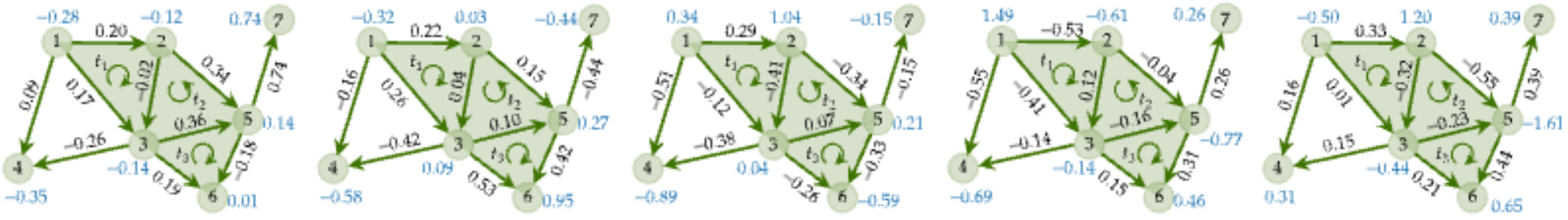


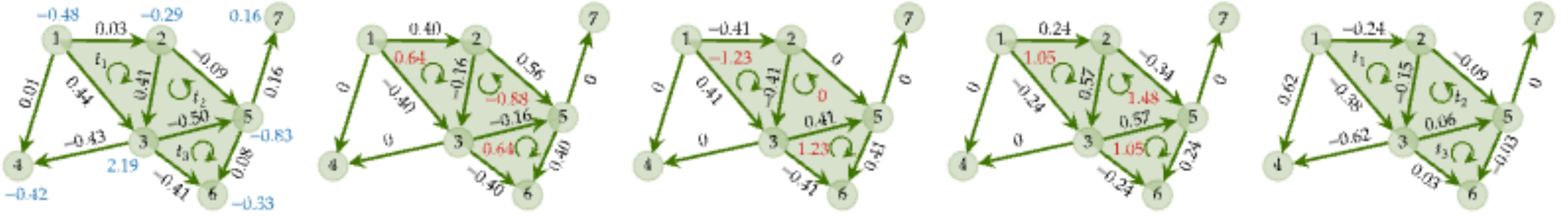
Fig. 2 Hodge/Helmholtz decomposition of pairwise rankings

- Ranking problems (Jiang et al. 2011)
- Condorcet paradox: cyclic

Full Eigenbasis of example SC

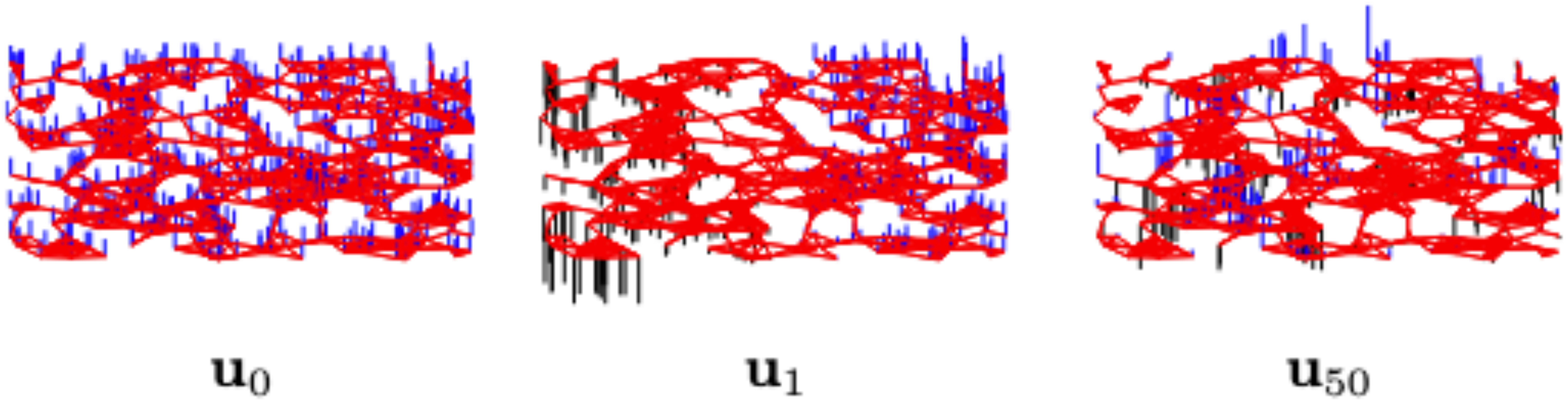


(a) $\boldsymbol{u}_{\mathrm{G},1}, \lambda_{\mathrm{G},1}(0.80)$ (b) $\boldsymbol{u}_{\mathrm{G},2}, \lambda_{\mathrm{G},2}(1.61)$ (c) $\boldsymbol{u}_{\mathrm{G},3}, \lambda_{\mathrm{G},3}(2.43)$ (d) $\boldsymbol{u}_{\mathrm{G},4}, \lambda_{\mathrm{G},4}(3.96)$ (e) $\boldsymbol{u}_{\mathrm{G},5}, \lambda_{\mathrm{G},5}(5.12)$



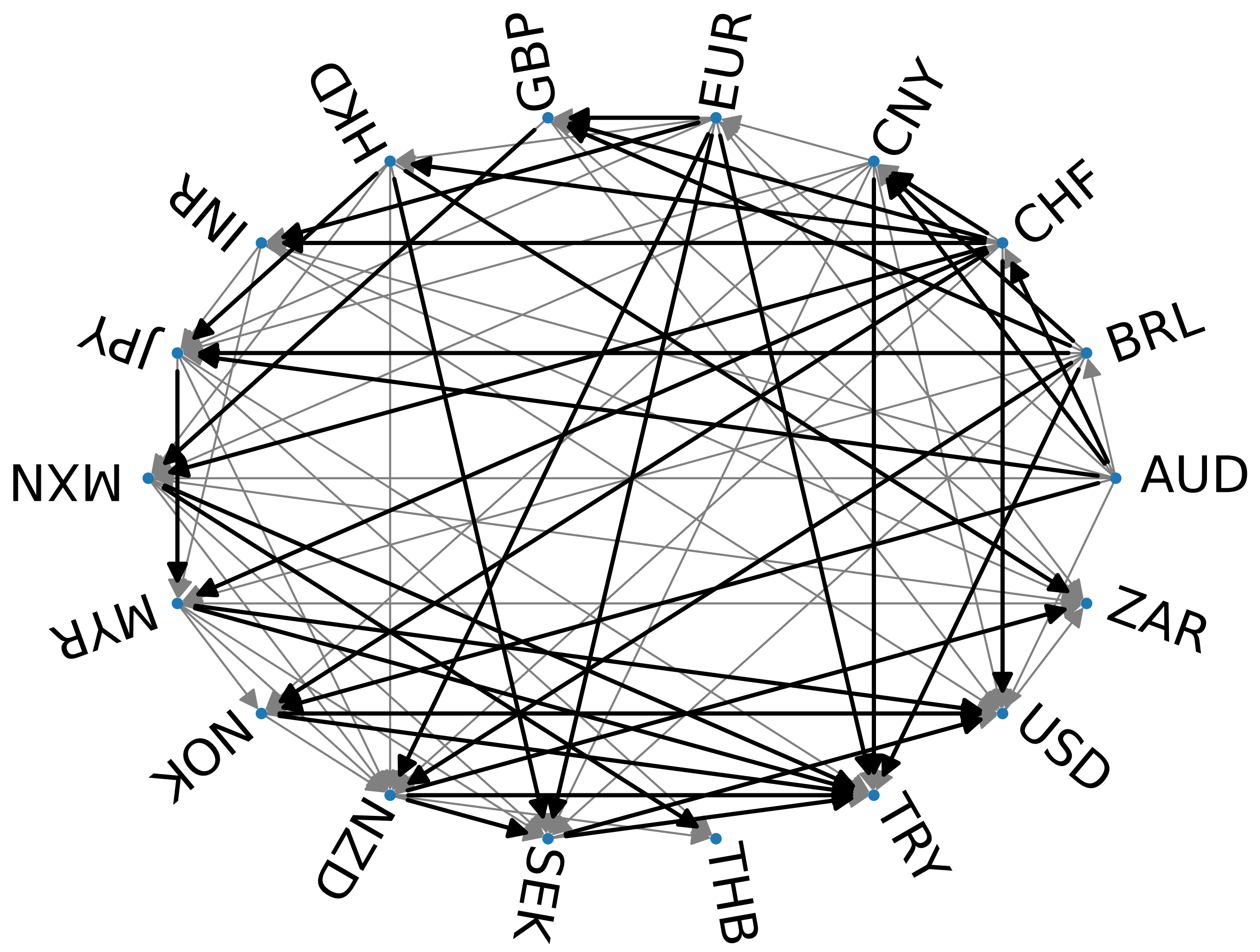
(f) $\boldsymbol{u}_{\mathrm{G},6}, \lambda_{\mathrm{G},6}(6.08)$ (g) $\boldsymbol{u}_{\mathrm{C},1}, \lambda_{\mathrm{C},1}(1.59)$ (h) $\boldsymbol{u}_{\mathrm{C},2}, \lambda_{\mathrm{C},2}(3.00)$ (i) $\boldsymbol{u}_{\mathrm{C},3}, \lambda_{\mathrm{C},3}(4.41)$ (j) $\boldsymbol{u}_{\mathrm{H}}, \lambda_{\mathrm{H}}(0)$

Spectrum of graph Laplacians



Shuman et al. (2013)

Learning for Forex



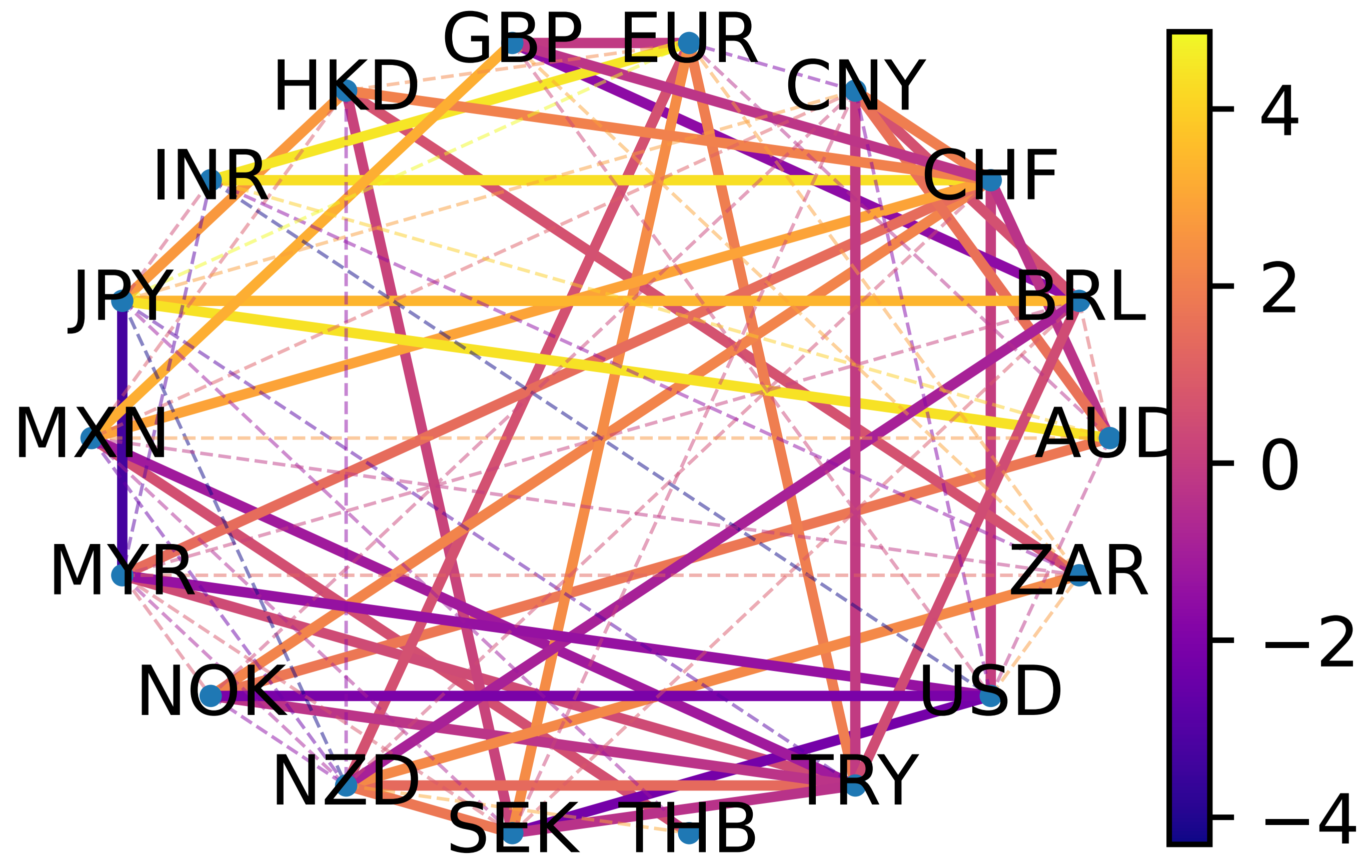


Table 1: Forex results (nmse|total arbitrage, ↓).

Methods	Random Noise	Curl Noise	Interpolation
Input	$0.119_{\pm 0.004}$|$29.19_{\pm 0.874}$	$0.552_{\pm 0.027}$|$122.4_{\pm 5.90}$	$0.717_{\pm .030}$|$106.4_{\pm 0.902}$
Baseline (ℓ_2 regularization)	$0.036_{\pm 0.005}$|$2.29_{\pm 0.079}$	$0.050_{\pm 0.002}$|$11.12_{\pm 0.537}$	$0.534_{\pm 0.043}$|$9.67_{\pm 0.082}$
SNN (Ebli et al., 2020)	$0.110_{\pm 0.005}$|$23.24_{\pm 1.03}$	$0.446_{\pm 0.017}$|$86.95_{\pm 2.20}$	$0.702_{\pm 0.033}$|$104.74_{\pm 1.04}$
PSNN (Roddenberry et al., 2021)	$0.008_{\pm 0.001}$|$0.984_{\pm 0.170}$	$0.000_{\pm 0.000}$|$0.000_{\pm 0.000}$	$0.009_{\pm 0.001}$|$1.13_{\pm 0.329}$
MPSN (Bodnar et al., 2021b)	$0.039_{\pm 0.004}$|$7.74_{\pm 0.88}$	$0.076_{\pm 0.012}$|$14.92_{\pm 2.49}$	$0.117_{\pm 0.063}$|$23.15_{\pm 11.7}$
SCCNN, id	$0.027_{\pm 0.005}$|$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$|$0.000_{\pm 0.000}$	$0.265_{\pm 0.036}$|$0.000_{\pm 0.000}$
SCCNN, tanh	$\mathbf{0.002_{\pm 0.000}}$|$\mathbf{0.325_{\pm 0.082}}$	$0.000_{\pm 0.000}$|$0.003_{\pm 0.003}$	$\mathbf{0.003_{\pm 0.002}}$|$\mathbf{0.279_{\pm 0.151}}$

Simplex prediction

Generalization of link prediction

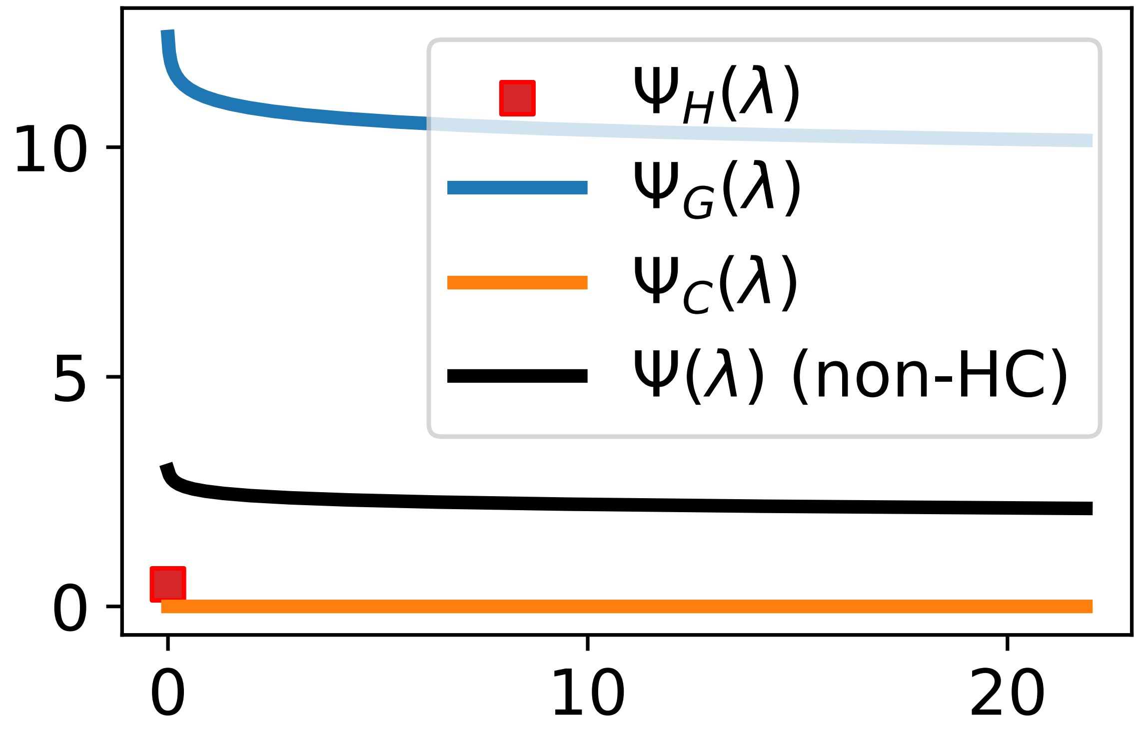
Table 2: Simplex prediction (AUC, ↑) .

Methods	2-simplex	3-simplex
Mean (Benson et al., 2018)	62.8±2.7	63.6±1.6
MLP	68.5±1.6	69.0±2.2
GNN (Defferrard et al., 2016)	93.9±1.0	96.6±0.5
SNN (Ebli et al., 2020)	92.0±1.8	95.1±1.2
PSNN (Roddenberry et al., 2021)	95.6±1.3	98.1±0.5
SCNN (Yang et al., 2022a)	96.5±1.5	98.3±0.4
Bunch (Bunch et al., 2020)	98.3±0.5	98.5±0.5
MPSN (Bodnar et al., 2021b)	98.1±0.5	99.2±0.3
SCCNN	**98.7±0.5**	**99.4±0.3**

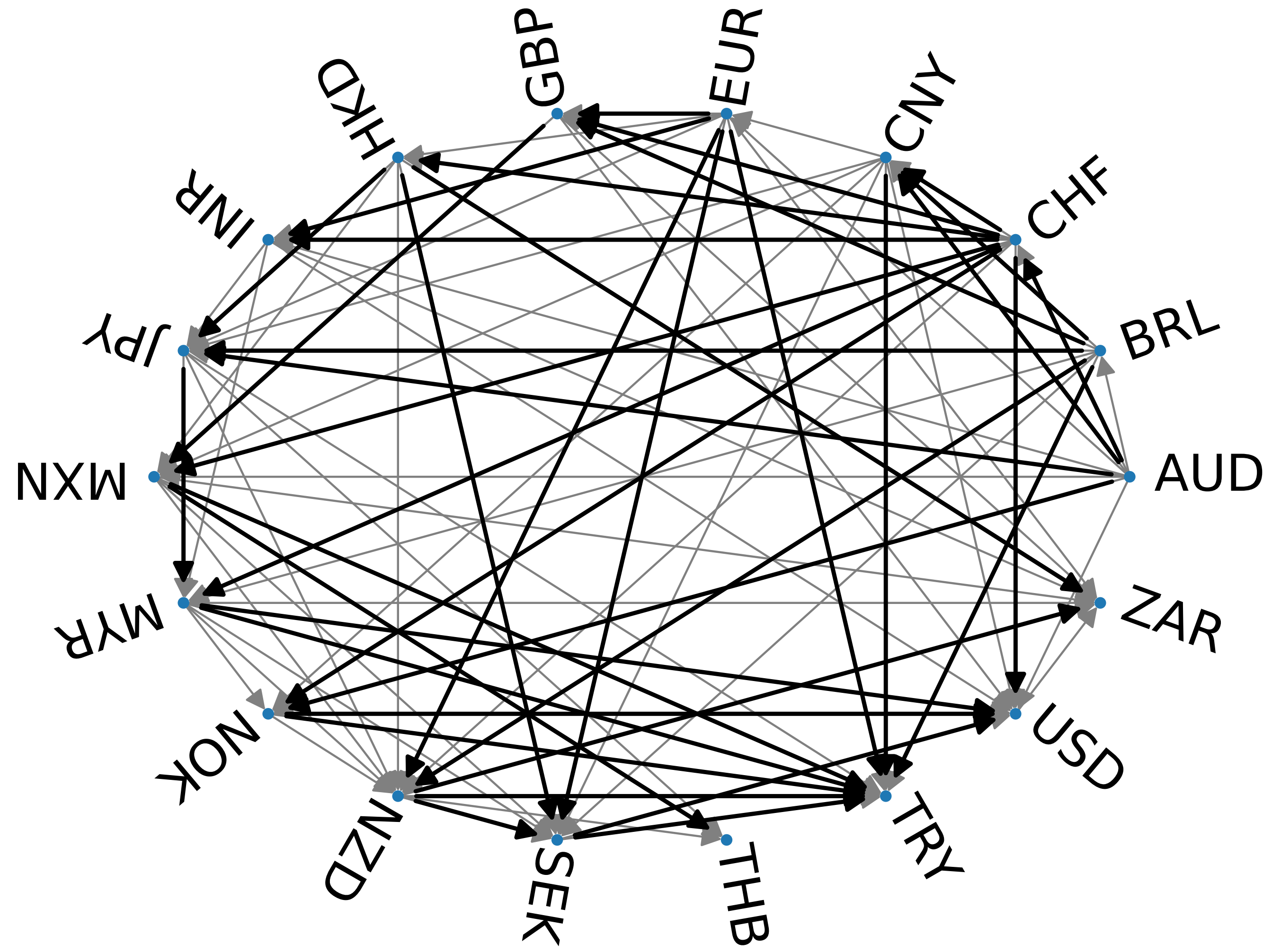
GPs based on Node-edge-triangle interactions

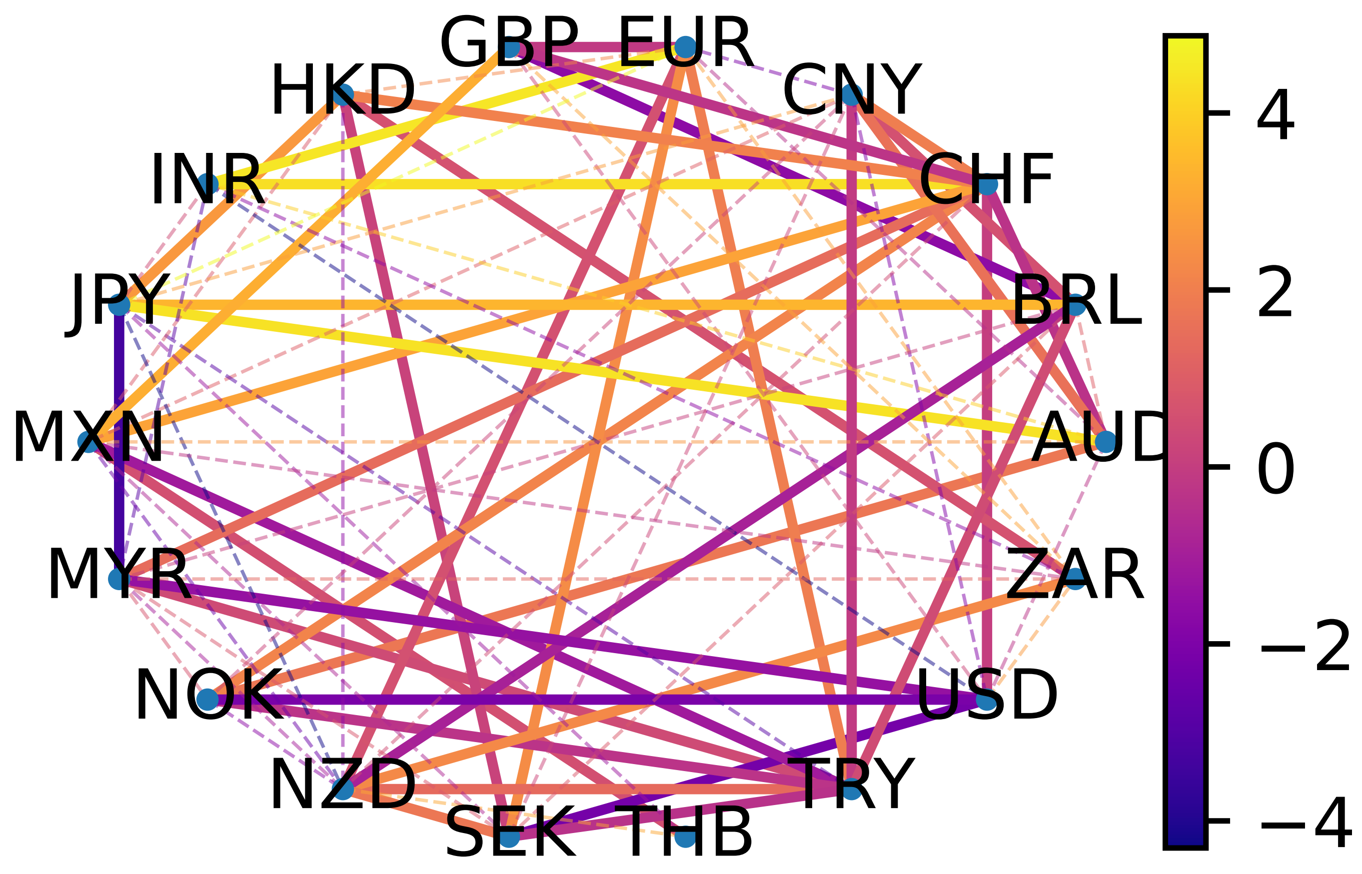
- Derivatives of GPs are also GPs
- Induce edge GPs from node GPs and triangle GPs
- $K_1 = K_H + B_1^\top K_0 B_1 + B_2 K_2 B_2^\top$
- Induce node GPs from edge GPs

GP based Forex prediction

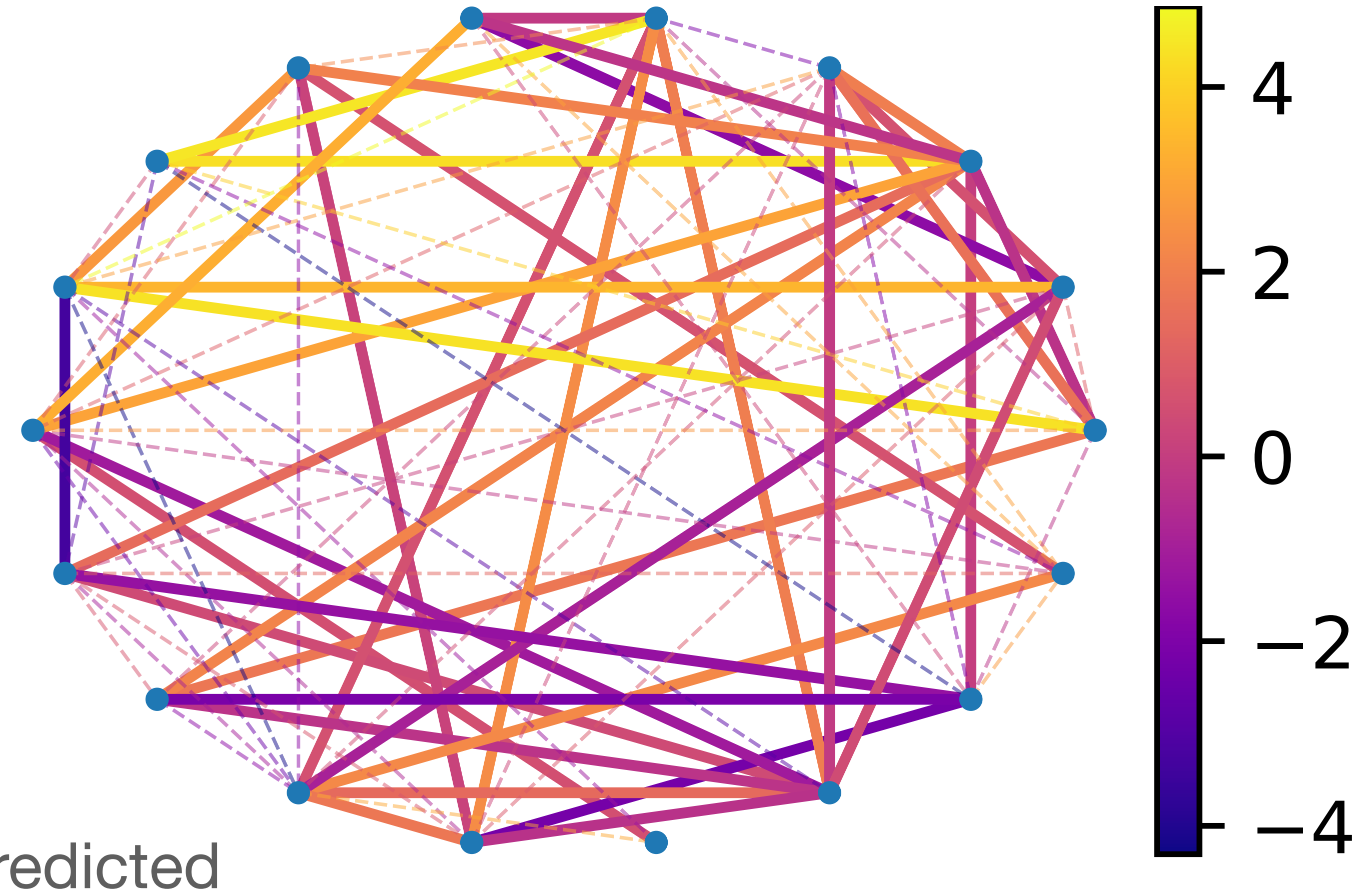


Learned kernels

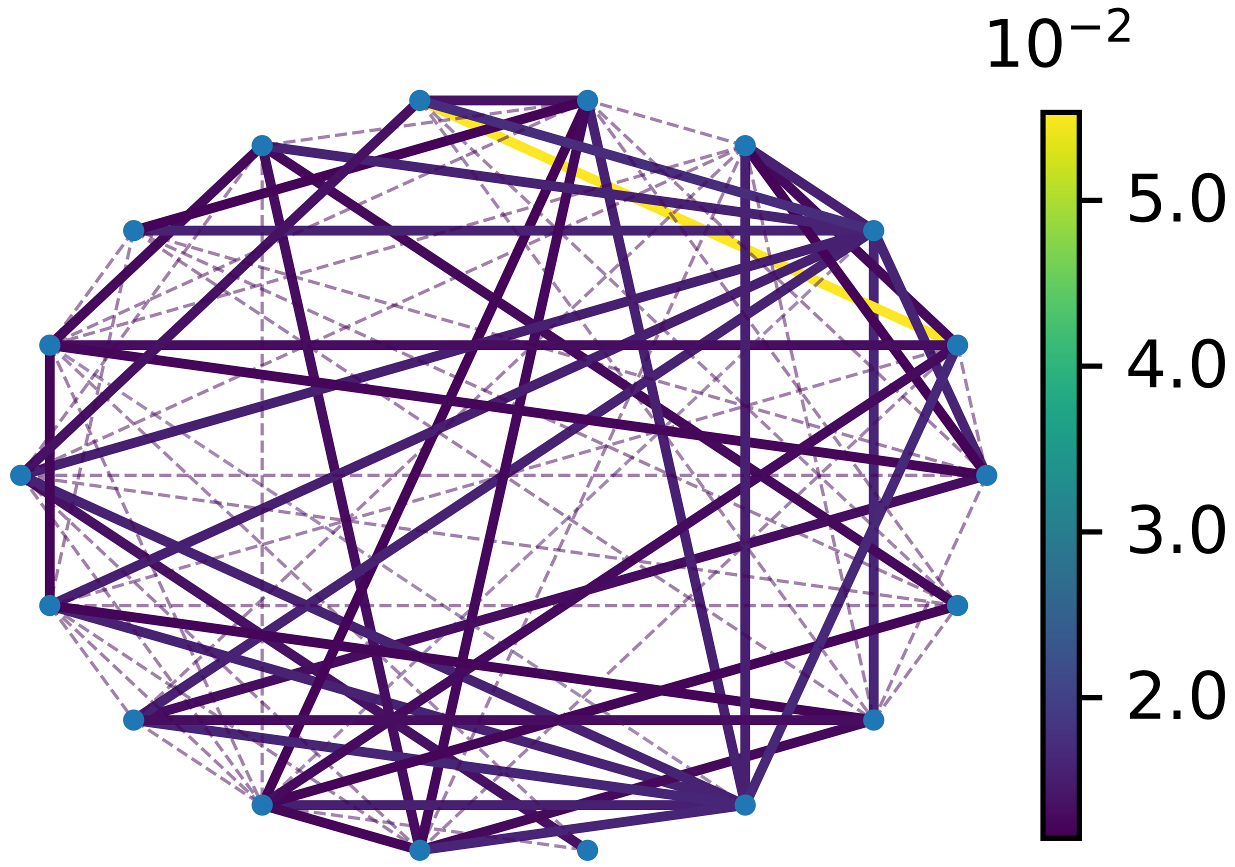




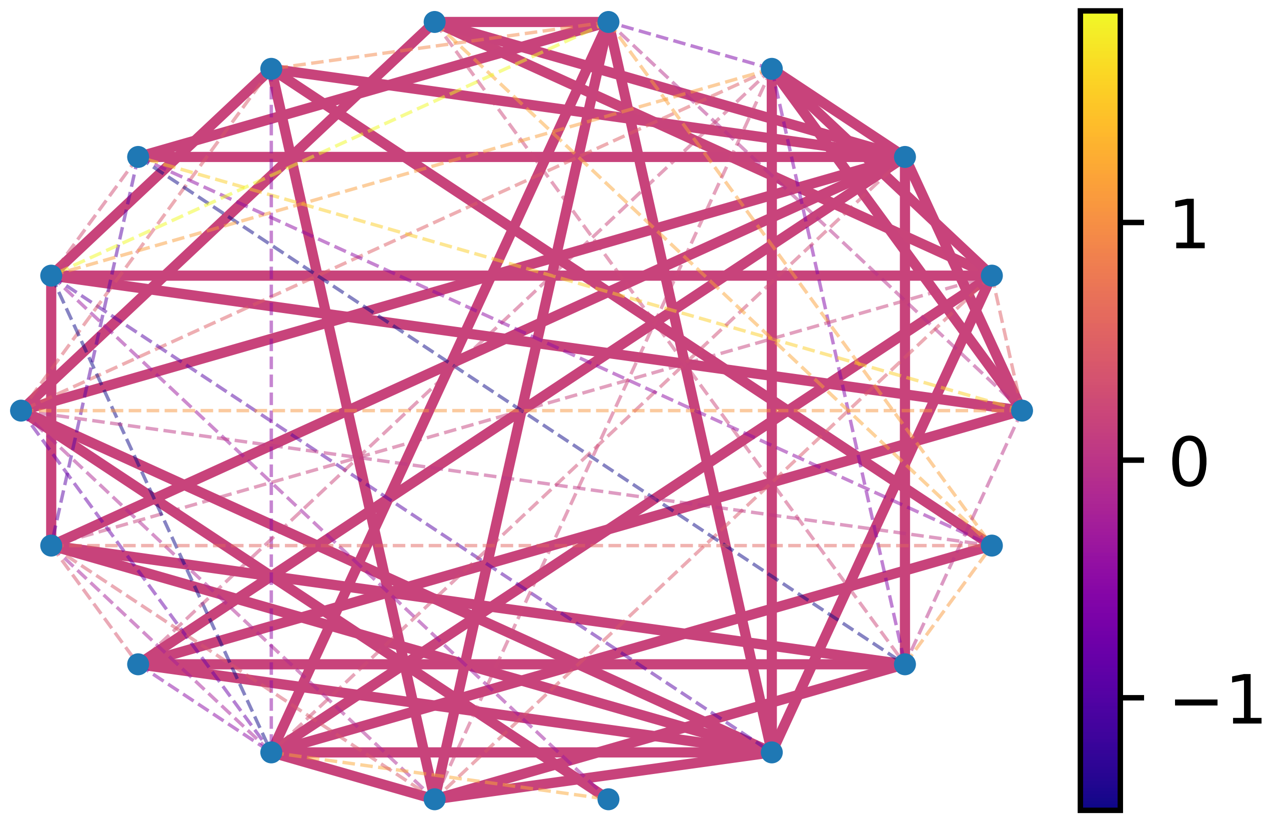
True



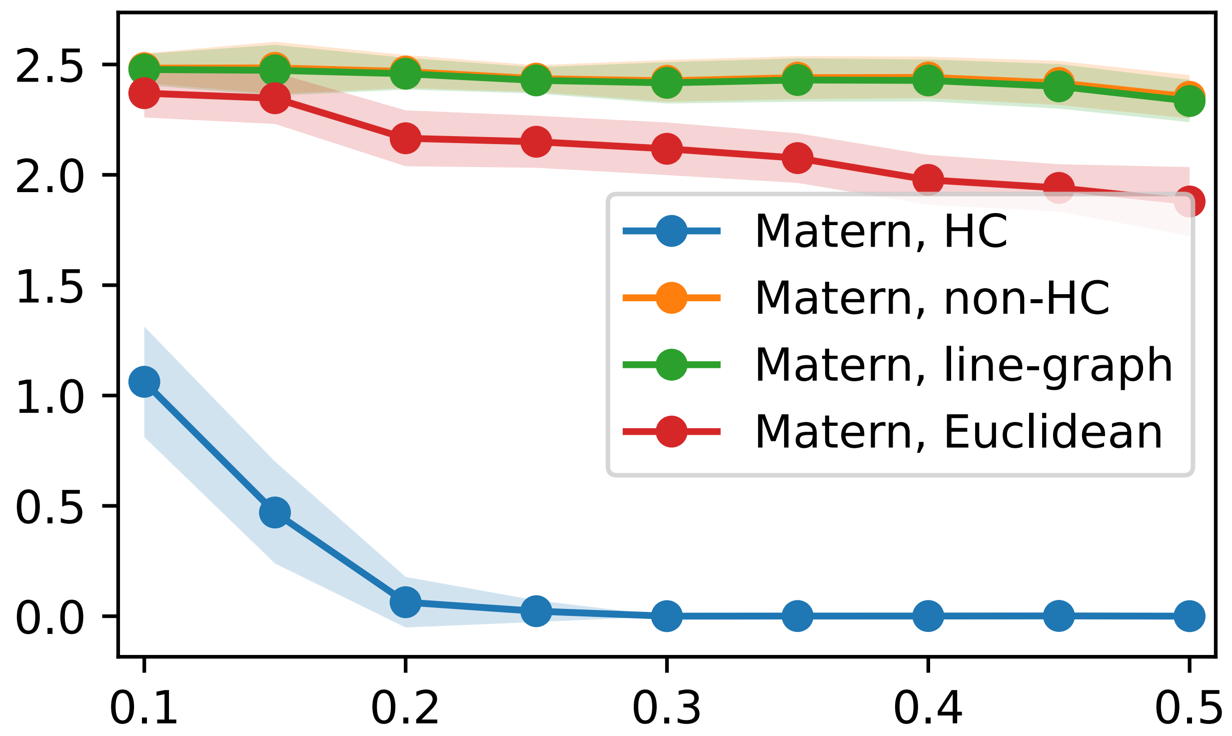
Predicted



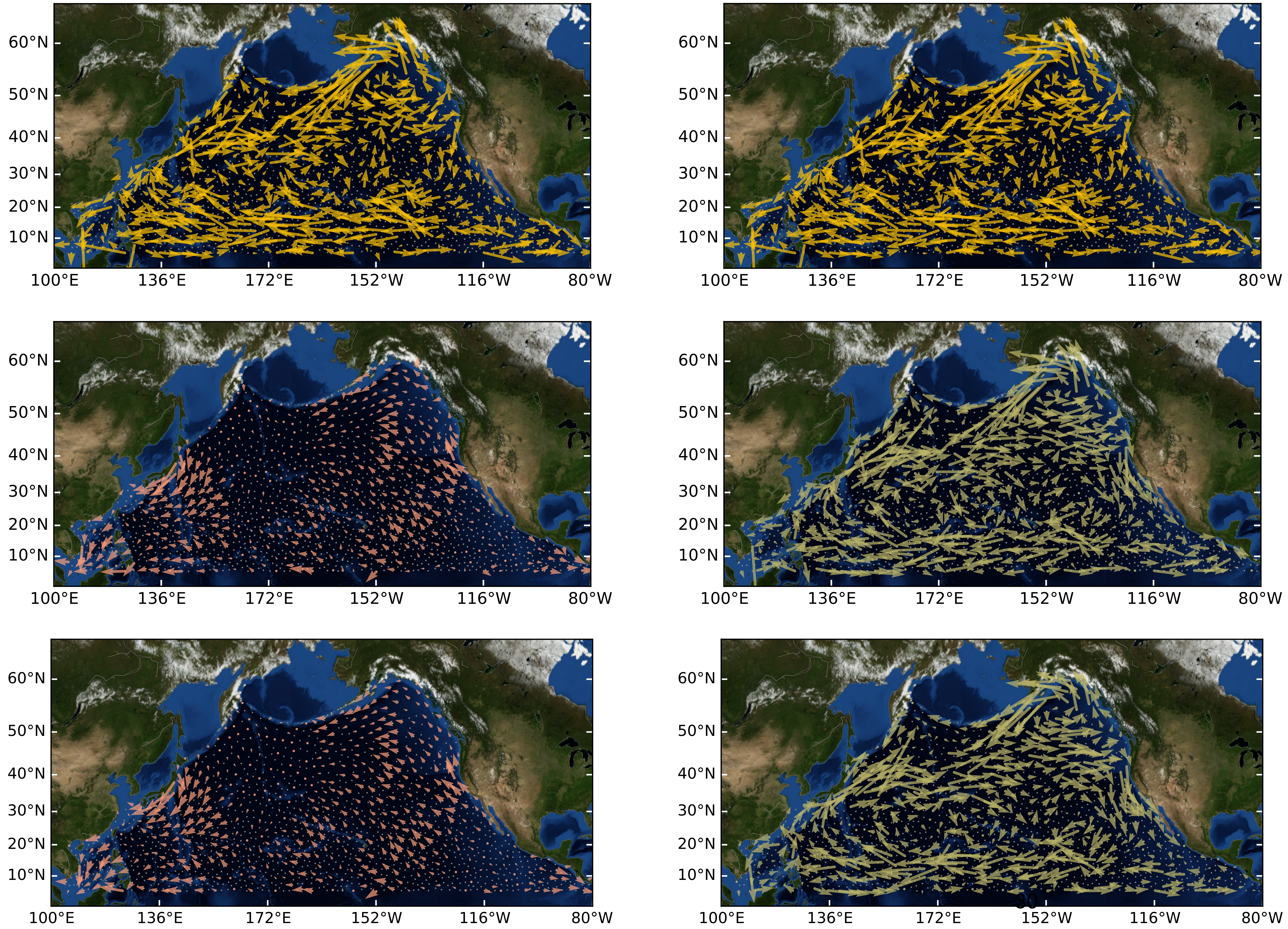
Std

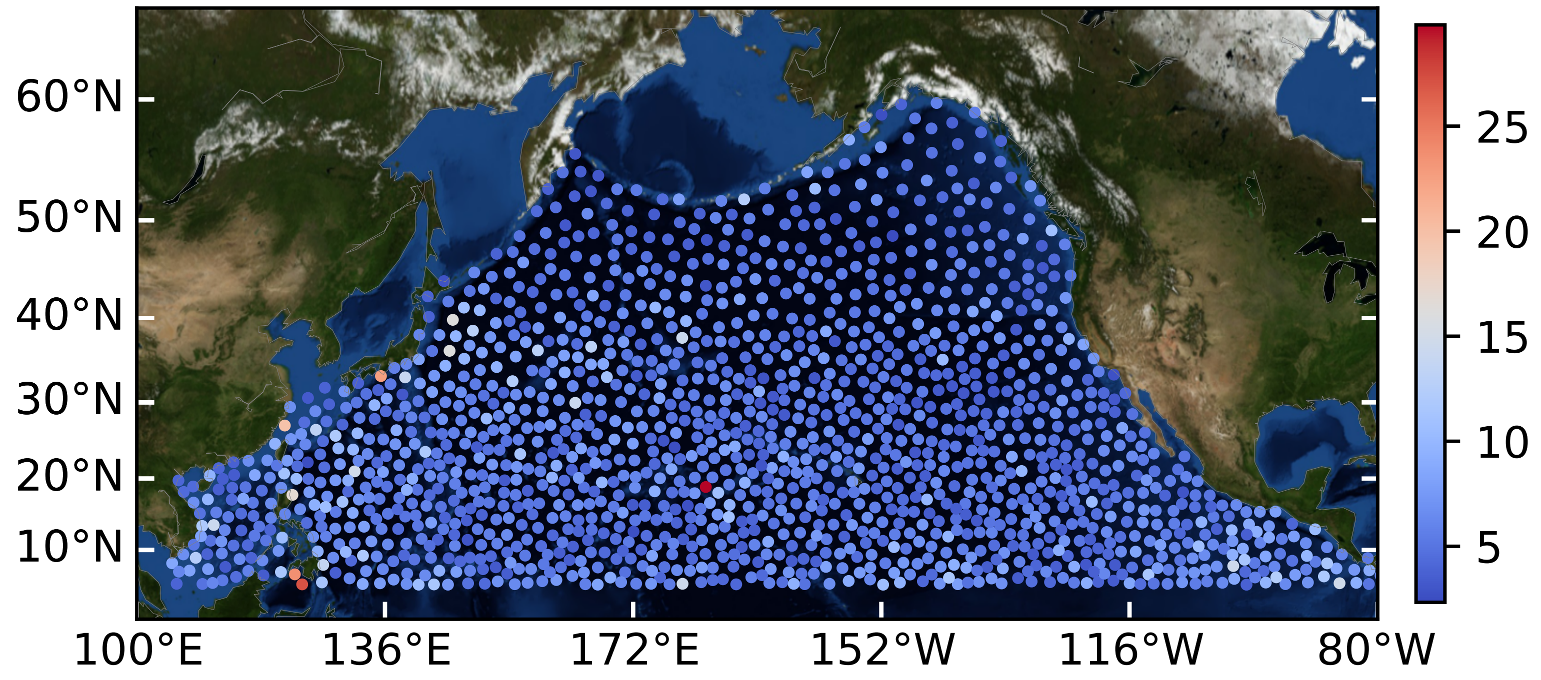


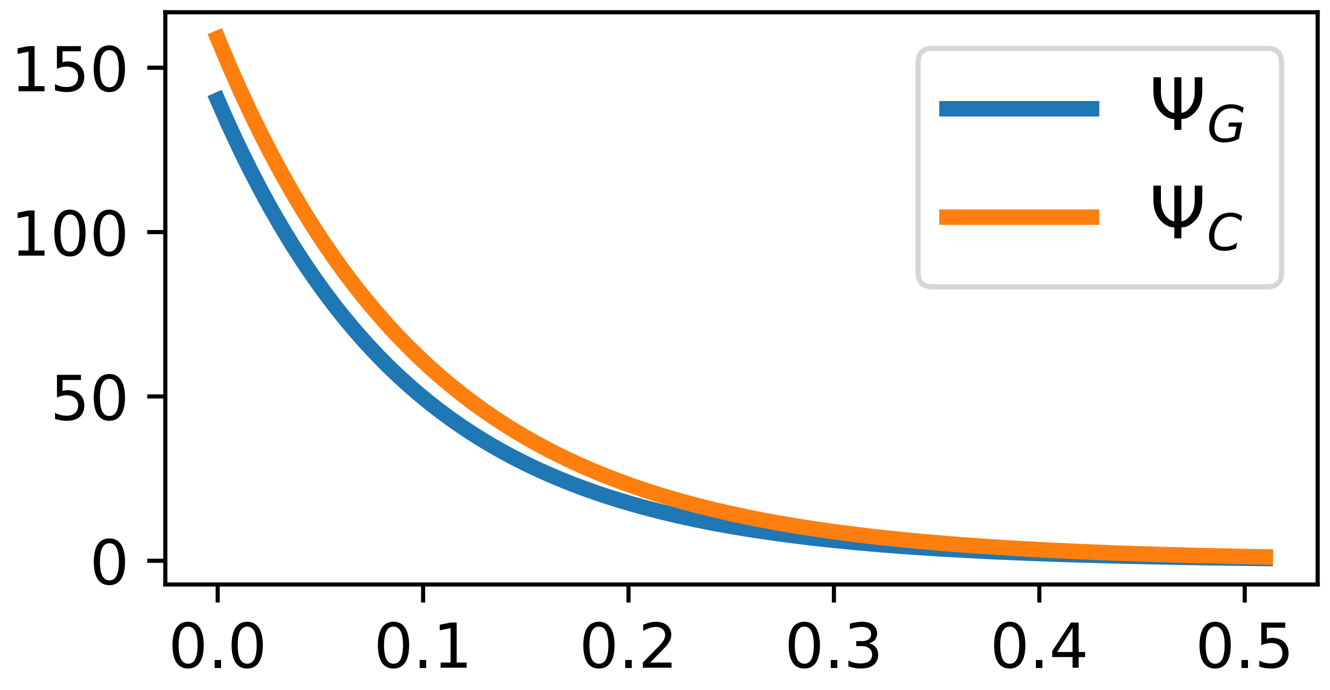
non-Hodge



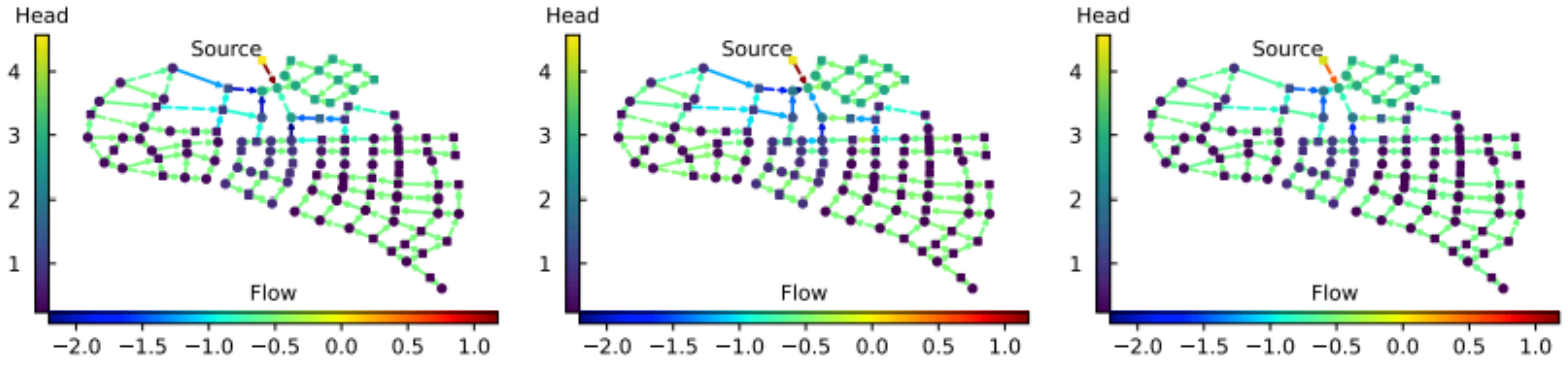
GP based Ocean current analysis



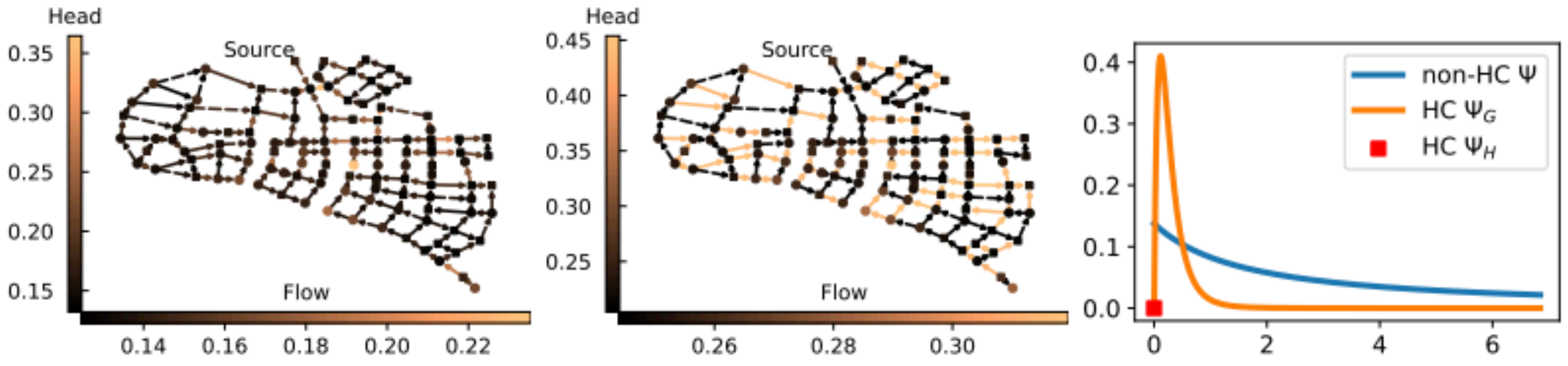




GP based state estimation in Water supply networks



(a) Ground truth (b) Mean, HC Matérn (c) Mean, non-HC Matérn



(d) Std, HC Matérn (e) Std, non-HC Matérn (f) Learned Matérn edge kernels