Spectral Graph Filters for Large Power Systems

Luke Lowery
Dept. of Elec. & Comp. Engineering
Texas A&M University
College Station, TX
lukel@tamu.edu

Adam B. Birchfield
Dept. of Elec. & Comp. Engineering
Texas A&M University
College Station, TX
abirchfield@tamu.edu

***Abstract*—The field of Graph Signal Processing (GSP) is relevant to power system stability studies, such as modal analysis, state estimation, and transient stability. In this work, we derive and detail a set of spectral graph filters that leverage the sparse graph convolution method of Kernel Fitting (KF) and the Cholesky decomposition to improve the scalability and time complexity of graph convolution for power system applications. In addition, low-rank updates to the Cholesky decomposition confer dynamic graph compatibility on the proposed filter banks. Detailed algorithms supplement the theoretical work of this paper. In the case studies, the time complexity of graph convolution with the proposed filters is benchmarked using synthetic networks (2k, 70k, 80k, and 82k buses), demonstrating remarkable speed and versatility.**

***Index Terms*—Cholesky, kernel fitting, large power systems, graph signal processing, vector fitting**

I. Introduction

GRAPH signal processing, a growing field in mathematics, has sparked interest in the study of large power systems through sparse implementation methods. The applications of GSP to power systems in the literature to date have been largely heuristic; practical analytical applications remain an open topic of discussion [1].

Signal reconstruction using GSP is a popular application of GSP in power systems [2], with various theorems demonstrating optimality for batched and online state estimation. In the author's prior work [3], low-frequency natural oscillation modes are identified using the Spectral Graph Wavelet Transformation (SGWT), the GSP analog to the continuous wavelet transformation. Other applications in literature include state and topology estimation.

In electromagnetic transients (EMT) research, vector fitting (VF) [4] is an unrivaled reduced-order modeling technique for approximating Laplace-domain functions to perform time-domain convolution. The recent work in [3] discovered that vector fitting also allows one to approximate functions in the *spectral graph domain* to perform *vertex domain* convolutions. In this paper, we show that a specific class of graph spectral filters exhibits desirable time complexity for applying GSP to power systems with tens to hundreds of thousands of buses.

The paper is structured as follows. Section II summarizes the preliminary mathematics and notation of GSP. Section III derives and details three closed-form spectral filters, and a vector fitting algorithm tailored for GSP using the Cholesky Decomposition. Section IV proposes a novel representation of GSP state tracking for power systems, followed by detailed algorithms for implementation in Section V. Lastly, the case study verifies the algorithms and their computational performance in Section VI.

To appear in the proceedings of TPEC 2026.

II. Graph Signal Processing Background

A. Graph Laplacian and Fourier Transform

Let an undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{A}, \mathbf{w}\}$ be defined by a set of verticies $|\mathcal{V}| = N$ and a set of edges $\mathcal{E}$ which are related by the arc-node incident matrix $\mathbf{A} \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{V}|}$ and the vector of branch weights $\mathbf{w} \in \mathbb{R}^{|\mathcal{E}|}$. The *graph Laplacian* (1) is denoted by $\mathscr{L} \in \mathbb{R}^{N\times N}$, a discrete analogue of the continuous Laplace-Beltrami operator.

$$\mathscr{L} := \mathbf{A}^\top \text{diag}\,(\mathbf{w})\mathbf{A} \tag{1}$$

The complex admittance matrix $\mathbf{Y}_{bus}$, in general, cannot be used in GSP as an orthonormal decomposition does not exist. Any real-valued branch weight can be used (e.g., admittance magnitude, physical or electrical distance). For the case study, the underlying graph used for convolution is identical to that in [3]. That is, the branch weights of the Laplacian are chosen to be the inverse squared length ℓ^{-2} so that the eigenvalues correspond to the squared wavenumber k^2.

B. Graph Fourier Transform

A *vertex domain* function on the graph $f : \mathcal{V} \to \mathbb{R}$ can be written as a vector $\mathbf{f} \in \mathbb{R}^N$, whose i^{th} element corresponds to the evaluation of f at the i^{th} vertex [5]. For a connected, undirected graph, (1) is positive semi-definite (PSD) with a real-valued eigenvalue decomposition $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$. The graph Fourier transform (GFT) $\hat{\mathbf{f}}$ of a function $\mathbf{f}$ is given by (2).

$$\hat{\mathbf{f}} := \mathbf{U}^\top \mathbf{f} \tag{2}$$

A function is said to be a *kernel function* $g : \mathbb{R} \to \mathbb{R}$ if it is spectrally defined (3). Let g_a denote the scaled kernel.

$$g_a(\mathscr{L}) = \mathbf{U} g(a\mathbf{\Lambda})\mathbf{U}^\top \tag{3}$$

The convolution of a vertex domain function and a generating kernel (4) corresponds to a product of the two functions in the graph Fourier domain using (3) and (2).

$$\mathbf{f} * g := g(\mathscr{L})\mathbf{f} \tag{4}$$

III. Cholesky Compatable GSP Filters

This section presents a detailed set of functions for applying GSP to large, sparse power systems, given their unique compatibility with Cholesky factorization.

A. Low-Pass Spectral Graph Filters

Also known as Tikhonov regularization, this filter (5) is commonly used for signal reconstruction and can be applied with sparse methods. The low-pass filter is *refinable* because it is a self-similar rational function, which makes it useful for signal smoothing and localization.

$$\phi(\mathscr{L}) = \frac{I}{\mathscr{L} + I} \tag{5}$$

This filter meets the admissibility criterion of a scaling kernel (6), which permits certain reconstruction theorems in spectral frameworks (e.g., [6]).

$$\phi(0) = 1 \qquad \text{and} \qquad \lim_{\lambda \to \infty} \phi(\lambda) = 0 \tag{6}$$

The voltage magnitude is smooth with respect to the underlying graph, and therefore can be constructed using scaling functions (blue trace in Fig. 1 and Fig. 2). This low-pass filter can be used to study the distribution of voltage magnitude in large systems.

B. High-Pass Spectral Graph Filters

The proposed high-pass filter (7) is zero-valued at $\lambda = 0$ (red trace in Fig. 1 and Fig. 2) and serves as a container for variations in the graph below a given spatial scale.

$$\mu(\mathscr{L}) = \frac{\mathscr{L}}{\mathscr{L} + I} \tag{7}$$

Admissibility criterion of the high-pass filter (8) is defined similarly to the scaling functions (6).

$$\mu(0) = 0 \qquad \text{and} \qquad \lim_{x \to \infty} \mu(x) = 1 \tag{8}$$

High-pass filters can be used to detect a steady-state injected network current. This observation enables us to use high-pass filters to study current flows in the transmission network.

C. Band-Pass Spectral Graph Filters

The closed-form band-pass filter (9) is useful for various GSP applications and serves as a wavelet-generation kernel. Power systems applications of this filter include mode identification and forced oscillation source location [3].

$$\Psi(\mathscr{L}) = \frac{4\mathscr{L}}{(\mathscr{L} + I)^2} \tag{9}$$

This filter qualifies as a wavelet generating kernel for the SGWT, since $\Psi(0) = 0$ and the admissibility condition is satisfied. The admissibility constant $C_\Psi = 8/3$ of this band-pass filter is detailed in Appendix A.

$$\Psi(0) = 0 \qquad \text{and} \qquad \int_0^\infty \frac{\Psi^2(x)}{x} \mathrm{d}x < \infty \tag{10}$$

Notice that the injected/emitted power from the network must take this form (gray trace in Fig. 1 and Fig. 2). This observation motivates the use of band-pass filters to study power transmission in the spectral domain.

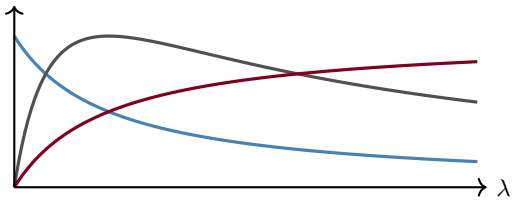

Fig. 1. A spectral plot of the low-pass (blue), high-pass (red), and band-pass (gray) filters as a function of the graph spectra.

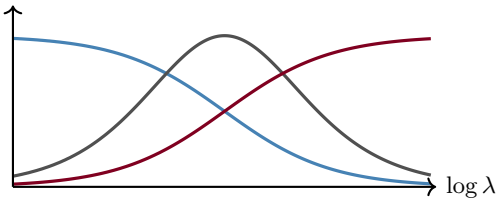

Fig. 2. A spectral plot of the low-pass (blue), high-pass (red), and band-pass (gray) filters as a function of the log-scale graph spectra.

D. Kernel Fitting

A vector fitting (VF) approach to graph convolution developed in [3] derived the generalized form of the Cholesky-based GSP filters, referred to as *Kernel Fitting* (KF). However, this can be slightly improved by adding two polynomial terms to the expansion as in [4].

Let $q \in Q$ be a real-valued pole, with a corresponding row vector in the residue matrix $\mathbf{R} \in \mathbb{R}^{|Q| \times |\mathcal{A}|}$ representing the individual contribution of each pole to each scale/dimension of a kernel (i.e., a filter bank), and let $\mathbf{d}, \mathbf{e} \in \mathbb{R}^{|\mathcal{A}|}$. A KF approximation of $\mathbf{g} : \mathbb{R} \to \mathbb{R}^{|\mathcal{A}|}$can fit multiple filters concurrently (11) using iterative pole reallocation. Once obtained, it can be reused for other graphs, as the spectral width of the graph Laplacian does not constrain the expansion.

$$\mathbf{g}(\lambda) \approx \mathbf{d} + \lambda \mathbf{e} + \sum_{q \in Q} \frac{\mathbf{r}_q}{\lambda + q} \tag{11}$$

The utility of (11) can not be overemphasized. Compared with other sparse GSP approximations, the ease of implementation and favorable computational speed have proven superior for large, sparse networks. Hence, KF performs graph-spectral approximations of kernel functions via iterative rational approximation to obtain a useful form for convolution in the vertex domain.

The vertex domain representation of (11) is obtained by substituting $\lambda \to \mathscr{L}$, which conceptually represents the domain of the function. Convolution is performed using the Cholesky decomposition to factor $\mathscr{L}$, then expressing convolution $\mathbf{f} * g$ as a sum of solutions to sparse linear systems [3]. In fact, this expanded representation is well supported by tools such as CHOLMOD [7], which can perform a symbolic factorization once and then perform repeated in-place numeric factorizations with a diagonal shift.

$$g_a(\mathscr{L}) \approx d_a I + e_a \mathscr{L} + \sum_{q \in Q} \frac{r_{q,a}}{\mathscr{L} + qI} \tag{12}$$

Crucially, we require $q > 0$ so that $\mathscr{L} + qI$ is positive definite to guarantee the existence of a Cholesky factorization. We speculate that this requirement can be eliminated by using an LU decomposition rather than Cholesky, which would permit complex poles in (12). Given the trade-off in time complexity, it remains unclear whether LU is ideal for GSP applications.

Impulse responses of a kernel at a given vertex (e.g., $\Psi_{n,a}$ for the SGWT) can be computed by convolving with δ_n (i.e., a vertex impulse). The KF approach has been tailored for the types of networks and GSP operations that engineers are likely to use, with high efficiency.

IV. Spectral Signals of Large Power Systems

This section details a known application of GSP to power systems state tracking and reconstruction, using GSP's convolutional notation.

A. Generalized State Tracking

For a smooth signal in the vertex domain, low-pass filters can be used to reconstruct from sparse or corrupted measurements. Let $\hat{\mathbf{f}}$ be our estimation of the signal $\mathbf{f}$, which could have missing or corrupted data.

$$\frac{\mathrm{d}\hat{\mathbf{f}}}{\mathrm{d}t} = (\mathbf{f} - \mathbf{J}\hat{\mathbf{f}}) * \phi \tag{13}$$

Where $\mathbf{J}$ is the diagonal sampling mask (i.e., sparsity pattern) indicating which elements of $\mathbf{f}$ have information. The mask can be time-varying, $\mathbf{J}(t)$, if the signal availability of $\mathbf{f}(t)$ varies over time. While this can also be solved using batched time series data, the online application is often more helpful in the context of monitoring and control.

We show that the optimal state tracking algorithm is equivalent to a graph convolutional representation with the scaling function ϕ (similar to Tikhonov regularization), as follows:

$$\begin{aligned} \hat{\mathbf{f}}_t - \hat{\mathbf{f}}_{t-1} &= \phi(\mathscr{L})(\mathbf{f}_t - \mathbf{J}_t\hat{\mathbf{f}}_{t-1}) \\ (I + \mathscr{L})(\hat{\mathbf{f}}_t - \hat{\mathbf{f}}_{t-1}) &= \mathbf{f}_t - \mathbf{J}_t\hat{\mathbf{f}}_{t-1} \\ (\mathbf{J}_t + \mathscr{L})\hat{\mathbf{f}}_t &= \mathbf{f}_t + \mathscr{L}\hat{\mathbf{f}}_{t-1} \\ \hat{\mathbf{f}}_t &= (\mathbf{J}_t + \mathscr{L})^{-1}(\mathbf{f}_t + \mathscr{L}\hat{\mathbf{f}}_{t-1}) \end{aligned} \tag{14}$$

This is precisely the optimal algorithm derived in [8] for online state tracking. In its convolutional representation (13), one can intuitively design the low-pass filter based on the requirements of the state estimation problem.

V. Implementation

For the three closed-form graph filters implemented in this section, the first-order pseudocode is provided. However, the k-th-order version of each filter can be computed by looping k times during the filter's application stage. This efficiently raises the kernel/filter to the k-th power.

A. Low-Pass Filter with Cholesky

Of all the filters in Section III, the low-pass filter is the easiest to implement using Cholesky solvers. The low-pass Cholesky algorithm for a single scale (Alg. 1) performs memory-efficient convolution, particularly when the graph Laplacian is prefactored for subsequent calls.

Algorithm 1 Low-Pass Convolution Algorithm

Input: $\mathscr{L}$ Graph Laplacian $\in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$
$\mathbf{f}$ Input signal $\in \mathbb{R}^{|\mathcal{V}|}$
a Scale parameter $\in \mathbb{R}_+$
Output: $\mathbf{w}$ (Low-pass coefficients)

Precompute symbolic analysis, which is independent of a
$\mathcal{S} \leftarrow \text{SymFactor}(\mathscr{L})$

Numeric factorization
$\mathcal{F} \leftarrow \text{NumFactor}(\mathscr{L} + a^{-1}\mathbf{I}, \mathcal{S})$

Apply Low-Pass Filter
$\mathbf{v} \leftarrow \text{Solve}(\mathcal{F}, \mathbf{f})$ ▷ Apply $(\mathscr{L} + a^{-1}\mathbf{I})^{-1}$
$\mathbf{w} \leftarrow \frac{1}{a}\mathbf{v}$
return $\mathbf{w}$

B. High-Pass Filter with Cholesky

The high-pass Cholesky algorithm (Alg. 2) can also be improved by pre-factoring the graph Laplacian if the function is used repeatedly. This is functionally identical to the low-pass implementation, with one additional matrix multiplication.

Algorithm 2 High-Pass Convolution Algorithm

Input: $\mathscr{L}$ Graph Laplacian $\in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$
$\mathbf{f}$ Input signal $\in \mathbb{R}^{|\mathcal{V}|}$
a Scale parameter $\in \mathbb{R}_+$
Output: $\mathbf{w}$ (High-pass coefficients)

Precompute symbolic analysis, which is independent of a
$\mathcal{S} \leftarrow \text{SymFactor}(\mathscr{L})$

Numeric factorization
$\mathcal{F} \leftarrow \text{NumFactor}(\mathscr{L} + a^{-1}\mathbf{I}, \mathcal{S})$

Apply High-Pass filter
$\mathbf{v} \leftarrow \text{Solve}(\mathcal{F}, \mathbf{f})$ ▷ Apply $(\mathscr{L} + a^{-1}\mathbf{I})^{-1}$
$\mathbf{w} \leftarrow \mathscr{L}\mathbf{v}$
return $\mathbf{w}$

C. Band-Pass Filter with Cholesky

The band-pass Cholesky algorithm (Alg. 3) can be used for the SGWT. In addition to the usual SGWT application, band-pass filters can be used analytically to extract and measure the scale of oscillation modes in large grid models [3].

Algorithm 3 Band-Pass Convolution Algorithm

Input: $\mathscr{L}$ Graph Laplacian $\in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$
$\mathbf{f}$ Input signal $\in \mathbb{R}^{|\mathcal{V}|}$
a Scale parameter $\in \mathbb{R}_+$
Output: $\mathbf{w}$ (Band-pass coefficients)

Precompute symbolic analysis, which is independent of a
$\mathcal{S} \leftarrow \text{SymFactor}(\mathscr{L})$

Numeric factorization
$\mathcal{F} \leftarrow \text{NumFactor}(\mathscr{L} + a^{-1}\mathbf{I}, \mathcal{S})$

Apply bandpass filter
$\mathbf{u} \leftarrow \text{Solve}(\mathcal{F}, \mathbf{f})$ ▷ Apply $(\mathscr{L} + a^{-1}\mathbf{I})^{-1}$
$\mathbf{v} \leftarrow \text{Solve}(\mathcal{F}, \mathbf{u})$ ▷ Apply $(\mathscr{L} + a^{-1}\mathbf{I})^{-1}$
$\mathbf{w} \leftarrow \frac{4}{a}\mathscr{L}\mathbf{v}$
return $\mathbf{w}$

D. General Convolution Implementation

Performing a graph convolution on a vertex domain function $\mathbf{f}$ with a KF function g is achieved with Alg. 4. This algorithm generalizes convolution on graphs to any KF kernel, including kernels/filter banks derived from empirical data.

Algorithm 4 General KF Convolution

Input: $\mathscr{L}$ Graph Laplacian $\in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$
$\mathbf{F}$ Input signal tensor $\in \mathbb{R}^{|\mathcal{V}|\times T}$
$\mathbf{d}, \mathbf{e}$ Polynomial terms $\in \mathbb{R}^{|\mathcal{A}|}$
$\mathbf{q}$ Pole locations $\in \mathbb{R}^{|\mathcal{Q}|}$
$\mathbf{R}$ Residue matrix $\in \mathbb{R}^{|\mathcal{Q}|\times|\mathcal{A}|}$
Output: $\mathbf{W}$ Output tensor $\in \mathbb{R}^{|\mathcal{V}|\times T\times|\mathcal{A}|}$

Precompute symbolic & numeric factorization
$\mathcal{S} \leftarrow \text{SymFactor}(\mathscr{L})$
for $i \leftarrow 1$ **to** $|\mathcal{Q}|$ **do**
  $\mathcal{F}_i \leftarrow \text{NumFactor}(\mathscr{L} + q_i\mathbf{I}, \mathcal{S})$ ▷ Reuse symbolic
end for

Initialize output
for $j \leftarrow 1$ **to** $|\mathcal{A}|$ **do**
  $\mathbf{W}_{:,:,j} \leftarrow d_j\mathbf{F} + e_j\mathscr{L}\mathbf{F}$ ▷ Weigh by $\mathbf{d} + \lambda\mathbf{e}$
end for

Rational convolution
for $i \leftarrow 1$ **to** $|\mathcal{Q}|$ **do**
  $\mathbf{X} \leftarrow \text{Solve}(\mathcal{F}_i, \mathbf{F})$ ▷ Batch solve
  Update filter channels
  for $j \leftarrow 1$ **to** $|\mathcal{A}|$ **do**
    $\mathbf{W}_{:,:,j} \leftarrow \mathbf{W}_{:,:,j} + R_{ij}\mathbf{X}$ ▷ Weigh by $\mathbf{R}$
  end for
end for
return $\mathbf{W}$

TABLE I
PER-SCALE BENCHMARK COMPUTATION TIME OF GRAPH CONVOLUTION

Filter	2k Buses	70k Buses	80k Buses	82k Buses
Low Pass ϕ	0.022 ms	0.804 ms	0.922 ms	0.942 ms
High Pass μ	0.047 ms	2.390 ms	2.773 ms	2.799 ms
Band Pass Ψ	0.049 ms	2.409 ms	2.802 ms	2.871 ms

E. Dynamic Graph Updates

All convolution algorithms in this work are implemented using the Cholesky factorization and a sparse direct solution method. We leverage this to support dynamic graph modifications (Alg. 5) via low-rank updates to the shifted factorizations.

Algorithm 5 Dynamic Graph Updates for Convolution

Input: $\{\mathcal{F}_k\}$ Set of cached Cholesky factors
u, v Branch endpoint indices
w Branch weight $\in \mathbb{R}_+$
Output: Updated set $\{\mathcal{F}_k\}$

Initialize the new branch as a vector
$\mathbf{c} \leftarrow \sqrt{w}(\delta_u - \delta_v)$

Update topology with a rank-1 update
for $k \leftarrow 1$ **to** $|\mathcal{Q}|$ **do**
  Updown($\mathcal{F}_k$, $\mathbf{c}$) ▷ Update $\mathcal{F}_k$ for $+\mathbf{c}\mathbf{c}^T$
end for

VI. Case Study

Large synthetic grid models [9] are used to validate and visualize the proposed convolution algorithms. The performance and time complexity of the KF implementation are further assessed in [3].

A. Computation Speed

This subsection measures the computation time of the proposed implementation for each corresponding graph filter. The convolution algorithms implemented in Section V are applied to the synthetic Texas case (2000 buses), the synthetic eastern case (70k buses), the synthetic east-west case (80k buses), and the synthetic USA case (82k buses). All of the proposed filter implementations (including KF) are compatible with any graph, so the kernel for the convolution procedure is identical for each graph.

The runtimes for a convolution of a randomly generated input signal with each filter are recorded in Table I. The performance of the proposed convolutional algorithms is evaluated by measuring the average computation time across 50 filter scales, computed concurrently. Experimentation revealed that the filter size did not affect performance.

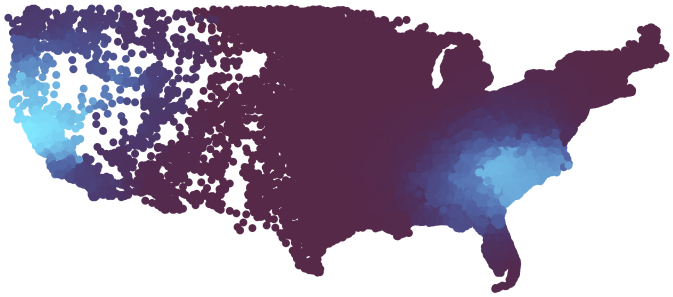

Fig. 3. A low-pass impulse response of two concurrent impulses, localized on the west and east coast on the 82k bus synthetic USA case. The intensity of the blue color depicts the intensity of the impulse.

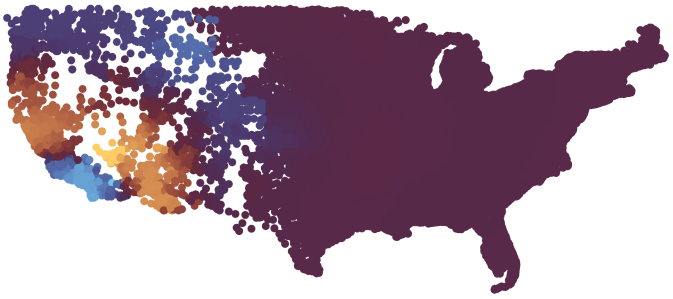

Fig. 4. Higher order spectral graph wavelet localized in the southwest of the 82k bus synthetic USA case. The intensities of blue and orange indicate the positive and negative components of the impulse, respectively.

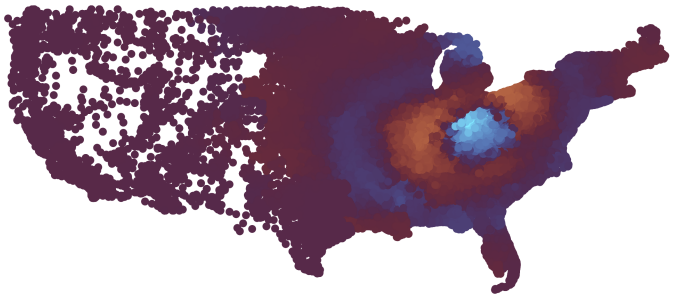

Fig. 5. Signal X: A wave on the synthetic USA grid originating from the Cincinnati area obtained with a band-pass convolution. The intensities of blue and orange indicate the positive and negative components of the impulse, respectively.

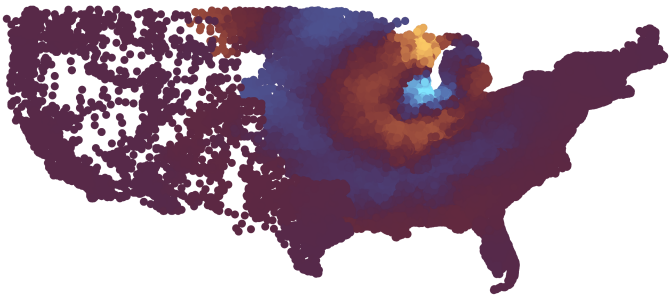

Fig. 6. Signal Y: A wave on the synthetic USA grid originating from the Chicago area obtained with a band-pass convolution. The intensities of blue and orange indicate the positive and negative components of the impulse, respectively.

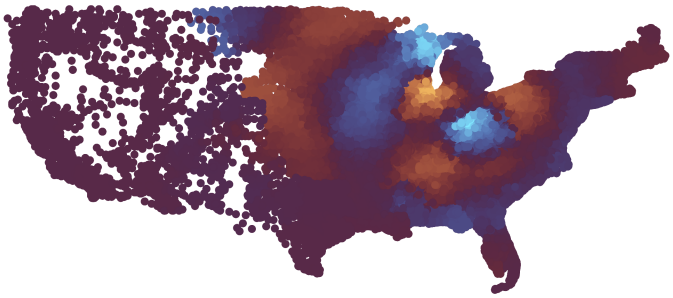

Fig. 7. Signal $Z = X - Y$, the interference pattern emergent from the wave sources at Cincinnati (Y) and Chicago (X) from Z. Destructive interference is most pronounced in the region near the state of Louisiana.

B. Voltage Potential Impulse Response

We visualize the impulse responses of two kernels (low- and band-pass filters) to illustrate how graph convolutional behavior varies across the network. We first consider the response of impulses at extreme ends of the network (Fig. 3). The scales are chosen so that the resulting signals have negligible shared energy; thus, we have effectively identified the set of buses and the regions associated with each impulse.

For the impulse response of a band-pass filter (Fig. 4), we see that the wavelet is smooth and attenuates from the incident bus in the radial direction. We also generate the impulse response of a higher-order (sharper) band-pass filter (Fig. 5), demonstrating that the response becomes more global in the phase space of the vertex domain than that of a kernel with softer spectral features. The resulting impulse response of a band-pass filter is equivalent to a localized wavelet function in the SGWT framework.

C. Interference Patterns

The interference patterns of two localized wavelets can be readily generated in the proposed framework. An injected current impulse δ_i at node i and impulse δ_j at node j have opposing polarities, and we seek to compute their interference pattern at each node in the network. Using the band-pass response of Fig. 5 and Fig. 6, we compute the interference by convolving the impulses concurrently using $\mathbf{f} = \delta_i - \delta_j$ as the function input. The interference pattern between the two buses (analogous to a transaction of current) in Fig. 7 consists of constructive and destructive patterns.

VII. Conclusion

This paper presents a practical framework for implementing graph convolutions on large electrical networks using modern GSP techniques. The proposed filters can be used for signal reconstruction, modal analysis, and other emerging analyses

in GSP. Compared with existing sparse applications of GSP in the literature, our framework is superior in speed and ease of implementation, thereby expanding the accessibility of GSP for power system researchers.

Future work should generalize the Cholesky-based KF method to support complex poles by using LU decomposition rather than Cholesky decomposition. While computing LU is slower than an LDL^T, an LU factorization permits a broader class of kernel functions for convolution.

Methods in GSP continue to aid stability studies by providing practical sparse methods for analyzing, synthesizing, and transforming signals in large power systems. To accelerate the adoption of GSP, this paper enhances its capabilities and accessibility for power systems engineers.

APPENDIX A
ADMISSIBILITY CRITERION

$$\begin{aligned} C_\Psi &= \int_0^\infty \frac{\Psi^2(x)}{x}\mathrm{d}x = \int_0^\infty \frac{1}{x}\left[\frac{4x}{(x+1)^2}\right]^2 \mathrm{d}x \\ &= \int_0^\infty \frac{16x}{(x+1)^4}\,\mathrm{d}x = \frac{8}{3} < \infty \end{aligned} \tag{15}$$

REFERENCES

[1] G. Leus, A. G. Marques, J. M. Moura, A. Ortega, and D. I. Shuman, "Graph signal processing: History, development, impact, and outlook," *IEEE Signal Processing Magazine*, vol. 40, no. 4, 2023.

[2] M. A. Hasnat and M. Rahnamay-Naeini, "Power System State Recovery using Local and Global Smoothness of its Graph Signals," in *2022 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 01–05, July 2022.

[3] L. Lowery, J. Baek, and A. Birchfield, "Using spectral graph wavelets to analyze large power system oscillation modes," *59th Hawaii International Conference on System Sciences (HICSS)*, 2026.

[4] B. Gustavsen, "Improving the pole relocating properties of vector fitting," *IEEE Transactions on Power Delivery*, vol. 21, no. 3, pp. 1587–1592, 2006.

[5] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, May 2013.

[6] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, 2011.

[7] Y. Chen, T. A. Davis, W. W. Hager, and S. Rajamanickam, "Algorithm 887: Cholmod, supernodal sparse cholesky factorization and update/downdate," *ACM Trans. Math. Softw.*, vol. 35, oct 2008.

[8] L. Stanković and E. Sejdić, *Vertex-frequency analysis of graph signals*. Springer, 2019.

[9] A. B. Birchfield, T. Xu, K. M. Gegner, K. S. Shetye, and T. J. Overbye, "Grid structural characteristics as validation criteria for synthetic networks," *IEEE Trans. on Power Sys.*, vol. 32, no. 4, 2017.