



HAL
open science

Learning Combination of Graph Filters for Graph Signal Modeling

Fei Hua, Cédric Richard, Chen Jie, Haiyan Wang, Pierre Borgnat, Paulo Gonçalves

► **To cite this version:**

Fei Hua, Cédric Richard, Chen Jie, Haiyan Wang, Pierre Borgnat, et al.. Learning Combination of Graph Filters for Graph Signal Modeling. *IEEE Signal Processing Letters*, 2019, 26 (12), pp.1912-1916. 10.1109/lsp.2019.2954981 . hal-02367868

HAL Id: hal-02367868

<https://inria.hal.science/hal-02367868>

Submitted on 14 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Designing Convex Combination of Graph Filters

Fei Hua, Cédric Richard, *Senior Member, IEEE*, Jie Chen, *Senior Member, IEEE*, Haiyan Wang, *Member, IEEE*, Pierre Borgnat, *Member, IEEE*, and Paulo Gonçalves, *Member, IEEE*

Abstract—In this letter, we study the problem of parametric modeling of network-structured signals with graph filters. Unlike the popular polynomial graph filters, which are based on a single graph shift operator, we consider convex combinations of graph shift operators **particularly adapted to directed graphs**. As the resulting modeling problem is not convex, we reformulate it as a convex optimization problem which can be solved efficiently. Experiments on real-world data structured by undirected and directed graphs are conducted. The results show the effectiveness of this method compared to other methods reported in the literature.

Index Terms—graph filter, shift operator, random walk, parametric modeling, convex combination.

I. INTRODUCTION

GRAPH signal processing (GSP) is a new mathematical tool dedicated to signals with irregular structures which has recently attracted growing research interest [1]–[4]. A key ingredient of GSP is the graph shift operator, which accounts for the topology of the graph in the vertex domain. It also provides a basis for the graph-Fourier transform, and allows to analyze graph signals in the frequency domain. Most operations on graph signals, e.g., filtering and sampling, rely on graph shift operators.

The Laplacian matrix and the adjacency matrix are typical choices of graph shift operators for undirected graphs [1] or directed graphs [5]. There is, however, no consensus about which one is the most appropriate according to an application. For instance, in [6], the authors introduce an isometric graph shift operator for stationary graph signals. In [7], the authors define a set of shift operators that preserve the energy content of graph signals in the frequency domain. The authors in [8] define an extension of the symmetric Laplacian matrix for directed graphs. Methods of constructing graph-Fourier type bases on directed graphs have also been proposed by means of minimizing the Lovász extension of the graph cut size [9], or minimizing the spectral dispersion function [10]. **The authors in [11]–[13] propose to use the random walk operator as a shift**

operator for directed graphs. They show in [12], [13] how a combination of two operators (here the random walk and the time reversed random walk) can perform better than a single shift operator, yet the combination has not been formulated so as to lead to a convex combination problem.

In this work, we consider generally the design of polynomial graph filters that consist of a combination of shift operators to model graph signals. This, however, gives rise to a challenging non-convex optimization problem. We address this issue by relaxing the original problem and reformulating it as a convex one. The new model has extra degrees of freedom that lead to better modeling performance. This paper is organized as follows. In Section II, we give some necessary preliminaries on graph signal processing and formulate the modeling problem. We relax this problem and introduce its convex counterpart in Section III. The algorithm is also provided in this section. Numerical results are reported in Section IV. We conclude this work in Section V.

Notation: We use normal font letters to denote scalars, boldface lowercase letters to denote column vectors, and boldface uppercase letters to denote matrices. The ℓ -th column of a matrix \mathbf{M} is denoted by $[\mathbf{M}]_{\cdot,\ell}$. We use $\text{diag}\{d_1, \dots, d_N\}$ to denote a diagonal matrix with entries d_1, \dots, d_N . Operator $\text{col}\{\cdot\}$ stacks the column vector entries on top of each other. Operator $\lambda_{\max}\{\cdot\}$ represents the maximum eigenvalue of its matrix argument.

II. PARAMETRIC MODELING VIA GRAPH FILTERS

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ denote a graph with a set \mathcal{V} of N vertices and a set \mathcal{E} of edges such that $\{k, \ell\} \in \mathcal{E}$ if vertices k and ℓ are connected. A graph \mathcal{G} can be represented by its adjacency matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$ whose (k, ℓ) -th entry $w_{k\ell}$ assigns a weight to the relation between vertices k and ℓ such that

$$w_{k\ell} > 0, \text{ if } \{k, \ell\} \in \mathcal{E}, \quad \text{and} \quad w_{k\ell} = 0, \text{ otherwise.} \quad (1)$$

We define the diagonal degree matrix $\mathbf{D} = \text{diag}\{d_1, \dots, d_N\}$ whose k -th diagonal entry is the sum of the k -th row of \mathbf{W} , i.e., $d_k = \sum_{\ell=1}^N w_{k\ell}$ for all $k = 1, \dots, N$. The combinatorial Laplacian matrix is defined as $\mathbf{L} \triangleq \mathbf{D} - \mathbf{W}$. For an undirected graph, \mathbf{L} is a symmetric positive semi-definite matrix. The normalized Laplacian matrix is given by $\mathbf{L}_{\text{norm}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}}$. For a directed graph, the random walk operator is an $N \times N$ probability transition matrix defined as $\mathbf{P} = \mathbf{D}^{-1} \mathbf{W}$. If the random walk is irreducible, it has a unique stationary distribution $\boldsymbol{\pi}$ that satisfies $\boldsymbol{\pi} \mathbf{P} = \boldsymbol{\pi}$. Its time reversed ergodic random walk is given by $\mathbf{P}^* = \boldsymbol{\Pi}^{-1} \mathbf{P}^T \boldsymbol{\Pi}$, where $\boldsymbol{\Pi} = \text{diag}\{\pi_1, \dots, \pi_N\}$.

This work was supported through the UCA JEDI Investments in the Future project managed by the ANR with the reference number ANR-15-IDEX-0001, the ACADEMICS grant given by the IDEXLYON project of the Université de Lyon, and by NSFC grants 61571365, 61671386, 61771396 and National Key R&D Program of China (2016YFC1400200).

F. Hua, J. Chen and H. Wang are with School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China. F. Hua is also with Université Côte d'Azur, OCA, CNRS, France. H. Wang is also with School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an, China (emails: fei.hua@oca.eu; dr.jie.chen@ieee.org; hywang@sust.edu.cn).

C. Richard is with Université Côte d'Azur, OCA, CNRS, France (email: cedric.richard@unice.fr).

P. Borgnat is with Université de Lyon, ENSL, CNRS, Laboratoire de Physique, Lyon, France. P. Gonçalves is with Université de Lyon, Inria, CNRS, ENSL, LIP, Lyon, France (emails: first.last@ens-lyon.fr).

A graph shift operator \mathbf{S} is defined as an $N \times N$ matrix which captures the graph topology such that its entries satisfy:

$$s_{k\ell} > 0, \text{ if } \{k, \ell\} \in \mathcal{E} \text{ or } k = \ell, \text{ and } s_{k\ell} = 0, \text{ otherwise.} \quad (2)$$

In that sense, any matrix which satisfies condition (2) can be used as a graph shift operator. **This property of locality allows a distributed implementation [14]–[16]. There are, however, other types of shift operators which do not satisfy (2); See, e.g., the isometric operator in [6].** Given a graph shift operator \mathbf{S} , we can define a linear shift-invariant graph filter as follows:

$$\mathbf{H} \triangleq \sum_{\ell=0}^{L-1} h_{\ell} \mathbf{S}^{\ell} \quad (3)$$

with $h_{\ell} \in \mathbb{R}$ the filter coefficients. Consider a graph signal defined as $\mathbf{x} = [x_1, \dots, x_N]^{\top} \in \mathbb{R}^N$ where x_k is the signal sample at vertex k . The filtering process can be expressed as:

$$\mathbf{y} = \mathbf{H}\mathbf{x} = \sum_{\ell=0}^{L-1} h_{\ell} \mathbf{S}^{\ell} \mathbf{x} = \mathbf{M}\mathbf{h}, \quad (4)$$

with $\mathbf{h} = \text{col}\{h_{\ell}\}_{\ell=0}^{L-1}$, \mathbf{y} the $N \times 1$ output vector, and \mathbf{M} the $N \times L$ matrix whose ℓ -th column is given by $[\mathbf{M}]_{\cdot, \ell} = \mathbf{S}^{\ell-1} \mathbf{x}$.

We consider a graph where each vertex k has access to a measurement y_k and a regression data x_k , assumed to be related by the linear model:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{e} = \mathbf{M}\mathbf{h} + \mathbf{e} \quad (5)$$

for some unknown $N \times 1$ vector \mathbf{h} , where \mathbf{e} denotes an i.i.d. zero-mean Gaussian noise. The model parameters \mathbf{h} can be learned by minimizing the following cost function:

$$J(\mathbf{h}) = \|\mathbf{y} - \mathbf{M}\mathbf{h}\|^2 = \sum_{k=1}^N (y_k - \mathbf{h}^{\top} \mathbf{m}_k)^2 \quad (6)$$

with \mathbf{m}_k^{\top} the k -th row vector of \mathbf{M} . As there is no rule on how to choose operator \mathbf{S} , some recent studies have investigated combinations of two shift operators [11], [12]. Following the same strategy in this letter, we consider convex combinations of shift operators:

$$\mathbf{S} = \alpha \mathbf{S}_1 + (1 - \alpha) \mathbf{S}_2 \quad (7)$$

where $\alpha \in (0, 1)$ is a combination factor to estimate while minimizing (6). Possible choices for \mathbf{S}_1 and \mathbf{S}_2 are, e.g., the adjacency \mathbf{W} and the Laplacian \mathbf{L} matrices for undirected graphs, the random walk matrix \mathbf{P} and its time reversed counterpart \mathbf{P}^* for directed graphs. **In particular, when $\mathbf{S} = \frac{\mathbf{P} + \mathbf{P}^*}{2}$, the random walk is reversible.** Substituting (7) into (4)–(6) leads to:

$$J(\mathbf{h}, \alpha) = \left\| \mathbf{y} - \sum_{\ell=0}^{L-1} h_{\ell} [\alpha \mathbf{S}_1 + (1 - \alpha) \mathbf{S}_2]^{\ell} \mathbf{x} \right\|^2 \quad (8)$$

which is not jointly convex with respect to α and \mathbf{h} . To address this issue, we observe that α^{ℓ} exponentially decreases to zero as ℓ increases since $\alpha \in (0, 1)$. Neglecting the higher orders of α , cost (8) can be reasonably approximated by:

$$J'(\mathbf{h}, \alpha) = \left\| \mathbf{y} - \left[\alpha \sum_{\ell=0}^{L-1} h_{\ell} \mathbf{S}_1^{\ell} + (1 - \alpha) \sum_{\ell=0}^{L-1} h_{\ell} \mathbf{S}_2^{\ell} \right] \mathbf{x} \right\|^2 \quad (9)$$

where the convex combination of the graph shift operators \mathbf{S}_1 and \mathbf{S}_2 has been replaced by a convex combination of two graph filters, say, $\mathbf{H}_1 = \sum_{\ell=0}^{L-1} h_{\ell} \mathbf{S}_1^{\ell}$ and $\mathbf{H}_2 = \sum_{\ell=0}^{L-1} h_{\ell} \mathbf{S}_2^{\ell}$. Cost function (9) is still not jointly convex w.r.t. α and \mathbf{h} since it is a function of $\alpha \mathbf{h}$. In the next section, we shall reformulate the optimization problem into a convex one which can be efficiently addressed.

III. JOINTLY ESTIMATING GRAPH FILTERS COEFFICIENTS AND THE COMBINATION COEFFICIENT

Before proceeding with (9), for clarity, we begin with (6) where we introduce an ℓ_2 -regularization term as follows:

$$J_{\text{reg}}(\mathbf{h}) = \frac{1}{2} \|\mathbf{h}\|^2 + \frac{1}{2\mu} \sum_{k=1}^N (y_k - \mathbf{h}^{\top} \mathbf{m}_k)^2 \quad (10)$$

with μ a small positive value to control the trade-off between the regularization and the fitting terms. The optimum \mathbf{h}^* can be obtained by solving the following optimization problem:

$$\mathbf{h}^* = \arg \min_{\mathbf{h} \in \mathbb{R}^L} \frac{1}{2} \|\mathbf{h}\|^2 + \frac{1}{2\mu} \sum_{k=1}^N e_k^2 \quad (11)$$

$$\text{subject to } e_k = y_k - \mathbf{h}^{\top} \mathbf{m}_k, \quad k \in \{1, \dots, N\}.$$

The dual problem can be derived by introducing Lagrange multipliers λ_k as follows:

$$\mathcal{L}(\mathbf{h}, \mathbf{e}, \boldsymbol{\lambda}) = \frac{1}{2} \|\mathbf{h}\|^2 + \frac{1}{2\mu} \sum_{k=1}^N e_k^2 - \sum_{k=1}^N \lambda_k (e_k - y_k + \mathbf{h}^{\top} \mathbf{m}_k). \quad (12)$$

The optimality conditions w.r.t. the primal variables lead to:

$$\begin{cases} \mathbf{h}^* &= \sum_{k=1}^N \lambda_k^* \mathbf{m}_k \\ e_k^* &= \mu \lambda_k^* \end{cases} \quad (13)$$

where the dual variables are estimated by solving:

$$\boldsymbol{\lambda}^* = \arg \max_{\boldsymbol{\lambda} \in \mathbb{R}^N} -\frac{1}{2} \boldsymbol{\lambda}^{\top} (\mathbf{M}\mathbf{M}^{\top} + \mu \mathbf{I}) \boldsymbol{\lambda} + \boldsymbol{\lambda}^{\top} \mathbf{y} \quad (14)$$

which is a quadratic programming (QP) problem. We consider now the cost function in (9). Graph filters \mathbf{H}_1 and \mathbf{H}_2 are coupled through the filter coefficients \mathbf{h} and α . We relax this coupling by considering the following combination:

$$\mathbf{H}_c \triangleq \sum_{\ell=0}^{L-1} (h_{1,\ell} \mathbf{S}_1^{\ell} + h_{2,\ell} \mathbf{S}_2^{\ell}). \quad (15)$$

We shall use $\mathbf{h}_1 = \text{col}\{h_{1,\ell}\}_{\ell=0}^{L-1}$, $\mathbf{h}_2 = \text{col}\{h_{2,\ell}\}_{\ell=0}^{L-1}$ to denote the corresponding filters coefficients.

The linear model (5) can be rewritten as:

$$\mathbf{y} = \mathbf{M}_1 \mathbf{h}_1 + \mathbf{M}_2 \mathbf{h}_2 + \mathbf{e} \quad (16)$$

where $[\mathbf{M}_1]_{\cdot, \ell} = \mathbf{S}_1^{\ell-1} \mathbf{x}$, $[\mathbf{M}_2]_{\cdot, \ell} = \mathbf{S}_2^{\ell-1} \mathbf{x}$, respectively. We shall further use $\mathbf{m}_{1,k}^{\top}$ and $\mathbf{m}_{2,k}^{\top}$ to denote the k -th row vectors of \mathbf{M}_1 and \mathbf{M}_2 , respectively. This reformulation allows to decouple filters \mathbf{H}_1 and \mathbf{H}_2 , and to introduce extra degrees of freedom. In order to balance their respective contributions,

$$J(\alpha) = \begin{cases} \min_{\mathbf{h}_1, \mathbf{h}_2 \in \mathbb{R}^L} F(\alpha, \mathbf{h}_1, \mathbf{h}_2) = \frac{1}{2} \left(\frac{1}{\alpha} \|\mathbf{h}_1\|^2 + \frac{1}{1-\alpha} \|\mathbf{h}_2\|^2 \right) + \frac{1}{2\mu} \sum_{k=1}^N e_k^2 \\ \text{subject to } e_k = y_k - \mathbf{h}_1^\top \mathbf{m}_{1,k} - \mathbf{h}_2^\top \mathbf{m}_{2,k}, \quad k \in \{1, \dots, N\} \end{cases} \quad (19)$$

we propose to consider the following optimization problem inherited from the multi-kernel learning literature [17]–[19]:

$$\mathbf{h}_1^*, \mathbf{h}_2^*, \alpha^* = \arg \min_{\mathbf{h}_1, \mathbf{h}_2, \alpha} \frac{1}{2} \left(\frac{\|\mathbf{h}_1\|^2}{\alpha} + \frac{\|\mathbf{h}_2\|^2}{1-\alpha} \right) + \frac{1}{2\mu} \sum_{k=1}^N e_k^2$$

subject to :

$$e_k = y_k - \mathbf{h}_1^\top \mathbf{m}_{1,k} - \mathbf{h}_2^\top \mathbf{m}_{2,k}, \quad k \in \{1, \dots, N\}$$

$$0 < \alpha < 1, \quad (17)$$

where α allows to adjust the balance between \mathbf{h}_1 and \mathbf{h}_2 via their norms. Indeed, the solution of problem (17) tends to that of problem (10) with \mathbf{h}_2 (resp., \mathbf{h}_1) as α tends to 0 (resp., 1). Note that function $\|\mathbf{h}\|^2/\alpha$, called the perspective function, is jointly convex w.r.t. \mathbf{h} and α [20]. It follows that problem (17) is convex w.r.t. $\mathbf{h}_1, \mathbf{h}_2$ and α .

In order to solve problem (17), we consider the following constrained optimization problem:

$$\min_{\alpha} J(\alpha) \quad \text{subject to } 0 < \alpha < 1 \quad (18)$$

where $J(\alpha)$ is given by (19). Problem (18) is an optimization problem that is jointly convex w.r.t. $\alpha, \mathbf{h}_1, \mathbf{h}_2$. It can be solved with a two-step procedure w.r.t. $\mathbf{h}_1, \mathbf{h}_2$ and α successively.

A. Solving w.r.t. $\mathbf{h}_1, \mathbf{h}_2$

The Lagrangian of problem (19) can be written as:

$$\mathcal{L}' = \frac{1}{2} \left(\frac{1}{\alpha} \|\mathbf{h}_1\|^2 + \frac{1}{1-\alpha} \|\mathbf{h}_2\|^2 \right) + \frac{1}{2\mu} \sum_{k=1}^N e_k^2 - \sum_{k=1}^N \lambda_k \left(e_k - y_k + \mathbf{h}_1^\top \mathbf{m}_{1,k} + \mathbf{h}_2^\top \mathbf{m}_{2,k} \right). \quad (20)$$

The optimality conditions for \mathcal{L}' w.r.t. the primal variables are:

$$\begin{cases} \mathbf{h}_1^* &= \alpha \sum_{k=1}^N \lambda_k^* \mathbf{m}_{1,k} \\ \mathbf{h}_2^* &= (1-\alpha) \sum_{k=1}^N \lambda_k^* \mathbf{m}_{2,k} \\ e_k^* &= \mu \lambda_k^* \end{cases} \quad (21)$$

Note that coefficients \mathbf{h}_1^* and \mathbf{h}_2^* are coupled through α in the dual domain. Substituting (21) into (20) yields:

$$\lambda^* = \arg \max_{\lambda \in \mathbb{R}^N} -\frac{1}{2} \lambda^\top (\mathbf{R}_\alpha + \mu \mathbf{I}) \lambda + \lambda^\top \mathbf{y}$$

with $\mathbf{R}_\alpha = \alpha \mathbf{M}_1 \mathbf{M}_1^\top + (1-\alpha) \mathbf{M}_2 \mathbf{M}_2^\top$. (22)

Problem (22) is a QP problem which can be efficiently solved. Given λ^* , coefficients \mathbf{h}_1^* and \mathbf{h}_2^* can be computed with (21).

B. Solving w.r.t. α

First, note that function:

$$f_{p,q}(\alpha) = \frac{p}{\alpha} + \frac{q}{1-\alpha} \quad \text{with } p, q \geq 0 \quad (23)$$

is convex over $0 < \alpha < 1$. It can be checked that its optimum is given by:

$$\alpha^* = (1 + \sqrt{q/p})^{-1}. \quad (24)$$

Then, considering (19), and substituting $\mathbf{h}_1^*, \mathbf{h}_2^*$ from (21) in (23), the optimum value α_i^* at iteration i is provided by:

$$\alpha_i^* = \left(1 + \frac{1 - \alpha_{i-1}^*}{\alpha_{i-1}^*} \sqrt{\frac{\lambda^{*\top} \mathbf{M}_2 \mathbf{M}_2^\top \lambda^*}{\lambda^{*\top} \mathbf{M}_1 \mathbf{M}_1^\top \lambda^*}} \right)^{-1} \quad (25)$$

where α_{i-1}^* is the optimum value obtained from the previous iteration. The algorithm can be stopped based on Karush-Kuhn-Tucker conditions, or the duality gap equals to zero. In practice, a numerical error tolerance can be specified by the user. The whole optimization procedure is summarized in Algorithm 1.

Algorithm 1

Input: $x, y, \mathcal{S}_1, \mathcal{S}_2, L$.

Initialize: randomly choose $0 < \alpha_{-1}^* < 1$, compute $\mathbf{M}_1, \mathbf{M}_2$.

Repeat:

- 1: solve (22) with a generic QP solver to get λ^*
compute $\mathbf{h}_1^*, \mathbf{h}_2^*$ from (21),
- 2: update α_i^* by using (25).

Until: stopping condition is satisfied.

Output: $\mathbf{h}_1^*, \mathbf{h}_2^*, \alpha^*$.

IV. NUMERICAL RESULTS

We shall now present a performance comparison of our approach and other methods reported in the literature. We tested graph filter models (4), (8), (9) and (17) for modelling real-world data. We considered the Molène temperature data set of hourly weather observations collected during January 2014 in Brittany (France) [21] for undirected graphs, and the data set of the political blogs of the 2004 US presidential election [22] for directed graphs. Operators \mathcal{S}_1 and \mathcal{S}_2 that were used with models (8), (9), (17) are given in Table I, where \mathbf{W}_{norm} denotes the normalized adjacency matrix $\mathbf{W}_{\text{norm}} = \mathbf{W} / |\lambda_{\max}\{\mathbf{W}\}|$. MATLAB function quadprog was used to solve the QP problem (22). The stopping criteria was set as the difference between two successive estimates of α smaller than 10^{-3} . The graph filter degree was set to $L = 10$.

Undirected graph: The Molène data set consists of 32 vertices, with 744 observations each. The undirected graph was generated by using GSPBOX [23]. Each vertex was connected to its 6 nearest neighbors. The modeling problem was to learn

graph filters \hat{H} based on models (4), (8), (9) and (17). Here y is the original graph signals, and x is the random sampling of y . We use p to denote the proportion of known signals. The reconstruction error is defined as:

$$\text{error} = \frac{\|\hat{H}x - y\|^2}{\|y\|^2}. \quad (26)$$

Figures 1 and 2 depict the reconstruction error for different proportions of known temperatures within Case 1 and Case 2, respectively. In Case 1, we observe in Figure 1 that the combination models (8) and (9) performed better than filters based on single matrix. In Case 2, the combination models (8) and (9) performed as well as the filter based on L_{norm} , which means that the optimal α in that case was close to 0. However, it can be observed that the proposed algorithm achieved the best performance in both cases.

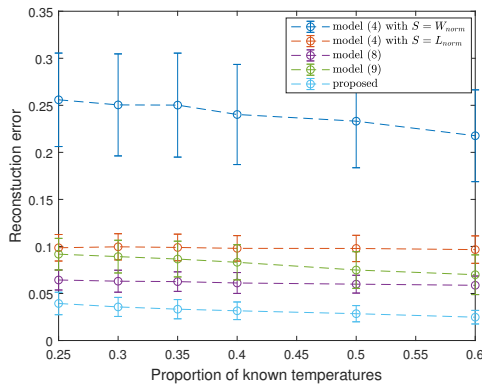


Fig. 1: Reconstruction errors for different proportions of known temperatures: Case 1.

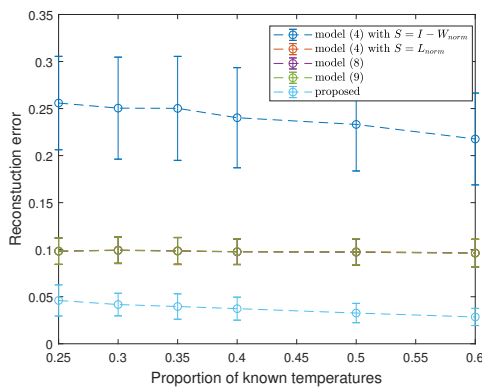


Fig. 2: Reconstruction errors for different proportions of known temperatures: Case 2.

TABLE I: Shift operators used in the experiments.

Graph type	S_1	S_2
Undirected, Case 1	W_{norm}	L_{norm}
Undirected, Case 2	$I - W_{\text{norm}}$	L_{norm}
Directed	P	P^*

Directed graph: The political blogs data set consists of 1224 blogs where each blog is either conservative, and labeled as +1, or liberal, and labeled as -1. This data set can be represented by a directed graph where vertices represent blogs, and a directed edge is considered to be present from vertex i to vertex j if there is a hyperlink from blog i to j . We consider a strongly connected part of this graph composed of 793 blogs, in which 351 are liberal and the remaining conservative. After learning the filter model \hat{H} , the reconstructed labels resulting from a random sampling x were given by $\hat{y} = \text{sgn}(\hat{H}x)$. Figure 3 reports the reconstruction accuracy for different proportions of known labels. The results are based on 100 realizations of random sampling for each proportion. Observe that the combination models performed better than the filters based on W_{norm} or random walk P , and the proposed model (17) performed slightly better than the two other combinations. In terms of computational cost, we compared the running times for designing the different combination models on MATLAB R2018a with Intel Core i5-8500 @ 3.00GHz and 8G RAM. Table II reports the averaged CPU time for a single learning.

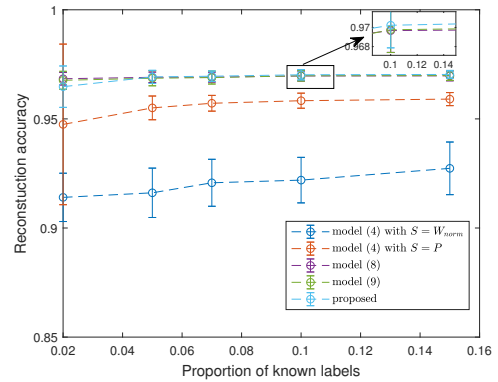


Fig. 3: Reconstruction accuracy for different proportions of known labels.

TABLE II: Averaged CPU time for different models.

Model	CPU time (sec.)
model (8)	2.6763
model (9)	5.6016
model (17)	0.7562

V. CONCLUSION

In this work, we investigated different models for combining graph filters. In particular, we introduced a convex combination of graph filters. We formulated the corresponding modeling problem as a convex optimization problem and derived a two-step optimization procedure in the dual domain. Numerical results on real-world data, for undirected and directed graphs, demonstrated the efficiency and robustness of the proposed method compared with models and methods reported in the literature. These experiments also showed that designing combinations of graph filters with the proposed algorithm is significantly more computationally efficient.

REFERENCES

- [1] 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 Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.
- [2] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 80–90, 2014.
- [3] A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.
- [4] P. Djurić and C. Richard, *Cooperative and Graph Signal Processing: Principles and Applications*. Academic Press, Elsevier, 2018.
- [5] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1644–1656, 2013.
- [6] B. Girault, P. Gonçalves, and É. Fleury, "Translation on graphs: An isometric shift operator," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2416–2420, 2015.
- [7] A. Gavili and X.-P. Zhang, "On the shift operator, graph frequency, and optimal filtering in graph signal processing," *IEEE Transactions on Signal Processing*, vol. 65, no. 23, pp. 6303–6318, 2017.
- [8] R. Singh, A. Chakraborty, and B. Manoj, "Graph Fourier transform based on directed laplacian," in *Proc. IEEE International Conference on Signal Processing and Communications (SPCOM)*, 2016, pp. 1–5.
- [9] S. Sardellitti, S. Barbarossa, and P. Di Lorenzo, "On the graph Fourier transform for directed graphs," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 6, pp. 796–811, 2017.
- [10] R. Shafipour, A. Khodabakhsh, G. Mateos, and E. Nikolova, "A directed graph Fourier transform with spread frequency components," *IEEE Transactions on Signal Processing*, vol. 67, no. 4, pp. 946–960, 2019.
- [11] A. Anis, A. Gadde, and A. Ortega, "Efficient sampling set selection for bandlimited graph signals using graph spectral proxies," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3775–3789, 2016.
- [12] H. Sevi, G. Rilling, and P. Borgnat, "Harmonic analysis on directed graphs and applications: from Fourier analysis to wavelets," *arXiv:1811.11636*, 2018.
- [13] —, "Modeling signals over directed graphs through filtering," in *Proc. IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2018, pp. 718–722.
- [14] S. Segarra, A. G. Marques, and A. Ribeiro, "Optimal graph-filter design and applications to distributed linear network operators," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 4117–4131, 2017.
- [15] R. Nassif, C. Richard, J. Chen, and A. H. Sayed, "Distributed diffusion adaptation over graph signals," in *Proc. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4129–4133.
- [16] F. Hua, R. Nassif, C. Richard, H. Wang, and A. H. Sayed, "A preconditioned graph diffusion lms for adaptive graph signal processing," in *Proc. 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 111–115.
- [17] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *Journal of Machine Learning Research*, vol. 9, pp. 2491–2521, 2008.
- [18] J. Chen, C. Richard, and P. Honeine, "Nonlinear unmixing of hyperspectral data based on a linear-mixture/nonlinear-fluctuation model," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 480–492, 2013.
- [19] J. Chen, C. Richard, and A. O. Hero, "Nonlinear unmixing of hyperspectral images using a semiparametric model and spatial regularization," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7954–7958.
- [20] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [21] B. Girault, "Stationary graph signals using an isometric graph translation," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2015, pp. 1516–1520.
- [22] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 US election: divided they blog," in *Proc. International workshop on Link discovery*, 2005, pp. 36–43.
- [23] N. Perraudin, J. Paratte, D. Shuman, L. Martin, V. Kalofolias, P. Vandergheynst, and D. K. Hammond, "GSPBOX: A toolbox for signal processing on graphs," *arXiv preprint arXiv:1408.5781*, 2014.