Transient Artifact Reduction Algorithm (TARA) Based on Sparse Optimization

Ivan W. Selesnick, *Senior Member, IEEE*, Harry L. Graber, *Member, IEEE*, Yin Ding, Tong Zhang, and Randall L. Barbour

***Abstract*—This paper addresses the suppression of transient artifacts in signals, e.g., biomedical time series. To that end, we distinguish two types of artifact signals. We define "Type 1" artifacts as spikes and sharp, brief waves that adhere to a baseline value of zero. We define "Type 2" artifacts as comprising approximate step discontinuities. We model a Type 1 artifact as being sparse and having a sparse time-derivative, and a Type 2 artifact as having a sparse time-derivative. We model the observed time series as the sum of a low-pass signal (e.g., a background trend), an artifact signal of each type, and a white Gaussian stochastic process. To jointly estimate the components of the signal model, we formulate a sparse optimization problem and develop a rapidly converging, computationally efficient iterative algorithm denoted TARA ("transient artifact reduction algorithm"). The effectiveness of the approach is illustrated using near infrared spectroscopic time-series data.**

***Index Terms*—Measurement artifact, artifact rejection, sparse optimization, wavelet, low-pass filter, total variation, lasso, fused lasso.**

I. Introduction

THIS paper addresses the suppression of artifacts in measured signals, where the artifacts are of unknown shape but are known to be transient in form. We are motivated in particular by the problem of attenuating artifacts arising in biomedical time series, such as those acquired using near infrared spectroscopic (NIRS) imaging devices [3]. Our approach is based on a signal model intended to capture the primary characteristics of the artifacts, and on the subsequent formulation of an optimization problem. We model the measured discrete-time series, $y(n)$, as

$$y(n) = f(n) + x_1(n) + x_2(n) + w(n) \quad n \in \mathbb{Z}, \qquad (1)$$

where f is a low-pass signal, x_i are two distinct types of artifact signals, and w is white Gaussian noise. Specifically, f is low-pass in the sense that when f is used as the input to an appropriately chosen high-pass filter, denoted by $\mathbf{H}$, the output is approximately the all-zero signal; i.e., $\mathbf{H}f \approx 0$.

The 'Type 1' artifact signal, x_1, is intended to model spikes and sharp, brief waves; while the 'Type 2' artifact signal, x_2, is intended to model additive step discontinuities. For the purpose of flexibility and generality, we avoid defining the artifact signals in terms of precise rules or templates. Instead, we use the notion of sparsity to define them in a less regimented way that facilitates the formulation of an optimization-based approach:

1) We define the 'Type 1' artifact signal, x_1, as being sparse and having a sparse derivative (actually, a discrete-time approximation of the derivative, here and subsequently).
2) We define the 'Type 2' artifact signal, x_2, as having a sparse derivative. This type of artifact signal is composed of step discontinuities (or approximate step discontinuities).

After the artifacts, x_i, are estimated, they are subtracted from the raw data to obtain a corrected time series.

To handle both types of artifacts simultaneously, in this work we develop an algorithm, denoted 'Transient Artifact Reduction Algorithm' (TARA). Complex artifacts often comprise both types; hence TARA performs joint optimization to maximize the effectiveness of the model to better reduce such artifacts. We devise TARA to have high computational efficiency and low memory requirements by constraining all matrices to be banded,[1] which allows us to leverage fast solvers for banded systems [29, Sect 2.4]. TARA does not require the user to specify auxiliary algorithmic parameters.

In addition to suppressing artifacts according to the model (1), we also consider the simplified model

$$y(n) = f(n) + x_1(n) + w(n), \quad n \in \mathbb{Z}, \qquad (2)$$

which contains artifacts of Type 1 only. We considered model (2) in previous work [34], where we used it to formulate what we called the 'LPF/CSD' problem. In this paper we present an improved algorithm for the LPF/CSD problem and provide an approach for setting the parameters. The new algorithm serves as the basis for the development of TARA, which suppresses artifacts according to model (1).

Although the suppression of Type 1 and Type 2 artifacts was addressed in our previous work [34], it was assumed that the measured time series is affected by the presence of either Type 1 or Type 2 artifacts, but not both. Two algorithms, one for each

Manuscript received April 22, 2014; revised August 04, 2014 and September 18, 2014; accepted September 18, 2014. Date of publication October 31, 2014; date of current version November 17, 2014. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hing Cheung So. This research was supported by the NSF under Grant No. CCF-1018020, the NIH under Grant Nos. R42NS050007, R44NS049734, and R21NS067278, and by DARPA project N66001-10-C-2008.

I. W. Selesnick, Y. Ding, and T. Zhang are with the Department of Electrical and Computer Engineering, Polytechnic School of Engineering, New York University Brooklyn, NY 11201 USA (e-mail: selesi@nyu.edu; tz535@nyu.edu; adamding0215@me.com).

H. L. Graber and R. L. Barbour are with the Department of Pathology, SUNY Downstate Medical Center, Brooklyn, NY 11203 USA (e-mail: Harry.Graber@downstate.edu; Randall.Barbour@downstate.edu).

This paper has supplementary downloadable material, available at http://ieeexplore.ieee.org, provided by the authors. The material includes software (MATLAB) implementing the algorithm and examples. This material is 372 KB in size.

Digital Object Identifier 10.1109/TSP.2014.2366716

[1] A matrix is banded if its non-zero elements lie only on its main diagonal and a few adjacent upper and lower diagonals.

artifact type, are described in [34]. The respective algorithms were illustrated on time series acquired using a NIRS system [23], which frequently have artifacts of both types [15]. The algorithms were compared with the algorithm used by the NAP software application for NIRS artifact suppression [15].

A. Related Work

Several approaches have been developed for the suppression of artifacts in biomedical time series [2], [9], [15], [20], [21], [25], [31], [37], [38]. Some methods, such as those based on independent component analysis (ICA) or adaptive filtering, require the availability of multiple channels or reference signals. However, if artifacts differ substantially among channels or if multiple channels are unavailable, then single-channel methods are needed [9], [21], [24]. Several methods for detecting and/or correcting motion artifacts in NIRS time series have been compared [9], [21], [24], [31], [32], leading to the conclusion that wavelet-based methods are more effective than other methods, especially for single-channel processing. Wavelets have also been shown effective for reducing ocular artifacts in EEG [2], [22]. Hence, we compare the sparse optimization and wavelet approaches below.

We remark that models (1) and (2) are prompted by the approach of morphological component analysis (MCA), in which all signal components are modeled as sparse with respect to distinct transforms [35]. However, the presence of the low-pass non-sparse component, f, which differs from MCA, makes models (1) and (2) more realistic for biomedical time-series analysis. Moreover, modeling the low-pass component enhances the prospective sparsity of the remaining components, on which sparse optimization and MCA rely.

Rather than developing optimization schemes for general linear inverse problems, this work emphasizes algorithms that exploit the banded structure of one-dimensional LTI operators to achieve high computational efficiency and fast convergence while avoiding additional algorithm parameters (e.g., step-size parameters), for the specific problems considered here. Yet, we note that general optimization algorithms, such as those based on proximity operators [8], [10], [12], [13], [28], [30], allow for consideration of non-smooth compound regularization problems more general than those considered here. Relevant surveys are also given in [4], [36].

B. Notation

Vectors and matrices are represented by lower- and upper-case bold (e.g., $\mathbf{x}$ and $\mathbf{H}$), respectively. The n-th component of a vector $\mathbf{x}$ is denoted $[\mathbf{x}]_n$. Finite-length discrete-time signals are represented as vectors in $\mathbb{R}^N$. The notation $x \in [a, b]$ means $a \leqslant x \leqslant b$.

II. Type 1 Artifacts (The LPF/CSD Problem)

In this section, we consider model (2), which contains artifacts of Type 1 only. The derivation of TARA in Section III for model (1) builds on the algorithm developed here. In previous work [34], an iterative algorithm based on the 'alternating direction method of multipliers' (ADMM) was derived for the 'LPF/CSD' problem. Here, we revisit the problem and present several improvements in comparison with [34].

1) A faster algorithm. In this work, we derive an algorithm based on majorization-minimization (MM). The new algorithm converges in fewer iterations in practice than the previous algorithm based on ADMM.
2) Fewer algorithm parameters. The algorithm of [34] required the user to specify a positive scalar, μ, analogous to a step-size parameter. A poor choice of μ leads to slow convergence. The new MM algorithm does not require any parameters beyond those in the objective function.
3) A method to set regularization parameters, λ_0 and λ_1, based on the noise variance (which we assume is known).
4) A more general problem formulation. In this work, we allow the penalty functions to be non-convex, whereas [34] considered convex penalty functions only. With convex penalty functions, the amplitudes of artifacts tend to be underestimated (the estimates are biased toward zero).
5) A method to set non-convexity parameters. In the case where non-convex penalty functions are utilized, we present a method to set the non-convexity parameters. The previous work considered only convex penalty functions.

A. Problem Formulation

We address the problem in the discrete-time setting. Signals are represented as vectors in $\mathbb{R}^N$. We write model (2) as

$$\mathbf{y} = \mathbf{f} + \mathbf{x} + \mathbf{w}, \qquad \mathbf{y},\ \mathbf{x},\ \mathbf{w} \in \mathbb{R}^N \tag{3}$$

where the signal $\mathbf{x}$ is modeled as sparse and having a sparse derivative. The derivative is approximated using the discrete-time first-order difference operator, $\mathbf{D}$, defined by $[\mathbf{D}\mathbf{x}]_n = [\mathbf{x}]_{n+1} - [\mathbf{x}]_n$. The matrix $\mathbf{D}$ has the form

$$\mathbf{D} = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{bmatrix}. \tag{4}$$

In order to estimate $\mathbf{x}$ from $\mathbf{y}$, we propose to solve the optimization problem

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \Big\{ F(\mathbf{x}) = \frac{1}{2}\|\mathbf{H}(\mathbf{y} - \mathbf{x})\|_2^2 + \lambda_0 \sum_n \phi_0([\mathbf{x}]_n) + \lambda_1 \sum_n \phi_1([\mathbf{D}\mathbf{x}]_n) \Big\}, \tag{5}$$

where $\lambda_i > 0$ are the regularization parameters. The low-pass signal is then estimated as

$$\hat{\mathbf{f}} = \mathbf{L}(\mathbf{y} - \hat{\mathbf{x}}) = \mathbf{y} - \hat{\mathbf{x}} - \mathbf{H}(\mathbf{y} - \hat{\mathbf{x}}), \tag{6}$$

where $\mathbf{L}$ denotes the low-pass filter defined as $\mathbf{L} = \mathbf{I} - \mathbf{H}$. The penalty functions, $\phi_i : \mathbb{R} \to \mathbb{R}$, are chosen to promote sparsity. We refer to (5) as the LPF/CSD (low-pass-filtering/compound-sparse-denoising) problem [34].

When ϕ_i is the absolute-value function, $\phi_i(x) = |x|$, for $i = 1, 2$, then problem (5) is the same as that considered in [34]. When, in addition, $\mathbf{H}$ is the identity operator, (5) is the same as the 'fused lasso signal approximator' in [17].

The high-pass filter, $\mathbf{H}$, is taken to be a zero-phase recursive discrete-time filter that we write as

$$\mathbf{H} = \mathbf{B}\mathbf{A}^{-1}, \tag{7}$$

where $\mathbf{A}$ and $\mathbf{B}$ are banded Toeplitz matrices, as described in Sec. VI of [34]. We further suppose that $\mathbf{B}$ admits the factorization

$$\mathbf{B} = \mathbf{B}_1\mathbf{D}, \tag{8}$$

where $\mathbf{B}_1$ is banded and $\mathbf{D}$ is the above-noted first-order difference matrix. (See [34] for derivations of the factorizations in (7) and (8).) The fact that $\mathbf{A}$ and $\mathbf{B}$ are banded is important for the computational efficiency of the algorithm to be developed. We also assume that $\mathbf{A}$, $\mathbf{B}_1$, and $\mathbf{D}$ commute. As linear time-invariant (LTI) filters, the commutativity of these operators is exactly true for infinite-length discrete-time signals defined on $\mathbb{Z}$. For finite-length signals, with which we deal in practice, the commutativity is approximately true, with the error being confined to the start and end of the signal, with a temporal extent that depends on the time-constant of the filter. We note that $\mathbf{H}$ was expressed as $\mathbf{A}^{-1}\mathbf{B}$ in our earlier work [34]. Here we express $\mathbf{H}$ as $\mathbf{B}\mathbf{A}^{-1}$, which the commutativity property permits, because the derivation of the computationally efficient MM algorithm in Section II-B relies on this ordering. We assume in this work that the signals of interest are sufficiently long that start and end transients are not problematic, in which case the commutativity assumption is justified.

The regularization parameters, λ_0 and λ_1, control the relative weight between the penalty terms. Their values should also be set according to the noise variance: higher noise calls for higher λ_i. As noted in [34], the regularization in problem (5) is an example of compound regularization [1], [6], wherein two or more regularizers are used to promote distinct properties of the signal to be recovered.

In (5), the functions $\phi_i : \mathbb{R} \to \mathbb{R}$ are chosen to be sparsity-promoting penalty functions more general than the ℓ_1 norm. Non-smooth penalties are given in lines 1a-3a of Table I; these are non-differentiable at zero. For the logarithmic (log) and arctangent (atan) penalties, the parameter $a > 0$ controls the extent to which the functions are non-convex. The log and atan penalties are strictly concave on $\mathbb{R}_+$ for $a > 0$. As $a \to 0$, the log and atan penalties approach the absolute value function. The atan penalty was derived to promote sparsity more strongly than the log penalty [33]. It will be useful below to define $\psi : \mathbb{R} \to \mathbb{R}$ as $\psi(u) := u/\phi'(u)$.

The algorithms derived in Sections II-B and III-A inevitably encounter 'divide-by-zero' errors when any of the penalties are non-differentiable at zero. To prevent this error condition, we introduce a small degree of smoothing so that the penalty functions become differentiable at $u = 0$; see lines 1b-3b of Table I. When the constant ϵ is sufficiently small, the minimizer of the smoothed objective function is negligibly different from the minimizer of the non-smoothed one. In practice, we use $\epsilon = 10^{-8}$.

TABLE I
SPARSITY-PROMOTING PENALTY FUNCTIONS

	Penalty, $\phi(u)$	$\psi(u) = u/\phi'(u)$
1a.	$\lvert u\rvert$ (i.e., ℓ_1 norm)	$\lvert u\rvert$
2a.	$\frac{1}{a}\log(1+a\lvert u\rvert)$	$\lvert u\rvert(1+a\lvert u\rvert)$
3a.	$\frac{2}{a\sqrt{3}}\left(\tan^{-1}\left(\frac{1+2a\lvert u\rvert}{\sqrt{3}}\right)-\frac{\pi}{6}\right)$	$\lvert u\rvert(1+a\lvert u\rvert+a^2\lvert u\rvert^2)$
1b.	$\sqrt{u^2+\epsilon}$	$\sqrt{u^2+\epsilon}$
2b.	$\frac{1}{a}\log\left(1+a\sqrt{u^2+\epsilon}\right)$	$\sqrt{u^2+\epsilon}\left(1+a\sqrt{u^2+\epsilon}\right)$
3b.	$\frac{2}{a\sqrt{3}}\left(\tan^{-1}\left(\frac{1+2a\sqrt{u^2+\epsilon}}{\sqrt{3}}\right)-\frac{\pi}{6}\right)$	$\sqrt{u^2+\epsilon}\left(1+a\sqrt{u^2+\epsilon}+a^2(u^2+\epsilon)\right)$

Notes: $u \in \mathbb{R}$, $a > 0$, $\epsilon > 0$. 1b-3b are smoothed penalties.

B. Algorithm

We use the majorization-minimization (MM) procedure [16] to minimize the objective (cost) function, $F : \mathbb{R}^N \to \mathbb{R}$, defined in (5). We use the MM procedure with a quadratic majorizer of the penalty function; each iteration of the MM procedure then requires the minimization of a quadratic function, which is performed by solving a system of linear equations. In this work, with suitable manipulations, the system matrix of the linear equations is banded. Hence, fast solvers for banded systems can be used to implement the algorithm with very high computational efficiency.

Under suitable restrictions on ϕ (symmetric, continuously differentiable, etc.), which are satisfied by the penalty functions in lines 1b-3b of Table I, a majorizer of ϕ is given by

$$g(u, v; \phi) := \frac{\phi'(v)}{2v}u^2 + \phi(v) - \frac{v}{2}\phi'(v). \tag{9}$$

That is,

$$g(u, v; \phi) \geqslant \phi(u), \text{ for all } u, v \in \mathbb{R} \tag{10}$$

$$g(v, v; \phi) = \phi(v), \text{ for all } v \in \mathbb{R}. \tag{11}$$

Note that $g(u, v; \phi)$ is quadratic in u. The majorizer g can be used, in turn, to obtain a majorizer of F in (5).

If $\mathbf{u}$ and $\mathbf{v}$ are vectors, then

$$\sum_n g([\mathbf{u}]_n, [\mathbf{v}]_n; \phi_0) \geqslant \sum_n \phi_0([\mathbf{u}]_n), \tag{12}$$

with equality if $\mathbf{u} = \mathbf{v}$. Using (9), we write the left-hand-side of (12) compactly as

$$\sum_n g([\mathbf{u}]_n, [\mathbf{v}]_n; \phi_0) = \frac{1}{2}\mathbf{u}^{\mathsf{T}}[\mathbf{\Lambda}(\mathbf{v}; \phi_0)]\mathbf{u} + c_0, \tag{13}$$

where $[\mathbf{\Lambda}(\mathbf{v}; \phi)]$ is defined as the diagonal matrix

$$[\mathbf{\Lambda}(\mathbf{v}; \phi)]_{n,n} := \frac{\phi'([\mathbf{v}]_n)}{[\mathbf{v}]_n} = 1/\psi([\mathbf{v}]_n) \tag{14}$$

and c_0 does not depend on $\mathbf{u}$. Similarly,

$$\sum_n g([\mathbf{D}\mathbf{u}]_n, [\mathbf{D}\mathbf{v}]_n; \phi_1) \geqslant \sum_n \phi_1([\mathbf{D}\mathbf{u}]_n), \tag{15}$$

with equality if $\mathbf{u} = \mathbf{v}$. The left-hand-side can be written as

$$\sum_n g([\mathbf{D}\mathbf{u}]_n, [\mathbf{D}\mathbf{v}]_n; \phi_1) = \frac{1}{2}\mathbf{u}^{\mathsf{T}}\mathbf{D}^{\mathsf{T}}[\mathbf{\Lambda}(\mathbf{D}\mathbf{v}; \phi_1)]\mathbf{D}\mathbf{u} + c_1,$$

where $\mathbf{\Lambda}$ is defined as in (14) and c_1 does not depend on $\mathbf{u}$.

Therefore, using (12) and (15), a majorizer of F in (5) is given by $G : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$, defined by

$$G(\mathbf{x},\mathbf{v}) = \frac{1}{2}\|\mathbf{B}\mathbf{A}^{-1}(\mathbf{y}-\mathbf{x})\|_2^2 + \frac{\lambda_0}{2}\mathbf{x}^{\mathsf{T}}[\mathbf{\Lambda}(\mathbf{v};\phi_0)]\mathbf{x} + \frac{\lambda_1}{2}\mathbf{x}^{\mathsf{T}}\mathbf{D}^{\mathsf{T}}[\mathbf{\Lambda}(\mathbf{D}\mathbf{v};\phi_1)]\mathbf{D}\mathbf{x} + c, \quad (16)$$

where c does not depend on $\mathbf{x}$. The MM update equation, wherein the majorizer is minimized, is

$$\mathbf{x}^{(i+1)} = \arg\min_{\mathbf{x}} \; G(\mathbf{x},\mathbf{x}^{(i)}), \quad (17)$$

where i is the iteration index. The update (17) leads to

$$\mathbf{x}^{(i+1)} = \left(\mathbf{A}^{-\mathsf{T}}\mathbf{B}^{\mathsf{T}}\mathbf{B}\mathbf{A}^{-1} + \mathbf{\Lambda}_0^{(i)} + \mathbf{D}^{\mathsf{T}}\mathbf{\Lambda}_1^{(i)}\mathbf{D}\right)^{-1} \times \mathbf{A}^{-\mathsf{T}}\mathbf{B}^{\mathsf{T}}\mathbf{B}\mathbf{A}^{-1}\mathbf{y}, \quad (18)$$

where $\mathbf{\Lambda}_0^{(i)}$ and $\mathbf{\Lambda}_1^{(i)}$ are the diagonal matrices

$$\mathbf{\Lambda}_0^{(i)} := \lambda_0\mathbf{\Lambda}(\mathbf{x}^{(i)};\phi_0), \qquad \mathbf{\Lambda}_1^{(i)} := \lambda_1\mathbf{\Lambda}(\mathbf{D}\mathbf{x}^{(i)};\phi_1). \quad (19)$$

Specifically,

$$[\mathbf{\Lambda}_0^{(i)}]_{n,n} := \lambda_0 \frac{\phi_0'([\mathbf{x}^{(i)}]_n)}{[\mathbf{x}^{(i)}]_n} = \lambda_0/\psi_0([\mathbf{x}^{(i)}]_n) \quad (20)$$

$$[\mathbf{\Lambda}_1^{(i)}]_{n,n} := \lambda_1 \frac{\phi_1'([\mathbf{D}\mathbf{x}^{(i)}]_n)}{[\mathbf{D}\mathbf{x}^{(i)}]_n} = \lambda_1/\psi_1([\mathbf{D}\mathbf{x}^{(i)}]_n). \quad (21)$$

Note that the system matrix in (18) (i.e., inside the parentheses), is not banded, because $\mathbf{A}^{-1}$ is not banded. Hence, fast solvers for banded systems cannot be used directly with (18). To utilize fast solvers, we write

$$\left(\mathbf{A}^{-\mathsf{T}}\mathbf{B}^{\mathsf{T}}\mathbf{B}\mathbf{A}^{-1} + \mathbf{\Lambda}_0^{(i)} + \mathbf{D}^{\mathsf{T}}\mathbf{\Lambda}_1^{(i)}\mathbf{D}\right)^{-1} = \mathbf{A}\left(\mathbf{B}^{\mathsf{T}}\mathbf{B} + \mathbf{A}^{\mathsf{T}}\left(\mathbf{\Lambda}_0^{(i)} + \mathbf{D}^{\mathsf{T}}\mathbf{\Lambda}_1^{(i)}\mathbf{D}\right)\mathbf{A}\right)^{-1}\mathbf{A}^{\mathsf{T}} \quad (22)$$

where the matrix within the inverse is banded. Then, using (22) in (18), we obtain the update equation

$$\mathbf{x}^{(i+1)} = \mathbf{A}\left(\mathbf{B}^{\mathsf{T}}\mathbf{B} + \mathbf{A}^{\mathsf{T}}\left(\mathbf{\Lambda}_0^{(i)} + \mathbf{D}^{\mathsf{T}}\mathbf{\Lambda}_1^{(i)}\mathbf{D}\right)\mathbf{A}\right)^{-1}\mathbf{B}^{\mathsf{T}}\mathbf{B}\mathbf{A}^{-1}\mathbf{y}.$$

The algorithm, summarized in Table II, can be implemented efficiently using fast solvers for banded systems because the matrix $\mathbf{Q}$ in line 5 is banded. The penalty functions ϕ_i arise only in lines 3 and 4 of the algorithm (Table II) and their role is encapsulated by ψ_i.

We note that, as the algorithm progresses, many elements of $[\mathbf{x}]_n$ and $[\mathbf{D}\mathbf{x}]_n$ generally go to zero when ϕ_0 and ϕ_1 are chosen to be any of the non-smooth penalties in lines 1a-3a of Table I, or other non-smooth sparsity-promoting penalties. Note also that, for such penalties, $\psi(u) \to 0$ as $u \to 0$. Therefore, as the algorithm converges to a sparse vector $\mathbf{x}$, elements of the $\mathbf{\Lambda}$ matrices go to infinity, and consequently the system of linear equations (line 6 in Table II) becomes ill-conditioned. For this reason, we use the smoothed versions of the penalty functions with a small value of ϵ to avoid 'divide-by-zero' errors. In practice, we set $\epsilon = 10^{-8}$. We have found that such slight smoothing of the penalty term has a negligible effect on the minimizer $\hat{\mathbf{x}}$, as validated by the comparison of ADMM and MM algorithms in Fig. 4 below.

TABLE II
MM ALGORITHM SOLVING THE LPF/CSD PROBLEM (5)

	Input: $\mathbf{y} \in \mathbb{R}^N,\ \lambda_i > 0,\ \phi_i, \mathbf{A}, \mathbf{B}$	
1.	$\bar{\mathbf{y}} = \mathbf{B}^{\mathsf{T}}\mathbf{B}\mathbf{A}^{-1}\mathbf{y}$	
2.	$\mathbf{x} = \mathbf{y}$	(initialization)
	Repeat	
3.	$[\mathbf{\Lambda}_0]_{n,n} = \lambda_0/\psi_0([\mathbf{x}]_n)$	($\mathbf{\Lambda}_0$ is diagonal)
4.	$[\mathbf{\Lambda}_1]_{n,n} = \lambda_1/\psi_1([\mathbf{D}\mathbf{x}]_n)$	($\mathbf{\Lambda}_1$ is diagonal)
5.	$\mathbf{Q} = \mathbf{B}^{\mathsf{T}}\mathbf{B} + \mathbf{A}^{\mathsf{T}}(\mathbf{\Lambda}_0 + \mathbf{D}^{\mathsf{T}}\mathbf{\Lambda}_1\mathbf{D})\mathbf{A}$	($\mathbf{Q}$ is banded)
6.	$\mathbf{x} = \mathbf{A}\mathbf{Q}^{-1}\bar{\mathbf{y}}$	(MM update)
	Until convergence	
	Output: $\mathbf{x}$	

C. Regularization Parameters

To use the LPF/CSD formulation to estimate transient artifacts, the λ_0 and λ_1 regularization parameters must be specified. The process of specifying appropriate values is governed by two considerations that have a substantial effect on the estimated artifact signal $\hat{\mathbf{x}}$. First, the values of both parameters nominally should be proportional to σ, where σ^2 is the noise variance, which we assume is known. In addition, we aim to set λ_i so that $\hat{\mathbf{x}}$ contains, with high probability, signal behavior that is due solely to the artifacts present in the data. This is because we seek to correct the observed data $\mathbf{y}$ by subtracting $\hat{\mathbf{x}}$ from it. If $\hat{\mathbf{x}}$ exhibits behavior that is not due to artifacts, then subtracting it from $\mathbf{y}$ generally leads to distortion of the signal of interest (e.g., oscillations of biomedical origin). Accordingly, the artifact characteristics of the data is the second factor that governs the selection of parameter values.

We begin by considering how to set λ_i according to the noise variance. First, note that if the λ_i are sufficiently large, then $\hat{\mathbf{x}}$, the minimizer of F, will be the all-zero vector, $\hat{\mathbf{x}} = \mathbf{0}$. Second, note that if the λ_i are near zero, then $\hat{\mathbf{x}}$ will approximate the noisy data, $\hat{\mathbf{x}} \approx \mathbf{y}$. We seek to set the λ_i large enough so that $\hat{\mathbf{x}}$ comprises solely artifact-related signal behavior, but not so large that $\hat{\mathbf{x}}$ is overly attenuated toward zero. Suppose that some realization of the data consists purely of noise, i.e., $\mathbf{y} = \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$. In this case, $\hat{\mathbf{x}}$ should be the all-zero vector, $\hat{\mathbf{x}} = \mathbf{0}$, at least with high probability. Hence λ_i should be chosen sufficiently large so that when $\mathbf{y}$ consists purely of noise, we obtain $\hat{\mathbf{x}} \approx \mathbf{0}$ with high probability. That is, a suitable pair, (λ_0, λ_1), is one for which we can write

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}) \Longrightarrow \hat{\mathbf{x}} \approx \mathbf{0} \quad \text{with high probability}, \quad (23)$$

where $\hat{\mathbf{x}}$ is obtained by solving (5). The suitable (λ_0, λ_1) thus depends on the noise standard deviation, σ. To find such (λ_0, λ_1) analytically is difficult due to the compound regularization. To tackle this problem, we consider two special cases where selection of λ_i that satisfy (23) is relatively straightforward.

In the first special case, we set λ_1 to zero and seek a value for λ_0 that satisfies (23). We denote this value λ_0^*; i.e., $(\lambda_0^*, 0)$ is a pair satisfying (23). Likewise, in the second special case, we set λ_0 to zero and seek a value for λ_1 that satisfies (23). We denote this value λ_1^*; i.e, $(0, \lambda_1^*)$ is a pair satisfying (23). Then for the general case we interpolate between these two pairs to obtain (λ_0, λ_1) approximately satisfying (23).

In order to find λ_0^* and λ_1^*, we work with two special cases of (5). We define the objective function, $F_0 : \mathbb{R}^N \to \mathbb{R}$, as

$$F_0(\mathbf{x}) = \frac{1}{2}\|\mathbf{H}(\mathbf{y} - \mathbf{x})\|_2^2 + \lambda_0 \sum_n \phi_0([\mathbf{x}]_n) \tag{24}$$

and the objective function, $F_1 : \mathbb{R}^N \to \mathbb{R}$, as

$$F_1(\mathbf{x}) = \frac{1}{2}\|\mathbf{H}(\mathbf{y} - \mathbf{x})\|_2^2 + \lambda_1 \sum_n \phi_1([\mathbf{D}\mathbf{x}]_n). \tag{25}$$

The functions F_0 and F_1 correspond to $\lambda_1 = 0$ and $\lambda_0 = 0$ in (5), respectively. Unlike F, the F_i do not involve compound regularization, which simplifies the analysis necessary to set λ_i. We denote the minimizers of F_0 and F_1 as:

$$\mathbf{x}_{F_0}^{\mathsf{opt}} = \arg\min_{\mathbf{x}} F_0(\mathbf{x}), \qquad \mathbf{x}_{F_1}^{\mathsf{opt}} = \arg\min_{\mathbf{x}} F_1(\mathbf{x}). \tag{26}$$

To find λ_0^* such that $(\lambda_0^*, 0)$ satisfies (23), we equivalently find λ_0^* such that

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}) \Longrightarrow \mathbf{x}_{F_0}^{\mathsf{opt}} \approx \mathbf{0} \text{ with high probability.} \tag{27}$$

That is, we seek to set λ_0 in (24) so that $\mathbf{x}_{F_0}^{\mathsf{opt}}$ is relatively noise-free with high probability. The value for λ_0^* accomplishing this is derived using optimality conditions from convex analysis. This general approach has been described by Fuchs [18] for the purpose of setting the false alarm rate in a target detection application. To find λ_1^* such that $(0, \lambda_1^*)$ satisfies (23), we will proceed in a similar manner; however, the presence of $\mathbf{D}$ in the penalty of F needs to be taken into account.

1) Obtaining λ_0^*: We assume here that ϕ_0 is one of the non-smooth penalty functions in Table I (lines 1a–3a). We will use a result from the theory of convex functions [5]: if a function $f : \mathbb{R}^N \to \mathbb{R}$ is convex, then $\mathbf{x}$ is a minimizer of f if and only if $\mathbf{0} \in \partial f(\mathbf{x})$, where ∂f is the subgradient of f.

As shown in [33], if ϕ_0 is chosen such that F_0 in (24) is convex, then $\mathbf{x}$ minimizes F_0 if and only if, for all n,

$$[\mathbf{H}^\mathsf{T}\mathbf{H}(\mathbf{y} - \mathbf{x})]_n \begin{cases} = \lambda_0 \phi_0'([\mathbf{x}]_n), & [\mathbf{x}]_n \neq 0 \\ \in [-\lambda_0,\ \lambda_0], & [\mathbf{x}]_n = 0. \end{cases} \tag{28}$$

When $\mathbf{x} = \mathbf{x}_{F_0}^{\mathsf{opt}}$ is the all-zero vector, we have

$$\mathbf{x}_{F_0}^{\mathsf{opt}} = \mathbf{0} \Longrightarrow [\mathbf{H}^\mathsf{T}\mathbf{H}\mathbf{y}]_n \in [-\lambda_0,\ \lambda_0],\ \forall n, \tag{29}$$

and the right-hand-side of (27) can be written as $[\mathbf{H}^\mathsf{T}\mathbf{H}\mathbf{y}]_n \in [-\lambda_0^*,\ \lambda_0^*],\ \forall n$ with high probability. Hence, λ_0^* should be chosen so that

$$\lambda_0^* \geqslant |[\mathbf{H}^\mathsf{T}\mathbf{H}\mathbf{y}]_n|,\ \forall n \text{ with high probability,} \tag{30}$$

where $\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$. Note that $\mathbf{H}^\mathsf{T}\mathbf{H}$ represents an LTI system. Then $\mathbf{H}^\mathsf{T}\mathbf{H}\mathbf{y}$ is a stationary stochastic process and $[\mathbf{H}^\mathsf{T}\mathbf{H}\mathbf{y}]_n \sim \mathcal{N}(0, \sigma_0^2)$, where σ_0 is given by

$$\sigma_0 := \mathrm{std}([\mathbf{H}^\mathsf{T}\mathbf{H}\mathbf{y}]_n) = \|\mathbf{p}_0\|_2\, \sigma$$

and $\mathbf{p}_0$ is the impulse response of the LTI filter $\mathbf{H}^\mathsf{T}\mathbf{H}$. That is, $\mathbf{p}_0 = \mathbf{h} * \mathbf{h}^r$, where $\mathbf{h}$ represents the impulse response of the LTI filter $\mathbf{H} := \mathbf{B}\mathbf{A}^{-1}$, and $\mathbf{h}^r$ is the time-reversed version of $\mathbf{h}$ (i.e., $h^r(-n) = h(n)$).

So, (30) can be expressed as

$$\lambda_0^* \geqslant |v|,\ \forall n \text{ with high probability, where } v \sim \mathcal{N}(0, \sigma_0^2).$$

A nominal value of λ_0^* is given by the 'three-sigma' rule,

$$\lambda_0^* = 3\,\sigma_0 = 3\,\|\mathbf{p}_0\|_2 \sigma. \tag{31}$$

Using the λ_0^* value given by (31), (30) is satisfied with a probability above 99%.

2) Obtaining λ_1^*: From (7) and (8), note that $\mathbf{H} = \mathbf{B}_1\mathbf{D}\mathbf{A}^{-1}$. Using commutativity, we have $\mathbf{H} = \mathbf{B}_1\mathbf{A}^{-1}\mathbf{D}$. Hence constant-valued signals are in the null space of both $\mathbf{H}$ and $\mathbf{D}$; i.e., the signal $[\mathbf{x}]_n = c$ is annihilated by both operators. Therefore, if $\mathbf{x}_2$ is defined as $[\mathbf{x}_2]_n = [\mathbf{x}]_n + c$, then $F_1(\mathbf{x}_2) = F_1(\mathbf{x})$; i.e., the value of the objective function F_1 is unaffected by a shift in the baseline of $\mathbf{x}$. Then the signal minimizing F_1 in (25) is unique only up to an additive constant. This issue is addressed by defining a change of variables, namely $\mathbf{u} = \mathbf{D}\mathbf{x}$, which facilitates the derivation of λ_1^*.

We define $\mathbf{H}_1 = \mathbf{B}_1\mathbf{A}^{-1}$. Then, $\mathbf{H}_1\mathbf{D} = \mathbf{H}$, and we can write

$$\mathbf{H}\mathbf{x} = \mathbf{H}_1\mathbf{u}, \quad \mathbf{u} = \mathbf{D}\mathbf{x}. \tag{32}$$

Hence, minimizing (25) is equivalent to the problem

$$\mathbf{u}_{\tilde{F}_1}^{\mathsf{opt}} = \arg\min_{\mathbf{u}} \left\{ \tilde{F}_1(\mathbf{u}) = \frac{1}{2}\|\mathbf{H}\mathbf{y} - \mathbf{H}_1\mathbf{u}\|_2^2 + \lambda_1 \sum_n \phi_1([\mathbf{u}]_n) \right\}.$$

Accordingly, to find λ_1^* such that $(\lambda_1^*, 0)$ satisfies (23), we equivalently find λ_1^* such that

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}) \Longrightarrow \mathbf{u}_{\tilde{F}_1}^{\mathsf{opt}} \approx \mathbf{0} \text{ with high probability.} \tag{33}$$

As above, if ϕ_1 is chosen such that $\tilde{F}_1$ is convex, then $\mathbf{u}$ minimizes $\tilde{F}_1$ if and only if, for all n,

$$[\mathbf{H}_1^\mathsf{T}(\mathbf{H}\mathbf{y} - \mathbf{H}_1\mathbf{u})]_n \begin{cases} = \lambda_1\, \phi_1'([\mathbf{u}]_n), & [\mathbf{u}]_n \neq 0 \\ \in [-\lambda_1,\ \lambda_1], & [\mathbf{u}]_n = 0. \end{cases} \tag{34}$$

Proceeding as above, we obtain a nominal value for λ_1^* of

$$\lambda_1^* = 3\,\sigma_1 = 3\,\|\mathbf{p}_1\|_2\,\sigma, \tag{35}$$

where

$$\sigma_1 := \mathrm{std}([\mathbf{H}_1^\mathsf{T}\mathbf{H}\mathbf{y}]_n) = \|\mathbf{p}_1\|_2\,\sigma$$

and $\mathbf{p}_1$ is the impulse response of the LTI filter $\mathbf{H}_1^\mathsf{T}\mathbf{H}$. That is, $\mathbf{p}_1 = \mathbf{h} * \mathbf{h}_1^r$, where $\mathbf{h}_1$ represents the impulse response of the

LTI filter $\mathbf{H}_1 := \mathbf{B}_1\mathbf{A}^{-1}$, and $\mathbf{h}_1^r$ is the time-reversed version of $\mathbf{h}_1$.

3) Setting (λ_0, λ_1): The parameter pair $(\lambda_0^*, 0)$ is appropriate for an artifact signal that is known to be sparse, i.e., departing only briefly from a baseline value of zero. In this case, the artifact signal can be modeled as consisting of pure impulses, i.e., isolated spikes of large deviation from baseline, such as 'salt and pepper' noise. This is because, when $\lambda_1 = 0$, the objective function imposes no continuity among the non-zero values of the artifact signal. On the other hand, the parameter pair $(0, \lambda_1^*)$ is suitable when it is known that the derivative of the artifact signal is sparse, i.e., the artifact signal consists of step discontinuities.

Real artifacts are generally not so easily classified as spikes or as additive step discontinuities. Therefore, λ_0 and λ_1 should be tuned according to the behavior of the artifacts in the data. Although the values λ_0^* and λ_1^* in (31) and (35) are ideally suited for two special cases only, they provide anchors for the selection of (λ_0, λ_1). We set

$$(\lambda_0, \lambda_1) = (\theta\lambda_0^*, (1-\theta)\lambda_1^*), \quad 0 \leqslant \theta \leqslant 1, \tag{36}$$

which restricts (λ_0, λ_1) to a line segment in the plane, reducing the two degrees of freedom to one. As one of $\{\lambda_0, \lambda_1\}$ is reduced, the other increases. Reducing one parameter without increasing the other would lead to a total reduction in the regularization, leading to potential noise contamination of $\hat{\mathbf{x}}$. Thereby, the interpolation (36) approximately satisfies (23); i.e., it takes into account the noise variance so that the estimated artifact signal $\hat{\mathbf{x}}$ is largely noise-free with high probability. In this approach, one of the two degrees of freedom in the LPF/CSD problem (5) is set according to the noise variance, and the other is used to tune the algorithm to the data.

D. Noise Model Deviation

Real time series are likely to deviate from the idealized model (2) on which LPF/CSD is based. The mid- and high-frequency spectral content of the data may comprise a mixture of biologically relevant signals, rather than white noise. In such cases, there is no well-defined noise standard deviation to use in formulas (31) and (35). However, the approach can still be utilized by using a 'pseudo-noise sigma' that serves as a substitute. The pseudo-sigma parameter then leads to values for λ_0^* and λ_1^*. The pseudo-σ parameter can be tuned using a representative data set. Then $\theta \in [0, 1]$ should be tuned such that $\hat{\mathbf{x}}$ captures the transient artifacts most effectively. In this way, the problem formulation (2) is parameterized in terms of (σ, θ) instead of (λ_0, λ_1). We have found this a more convenient parameterization for setting parameter values.

E. Setting the Non-Convexity Parameters

The use of non-convex penalties in (5) can be advantageous because, in comparison with convex penalties, they generally produce estimates that are less biased toward zero; i.e., the amplitudes of the estimated transients are less attenuated than those produced by convex penalties [14]. However, when using non-convex penalties, optimization algorithms may get trapped in sub-optimal local minima. Hence, non-convex penalties should be specified with care. One approach to avoid the issue of entrapment in local minima is to specify non-convex penalties such that the total objective function, F, is convex [7], [26], [27], [33]. Then the total objective function, owing to its convexity, does not posses sub-optimal local minima and a global optimal solution can be reliably found. The design of non-convex penalties according to this principle is formulated as a semidefinite program (SDP) in [33]. Here, we make simplifying assumptions to avoid the high computational cost of SDP.

When we use the logarithmic or arctangent penalty functions, which are non-convex, we need to set the non-convexity parameter, a, for each of ϕ_0 and ϕ_1. We denote the respective values by a_0 and a_1, and write the penalties as $\phi_0(u, a_0)$ and $\phi_1(u, a_1)$ to emphasize the dependence of the penalties on a_i.

To derive a heuristic for setting the non-convexity parameters, we assume that the sparse vectors $\mathbf{x}_{F_0}^{\text{opt}}$ and $\mathbf{u}_{F_1}^{\text{opt}}$ contain only a single non-zero entry. While this assumption is not satisfied in practice, with it we obtain values of a_i for which F is definitely non-convex. Using corollary 1 of [33], this assumption leads to upper bounds on a_0 and a_1 of $\|\mathbf{h}\|_2^2/\lambda_0$ and $\|\mathbf{h}_1\|_2^2/\lambda_1$, respectively, where $\mathbf{h}$ and $\mathbf{h}_1$ represent the impulse responses of the systems $\mathbf{H} := \mathbf{B}\mathbf{A}^{-1}$ and $\mathbf{H}_1 := \mathbf{B}_1\mathbf{A}^{-1}$, respectively. Because the assumption is idealized, the upper bounds are too high in general (i.e., they do not guarantee convexity of F). Therefore, in the examples below, we halve these values, i.e., we set

$$a_0 = 0.5\,\|\mathbf{h}\|_2^2/\lambda_0, \qquad a_1 = 0.5\,\|\mathbf{h}_1\|_2^2/\lambda_1. \tag{37}$$

In the non-convex case, we initialize the algorithm with the ℓ_1-norm solution to reduce the likelihood the algorithm becomes trapped in a poor local minimizer. For the ℓ_1-norm penalty, the initialization does not matter, due to its convexity.

We also note that it was assumed in the derivation of (31) and (35) that the total objective function, F, is convex. Hence, suitably constraining the penalties so that F is at least approximately convex is further advantageous, as it approximately justifies the use of (31) and (35) in setting λ_i.

F. LPF/CSD Example 1

This example shows LPF/CSD processing as applied to a simulated signal illustrated in Fig. 1(a). This signal consists of two low-frequency sinusoids, several additive step-transients, and additive white Gaussian noise ($\sigma = 0.5$).

In this example, we compute the solution to the LPF/CSD problem using both the ℓ_1-norm and arctangent penalties. For the filter, we use the second-order zero-phase Butterworth filter described in [34]. We use (31) and (35) to obtain λ_0^* and λ_1^*, and (37) to specify the non-convexity parameters for the arctangent penalty functions. We run the algorithm in Table II for 50 iterations.

With $(\lambda_0, \lambda_1) = (\lambda_0^*, 0)$ and the arctangent penalty, we obtain the estimated transient signal $\hat{\mathbf{x}}$ (i.e., $\hat{\mathbf{x}} = \mathbf{x}_{F_0}^{\text{opt}}$) shown in Fig. 2(a), which can be seen to deviate infrequently from the baseline value of zero. With $(\lambda_0, \lambda_1) = (0, \lambda_1^*)$ we obtain the estimated signal $\hat{\mathbf{x}}$ (i.e., $\hat{\mathbf{x}} = \mathbf{x}_{F_1}^{\text{opt}}$) shown in Fig. 2(b), which does not follow a baseline of zero, but is approximately piecewise constant (i.e., has a sparse derivative). To obtain an estimated transient signal, $\hat{\mathbf{x}}$, which both largely adheres to a baseline of zero and is piecewise constant, we set (λ_0, λ_1) according to (36), with θ manually tuned to a value of 0.3. The result, shown in Fig. 2(c), is reasonably sparse, has a sparse derivative, and is not contaminated by the additive white Gaussian noise of

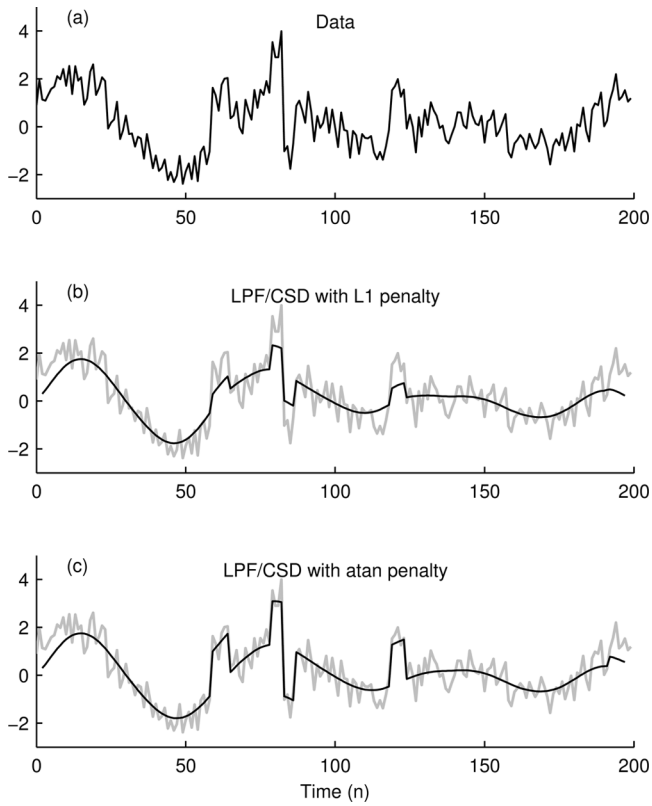


Fig. 1. LPF/CSD Example 1. (a) Simulated data. Denoising using LPF/CSD with the ℓ_1-norm penalty (b) and the arctangent penalty (c).

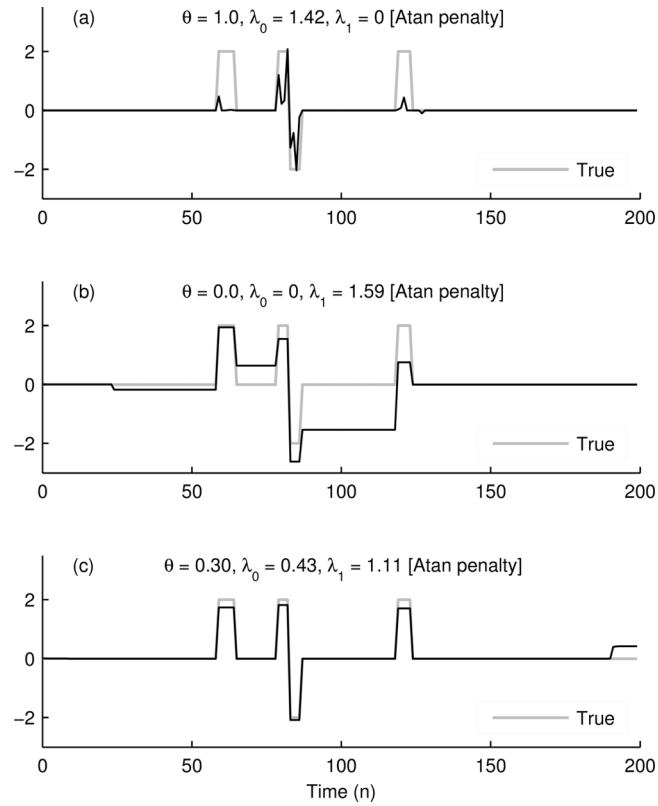


Fig. 2. LPF/CSD Example 1. Estimated transient signals, $\hat{\mathbf{x}}$, obtained by LPF/CSD processing with various (λ_0, λ_1). (a) $\theta = 1$. (b) $\theta = 0$. (c) $\theta = 0.3$.

the data. Reducing both λ_0 and λ_1 leads to a noisy under-regularized artifact signal. Increasing both λ_0 and λ_1 leads to an over-regularized artifact signal where the transient pulses are attenuated in amplitude. The interpolation given by (36) provides a trade-off between these two cases while keeping the total regularization at an appropriate level. Note that the interpolation (36) is in the domain of the regularization parameters, not the estimated signal $\hat{\mathbf{x}}$ itself.

With (λ_0, λ_1) as obtained in Fig. 2(c), the lowpass signal $\hat{\mathbf{f}}$ is obtained using (6). The total signal, $\hat{\mathbf{f}} + \hat{\mathbf{x}}$, is shown in Figs. 1(b) and 1(c) for the ℓ_1-norm and arctangent penalties, respectively. It can be seen that the non-convex arctangent penalty estimates the amplitude of the transients more accurately than the ℓ_1-norm penalty.

G. LPF/CSD Example 2

This example shows artifact reduction using LPF/CSD as applied to the near infrared spectroscopic (NIRS) time series. The NIRS neuroimaging data in this and later examples was acquired at a rate of 6.25 samples/second. Subjects were seated in a fixed chair (no wheels or reclining seat back, etc.), but were otherwise unrestrained. Participants were asked to remain as still and quiet as possible (no talking, etc.).

The data shown in Fig. 3(a) were acquired using a pair of optodes (one source and one detector) on the subject's forehead near the left eye. As such, it is susceptible to artifacts due to eye blinks (in addition to other artifacts ordinarily present). Fig. 3(a) shows transient artifacts of variable amplitude, width, and shape. The time series has a length of 1900 samples, of which 900 samples are shown to better reveal the signal details. The same time series was used in [34].

To apply LPF/CSD processing, we used a second-order zero-phase Butterworth filter with a cut-off frequency of 0.05 cycles/sample, and the arctangent penalty with λ_0^* and λ_1^* set as described in Section II-C, with a pseudo-noise standard deviation of $\sigma = 0.65$. The signals corresponding to $(\lambda_0^*, 0)$ and $(0, \lambda_1^*)$ are shown in Figs. 3(b) and 3(c), respectively. We then set λ_0 and λ_1 using (36) with θ manually tuned to 0.05, which produces the signal, $\hat{\mathbf{x}}$, shown in Fig. 3(d). Only a small value of θ is needed to obtain a signal that adheres to a baseline value of zero. The result provides an apparently accurate estimate of the transient artifacts. The corrected time series, obtained by subtracting the estimated artifact signal $\hat{\mathbf{x}}$ from the original data, is shown in Fig. 3(e). Compared with the result of [34], which used the ℓ_1-norm penalty exclusively, the artifacts appear more accurately estimated here, due to the use of the arctangent penalty in place of the ℓ_1 norm. The MM algorithm was run for 50 iterations, with a run time of about 80 milliseconds.

As noted above, if the LPF/CSD problem (5) is used with the ℓ_1-norm penalty, then 'Algorithm 2' in [34] (derived using ADMM) can be used instead of the algorithm in Table II (derived using MM). However, the ADMM algorithm has the disadvantage that it exhibits slower initial convergence than the MM algorithm, and it requires the user to specify a parameter, μ, which, if improperly specified, can further reduce the initial convergence rate. To illustrate the comparative convergence behavior of the ADMM and MM algorithms, Fig. 4 shows the cost-function history for both. For the ADMM algorithm, the value of μ was optimized to give the smallest cost function value at iteration 50. We denote this optimal value by μ^*. Increasing

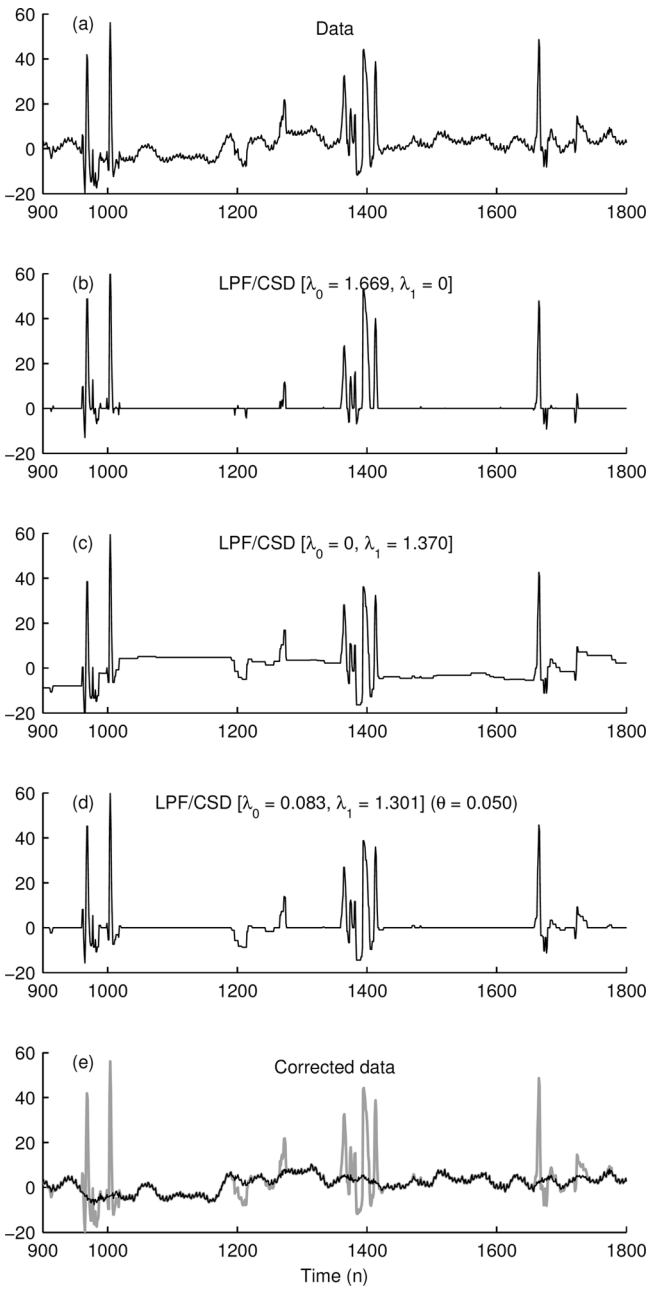


Fig. 3. Reduction of transient artifacts in a NIRS time series using LPF/CSD with the arctangent penalty. (a) Raw data. (b), (c), (d) Output of the LPF/CSD problem with (λ_0, λ_1) given by $(\lambda_0^*, 0)$, $(0, \lambda_1^*)$, and (36). (e) Corrected data (CSD signal (d) subtracted from raw data).

or decreasing μ adversely affects the behavior of the algorithm (with respect to a 50-iteration run). To illustrate this, we run the ADMM algorithm with μ set to $2\mu^*$ and $\mu^*/2$, in addition to μ^*. Fig. 4 shows that $2\mu^*$ has better initial convergence, but the long-term convergence is slower. For $\mu^*/2$, the initial convergence is slower, with no benefit for the first 50 iterations (the value of $\mu^*/2$ benefits the long term convergence). As shown in Fig. 4, when both algorithms are run for 50 iterations, the MM algorithm converges faster than the ADMM algorithm for any value of μ. Moreover, the MM algorithm does not require user-specification of any parameter beyond those that define the cost function F in (5).

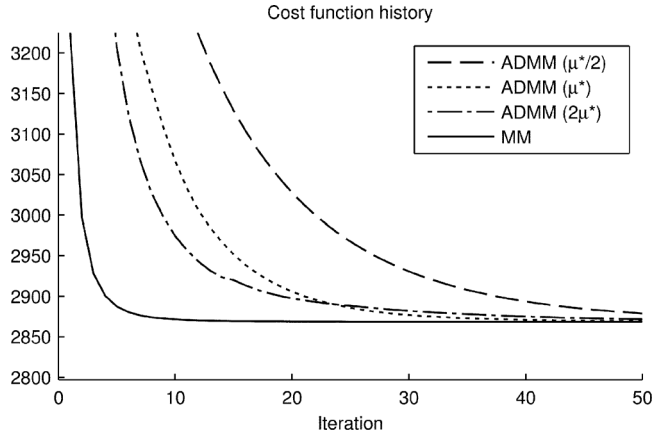


Fig. 4. Comparison of convergence behavior of ADMM and MM algorithms for the LPF/CSD problem.

Note that the MM algorithm minimizes the smooth cost function. To make a fair, direct comparison with the ADMM algorithm, which minimizes the non-smooth cost function, we evaluate the non-smooth cost function for both algorithms in Fig. 4. As the figure shows, even though the MM algorithm minimizes the smooth cost function, it reduces the non-smooth cost function faster than the ADMM algorithm over the first 50 iterations. (Eventually, the ADMM cost will cross below that of the MM cost, but at that point the solutions are negligibly different.) Hence, the slight smoothing of the penalty has an insignificant adverse impact in practice.

III. Transient Artifact Reduction Algorithm

We address problem (1) in the discrete-time setting, and write the model as

$$\mathbf{y} = \mathbf{f} + \mathbf{x}_1 + \mathbf{x}_2 + \mathbf{w}, \qquad \mathbf{y},\ \mathbf{x}_1,\ \mathbf{x}_2,\ \mathbf{w} \in \mathbb{R}^N \tag{38}$$

where $\mathbf{f}$ is a low-pass discrete-time signal, $\mathbf{x}_1$ and $\mathbf{x}_2$ comprise Type 1 and Type 2 artifacts, respectively, and $\mathbf{w}$ is additive white Gaussian noise. That is, $\mathbf{x}_1$, $\mathbf{D}\mathbf{x}_1$, and $\mathbf{D}\mathbf{x}_2$ are modeled as sparse. Similar to Section II, the discrete-time formulation leads to the optimization problem

$$\begin{aligned}\{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2\} = \arg\min_{\mathbf{x}_1,\mathbf{x}_2} \Big\{ &\frac{1}{2}\|\mathbf{H}(\mathbf{y} - \mathbf{x}_1 - \mathbf{x}_2)\|_2^2 \\ &+ \lambda_0 \sum_n \phi_0([\mathbf{x}_1]_n) + \lambda_1 \sum_n \phi_1([\mathbf{D}\mathbf{x}_1]_n) \\ &+ \lambda_2 \sum_n \phi_2([\mathbf{D}\mathbf{x}_2]_n) \Big\}, \end{aligned} \tag{39}$$

where $\mathbf{H}$ denotes the high-pass filter annihilating (approximately) the low-pass signal $\mathbf{f}$. The low-pass signal is then estimated as

$$\hat{\mathbf{f}} = \mathbf{L}(\mathbf{y} - \hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_2) = \mathbf{y} - \hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_2 - \mathbf{H}(\mathbf{y} - \hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_2),$$

where $\mathbf{L}$ denotes the low-pass filter given by $\mathbf{L} = \mathbf{I} - \mathbf{H}$.

As in Section II, we define the filter $\mathbf{H}$ as $\mathbf{H} = \mathbf{B}\mathbf{A}^{-1}$, where $\mathbf{A}$ and $\mathbf{B}$ are banded (see Sec. VI of [34]). We assume $\mathbf{B}$ can be factored as $\mathbf{B} = \mathbf{B}_1\mathbf{D}$. Because $\mathbf{D}$ and $\mathbf{A}^{-1}$ are matrix

representations of LTI systems, they approximately commute. Hence, we formulate the problem as

$$\{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2\} = \arg\min_{\mathbf{x}_1,\mathbf{x}_2} \Big\{ \frac{1}{2}\|\mathbf{H}\mathbf{y} - \mathbf{B}\mathbf{A}^{-1}\mathbf{x}_1 - \mathbf{B}_1\mathbf{A}^{-1}\mathbf{D}\mathbf{x}_2\|_2^2 + \lambda_0 \sum_n \phi_0([\mathbf{x}_1]_n) + \lambda_1 \sum_n \phi_1([\mathbf{D}\mathbf{x}_1]_n) + \lambda_2 \sum_n \phi_2([\mathbf{D}\mathbf{x}_2]_n) \Big\}. \quad (40)$$

The advantage of this form compared to (39) is that it leads to an optimization algorithm that involves banded matrices exclusively, and hence admits a fast implementation.

Note that $\mathbf{x}_2$ is not uniquely determined because adding a constant (dc offset) to $\mathbf{x}_2$ does not change the value of the objective function. Hence, it is sufficient to solve for the derivative (first-order difference), which we then integrate to obtain $\mathbf{x}_2$. We denote the discrete-time integrator by $\mathbf{S}$ (defined such that $\mathbf{D}\mathbf{S} = \mathbf{I}$, as in [34]), and make the change of variables

$$\mathbf{x}_1 = \mathbf{A}\mathbf{u}_1, \qquad \mathbf{D}\mathbf{x}_2 = \mathbf{A}\mathbf{u}_2. \quad (41)$$

Then the data fidelity term in (40) can be written as

$$F_1(\mathbf{u}_1, \mathbf{u}_2) = \frac{1}{2}\|\mathbf{H}\mathbf{y} - \mathbf{B}\mathbf{u}_1 - \mathbf{B}_1\mathbf{u}_2\|_2^2 \quad (42)$$

and the regularization term in (40) can be written as

$$F_2(\mathbf{u}_1, \mathbf{u}_2) = \lambda_0 \sum_n \phi_0([\mathbf{A}\mathbf{u}_1]_n) + \lambda_1 \sum_n \phi_1([\mathbf{D}\mathbf{A}\mathbf{u}_1]_n) + \lambda_2 \sum_n \phi_2([\mathbf{A}\mathbf{u}_2]_n). \quad (43)$$

Then $F = F_1 + F_2$, and in terms of $\mathbf{u}_1$ and $\mathbf{u}_2$ we have the optimization problem

$$\{\hat{\mathbf{u}}_0, \hat{\mathbf{u}}_1\} = \arg\min_{\mathbf{u}_1,\mathbf{u}_2} \big\{F_1(\mathbf{u}_1, \mathbf{u}_2) + F_2(\mathbf{u}_1, \mathbf{u}_2)\big\}. \quad (44)$$

A. Algorithm

To solve (44), we use the majorization-minimization principle. We majorize F_1 with G_1, and F_2 by G_2. Then F is majorized with $G = G_1 + G_2$. Following the MM principle, we then iteratively minimize G. With suitably chosen majorizers G_i, the minimization of G can be implemented with high computational efficiency. After the minimizers $\hat{\mathbf{u}}_1$ and $\hat{\mathbf{u}}_2$ are obtained, we compute $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ as $\hat{\mathbf{x}}_1 = \mathbf{A}\hat{\mathbf{u}}_1$ and $\hat{\mathbf{x}}_2 = \mathbf{S}\mathbf{A}\hat{\mathbf{u}}_2$, in accordance with (41).

To facilitate the following derivation, we define

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}, \mathbf{v} = \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix}. \quad (45)$$

We then write the data fidelity term, F_1, as

$$F_1(\mathbf{u}) = \frac{1}{2}\|\mathbf{H}\mathbf{y} - \begin{bmatrix} \mathbf{B} & \mathbf{B}_1 \end{bmatrix} \mathbf{u}\|_2^2. \quad (46)$$

The matrix of the quadratic term of F_1 (i.e., $\mathbf{C}$ in $\mathbf{u}^\mathsf{T}\mathbf{C}\mathbf{u}$ after expansion of $F_1(\mathbf{u})$) is not banded. Hence, the use of direct minimization by fast banded solvers is precluded. However, a quadratic majorizer of F_1 is given by

$$G_1(\mathbf{u}, \mathbf{v}) = F_1(\mathbf{u}) + \frac{1}{2}(\mathbf{u} - \mathbf{v})^\mathsf{T}\mathbf{P}(\mathbf{u} - \mathbf{v}), \quad (47)$$

where $\mathbf{P}$ is a positive semidefinite matrix. Note that $G_1(\mathbf{u}, \mathbf{u}) = F_1(\mathbf{u})$. Since $\mathbf{P}$ is positive semidefinite, $G_1(\mathbf{u}, \mathbf{v}) \geqslant F_1(\mathbf{u})$ for all $\mathbf{u}$, $\mathbf{v}$. Hence, G_1 is a majorizer of F_1. The matrix $\mathbf{P}$ should, furthermore, be chosen so that G is easily minimized. For that purpose, we set $\mathbf{P}$ so that G_1 has a banded second-order term. Such a majorizer is obtained by taking $\mathbf{P}$ to be

$$\mathbf{P} := \alpha \begin{bmatrix} \mathbf{B}^\mathsf{T}\mathbf{B} & \\ & \mathbf{B}_1^\mathsf{T}\mathbf{B}_1 \end{bmatrix} - \begin{bmatrix} \mathbf{B}^\mathsf{T} \\ \mathbf{B}_1^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathbf{B} & \mathbf{B}_1 \end{bmatrix}, \quad (48)$$

where α is chosen so that $\mathbf{P}$ is positive semidefinite. With $\mathbf{P}$ chosen as in (48), the second-order term of F_1 is canceled and the second-order term of G_1 is banded. The matrix $\mathbf{P}$ can be written as

$$\mathbf{P} = \begin{bmatrix} \mathbf{B}^\mathsf{T} & \\ & \mathbf{B}_1^\mathsf{T} \end{bmatrix} \left(\alpha \begin{bmatrix} \mathbf{I} & \\ & \mathbf{I} \end{bmatrix} - \begin{bmatrix} \mathbf{I} \\ \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{I} \end{bmatrix} \right) \begin{bmatrix} \mathbf{B} & \\ & \mathbf{B}_1 \end{bmatrix} \quad (49)$$

$$= \begin{bmatrix} \mathbf{B}^\mathsf{T} & \\ & \mathbf{B}_1^\mathsf{T} \end{bmatrix} \left(\begin{bmatrix} \alpha - 1 & -1 \\ -1 & \alpha - 1 \end{bmatrix} \otimes \mathbf{I} \right) \begin{bmatrix} \mathbf{B} & \\ & \mathbf{B}_1 \end{bmatrix}, \quad (50)$$

where $\otimes$ denotes the Kronecker product. Hence, $\mathbf{P}$ is positive semidefinite if the 2×2 matrix in (50) is positive semidefinite. Its eigenvalues are α and $\alpha - 2$; hence, it is positive semidefinite for $\alpha \geqslant 2$. Therefore, we set $\alpha = 2$ in the following. With this α value, the matrix $\mathbf{P}$ is given by

$$\mathbf{P} = \begin{bmatrix} \mathbf{B}^\mathsf{T}\mathbf{B} & -\mathbf{B}^\mathsf{T}\mathbf{B}_1 \\ -\mathbf{B}_1^\mathsf{T}\mathbf{B} & \mathbf{B}_1^\mathsf{T}\mathbf{B}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{B}^\mathsf{T} \\ -\mathbf{B}_1^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathbf{B} & -\mathbf{B}_1 \end{bmatrix}, \quad (51)$$

and $G_1(\mathbf{u}, \mathbf{v})$ is given by

$$G_1(\mathbf{u}, \mathbf{v}) = F_1(\mathbf{u}) + \frac{1}{2}(\mathbf{u} - \mathbf{v})^\mathsf{T} \begin{bmatrix} \mathbf{B}^\mathsf{T} \\ -\mathbf{B}_1^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathbf{B} & -\mathbf{B}_1 \end{bmatrix} (\mathbf{u} - \mathbf{v}).$$

A majorizer of the regularization term F_2 is given by

$$G_2(\mathbf{u}, \mathbf{v}) = \frac{\lambda_0}{2}\mathbf{u}_1^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{\Lambda}(\mathbf{A}\mathbf{v}_1; \phi_0)\mathbf{A}\mathbf{u}_1 + \frac{\lambda_1}{2}\mathbf{u}_1^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{\Lambda}(\mathbf{D}\mathbf{A}\mathbf{v}_1; \phi_1)\mathbf{D}\mathbf{A}\mathbf{u}_1 + \frac{\lambda_2}{2}\mathbf{u}_2^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{\Lambda}(\mathbf{A}\mathbf{v}_2; \phi_2)\mathbf{A}\mathbf{u}_2 + C, \quad (52)$$

where $\mathbf{\Lambda}(\mathbf{v}; \phi)$ is the diagonal matrix defined in (14) and C does not depend on $\mathbf{u}$. We can write G_2 as

$$G_2(\mathbf{u}, \mathbf{v}) = \frac{1}{2}\mathbf{u}^\mathsf{T} \begin{bmatrix} \mathbf{R}_1(\mathbf{v}_1) & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2(\mathbf{v}_2) \end{bmatrix} \mathbf{u} + C, \quad (53)$$

where $\mathbf{R}_i(\mathbf{v}_i)$ are the banded matrices

$$\mathbf{R}_1(\mathbf{v}_1) = \mathbf{A}^\mathsf{T}[\lambda_0\mathbf{\Lambda}(\mathbf{A}\mathbf{v}_1; \phi_0) + \lambda_1\mathbf{D}^\mathsf{T}\mathbf{\Lambda}(\mathbf{D}\mathbf{A}\mathbf{v}_1; \phi_1)\mathbf{D}]\mathbf{A}$$
$$\mathbf{R}_2(\mathbf{v}_2) = \lambda_2\mathbf{A}^\mathsf{T}\mathbf{\Lambda}(\mathbf{A}\mathbf{v}_2; \phi_2)\mathbf{A}.$$

Consequently, a quadratic majorizer of F is given by

$$G(\mathbf{u}, \mathbf{v}) = G_1(\mathbf{u}, \mathbf{v}) + G_2(\mathbf{u}, \mathbf{v}). \quad (54)$$

To find $\mathbf{u}$ minimizing G, we set to zero the gradient of G with respect to $\mathbf{u}$, to obtain a system of linear equations. Due to the way we have defined the objective function and the majorizer, the system of linear equations will be banded. We have

$$\nabla_{\mathbf{u}} G_1(\mathbf{u},\mathbf{v}) = \nabla_{\mathbf{u}} F_1(\mathbf{u}) + \begin{bmatrix} \mathbf{B}^{\mathsf{T}} \\ -\mathbf{B}_1^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} \mathbf{B} & -\mathbf{B}_1 \end{bmatrix} (\mathbf{u}-\mathbf{v}), \tag{55}$$

with

$$\nabla_{\mathbf{u}} F_1(\mathbf{u}) = \begin{bmatrix} \mathbf{B}^{\mathsf{T}} \\ \mathbf{B}_1^{\mathsf{T}} \end{bmatrix} \left(\begin{bmatrix} \mathbf{B} & \mathbf{B}_1 \end{bmatrix} \mathbf{u} - \mathbf{H}\mathbf{y} \right).$$

We also have

$$\nabla_{\mathbf{u}} G_2(\mathbf{u},\mathbf{v}) = \begin{bmatrix} \mathbf{R}_1(\mathbf{v}_1) & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2(\mathbf{v}_2) \end{bmatrix} \mathbf{u}. \tag{56}$$

Hence, $\nabla_{\mathbf{u}} G = \nabla_{\mathbf{u}} G_1 + \nabla_{\mathbf{u}} G_2 = \mathbf{0}$ leads to the linear system

$$\begin{bmatrix} 2\mathbf{B}^{\mathsf{T}}\mathbf{B} + \mathbf{R}_1(\mathbf{v}_1) & \\ & 2\mathbf{B}_1^{\mathsf{T}}\mathbf{B}_1 + \mathbf{R}_2(\mathbf{v}_2) \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{B}^{\mathsf{T}} \\ \mathbf{B}_1^{\mathsf{T}} \end{bmatrix} \mathbf{H}\mathbf{y} + \begin{bmatrix} \mathbf{B}^{\mathsf{T}} \\ -\mathbf{B}_1^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} \mathbf{B} & -\mathbf{B}_1 \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix}. \tag{57}$$

Note that the system matrix is banded. The solution to (57) is given by

$$\mathbf{g} = \mathbf{B}\mathbf{v}_1 - \mathbf{B}_1\mathbf{v}_2 \tag{58}$$

$$\mathbf{u}_1 = \left[2\mathbf{B}^{\mathsf{T}}\mathbf{B} + \mathbf{R}_1(\mathbf{v}_1)\right]^{-1} \left(\mathbf{B}^{\mathsf{T}}\mathbf{H}\mathbf{y} + \mathbf{B}^{\mathsf{T}}\mathbf{g}\right) \tag{59}$$

$$\mathbf{u}_2 = \left[2\mathbf{B}_1^{\mathsf{T}}\mathbf{B}_1 + \mathbf{R}_2(\mathbf{v}_2)\right]^{-1} \left(\mathbf{B}_1^{\mathsf{T}}\mathbf{H}\mathbf{y} - \mathbf{B}_1^{\mathsf{T}}\mathbf{g}\right). \tag{60}$$

Hence, minimizing the majorizer G according to the MM update (17) leads to

$$\mathbf{g}^{(i)} = \mathbf{B}\mathbf{u}_1^{(i)} - \mathbf{B}_1\mathbf{u}_2^{(i)} \tag{61}$$

$$\mathbf{u}_1^{(i+1)} = \left[2\mathbf{B}^{\mathsf{T}}\mathbf{B} + \mathbf{R}_1(\mathbf{u}_1^{(i)})\right]^{-1} \left(\mathbf{B}^{\mathsf{T}}\mathbf{H}\mathbf{y} + \mathbf{B}^{\mathsf{T}}\mathbf{g}^{(i)}\right) \tag{62}$$

$$\mathbf{u}_2^{(i+1)} = \left[2\mathbf{B}_1^{\mathsf{T}}\mathbf{B}_1 + \mathbf{R}_2(\mathbf{u}_2^{(i)})\right]^{-1} \left(\mathbf{B}_1^{\mathsf{T}}\mathbf{H}\mathbf{y} - \mathbf{B}_1^{\mathsf{T}}\mathbf{g}^{(i)}\right). \tag{63}$$

Equations (61)–(63) constitute the iterative algorithm, TARA, summarized in Table III. Note that the system matrices are banded; hence, the algorithm can be implemented using fast solvers for banded systems. The vectors $\mathbf{B}^{\mathsf{T}}\mathbf{H}\mathbf{y}$ and $\mathbf{B}_1^{\mathsf{T}}\mathbf{H}\mathbf{y}$ need to be computed one time only. The algorithm does not require any parameters other than the ones in (40). After $\mathbf{u}_1$ and $\mathbf{u}_2$ are obtained upon convergence of the algorithm, $\mathbf{x}_1$ and $\mathbf{x}_2$ are obtained using (41).

Run times of the new LPF/CSD algorithm and of TARA are shown in Fig. 5, as measured using a 2013 MacBook Pro (2.5 GHz Intel Core i5) running Matlab R2011a. The algorithms were run with a second order filter $\mathbf{H}$ and for 50 iterations. The run times are linear in the signal length, N.

B. Parameters

To use TARA (i.e., to solve (40)), three regularization parameters λ_i must be specified; if non-convex penalties are utilized, then three non-convexity parameters a_i must be specified as well. If λ_2 is appropriately set (i.e., so that $\hat{\mathbf{x}}_2 \approx \mathbf{x}_2$), then $\mathbf{y} - \hat{\mathbf{x}}_2$ can be modeled as having Type 1 artifacts only, and λ_0 and λ_1 can be nominally set as in Section II-C. In the examples below, we set λ_0 and λ_1 using (36) with $\theta \in (0,1)$ manually tuned so that $\hat{\mathbf{x}}_0$ adheres to a baseline value of zero.

TABLE III
TRANSIENT ARTIFACT REDUCTION ALGORITHM (TARA)

	Input: $\mathbf{y} \in \mathbb{R}^N,\ \lambda_i > 0,\ \phi_i, \mathbf{A}, \mathbf{B}$	
1.	$\mathbf{y}_1 = \mathbf{B}^{\mathsf{T}}\mathbf{B}\mathbf{A}^{-1}\mathbf{y}$	($\mathbf{B}^{\mathsf{T}}\mathbf{H}\mathbf{y}$)
2.	$\mathbf{y}_2 = \mathbf{B}_1^{\mathsf{T}}\mathbf{B}\mathbf{A}^{-1}\mathbf{y}$	($\mathbf{B}_1^{\mathsf{T}}\mathbf{H}\mathbf{y}$)
3.	$\mathbf{u}_1 = \mathbf{0},\ \mathbf{u}_2 = \mathbf{0}$	(initialization)
	Repeat	
4.	$[\mathbf{\Lambda}_0]_{n,n} = \lambda_0/\psi_0([\mathbf{A}\mathbf{u}_1]_n)$	($\mathbf{\Lambda}_0$ is diagonal)
5.	$[\mathbf{\Lambda}_1]_{n,n} = \lambda_1/\psi_1([\mathbf{D}\mathbf{A}\mathbf{u}_1]_n)$	($\mathbf{\Lambda}_1$ is diagonal)
6.	$[\mathbf{\Lambda}_2]_{n,n} = \lambda_2/\psi_2([\mathbf{A}\mathbf{u}_2]_n)$	($\mathbf{\Lambda}_2$ is diagonal)
7.	$\mathbf{Q}_1 = 2\mathbf{B}^{\mathsf{T}}\mathbf{B} + \mathbf{A}^{\mathsf{T}}(\mathbf{\Lambda}_0 + \mathbf{D}^{\mathsf{T}}\mathbf{\Lambda}_1\mathbf{D})\mathbf{A}$	($\mathbf{Q}_1$ is banded)
8.	$\mathbf{Q}_2 = 2\mathbf{B}_1^{\mathsf{T}}\mathbf{B}_1 + \mathbf{A}^{\mathsf{T}}\mathbf{\Lambda}_2\mathbf{A}$	($\mathbf{Q}_2$ is banded)
9.	$\mathbf{g} = \mathbf{B}\mathbf{u}_1 - \mathbf{B}_1\mathbf{u}_2$	
10.	$\mathbf{u}_1 = \mathbf{Q}_1^{-1}\left(\mathbf{y}_1 + \mathbf{B}^{\mathsf{T}}\mathbf{g}\right)$	
11.	$\mathbf{u}_2 = \mathbf{Q}_2^{-1}\left(\mathbf{y}_2 - \mathbf{B}_1^{\mathsf{T}}\mathbf{g}\right)$	
	Until convergence	
12.	$\mathbf{x}_1 = \mathbf{A}\mathbf{u}_1,\ \ \mathbf{x}_2 = \mathbf{S}\mathbf{A}\mathbf{u}_2$	
	Output: $\mathbf{x}_1, \mathbf{x}_2$	

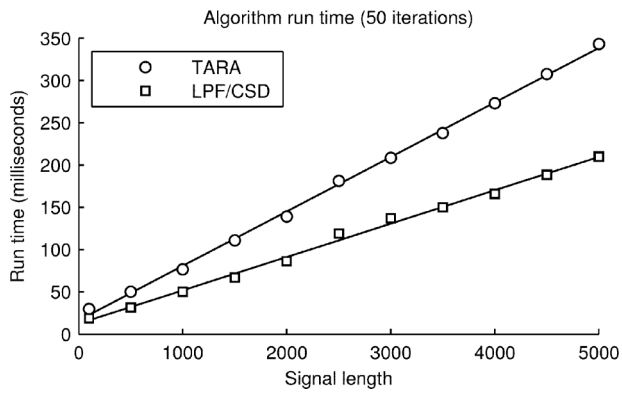


Fig. 5. Run times of LPF/CSD and TARA with linear fits.

The λ_2 parameter should be set, in part, according to the noise variance. A constraint on λ_2 is obtained via the noise analysis described in Section II-C. Namely, we obtain the same nominal value (35) as for λ_1^*. Hence, to prevent noise contamination of $\hat{\mathbf{x}}_2$, we require $\lambda_2 \geqslant \lambda_1^*$.

The Type 1 and Type 2 artifacts are implicitly and operationally defined through the λ_i parameters. Their relative values affects the apportionment of signal features between $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$. However, the total artifact signal $\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$, and $\hat{\mathbf{f}}$, are quite robust to small changes in λ_i, as will be illustrated in the Example in Section III-C.

We note that it is reasonable to set $\lambda_2 > \lambda_1$. Because, if we set $\lambda_2 \leqslant \lambda_1$ and $\lambda_0 > 0$ in (40), then $\mathbf{x}_1$ is strictly more regularized than $\mathbf{x}_2$ and, consequently, the solution $\hat{\mathbf{x}}_1$ in (40) will always be identically zero. It can be said that setting λ_i in this way leads to all the transient artifacts being classified as Type 2. In this case,

the $\mathbf{x}_1$ component may as well be omitted from the problem formulation.

The parameter λ_2 may be set so as to distinguish between Type 1 and Type 2 artifacts. To consider how λ_2 influences the implicit distinction between Type 1 and Type 2 artifacts, consider problem (40) with ℓ_1 norm penalties,

$$F(\mathbf{x}_1,\mathbf{x}_2)=\frac{1}{2}\|\mathbf{H}(\mathbf{y}-\mathbf{x}_1-\mathbf{x}_2)\|_2^2+\lambda_0\|\mathbf{x}_1\|_1 +\lambda_1\|\mathbf{D}\mathbf{x}_1\|_1+\lambda_2\|\mathbf{D}\mathbf{x}_2\|_1. \quad (64)$$

Suppose $\mathbf{y}$ is all zero except for some transient pulse that we consider a Type 1 artifact. It can be expected that the minimizer of F is likewise some transient pulse, which we denote by $\mathbf{p}$; i.e., $\hat{\mathbf{x}}_1+\hat{\mathbf{x}}_2=\mathbf{p}$. If $\mathbf{p}$ is considered a Type 1 artifact, then λ_i should be set so that $\hat{\mathbf{x}}_1=\mathbf{p}$ and $\hat{\mathbf{x}}_2=\mathbf{0}$. To find a rule for setting the parameter values, we evaluate the objective function F for two candidate solutions:

$$S_1=\{\mathbf{x}_1=\mathbf{p},\ \mathbf{x}_2=\mathbf{0}\},\quad S_2=\{\mathbf{x}_1=\mathbf{0},\ \mathbf{x}_2=\mathbf{p}\}. \quad (65)$$

If S_1 minimizes F, then TARA correctly classifies $\mathbf{p}$ as a Type 1 artifact; while if S_2 minimizes F, then TARA incorrectly classifies $\mathbf{p}$ as a Type 2 artifact. Solution S_1 can be the optimal solution only if $F(S_1)<F(S_2)$. Because $(\mathbf{x}_1+\mathbf{x}_2)$ is the same for solutions S_1 and S_2, the data fidelity term of F is equal for S_1 and S_2. Hence, the relative cost depends only on the penalty terms. Therefore, we have

$$\lambda_0\|\mathbf{p}\|_1+\lambda_1\|\mathbf{D}\mathbf{p}\|_1 \underset{S_2}{\overset{S_1}{\lessgtr}} \lambda_2\|\mathbf{D}\mathbf{p}\|_1 \quad (66)$$

or

$$\lambda_0\frac{\|\mathbf{p}\|_1}{\|\mathbf{D}\mathbf{p}\|_1}+\lambda_1 \underset{S_2}{\overset{S_1}{\lessgtr}} \lambda_2. \quad (67)$$

The notation $\lessgtr$ means S_1 is the optimal solution if the left-hand side is the smaller value, and vice-versa. Hence, for $\mathbf{p}$ to be classified as a Type 1 artifact by TARA, λ_2 must be at least as great as the quantity on the left-hand side of (67). Note that condition (67) is invariant to amplitude scaling of $\mathbf{p}$; i.e., only the shape of $\mathbf{p}$ matters.

As an example, suppose that $\mathbf{p}$ is taken to be a rectangular pulse of length M samples and amplitude A. Then $\|\mathbf{p}\|_1=MA$ and $\|\mathbf{D}\mathbf{p}\|_1=2A$, so condition (67) can be written as

$$0.5M\lambda_0+\lambda_1 \underset{S_2}{\overset{S_1}{\lessgtr}} \lambda_2. \quad (68)$$

Hence, for the M-point pulse to be exhibited in $\mathbf{x}_1$, the parameter λ_2 must exceed λ_1 by $0.5M\lambda_0$. When $(\lambda_0,\lambda_1)=(\theta\lambda_0^*,(1-\theta)\lambda_1^*)$, as suggested in (36), then (68) gives a condition in terms of θ,

$$\theta(0.5M\lambda_0^*-\lambda_1^*)+\lambda_1^* \underset{S_2}{\overset{S_1}{\lessgtr}} \lambda_2. \quad (69)$$

We noted above that λ_2 should satisfy $\lambda_2\geqslant\lambda_1^*$. Hence, λ_2 should be set according to

$$\lambda_2\geqslant\max\{\lambda_1^*,\ \lambda_1^*+\theta(0.5M\lambda_0^*-\lambda_1^*)\}. \quad (70)$$

It experiments, we have found that $0.5M\lambda_0^*-\lambda_1^*$ is usually positive, so the second term dominates. Moreover, θ is often relatively small in practice (sufficient so that $\mathbf{x}_1$ adheres to a baseline of zero). Hence, it will often be sufficient that λ_2 be only slightly larger than λ_1^*. The use of condition (68) to control the behavior of TARA is illustrated in Section III-C.

Based on the forgoing considerations, we suggest writing $\lambda_2=\beta\lambda_1^*$ and taking β as a tuning parameter with a nominal range of $\beta\in[1,2]$. In conjunction with the discussion in Section II-D, we obtain a parameterization of the TARA problem (40) in terms of (σ,θ,β) instead of $(\lambda_0,\lambda_1,\lambda_2)$. We find this parameterization more useful in practice because the influence of each parameter can be more readily understood. In particular, we consider (θ,β) to be *shape* parameters; they influence the shape of the estimated transients.

When non-convex penalties are utilized, a_0 and a_1 can be set as in (37). Following the same considerations as in Section II-E, we set $a_2=0.5\|\mathbf{h}_1\|_2^2/\lambda_2$ like for a_1.

C. TARA Example 1

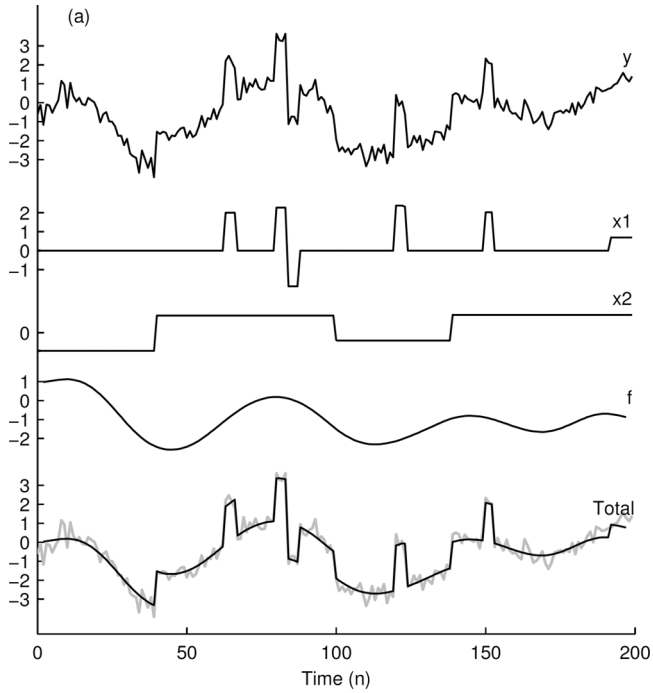
This example shows TARA as applied to the simulated time series $\mathbf{y}$ shown in Fig. 6. The signal consists of two low-frequency sinusoids, several additive rectangular pulses of short duration, several additive step discontinuities, and additive white Gaussian noise ($\sigma=0.3$). Each of the rectangular pulses has a length of four samples, except for the last pulse (at $n=150$) which has a length of three samples.

In this example, we use a fourth-order zero-phase Butterworth filter with $f_c=0.03$ cycles/sample. We also use the non-convex arctangent penalty, set λ_0^* and λ_1^* according to (31) and (35) in Section II-C, and set $\theta=0.3$ by manual tuning as in Section II-F.

To demonstrate the influence of λ_2 as discussed in Section III-B, we consider the question of how to set λ_2 to ensure that the brief rectangular pulses appear in $\hat{\mathbf{x}}_1$ rather than in $\hat{\mathbf{x}}_2$ (i.e., to ensure TARA classifies these pulses as Type 1 artifacts). Since all the brief pulses are of length 4 or less, we set $M=4$ in (68) to find that $2\lambda_0+\lambda_1$ is the critical value for λ_2. Hence, we must set $\lambda_2>2\lambda_0+\lambda_1$ to ensure that the pulses appear in $\hat{\mathbf{x}}_1$. Therefore, we set $\lambda_2=1.1\times(2\lambda_0+\lambda_1)$; i.e., $\beta=1.1$. The output of TARA for this λ_2 is shown in Fig. 6(a). In conformity with our expectation, all the brief pulses are exhibited in $\hat{\mathbf{x}}_1$, i.e., they are classified by TARA as Type 1 artifacts. The signal $\hat{\mathbf{x}}_2$ is piecewise constant and contains no brief pulses. (The small step at the end of $\hat{\mathbf{x}}_1$ is a boundary artifact due to applying the recursive filter $\mathbf{H}$ to a finite-length signal.)

To further illustrate the role of λ_2, we set $\lambda_2=0.9\times(2\lambda_0+\lambda_1)$. This value is less than the critical value needed to classify a length-4 pulse as a Type 1 artifact. Accordingly, it is expected that TARA will classify pulses of length 4 and longer as Type 2 artifacts and that they will be exhibited in $\hat{\mathbf{x}}_2$. The output of TARA for this value of λ_2 is shown in Fig. 6(b). As predicted, the length-4 pulses are exhibited in $\hat{\mathbf{x}}_2$. The only pulse exhibited in $\hat{\mathbf{x}}_1$ is the final one (at $n=150$), which is of length 3. This example validates the use of λ_2 for the disambiguation of pulses based on their duration.

This example uses rectangular pulses because of the availability of the simple formula (68). TARA does not explicitly



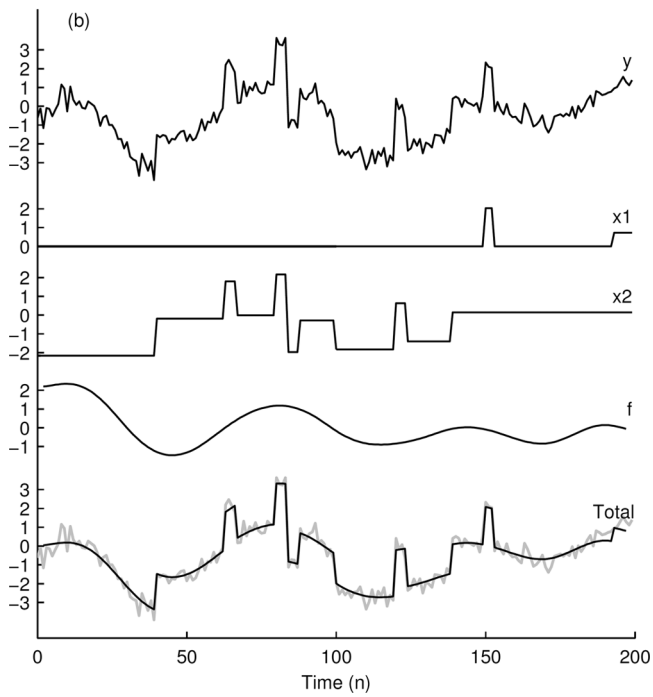


Fig. 6. Signal decomposition and filtering with TARA. (a) Pulses 4 samples and shorter appear in $\mathbf{x}_1$. (b) Pulses 4 samples and longer appear in $\mathbf{x}_2$.

model a signal in terms of rectangular pulses, and its effectiveness is not limited to rectangular artifacts. For real data with transient artifacts of complex shape, it is not expected that a simple formula for a critical value will be available; however, the general influence of λ_2 on the relative properties of $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ holds. Namely, decreasing λ_2 results in more waveforms being classified as Type 2 artifacts.

Note that the low-pass signal, $\hat{\mathbf{f}}$, is essentially the same in Figs. 6(a) and 6(b). Likewise, the total signal, $\hat{\mathbf{x}} = \hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2 + \hat{\mathbf{f}}$, is approximately the same in both figures. The small change in λ_2 produced only a small change in the total signal, even though it produced a large change in $\hat{\mathbf{x}}_i$. Hence, for the purpose of denoising, the total signal is not overly sensitive to the exact value of λ_2. Note that TARA provides a reasonable denoising result: the total signal is relatively noise-free, preserves the discontinuities in the data, and does not exhibit ringing around the discontinuities.

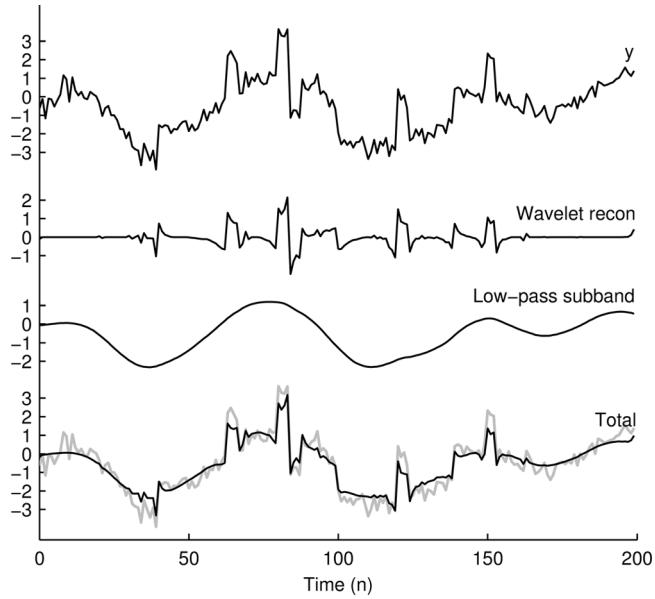


Fig. 7. Wavelet-based decomposition of a signal into transient and low-pass components.

Since wavelet methods have been successfully used for the correction of transient artifacts [9], [21], [24], we illustrate a wavelet-based decomposition in Fig. 7 of the same example signal as was considered in Fig. 6. We use the stationary (un-decimated) wavelet transform [11] with the Haar wavelet filter and the non-negative garrote threshold function [19] (same as in [21]). As shown, the reconstruction of the signal from the thresholded wavelet coefficients (excluding the low-pass component) adheres to a baseline of zero, captures the transient pulses, and is approximately noise-free. The reconstruction of the signal from the low-pass (non-thresholded) wavelet coefficients is a smooth noise-free signal. However, using wavelet transforms, the step-changes (Type 2 transients) cannot be easily isolated from the rest of the signal, because the step discontinuities are represented by both high-frequency and low-frequency wavelet coefficients. For example, the step-discontinuity at time index 40 appears as a bi-phasic pulse in the wavelet-reconstructed signal, and the change in baseline value induced by the step-discontinuity is absorbed into the low-pass component. As a consequence, the wavelet-estimate of the signal exhibits spurious ripples and cusps where the signal has discontinuities. In addition, the low-pass component of the wavelet decomposition less accurately recovers the low-pass component of the data. In this example, the low-pass signal as estimated by the wavelet method and TARA have RMSEs of 0.85 and 0.29, respectively; i.e., TARA recovers the low-pass component more accurately.

D. TARA Example 2

This example illustrates TARA as applied to the NIRS time series shown in Fig. 8, which comes from the same measure-

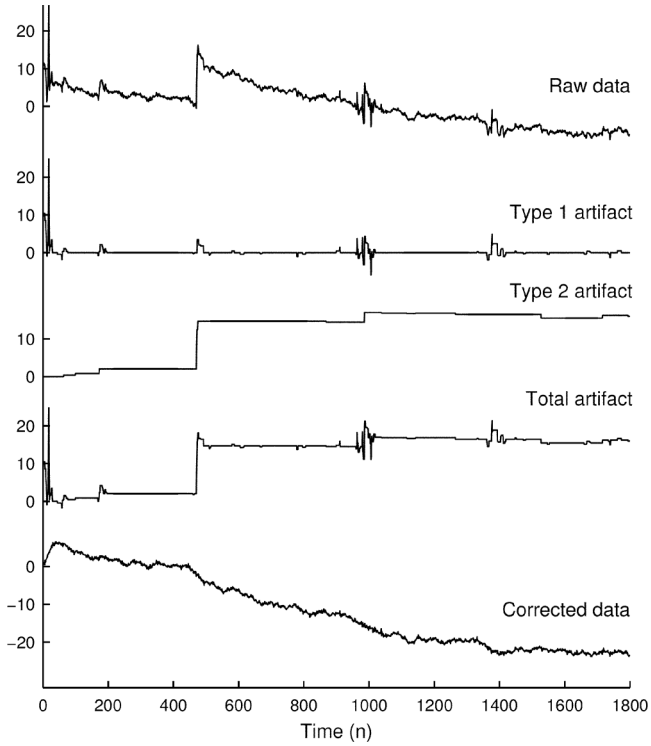


Fig. 8. Artifact reduction with TARA using the arctangent penalty, as applied to a NIRS time series.

ment that produced the data used shown in Fig. 3. The time series was acquired using a pair of optodes on the back of a subject's head; as such, the data is susceptible to motion artifacts which can cause abrupt shifts of the baseline. A prominent baseline shift can be seen at time index 470 in Fig. 8(a). Other motion artifacts also are visible. This data was also used in [34].

To apply TARA for artifact suppression, we must specify the filter $\mathbf{H}$, the three regularization parameters, and the penalty functions. We used a second-order zero-phase Butterworth filter with $f_c = 0.06$ cycles/sample. The tuning procedure described in Section II-D was used to set λ_0^* and λ_1^* with a pseudo-noise standard deviation of $\sigma = 0.25$. We manually tuned the shape parameters to $\theta = 0.05$ and $\beta = 1.4$. The arctangent penalty was used, with non-convexity parameters set according to (37). We ran TARA for 100 iterations with a run time of about 0.28 seconds.

The Type 1 and Type 2 artifact signals estimated by TARA, $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$, shown in Fig. 8, are sparse and approximately piecewise constant, as intended. The estimated total artifact signal, $\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$, which comprises additive step discontinuities and transient spikes, appears to accurately model the artifacts present in the data. Note that the corrected time series, obtained by subtracting the total estimated artifact signal from the original time series, has both low-frequency and high-frequency spectral content. Compared with [34], in which LPF/TVD processing is applied to the same data, the artifacts appear to be more accurately estimated here.

E. Wavelet-based Artifact Reduction

It has been found that wavelet methods compare favorably to other methods for the correction of motion artifacts in single-channel NIRS time series [9], [21], [24]. Fig. 9 compares wavelet transient artifact reduction (WATAR) and TARA as applied to the NIRS data from Fig. 8. As in [21], we use the stationary (un-decimated) wavelet transform [11] with the Haar wavelet filter and the non-negative garrote threshold function [19]. We apply thresholding to all subbands except the low-pass one. The wavelet-corrected time series does not have the long-term drift that the TARA-corrected time series has; however, that is easily removed by LTI filtering and its removal is not an objective of TARA. Moreover, some biological information may be present at low frequencies (see Section III-G).

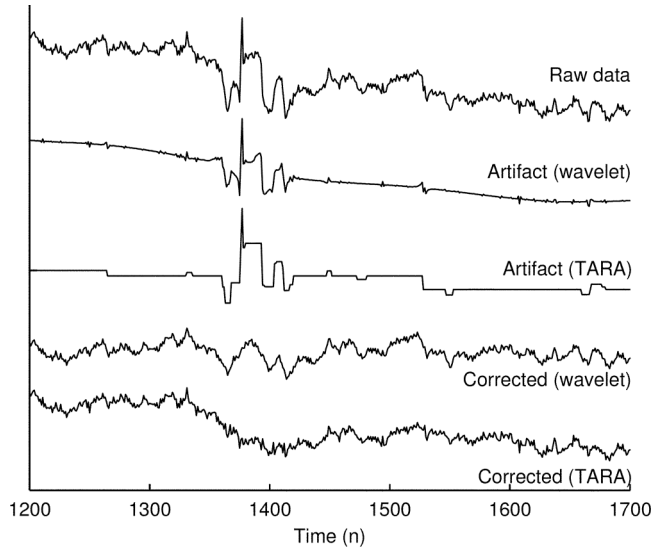


Fig. 9. Artifact estimation and correction using wavelets and TARA.

It can be seen that both methods otherwise give generally similar results, but the TARA-estimated artifact signal captures abrupt changes in the data, unlike the wavelet-estimated artifact signal. The artifact in the interval 1370–1420 is estimated by TARA with distinct pre- and post-artifact baseline values; whereas the wavelet-estimated artifact signal exhibits a small change due to the slowly-varying low-pass component (coming from the low-pass subband of the wavelet transform). In addition, TARA finds an abrupt change at time index 1530; while the wavelet method exhibits only a small bi-phasic (zero-mean) pulse at that instant. TARA is better able to estimate abrupt step-changes than the wavelet method because it is explicitly based on a two-component model. In NIRS time series analysis, motion artifacts often cause step-changes, and this motivates the accurate estimation thereof.

That TARA and wavelet methods give similar results can be explained by their being based on similar underlying models. The wavelet method implicitly models transient artifacts as piecewise smooth. TARA is based on a similar model, but uses an optimization approach instead of a fixed transform.

F. Multichannel Data

Physiological time-series data (e.g., NIRS, EEG) are often acquired in multichannel form. If different regularization parameters are to be required for each channel, then setting the parameters will be a problematic issue. In this example, we apply TARA to multichannel data (Fig. 10, black) and use the same shape parameters (α, β) and filter $\mathbf{H}$ for all channels. The psuedo-noise

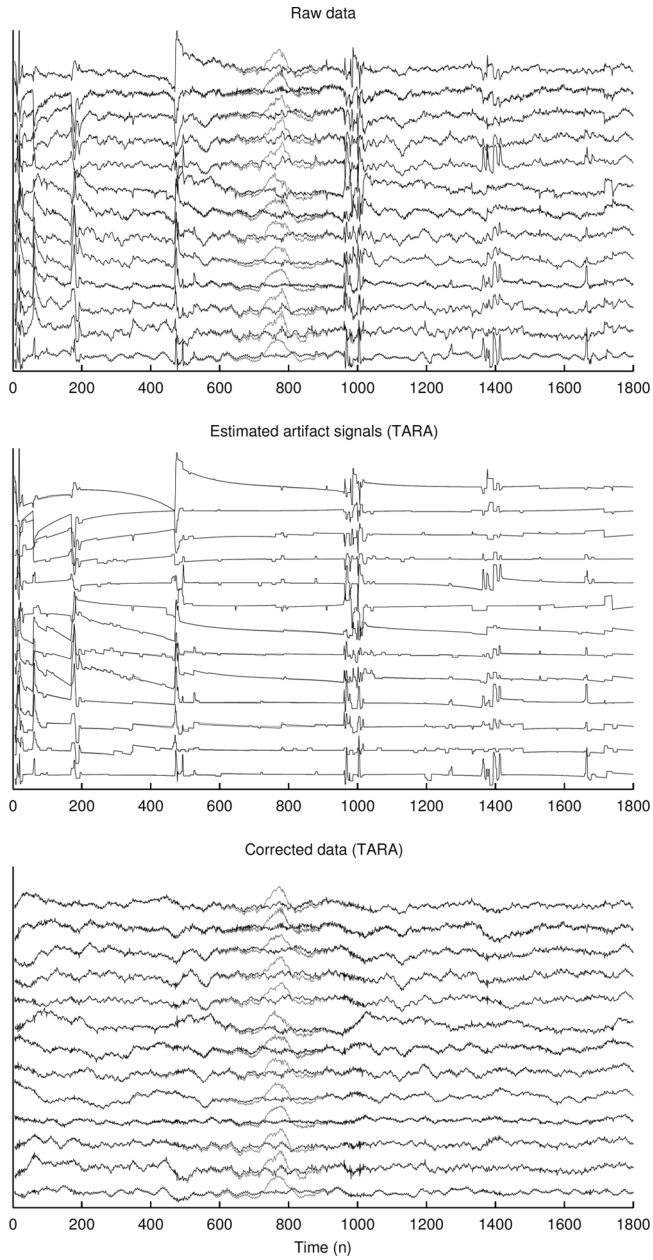


Fig. 10. TARA applied to multichannel data.

parameter σ should, however, be set channel-by-channel, because the channels are not equally normalized. For this illustration, we set σ for each channel based on the artifact-free segment of the data extending from 550 to 950 time frames (see Fig. 10). The σ values were set by applying a high-pass filter to each channel, and then computing the standard deviation of its output in the artifact-free segment.

The artifacts and corrected data obtained using TARA are shown in Fig. 10. For the purpose of display, to avoid cumulative baseline drift, each corrected time series has been filtered with a zero-phase second-order recursive dc-notch filter. TARA effectively reduces transient artifacts in most channels, without introducing substantial distortion. Some channels of the corrected data exhibit slow waves around the removed transients; these are consistent with the assumed signal model, wherein the signal is modeled as comprising a low-pass signal. This example suggests that with a fixed pair of shape parameter values (α, β) and filter $\mathbf{H}$, TARA may be quite effective for multiple channels and that, given an artifact-free segment (obtained here by visual inspection), it may be possible to use TARA in a data-driven automated manner.

G. Preservation of Hemodynamic Response

In the course of suppressing artifacts, biological information of interest should not be distorted or attenuated. For NIRS specifically, any hemodynamic response (HR) waveforms present should be preserved. Equivalently, the artifact signals should not be affected by HRs, because the HRs should not be mistaken for artifacts. To test TARA in this regard, we add a simulated HR to each channel of the considered multichannel data. The data with the simulated HR is shown in gray in Fig. 10. The artifact signals and corrected data, are likewise shown in gray. The gray-colored artifact signals, obtained from the HR-added data, are nearly indistinguishable from the original artifact signals. Accordingly, TARA accurately preserves the HR in the corrected data.

To compare TARA and WATAR (i.e., wavelets), we likewise apply WATAR to the same multichannel data, with and without the added HR. The WATAR-estimated artifact signals (Fig. 11) exhibit a noticeable portion of the HR in about half the channels. Consequently, the HR is more attenuated and distorted in the WATAR-corrected data than in the TARA-corrected data. In quantitative terms, we measure the root-mean-square deviation over the HR interval (700–880) between the HR and non-HR artifact signals. The value is 0.18 for TARA and 0.53 for WATAR. By this measure, WATAR is affected by the HR 2.9 times more than TARA, and so TARA better preserves the HR than WATAR. This is intended as an illustrative example; a more thorough investigation such as in [15], [21], [31] is needed before definitive statements can be made.

IV. Conclusion

For the purpose of identifying and isolating transient artifacts in time series, this work distinguishes two types of artifact signals: one type that consists of infrequent transient pulses and otherwise adheres to a baseline value of zero, and a second type which consists of abrupt shifts of the baseline (i.e., additive step discontinuities). In this work, the observed time series is modeled as the sum of an artifact signal of each type, a low-pass signal (e.g., a background trend), and a white Gaussian stochastic process. To jointly estimate the components of the signal model, we formulate an optimization problem and develop a rapidly converging, computationally efficient iterative algorithm, denoted TARA (for 'transient artifact reduction algorithm'). We also address the selection of the regularization and non-convexity parameters. The effectiveness of the approach is illustrated using both simulated and NIRS time-series data. The presented approach is also useful for denoising piecewise smooth signals (Figs. 1 and 6).

Some artifacts arising in biomedical time series, such as oscillatory transients, are not of the form considered here. In this case, the approach of TARA should be modified to account for the artifact characteristics, or other methods should be used, e.g., [15], [21]. Additionally, if the available data is corrupted by high amplitude oscillatory noise or interference, then the Gaussian

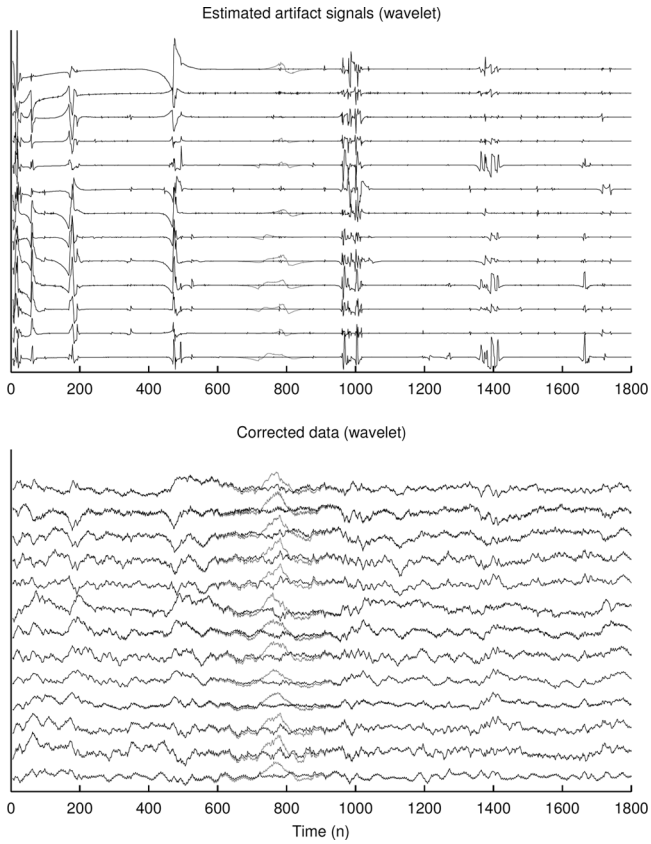


Fig. 11. Wavelet transient artifact reduction (WATAR) of multichannel data.

noise assumption is not satisfied, and the approach should be modified accordingly.

To exploit TARA for two-dimensional (2-D) data, one issue is that the sparse filter matrices ($\mathbf{A}$ and $\mathbf{B}$) may not be exactly banded, depending on how the 2-D case is formulated. In that case, fast solvers for banded systems may not be so readily utilized. Hence, the high-computational efficiency of TARA might not completely carry over to the 2-D case. As a result, an extension of TARA to the 2-D case remains of interest as future work.

ACKNOWLEDGMENT

The authors gratefully acknowledge Justin R. Estepp (Air Force Research Laboratory, Wright-Patterson AFB, OH, USA) and Sean M. Weston (Oak Ridge Institute for Science and Education, TN, USA) for experimental NIRS data. The authors also thank the reviewers for their constructive comments.

REFERENCES

[1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian approach to linear inverse problems with compound regularization," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 4169–4172.

[2] M. T. Akhtar, W. Mitsuhashi, and C. J. James, "Employing spatially constrained ICA and wavelet denoising, for automatic removal of artifacts from multichannel EEG data," *Signal Process.*, vol. 92, no. 2, pp. 401–416, 2012.

[3] R. Al abdi, H. L. Graber, Y. Xu, and R. L. Barbour, "Optomechanical imaging system for breast cancer detection," *J. Opt. Soc. Amer. A*, vol. 28, no. 12, pp. 2473–2493, Dec. 2011.

[4] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Found. Trends Mach. Learn.*, vol. 4, no. 1, pp. 1–106, 2012.

[5] D. P. Bertsekas, *Convex Optimization Theory*. Belmont, MA, USA: Athena Scientific, 2009.

[6] J. M. Bioucas-Dias and M. A. T. Figueiredo, "An iterative algorithm for linear inverse problems with compound regularizers," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 685–688.

[7] A. Blake and A. Zisserman, *Visual Reconstruction*. Cambridge, MA, USA: MIT Press, 1987.

[8] L. M. Briceño-Arias and P. L. Combettes, "A $\mathrm{monotone}+\mathrm{skew}$ splitting model for composite monotone inclusions in duality," *SIAM J. Optim.*, vol. 21, no. 4, pp. 1230–1250, Oct. 2011.

[9] S. Brigadoi, L. Ceccherini, S. Cutini, F. Scarpa, P. Scatturin, J. Selb, L. Gagnon, D. A. Boas, and R. J. Cooper, "Motion artifacts in functional near-infrared spectroscopy: A comparison of motion correction techniques applied to real cognitive data," *NeuroImage*, vol. 85, pp. 181–191, 2014.

[10] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging. J. math," *J. Math. Vis.*, vol. 40, no. 1, pp. 120–145, 2011.

[11] R. R. Coifman and D. L. Donoho, "Translation-invariant de-noising," in *Wavelet and Statistics*, A. Antoniadis and G. Oppenheim, Eds. New York, NY, USA: Springer-Verlag, 1995, pp. 125–150.

[12] P. L. Combettes and J.-C. Pesquet, "Primal-dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators," *Set-Valued Variation. Anal.*, vol. 20, no. 2, pp. 307–330, Jun. 2012.

[13] L. Condat, "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," *J. Optim. Theory Appl.*, vol. 158, no. 2, pp. 460–479, 2013.

[14] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *J. Amer. Statist. Assoc.*, vol. 96, no. 456, pp. 1348–1360, 2001.

[15] T. Fekete, D. Rubin, J. M. Carlson, and L. R. Mujica-Parodi, "The NIRS analysis package: Noise reduction and statistical inference," *PLoS ONE*, vol. 6, no. 9, p. E24322, 2011.

[16] M. Figueiredo, J. Bioucas-Dias, and R. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2980–2991, Dec. 2007.

[17] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani, "Pathwise coordinate optimization," *Ann. Appl. Statist.*, vol. 1, no. 2, pp. 302–332, 2007.

[18] J.-J. Fuchs, "Convergence of a sparse representations algorithm applicable to real or complex data," *IEEE. J. Sel. Top. Signal Process.*, vol. 1, no. 4, pp. 598–605, Dec. 2007.

[19] H. Gao, "Wavelet shrinkage denoising using the nonnegative garrote," *J. Comput. Graph. Statist.*, vol. 7, pp. 469–488, 1998.

[20] T. J. Huppert, S. G. Diamond, M. A. Franceschini, and D. A. Boas, "HomER: a review of time-series analysis methods for near-infrared spectroscopy of the brain," *Appl. Opt.*, vol. 48, no. 10, pp. D280–D298, Apr. 2009.

[21] M. K. Islam, A. Rastegarnia, A. T. Nguyen, and Z. Yang, "Artifact characterization and removal for in vivo neural recording," *J. Neurosci. Methods*, vol. 226, pp. 110–123, 2014.

[22] N. Mammone, F. L. Foresta, and F. C. Morabito, "Automatic artifact rejection from multichannel scalp EEG by wavelet ICA," *IEEE J. Sensors*, vol. 12, no. 3, pp. 533–542, Mar. 2012.

[23] J. Mehnert, M. Brunetti, J. Steinbrink, M. Niedeggen, and C. Dohle, "Effect of a mirror-like illusion on activation in the precuneus assessed with functional near-infrared spectroscopy," *J. Biomed. Opt.*, vol. 18, no. 066001, 2013.

[24] B. Molavi and G. A. Dumont, "Wavelet-based motion artifact removal for functional near-infrared spectroscopy," *Physio. Meas.*, vol. 33, no. 2, p. 259, 2012.

[25] M. K. I. Molla, T. Tanaka, and T. M. Rutkowski, "Multivariate EMD based approach to EOG artifacts separation from EEG," in *In Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Mar. 2012, pp. 653–656.

[26] M. Nikolova, "Estimation of binary images by minimizing convex criteria," in *Proc. IEEE Int. Conf. Image Process.*, 1998, vol. 2, pp. 108–112.

[27] M. Nikolova, M. K. Ng, and C.-P. Tam, "Fast nonconvex nonsmooth minimization methods for image restoration and reconstruction," *IEEE trans. Image Process.*, vol. 19, no. 12, pp. 3073–3088, Dec. 2010.

[28] J.-C. Pesquet and N. Pustelnik, "A parallel inertial proximal optimization method," *Pacific J. Optim.*, vol. 8, no. 2, pp. 273–305, Apr. 2012.

[29] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1992.

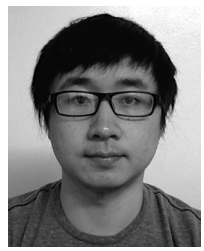
[30] H. Raguet, J. Fadili, and G. Peyré, "A generalized forward-backward splitting," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1199–1226, 2013.
[31] F. C. Robertson, T. S. Douglas, and E. M. Meintjes, "Motion artifact removal for functional near infrared spectroscopy: A comparison of methods," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 6, pp. 1377–1387, Jun. 2010.
[32] H. Sato, N. Tanaka, M. Uchida, Y. Hirabayashi, M. Kanai, T. Ashida, I. Konishi, and A. Maki, "Wavelet analysis for detecting body-movement artifacts in optical topography signals," *NeuroImage*, vol. 33, no. 2, pp. 580–587, 2006.
[33] I. W. Selesnick and I. Bayram, "Sparse signal estimation by maximally sparse convex optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1078–1092, Mar. 2014.
[34] I. W. Selesnick, H. L. Graber, D. S. Pfeil, and R. L. Barbour, "Simultaneous low-pass filtering and total variation denoising," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1109–1124, Mar. 2014.
[35] J.-L. Starck, M. Elad, and D. Donoho, "Redundant multiscale transforms and their application for morphological component analysis," *Adv. Imag. Electron Phys.*, vol. 132, pp. 287–348, 2004.
[36] J.-L. Starck, F. Murtagh, and J. M. Fadili, *Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
[37] K. T. Sweeney, S. F. McLoone, and T. E. Ward, "The use of ensemble empirical mode decomposition with canonical correlation analysis as a novel artifact removal technique," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 1, pp. 97–105, Jan. 2013.
[38] H. Zeng, A. Song, R. Yan, and H. Qin, "EOG artifact correction from EEG recording using stationary subspace analysis and empirical mode decomposition," *Sensors*, vol. 13, no. 11, pp. 14839–14859, 2013.

Ivan W. Selesnick (S'91–M'98–SM'08) received the B.S., M.E.E., and Ph.D. degrees in Electrical Engineering in 1990, 1991, and 1996 from Rice University, Houston, TX. In 1997, he was a visiting professor at the University of Erlangen-Nurnberg, Germany. He then joined the Department of Electrical and Computer Engineering, NYU Polytechnic School of Engineering, New York (then Polytechnic University), where he is associate Professor.

His current research interests are in the area of digital signal and image processing, wavelet-based signal processing, sparsity techniques, and biomedical signal processing. He has been an associate editor for the IEEE Transactions. ON IMAGE PROCESSING, the IEEE SIGNAL PROCESSING LETTERS, and the IEEE TRANSACTIONS ON SIGNAL PROCESSING.

Harry L. Graber (M'95) received the A.B. degree in chemistry from Washington University, St. Louis, MO, in 1983, and the Ph.D. degree in physiology and biophysics from SUNY Downstate Medical Center, Brooklyn, NY, in 1998. He subsequently became a Research Associate (1998–2001) and then a Research Assistant Professor (2001–2014) at SUNY Downstate Medical Center. Since 2001 he also has been the Senior Applications Specialist for NIRx Medical Technologies.

His research interests include diffuse optical imaging algorithms, and application of feature-extraction and time-series analysis methods for interpretation of biological signals.



Yin Ding received the B.S. degree from Nanjing University of Posts and Telecommunications, China in 2008, and the M.S. degree in Electrical Engineering from New York University Polytechnic School of Engineering, NY, in 2011. From 2011 to 2012 he was a research engineer at Li Creative Technologies Inc., NJ. He is currently pursuing the Ph.D. degree in Electrical Engineering at the NYU Polytechnic School of Engineering.

His research interests include digital signal and image processing, biomedical signal processing, audio signal processing, and biometrics. His recent research is on the development of sparse signal processing algorithms for biomedical signal analysis.

Tong Zhang received the B.E. degree from Beihang University, Beijing, China in 2011, and the M.S. degree in Electrical Engineering from the New York University Polytechnic School of Engineering in 2014. His research interests include digital signal and image processing, machine learning, and computer vision. He is a member of Eta Kappa Nu.

Randall L. Barbour received the Ph.D. degree in biochemistry from Syracuse University, Syracuse, NY, in 1981. This was followed by a postdoctoral fellowship in Laboratory Medicine at SUNY at Buffalo.

He is currently Professor of Pathology at SUNY Downstate Medical Center, and Research Professor of Electrical Engineering at Polytechnic University, Brooklyn, NY. He is an originator of the field of diffuse optical tomography and is co-founder of NIRx Medical Technologies, LLC. He has an extensive background in a broad range of medical and scientific and technical fields.