Periodic Artifact Removal With Applications to Deep Brain Stimulation

Paula Chen, Taewoo Kim, Evan Dastin-van Rijn, Nicole R. Provenza, Sameer A. Sheth, Wayne K. Goodman, David A. Borton, and Jérôme Darbon

Abstract—Deep brain stimulation (DBS) therapies have shown clinical success in the treatment of a number of neurological illnesses, including obsessive-compulsive disorder, epilepsy, and Parkinson's disease. An emerging strategy for increasing the efficacy of DBS therapies is to develop closed-loop, adaptive DBS systems that can sense biomarkers associated with particular symptoms and in response, adjust DBS parameters in real-time. The development of such systems requires extensive analysis of the underlying neural signals while DBS is on, so that candidate biomarkers can be identified and the effects of varying the DBS parameters can be better understood. However, DBS creates high amplitude, high frequency stimulation artifacts that prevent the underlying neural signals and thus the biological mechanisms underlying DBS from being analyzed. Additionally, DBS devices often require low sampling rates, which alias the artifact frequency, and rely on wireless data transmission methods that can create signal recordings with missing data of unknown length. Thus, traditional artifact removal methods cannot be applied to this setting. We present a novel periodic artifact removal algorithm for DBS applications that can accurately remove stimulation artifacts in the presence of missing data and in some cases where the stimulation frequency exceeds the Nyquist frequency. The numerical examples suggest that, if implemented on dedicated hardware, this algorithm has the potential to be used in embedded closed-loop DBS therapies to remove DBS stimulation artifacts and hence, to aid in the discovery of candidate biomarkers in real-time. Code for our proposed algorithm is publicly available on Github.

Index Terms—Deep brain stimulation, optimization, online periodic artifact removal algorithms.

Paula Chen, Taewoo Kim, Matthew T. Harrison, and Jérôme Darbon are with the Division of Applied Mathematics, Brown University, Providence, RI 02912 USA (e-mail: paula_chen@brown.edu; taewoo_kim@brown.edu; matthew_harrison@brown.edu; jerome_darbon@brown.edu).

Evan Dastin-van Rijn is with the Department of Biomedical Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: dasti006@umn.edu).

Nicole R. Provenza and Sameer A. Sheth are with the Department of Neurosurgery, Baylor College of Medicine, Houston, TX 77030 USA (e-mail: nicole.provenza@bcm.edu; sameer.sheth@bcm.edu).

Wayne K. Goodman is with the Menninger Department of Psychiatry and Behavioral Sciences, Baylor College of Medicine, Houston, TX 77030 USA (e-mail: wayne.goodman@bcm.edu).

David A. Borton is with the School of Engineering, Carney Institute for Brain Science, Brown University, Providence, RI 02912 USA, and also with the Center for Neurorestoration and Neurotechnology, Rehabilitation Research and Development Service, Department of Veterans Affairs, Providence, RI 02908 USA (e-mail: david_borton@brown.edu).

Digital Object Identifier 10.1109/TNSRE.2022.3205453

I. Introduction

DEEP brain stimulation (DBS) is a treatment that regulates neural activity by sending electrical impulses through electrodes implanted in the brain. Currently, DBS is an existing therapy for obsessive-compulsive disorder (OCD), epilepsy, and Parkinson's disease, among many other neurological illnesses, but strategies for maximizing the therapeutic effects of DBS while minimizing its side effects remain an active area of research [1]. To that end, one emerging area of interest is the discovery of biomarkers (i.e., neural signals correlated with symptoms) (see, for instance, [2], [3]). The discovery of biomarkers might allow for the development of closed-loop, adaptive DBS systems that are able to sense a biomarker and in response, adjust DBS parameters (e.g., stimulation frequency, amplitude, etc.) in real-time to effectively target symptoms as they arise, while minimizing adverse effects. In order to discover a biomarker and better understand the neurological responses to varying the DBS parameters, the underlying neural signals must be able to be studied while DBS is on. However, DBS creates high amplitude, high frequency stimulation artifacts that contaminate the electrical recordings of the brain. As a result, periodic artifact removal is essential for recovering the underlying neural signal and thus, for understanding the biological mechanisms underlying the qualitative successes of DBS. Furthermore, while biomarker discovery (and thus the corresponding artifact removal) may be done offline, closed-loop, adaptive DBS therapies necessitate the removal of these stimulation artifacts in real-time.

In DBS applications, the artifact removal problem is further complicated by the following: (1) while the stimulation frequency can be set by the DBS device, the device setting provides an estimate of the stimulation frequency that is not accurate enough for successful artifact removal; (2) power constraints of DBS devices often require low sampling rates

(e.g., 200-250 Hz), which in turn alias the stimulation frequency near the frequencies of underlying signals of interest; and (3) DBS recordings are often broken into many time segments with unknown phases, e.g., due to modulation of device settings or missing data of unknown length, the latter of which is a common limitation of the wireless data transmission methods used by some DBS devices [4]. While many methods for removing DBS stimulation artifacts exist (see, for instance, [5], [6], [7], [8], [9]), no single method, to the authors' knowledge, addresses all of the aforementioned complications, which are often found in real data. We propose a new method for periodic artifact removal that addresses all of these complications.

The general formulation of our problem of interest is as follows. Given $n + 1$ segments of data, we model the i-th segment of the observed signal S_i as:

$$S_i(t) = A\left(t + \frac{\delta_i^*}{\xi^*}\right) + B_i(t) + \eta_i(t), \quad i = 0, \ldots, n, \quad (1)$$

where δ_i^* is the (unknown) phase shift between the 0-th and i-th segments, A is a periodic artifact with (unknown) period $\frac{1}{\xi^*}$, B_i is the neural signal in segment i, and η_i is the noise in segment i.

Our goal is to estimate and remove A from each S_i. As a result, the underlying signals that we seek to recover are $B_i + \eta_i, i = 0, \ldots, n$. In practice, S_i is sampled at some collection of discrete times T_j. Then, the following loss function can be used to reconstruct and remove the artifact:

$$\mathcal{L}(\xi, \delta_i, \boldsymbol{\theta}) = \sum_{i=0}^{n} \sum_{t \in T_i} (S_i(t) - a(t|\xi, \delta_i, \boldsymbol{\theta}))^2, \quad (2)$$

where $a(\cdot|\xi, \delta, \boldsymbol{\theta}) \approx A(\cdot + \frac{\delta}{\xi^*})$ is a model for the artifact that depends on the artifact frequency ξ (or equivalently the artifact period $1/\xi$) and a set (possibly empty) of other parameters $\boldsymbol{\theta}$. Equation (2) generalizes the loss function in [6] and [10] to allow for (multiple) unknown phases. In this paper, we consider the following parametric model for the artifact:

$$a(t|\xi, \delta, \alpha_0, \alpha_k, \beta_k, K)$$
$$= \alpha_0 + \sum_{k=1}^{K} \alpha_k \cos(2\pi k(\xi t + \delta)) + \beta_k \sin(2\pi k(\xi t + \delta)),$$
$$(3)$$

where, in practice, the number of harmonics K must be chosen to be finite. Using this parametric model allows for fast computations in our proposed algorithms. Reference [6] uses the same model (3) and [10] experiments with a variety of models for A, including (3), but both without phase shifts. Meanwhile, [4] uses the same framework described above to estimate a single unknown phase from two segments ($n = 1$), but does not consider simultaneous inference of the model parameters. Instead, [4] first estimates the artifact and its frequency from a single segment using [6] and then selects among a small finite collection of candidate phases corresponding to an unknown number of missing samples. This process is repeated sequentially to handle multiple phases ($n > 1$). Our method is not sequential; rather, it is based on optimization of (2).
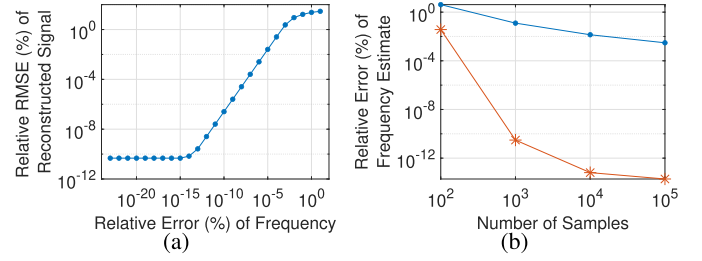
Fig. 1. (a) Relative error of frequency vs. relative RMSE of reconstructed signal using harmonic regression. Each point represents the average over 40 trials. For each trial, we use harmonic regression to fit the observed signal with a sinusoid with 5 harmonics and a given frequency estimate. (b) Number of samples vs. relative error of the frequency estimated using a DFT-based method (—·) or Algorithm 1 (-*). Each point represents the average over 10 trials. In both (a) and (b), each trial uses an observed signal sampled at 1000 Hz that consists only of an artifact of the form (3) with no phase shifts, $K = 5$, and a random true fundamental frequency $\xi \in (25, 200)$.

Using the above model (3), a natural way to remove the artifact with the above framework is harmonic regression [11, Ch. 2], which for fixed (ξ, δ_i) is equivalent to minimizing (2) with respect to $\boldsymbol{\theta} = \{\alpha_0, \alpha_k, \beta_k, k = 1, \ldots, K\}$. However, in Fig. 1a, we see that harmonic regression requires accurate estimates of the frequency ξ. For example, using harmonic regression, 0.001% relative error in the frequency estimate, on average, corresponds to a relative root mean squared error (RMSE) in the reconstructed signal $S_i - a(\cdot|\xi, \delta_i, \boldsymbol{\theta})$ of almost 10%. Relative RMSE is defined as follows:

$$\text{relative RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(f(t_i) - \hat{f}(t_i))^2}{\sum_{i=1}^{N} f(t_i)^2}}, \quad (4)$$

where f is the true signal of interest (e.g., $f = B + \eta$), $\hat{f}$ is the estimated signal (e.g., $\hat{f} = S - a(\cdot|\xi, \delta_i, \boldsymbol{\theta})$), and t_i are the sample times. Thus, even somewhat small errors in the frequency can lead to large reconstruction errors.

Many periodic artifact removal methods rely on the discrete Fourier transform (DFT) to estimate the frequency. However, DFT-based methods generally cannot handle phase shifts and, even segmentwise, are limited in their accuracy for frequency estimation. For example, consider the following simple DFT-based frequency detection method, where the artifact frequency is estimated as the frequency that maximizes the energy (magnitude squared of the DFT) of the observed signal. In Fig. 1b, we observe that even in the case where the observed signal only consists of the periodic artifact and the sampling rate is relatively high ($f_s = 1000$ Hz), this DFT-based method is unable to accurately estimate the artifact frequency when the frequency does not lie on the DFT grid $\{i\frac{N}{f_s} : i = 0, \ldots, \frac{N}{2} - 1\}$, which would be the case for most real-life applications. Specifically, we see that even with 10^5 samples, this method achieves, on average, greater than 0.001% relative error in the frequency estimate, which, as discussed above, is insufficiently accurate for harmonic regression. In contrast, Fig. 1b shows that our proposed Algorithm 1 (detailed in

Section II) achieves much more accurate frequency estimates, e.g., close to machine precision with 10^5 samples.

The main contribution of this paper is an iterative periodic artifact removal algorithm that can accurately estimate the artifact frequency in the presence of unknown phase shifts and that involves few (i.e., one) tuning parameters. We also provide an initialization algorithm that experimentally provides robust initial estimates of the frequency and phase shifts for our artifact removal algorithm. We demonstrate the performance of our algorithms on the following cases: a simulated artifact with no underlying signal, a simulated artifact with a chirp, an aliased simulated artifact with a simulated underlying signal and missing data, and a human local field potential (LFP) recording. Both algorithms use relatively simple computations and demonstrate relatively short runtimes (in comparison to the length in time of the data processed), which show their potential to be used in embedded closed-loop DBS therapies in real-time.

The remainder of this paper is organized as follows. In Section II, we describe the details of our method. In Section II-A, we describe our artifact removal method, which fits the observed signal using harmonic regression while jointly estimating the frequency and phase shifts via a least squares regression. In Section II-B, we provide an initialization algorithm for the method described in Section II-A. In Section III, we provide some numerical examples, which demonstrate some of the capabilities of our proposed method. Finally, in Section IV, we provide some concluding remarks.

II. PROPOSED ALGORITHM

In this section, we describe our algorithm. First, we describe our artifact removal algorithm (Algorithm 1), which removes the artifact by estimating the frequency and phase shifts while jointly fitting the artifact using harmonic regression. Second, we describe an initialization algorithm (Algorithm 2), which finds an initial estimate of the frequency and phase shifts via an energy maximization method. The code for both algorithms and implementation details are available at https://github.com/pxchen95/artifact_removal.

Our model is as follows. We observe $n + 1$ segments of data $S_i, i = 0, \ldots, n$, where each S_i is defined by (1) and δ_i^* is the (unknown) true phase shift between the 0-th and i-th segments (by convention, $\delta_0^* = 0$). We assume that the artifact can be represented by (3), where the true fundamental frequency ξ^*, the true amplitudes $\alpha_0^*, \alpha_k^*, \beta_k^*$, and the true number of harmonics K are unknown. A key assumption for our method is that the energy of the artifact at its fundamental frequency and harmonics is sufficiently larger than the energy of the brain at those frequencies (or at the frequencies that they are aliased to). In practice, we do not have access to the continuous signals and only access each S_i at some discrete sample times. While these sample times may be irregularly spaced, we only consider uniformly spaced sample times as, in practice, most DBS devices use uniformly spaced sampling based on relatively stable system clocks; i.e., we assume that the i-th segment is sampled at the times $\{\frac{j}{f_s} : j = 0, \ldots, N_i - 1\}$, where f_s is the sampling rate.

Algorithm 1 Artifact Removal Algorithm

Inputs :
  $\hat{\xi}$: frequency initialization (e.g., the frequency estimate from Algorithm 2),
  $\hat{\delta}_i$: phase shift initializations (e.g., the phase shift estimates from Algorithm 2),
  $\hat{K}$: number of harmonics to fit,
  $\left\{ S_i \left(\frac{j}{f_s} \right) : j = 0, \ldots, N_i - 1 \right\}$: $n + 1$ segments of observed data,
  maxIter: maximum number of iterations,
  tol_1, tol_2: tolerances for stopping criteria.

Outputs:
  $\hat{\hat{\xi}}$: new frequency estimate,
  $\hat{\hat{\delta}}_i$: new phase shift estimates,
  $\left\{ \hat{R}_i \left(\frac{j}{f_s} \right) : j = 0, \ldots, N_i - 1 \right\}$: $n + 1$ segments of recovered signal.

1 Initialize $\boldsymbol{x}^0 = [x_0^0, \ldots, x_n^0]'$, where $x_0^0 = \hat{\xi}$ and
  $x_i^0 = \hat{\delta}_i, i = 1, \ldots, n$.
2 **for** $j = 0, \ldots, maxIter$ **do**
3   Compute direction of descent
      $d_j = -(\nabla^2 g(\boldsymbol{x}^j))^{-1} \nabla g(\boldsymbol{x}^j)$.
4   Update iterate by $\boldsymbol{x}^{j+1} = \boldsymbol{x}^j + d_j$.
5   **if** $\|\nabla g(\boldsymbol{x}^{j+1})\|_2 < tol_1$ *or* $\|\boldsymbol{x}^{j+1} - \boldsymbol{x}^j\|_2 < tol_2$ **then**
6   Set $\boldsymbol{x}^* = \boldsymbol{x}^j$.
7   break
8   **end**
9 **end**
10 Output the new estimated frequency by $\hat{\hat{\xi}} = x_0^*$, the new estimated phase shifts by $\hat{\hat{\delta}}_i = x_i^*$, and the recovered signal by

$$\hat{R}_i \left(\frac{j}{f_s} \right) = R_i \left(\frac{j}{f_s}, \hat{\alpha}_0, \hat{\alpha}_k, \hat{\beta}_k \right), \qquad (5)$$

where the right-hand side is the function defined in (8) and $\hat{\alpha}_0, \hat{\alpha}_k, \hat{\beta}_k$ are the minimizing $\alpha_0, \alpha_k, \beta_k$ in the definition of $g(\boldsymbol{x}^*)$.

A. Artifact Removal Algorithm

Based on the assumed form of the artifact (3), harmonic regression [11, Ch. 2] is a natural choice for removing the artifact given an estimate of the frequency. However, as shown in Fig. 1a, small perturbations in the frequency estimate can result in large errors in the signal reconstructed by harmonic regression. To mitigate this issue, our method fits the observed signal using harmonic regression while jointly estimating the frequency and phase shifts. Specifically, we remove the artifact by solving the following optimization problem:

$$\min_{\omega, \delta_i, i=1,\ldots,n} g(\omega, \delta_1, \ldots, \delta_n), \qquad (6)$$

where the objective function g is defined by:

$$g(\omega, \delta_1, \ldots, \delta_n)$$
$$= \min_{\alpha_0, \alpha_k, \beta_k} \sum_{i=0}^{n} \sum_{j=0}^{N_i-1} \left(R_i \left(\frac{j}{f_s}, \alpha_0, \alpha_k, \beta_k \right) \right)^2, \qquad (7)$$

and we denote the i-th segment of the recovered signal by:

$$R_i(t, \alpha_0, \alpha_k, \beta_k) = S_i(t) - a(t|\omega, \delta_i, \alpha_0, \alpha_k, \beta_k, \hat{K}), \quad (8)$$

where $a(t|\omega, \delta_i, \alpha_0, \alpha_k, \beta_k, \hat{K})$ is defined by (3). For each fixed $(\omega, \delta_1, \ldots, \delta_n)$, $g(\omega, \delta_1, \ldots, \delta_n)$ is the result of using harmonic regression to remove a periodic artifact of the form (3) with $K = \hat{K}$ harmonics, fundamental frequency $\xi^* = \omega$, and phase shifts δ_i from the 0-th observed segment S_0. For our algorithms, the number of harmonics to fit $\hat{K}$ is effectively the only parameter that needs to be tuned. We note that minimizing over α_0 in (7) means that the recovered signal will have mean 0. Including this amplitude is optional, and if α_0 is omitted in (8), then the recovered signal may have nonzero mean, but the reconstructed artifact will necessarily have mean 0. We also note that for each fixed $(\omega, \delta_1, \ldots, \delta_n)$, $g(\omega, \delta_1, \ldots, \delta_n)$ can then be computed exactly up to numerical precision errors. Specifically, we set the gradient of the objective function in (7) to zero and solve the resulting linear system using any appropriate numerical linear algebra solver.

In Algorithm 1, we use a Newton's descent method to solve (6). The numerical complexity of Algorithm 1 is dominated by the computation of the descent direction, which requires us to solve an $(n+1) \times (n+1)$ and a $\hat{K} \times \hat{K}$ linear system. Thus, the numerical complexity of the algorithm is $O(n^3 + \hat{K}^3)$. In practice, we expect both n and $\hat{K}$ to be relatively small since the data can always be segmented so as not to consider too many gaps at a time and many DBS devices have built-in low-pass filters that effectively bandlimit the recorded signals. Note that we can apply Newton's descent to this problem since g is twice differentiable, where the differentiability of g follows from [12, Ch. 4.3, Th. 4.13]. Also note that while g is not convex, it is locally convex at the global minimizer. Hence, the initialization for Algorithm 1 is crucial for convergence. In the next section, we introduce an initialization algorithm (Algorithm 2) for Algorithm 1. Based on our numerical experiments, we found that Algorithm 2 typically has relatively short computational runtime and provides sufficiently accurate initializations for Algorithm 1.

B. An Initialization Algorithm

In this section, we present an initialization algorithm for Algorithm 1. Specifically, our algorithm locally maximizes the energy of the observed signal using discrete samples but without being constrained to frequencies on the DFT grid.

Define the Fourier transform for our problem as

$$\mathcal{F}(\omega, \delta_1, \ldots, \delta_n)$$
$$= \int_0^{\frac{N_0}{f_s}} S_0(t) e^{-2\pi i \omega t} dt + \sum_{\ell=1}^{n} \int_0^{\frac{N_\ell}{f_s}} S_\ell(t) e^{-2\pi i (\omega t + \delta_\ell)} dt.$$
$$(9)$$

We interpret (9) as trying to align all of the segments S_ℓ ($\ell = 0, 1, \ldots, n$) in time to be in phase with a single wave $e^{-2\pi i \omega t}$. Next, define the energy as

$$E(\omega, \delta_1, \ldots, \delta_n) = |\mathcal{F}(\omega, \delta_1, \ldots, \delta_n)|^2. \quad (10)$$

Algorithm 2 Initialization Algorithm for Algorithm 1

Inputs :
  ω_0: frequency initialization,
  Δ_ω: width of frequency range to be tested,
  $\left\{ S_\ell\left(\frac{j}{f_s}\right) : j = 0, \ldots, N_\ell - 1 \right\}$: $n+1$ segments of observed data,
  numInit: number of random initializations,
  maxIter: maximum number of iterations,
  tol_1, tol_2: tolerances for stopping criteria.
Outputs:
  $\hat{\xi}$: frequency estimate,
  $\hat{\delta}_i$: phase shift estimates.

1 **for** $i = 1, \ldots, numInit$ **do**
2   Initialize $\boldsymbol{x}^0 = [x_0^0, \ldots, x_n^0]'$, using
     $x_0^0 \sim \text{Unif}(\omega_0 - \Delta_\omega, \omega_0 + \Delta_\omega)$ and
     $x_j^0 \sim \text{Unif}(0, 1)$, $j = 1, \ldots, n$.
3   **for** $j = 1, \ldots, maxIter$ **do**
4   Use modified Cholesky to compute
        $M = \nabla^2 E(\boldsymbol{x}^{j-1}) - \tau I_{(n+1)\times(n+1)}$, such that M is
        negative definite.
5   Compute direction of ascent $d_j = M^{-1} \nabla E(\boldsymbol{x}^{j-1})$.
6   Use backtracking line search to compute stepsize η.
7   Update iterate by $\boldsymbol{x}^j = \boldsymbol{x}^{j-1} - \eta d_j$.
8   **if** $\|\nabla E(\boldsymbol{x}^j)\|_2^2 < tol_1$ *and* $\|\boldsymbol{x}^j - \boldsymbol{x}^{j-1}\|_2^2 < tol_2$
        then
9   Set $\boldsymbol{v}^i = \boldsymbol{x}^j$.
10   break
11   **end**
12   **end**
13 **end**
14 Compute $\boldsymbol{v}^* = \arg\max_{\boldsymbol{v} \in \{\boldsymbol{v}^1, \ldots, \boldsymbol{v}^{numInit}\}} E(\boldsymbol{v})$.
15 Output the estimated frequency by $\hat{\xi} = \boldsymbol{v}_0^*$ and the estimated phase shifts by $\hat{\delta}_i = \boldsymbol{v}_i^*$.


By straightforward calculation, it is easy to verify that E is twice differentiable and nonconcave. We propose to recover the true frequency ξ^* and true phase shifts δ_i^* by solving the following optimization problem:

$$\max_{\omega, \delta_1, \ldots, \delta_n} E(\omega, \delta_1, \ldots, \delta_n). \quad (11)$$

Note that for fixed ω, δ_j, $j \neq i$, the function $\delta_i \mapsto \mathcal{F}(\omega, \delta_1, \ldots, \delta_n)$ (and hence, the function $\delta_i \mapsto E(\omega, \delta_1, \ldots, \delta_n)$) is 1-periodic. Thus, solving (11) can only recover the phase shifts δ_i^* up to a multiple of 1. Moreover, since E is nonconcave, the maximizers of (11) are not unique.

In Algorithm 2, we solve (11) numerically by applying a modified Newton's ascent method (based on modified Cholesky [13]) with backtracking line search and random initialization. Since we only have access to discrete samples (and not the continuous signals), we approximate all of the integrals in the definitions of the energy E (10), its gradient ∇E, and its Hessian $\nabla^2 E$ using trapezoidal rule [14, Ch. 8.1]. The computational effort for each iteration of Algorithm 2 is proportional to $(n+1)^3$. As before, we note that, in practice, the data can always be segmented so that n is relatively
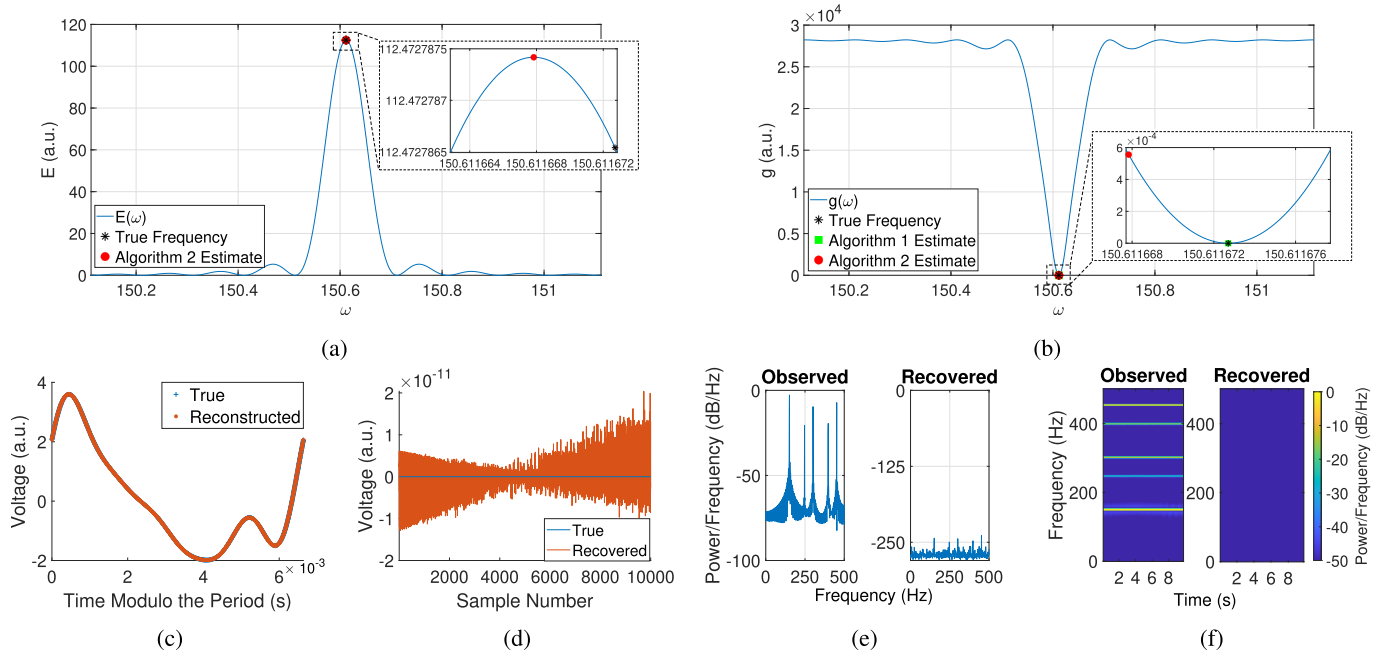
Fig. 2. Removing a well-sampled artifact with no underlying signal. (a) Plot of the energy E, as computed using trapezoidal rule. (b) Plot of the least squares objective function g, as defined by (7). (c) Plot of the true and reconstructed artifacts (with time modulo the true and Algorithm 1 estimate of the period, respectively). (d) Plot of the true and recovered underlying signals. (e) Welch's power spectral density. (f) Spectrogram using short-time Fourier transforms. (a.u. = arbitrary units).

small. Additionally, the implementation of Algorithm 2 can be made more efficient by parallelizing the computations for each random initialization.

III. RESULTS

In this section, we present several numerical examples that demonstrate some of the capabilities of our proposed algorithms. In examples 1-3, we set the artifact A to be of the form (3) with 5 harmonics and true fundamental frequency $\xi^* \approx 150.6117$ Hz. In example 4, we use a human local field potential (LFP) dataset, where the stimulation frequency is set by the DBS device to 150.6 Hz. Examples 3-4 contain missing data. For these examples, the phase shifts δ_i correspond to the time gaps in the data, and we define the segments S_i to be the contiguous segments of observed data. In all of the examples, we remove the reconstructed artifact from S_i to recover the underlying signals $B_i + \eta_i$ using Algorithm 1 with $\hat{K} = 5$, $maxiter = 1000$, $tol_1 = 1e - 8$, and $tol_2 = 1e - 16$ and initialized using the outputs of Algorithm 2 with $\omega_0 = 150.6$, $\Delta_\omega = 5$, $numInit = 25$, $maxiter = 5000$, and $tol_1 = tol_2 = 1e - 8$. Since Examples 1-3 all use simulated data, we report the corresponding results in arbitrary units (a.u.); we note that, unless stated otherwise, the scaling of the units are consistent between the subfigures of each figure corresponding to these simulated examples. We run all of the numerical examples using a simple MATLAB implementation on an 8th Gen Intel Laptop Core i5-8250U with a 1.60GHz processor.

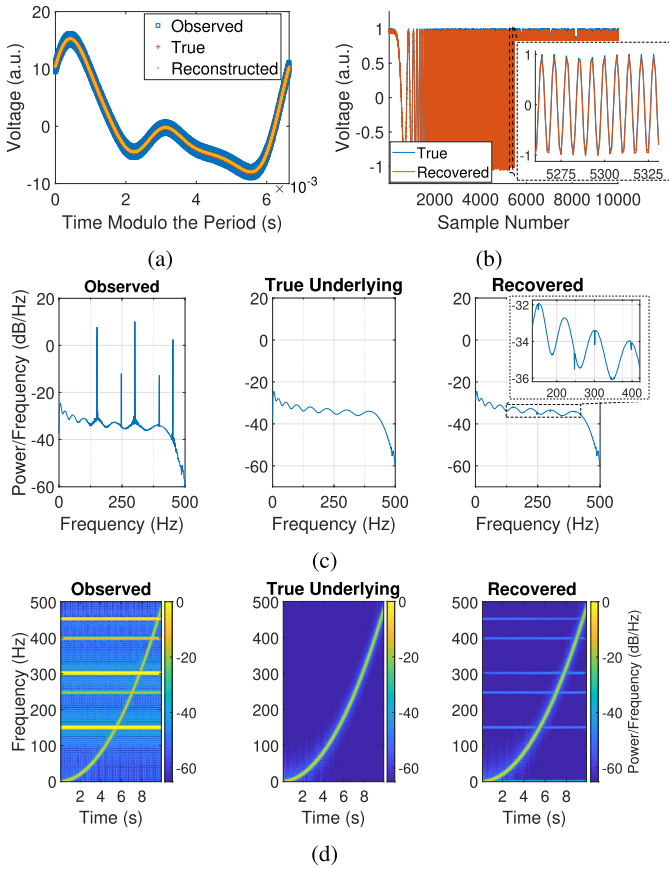### A. Example 1: A Simulated Artifact With No Underlying Signal and No Phase Shifts

In this example, we set the underlying signals $B_0 = \eta_0 = 0$ and use the sampling rate $f_s = 1000$ Hz, 10^4 samples, and no phase shifts ($n = 0$). Since there are no phase shifts in

this example, it is possible to initialize Algorithm 1 using the output of other existing frequency estimation algorithms. However, we use the initialization from Algorithm 2 to demonstrate our algorithms' ability to perform basic frequency detection and to visualize the key functions being optimized by our algorithms.

Fig. 2a depicts plots of the energy E, which is computed using trapezoidal rule to compute the integrals in (10). In Fig. 2a, we see that E is nonconcave and that Algorithm 2 converges to the numerical maximizer of E, but the approximation errors resulting from the trapezoidal rule cause the numerical maximizer of E to be slightly offset from the true frequency of the artifact. Fig. 2b depicts plots of the least squares objective function g, as defined by (7). In Fig. 2b, we see that g is nonconvex and that Algorithm 1 converges to the true frequency and obtains a much more accurate estimate in comparison to Algorithm 2. Specifically, the frequency estimate resulting from Algorithm 1 has a relative error of $3.7742 \times 10^{-14}\%$, whereas the frequency estimate resulting from Algorithm 2 has a relative error of $3.2337 \times 10^{-6}\%$. In Fig. 2c, we observe that the artifact reconstructed by our algorithms matches the true artifact very well with a relative RMSE of $1.7918 \times 10^{-10}\%$. In Fig. 2d, we see that the signal recovered by our algorithms is approximately 0 with an RMSE of 3.0106×10^{-10}. Finally, in Figs. 2e and 2f, we observe that in the frequency domain, the artifact is no longer detectable in the recovered signal.

B. Example 2: A Simulated Artifact With a Chirp and No Phase Shifts

In this example, we set the underlying signals B_0 to be a chirp with frequencies ranging from 0 Hz to 500 Hz and

Fig. 3.   Removing a well-sampled artifact from a chirp. (a) Plot of the observed signal and reconstructed artifact (with time modulo the period estimated by Algorithm 1) and the true artifact (with time modulo the true period). (b) Plot of the true and recovered underlying signals. (c) Welch's power spectral density. (d) Spectrogram using short-time Fourier transforms. (a.u. = arbitrary units).

$\eta_0 = 0$ and use the sampling rate $f_s = 1000$ Hz, 10^4 samples, and no phase shifts ($n = 0$). In this example, the amplitude of the simulated artifact is approximately 15 times larger than that of the chirp. We use this example to demonstrate the performance of our algorithms in the presence of underlying signals consisting of short snippets with frequencies at or near the frequency of the artifact.

In Fig. 3a, our reconstructed artifact matches the true artifact with a relative RMSE of 0.5837%, where our frequency estimate has relative error 7.7068×10^{-6}%. In Fig. 3b, the recovered underlying signal matches the true underlying signal with a relative RMSE of 5.5508%. In Fig. 3c, we see that most of the power at the artifact frequencies is removed by our algorithms, but some small peaks are present near the frequencies corresponding to the artifact harmonics. Similarly, in Fig. 3d, we observe that the artifact is mostly removed (the difference in the power at the artifact frequencies in the observed signal versus in the recovered signal is about 50 dB), while the power of the chirp remains intact. These results show that our algorithms are able to accurately recover the underlying signal even when it contains frequencies at or near the artifact frequency.

C. Example 3: An Aliased Simulated Artifact With a Simulated Neural Signal, Noise, and Missing Data

In this example, we set the underlying signals B_i to be a simulated neural signal (computed as the sum of short snippets of sinusoidal waves with random frequencies and random lengths and demeaned) and η_i to be independent and identically distributed (iid) Gaussian noise and use the sampling rate $f_s = 250$ Hz. In this example, the amplitude of the simulated artifact is approximately 1.4 times larger than that of the underlying signal $B_i + \eta_i$. To simulate missing data, we create 10 segments of contiguous data containing 250 samples each, where the gaps between segments are of random length. Recall that many real-life applications require low sampling rates to handle the battery constraints of DBS devices. In these cases, the fundamental frequency of the stimulation artifact often exceeds the Nyquist frequency [14] and is aliased to potential frequency ranges of interest. Here, the fundamental frequency of the artifact is aliased to approximately 99.3883 Hz. Additionally, recall that the recording methods used by DBS devices can lead to missing data, where the length of the missing data is unknown and the length of the segments of contiguous data is relatively short. Thus, we test the performance of our algorithms in this worst-case scenario, which includes aliasing, short segments of recorded data, and missing data.

In Fig. 4a, the reconstructed artifact matches the true artifact with a relative RMSE of 5.5521%, while our frequency estimate has relative error 2.3023×10^{-3}%. In Fig. 4b, we see that in the time domain, the recovered underlying signal matches the true underlying signal with a relative RMSE of 11.0553%, but the two signals visually match very well despite the relative RMSE being somewhat high. Note that in Fig. 4b, we overlay each sample of the recovered signal on top of the corresponding sample of the true underlying signal. In other words, the length of the gaps in the recovered signal in this figure do not necessarily correspond to the lengths estimated by Algorithm 1 (recall that Algorithm 1 can only recover phase shifts up to some multiple of 1). In Fig. 4c, we observe that in the frequency domain, the power of the artifact frequencies is no longer detectable in the recovered signal, and the recovered signal and true underlying signal are visually indistinguishable. These results show that our algorithms are able to remove stimulation artifacts even in this worst-case scenario.

D. Example 4: Human LFP Recording With Missing Data

In this example, we consider a real recording of a resting state human LFP signal with DBS from a clinical trial (ClinicalTrials.gov #NCT04281134). The research participant (37y/o female) had a history of long-standing OCD and underwent clinically indicated DBS surgery for treatment of OCD using a Summit RC+S (Medtronic, Minneapolis, MN, USA) device. DBS leads (Model 3778) were intracranially placed bilaterally in the VC/VS. The participant gave fully informed consent according to study sponsor guidelines, and all procedures were approved by the local institutional review board at Baylor College of Medicine (H-40255, H-44941,
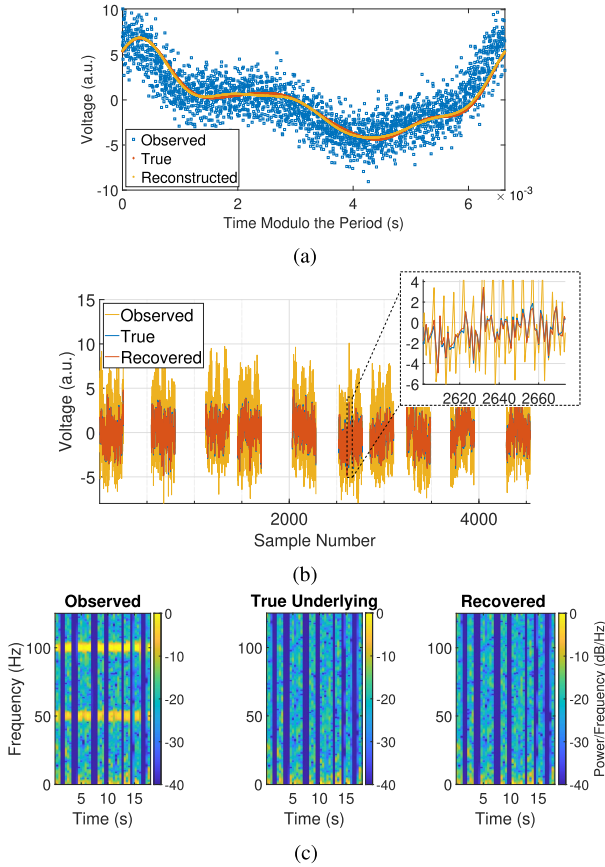
(a)

(b)

(c)

Fig. 4. Removing an aliased artifact from a simulated neural signal with missing data. (a) Plot of the observed signal and reconstructed artifact (with time modulo the period estimated by Algorithm 1) and the true artifact (with time modulo the true period). (b) Plot of the true and recovered underlying signals. (c) Spectrogram using short-time Fourier transforms. (a.u. = arbitrary units).

5/4/2021). LFP was sensed with bipolar contacts around the stimulation contact at a sampling rate of $f_s = 1000$ Hz. The stimulation parameters were set by the device as follows: stimulation frequency of 150.6 Hz, stimulation amplitude of 4.5 mA, and pulse width of 90 μs. We simulated missing data by segmenting the recorded data into 10 segments of contiguous data containing 250 samples each with gaps between the segments that are of random length.

In this example, Algorithm 1 gives us a frequency estimate of approximately 150.6093. In Fig. 5a, we see that using our estimates of the artifact period and the time gaps, the observed signal generally looks like a noisier version of our reconstructed artifact. In Fig. 5b, we observe that the bands corresponding to the frequencies of the DBS artifact are removed by our algorithm. Note that in this figure, the "No DBS" plot cannot be directly compared with the "Recovered" plot since they each represent signals that were recorded at different times and under different conditions (i.e., with DBS off or on, respectively); however, since we do not have access to the true underlying signal, the former plot provides a baseline for what the plot for the true artifact-free underlying signal may look like. These results demonstrate that our algorithms work on real datasets of interest, even in the presence of many gaps of unknown length.
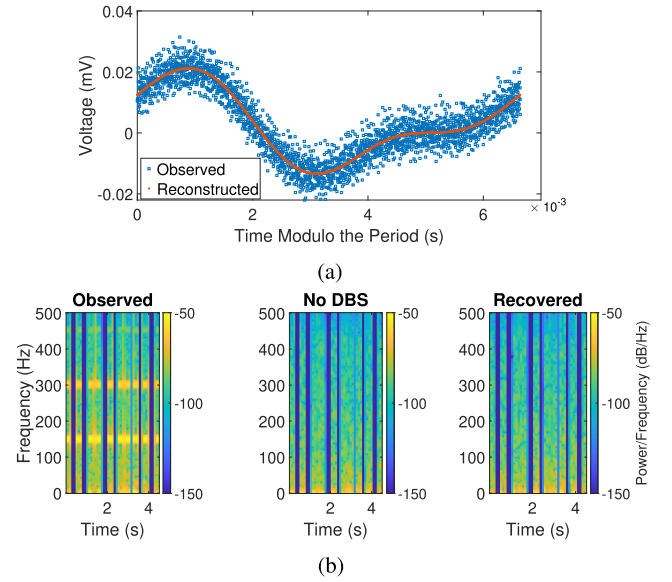

(a)

(b)

Fig. 5. Removing a real DBS artifact from a human LFP recording with missing data. (a) Plot of the observed signal and reconstructed artifact (with time modulo the period estimated by Algorithm 1). (b) Spectrogram using short-time Fourier transforms. Note that the "Recovered" and the "No DBS" plots are not directly comparable as they represent signals that were recorded at different times and under different conditions (i.e., with DBS on or off, respectively).

IV. DISCUSSION AND FUTURE DIRECTIONS

We have proposed a periodic artifact removal algorithm (and a corresponding initialization algorithm) that is able to accurately remove stimulation artifacts in the presence of unknown phase shifts (e.g., missing data of unknown length) and in some cases where the stimulation frequency exceeds the Nyquist frequency. Our algorithms effectively require only one tuning parameter – the number of harmonics $\hat{K}$ to fit. Choosing $\hat{K}$ too small corresponds to underfitting the artifact, while choosing $\hat{K}$ too large corresponds to overfitting the artifact. Thus, the parameter $\hat{K}$ is easily interpretable and could be roughly estimated by counting the number of bands corresponding to the harmonics of the artifact that are visible in a spectogram of the observed signal, although using more sophisticated techniques (e.g., information criteria, such as AIC or BIC [15]) to pick $\hat{K}$ could be of interest for future study. However, we also observed that picking a slightly higher $\hat{K}$ makes little difference in the results. For instance, if we instead use $\hat{K} = 10$ in Example 1 (Section III-A), then all of the resulting errors are identical to the corresponding reported errors up to three decimal places; meanwhile, using $\hat{K} = 10$ in Example 3 (Section III-C) results in relative errors that differ from the corresponding reported values by less than 1%. We also note that, in some sense, $\hat{K}$ can be treated as a fixed parameter since the true number of harmonics of the artifact is fixed for any given set of device settings of any given DBS device. As such, for each fixed set of DBS device settings, $\hat{K}$ only needs to be tuned once offline and the corresponding online artifact removal algorithm becomes effectively parameterless. Additionally, while the numerical examples provided in this paper provide promising results, they are not exhaustive. For example, many real-life DBS

artifacts consist of semi-rectangular waves (i.e., have infinitely many harmonics), which were not considered in this paper and could be explored in a future study.

Since our algorithms are relatively simple computationally, they have the potential for efficient implementation on dedicated hardware, such as digital signal processors (DSPs) or field-programmable gate array (FPGAs). In each of the examples in Section III, each of which corresponds to roughly 4-18 seconds of DBS recording, our simple MATLAB implementation of Algorithm 1 took 0.6-1.5 seconds to run, while Algorithm 2 took 0.4-1.0 seconds to run. However, these runtimes would be greatly shortened by more efficient implementations of our algorithms. As such, our algorithms have the potential to be used in embedded closed-loop DBS therapies to remove stimulation artifacts and thus to discover candidate biomarkers in real-time. Our algorithms may also be applicable to more general settings. For example, we defined our algorithms using uniform sample times, but they could be easily extended to handle nonuniform sample times. Furthermore, (7) could be reformulated to minimize over the mean α_0 segmentwise to handle cases where the mean of the underlying signal varies between segments.

ACKNOWLEDGMENT

REFERENCES

[1] C. O. Oluigbo, A. Salma, and A. R. Rezai, "Deep brain stimulation for neurological disorders," *IEEE Rev. Biomed. Eng.*, vol. 5, pp. 88–99, 2012.

[2] N. R. Provenza *et al.*, "Long-term ecological assessment of intracranial electrophysiology synchronized to behavioral markers in obsessive-compulsive disorder," *Nature Med.*, vol. 27, no. 12, pp. 2154–2164, Dec. 2021.

[3] K. P. Michmizos, P. Frangou, P. Stathis, D. Sakas, and K. S. Nikita, "Beta-band frequency peaks inside the subthalamic nucleus as a biomarker for motor improvement after deep brain stimulation in Parkinson's disease," *IEEE J. Biomed. Health Informat.*, vol. 19, no. 1, pp. 174–180, Jan. 2015.

[4] E. M. Dastin-van Rijn *et al.*, "PELP: Accounting for missing data in neural time series by periodic estimation of lost packets," *Frontiers Hum. Neurosci.*, vol. 16, pp. 1–9, Jul. 2022.

[5] G. Lio, S. Thobois, B. Ballanger, B. Lau, and P. Boulinguez, "Removing deep brain stimulation artifacts from the electroencephalogram: Issues, recommendations and an open-source toolbox," *Clin. Neurophysiol.*, vol. 129, no. 10, pp. 2170–2185, Oct. 2018.

[6] E. M. Dastin-Van Rijn *et al.*, "Uncovering biomarkers during therapeutic neuromodulation with PARRM: Period-based artifact reconstruction and removal method," *Cell Rep. Methods*, vol. 1, no. 2, pp. 1–21, Jun. 2021.

[7] A. Santillán-Guzmán, U. Heute, M. Muthuraman, U. Stephani, and A. Galka, "DBS artifact suppression using a time-frequency domain filter," in *Proc. 35th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2013, pp. 4815–4818.

[8] X. Qian, Y. Chen, Y. Feng, B. Ma, H. Hao, and L. Li, "A method for removal of deep brain stimulation artifact from local field potentials," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 12, pp. 2217–2226, Dec. 2017.

[9] C. Waddell, J. A. Pratt, B. Porr, and S. Ewing, "Deep brain stimulation artifact removal through under-sampling and cubic-spline interpolation," in *Proc. 2nd Int. Congr. Image Signal Process.*, Oct. 2009, pp. 1–5.

[10] P. Hall, "Nonparametric estimation of a periodic function," *Biometrika*, vol. 87, no. 3, pp. 545–557, Sep. 2000.

[11] P. Bloomfield, *Fourier Analysis of Time Series: An Introduction*, 2nd ed. New York, NY, USA: Wiley, 2000.

[12] J. F. Bonnans and A. Shaprio, *Perturbation Analysis of Optimization Problems*, 1st ed. New York, NY, USA: Springer, 2000.

[13] M. Bierlaire, *Optimization: Principles and Algorithms*, 2nd ed. Lausanne, Switzerland: EPFL, 2018.

[14] C. Gasquet and P. Witomski, *Fourier Analysis and Applications: Filtering, Numerical Computation*, 1st ed. New York, NY, USA: Springer, 1999.

[15] S. Konishi and G. Kitagawa, *Information Criteria and Statistical Modeling*, 1st ed. New York, NY, USA: Springer, 2008.