



Published in final edited form as:

*IEEE Trans Signal Process.* 2018 February 15; 66(4): 1037–1050. doi:10.1109/TSP.2017.2775591.

## Bispectrum Inversion with Application to Multireference Alignment

**Tamir Bendory**<sup>†</sup>,

Princeton University in PACM and the Mathematics Department

**Nicolas Boumal**<sup>†</sup>,

Princeton University in PACM and the Mathematics Department

**Chao Ma,**

Princeton University in PACM and the Mathematics Department

**Zhizhen Zhao,** and

University of Illinois at Urbana-Champaign in the Department of Electrical and Computer Engineering

**Amit Singer**

Princeton University in PACM and the Mathematics Department

### Abstract

We consider the problem of estimating a signal from noisy circularly-translated versions of itself, called *multireference alignment* (MRA). One natural approach to MRA could be to estimate the shifts of the observations first, and infer the signal by aligning and averaging the data. In contrast, we consider a method based on estimating the signal directly, using features of the signal that are invariant under translations. Specifically, we estimate the power spectrum and the bispectrum of the signal from the observations. Under mild assumptions, these invariant features contain enough information to infer the signal. In particular, the bispectrum can be used to estimate the Fourier phases. To this end, we propose and analyze a few algorithms. Our main methods consist of non-convex optimization over the smooth manifold of phases. Empirically, in the absence of noise, these non-convex algorithms appear to converge to the target signal with random initialization. The algorithms are also robust to noise. We then suggest three additional methods. These methods are based on frequency marching, semidefinite relaxation and integer programming. The first two methods provably recover the phases exactly in the absence of noise. In the high noise level regime, the invariant features approach for MRA results in stable estimation if the number of measurements scales like the cube of the noise variance, which is the information-theoretic rate. Additionally, it requires only one pass over the data which is important at low signal-to-noise ratio when the number of observations must be large.

---

<sup>†</sup>These two authors contributed equally to the work.

## Index Terms

bispectrum; multireference alignment; phase retrieval; non-convex optimization; optimization on manifolds; semidefinite relaxation; phase synchronization; frequency marching; integer programming; cryo-EM

## I. Introduction

We consider the problem of estimating a discrete signal from multiple noisy and translated (i.e., circularly shifted) versions of itself, called *multireference alignment* (MRA). This problem occurs in a variety of applications in biology [1], [2], [3], [4], radar [5], [6], image registration and super-resolution [7], [8], [9], and has been the subject of recent theoretical analysis [10], [11]. The MRA model reads

$$\xi_j = R_{r_j} x + \varepsilon_j, \quad j = 1, \dots, M, \quad (\text{I.1})$$

where  $\varepsilon_j$  are i.i.d. normal random vectors with variance  $\sigma^2$  and the underlying signal  $x$  is in  $\mathbb{R}^N$  or in  $\mathbb{C}^N$ . Operator  $R_{r_j}$  rotates the signal  $x$  circularly by  $r_j$  locations, namely,  $(R_{r_j} x)[n] = x[n - r_j]$ , where indexing is zero-based and considered modulo  $N$  (throughout the paper). While both  $x$  and the translations  $\{r_j\}$  are unknown, we stress that the goal here is merely to estimate  $x$ . This estimation is possible only up to an arbitrary translation.

A chief motivation for this work arises from the imaging technique called single particle Cryo-Electron Microscopy (Cryo-EM), which allows to visualize molecules at near-atomic resolution [12], [13]. In Cryo-EM, we aim to estimate a three dimensional (3D) object from its two-dimensional (2D) noisy projections, taken at unknown viewing directions [14], [15]. While typically the recovery process involves alignment of multiple observations in a low signal-to-noise ratio (SNR) regime, the underlying goal is merely to estimate the 3D object. In this manner, with the unknown shifts corresponding to the unknown viewing directions, MRA can be understood as a simplified model for Cryo-EM.

Existing approaches for MRA can be classified into two main categories. The first class of methods aims to estimate the set of translations  $\{r_j\}$  first. Given this set, estimating  $x$  can be achieved easily by aligning all observations  $\xi_j$  and then averaging to reduce the noise. The second class, which we favor in this paper, consists of methods which aim to estimate the signal directly, without estimating the shifts.

Considering the first class, one intuitive approach to estimating the translations is to fix a template observation, say  $\xi_1$ , and to estimate the relative translations by cross-correlation. This is called template matching. Specifically,  $r_j$  is estimated as

$$\hat{r}_j = \arg \max_k \Re \left\{ \sum_{n=0}^{N-1} \xi_1[n] \overline{\xi_j[n+k]} \right\}, \quad j = 2, \dots, M,$$

where  $\Re\{z\}$  and  $\bar{z}$  denote the real part and the conjugate of a complex number  $z$ . This approach requires only one pass over the data: for each observation, the best shift can be computed in  $\mathcal{O}(N \log N)$ , and the aligned observations can be averaged online. This results in a total computational cost of  $\mathcal{O}(MN \log N)$ , see Table I. While this approach is simple and efficient, it necessarily fails below a critical SNR—see Figure I.1 for a representative example.

The issue with template matching is that we rely on aligning each observation to only one template: this is error prone at low SNR. Instead, to derive a more robust estimator, one can look for the most suitable alignment among all pairs of observations. The  $M^2$  relative shifts thus computed must then be reconciled into a compatible choice of  $M$  shifts for the individual observations. This is a discrete version of the *angular synchronization* problem, see [16], [17], [18], [19], [20], [21]. The computational complexity of aligning all pairs individually is  $\mathcal{O}(M^2 N \log N)$ , while storing the results uses  $\mathcal{O}(M^2)$  memory.

Alternative algorithms for estimating the translations are based on different SDP relaxations [22], [23], iterative template alignment [24], zero phase representations [5] and neural networks [6]. The statistical limits of alignment tasks were derived for a variety of setups and noise models, see for instance [25], [26], [27], [28]. For example, for a continuous, 2D version of the MRA model, it was shown that the Cramér–Rao lower bound (CRLB) for translation estimation is proportional to the noise variance  $\sigma^2$  [25]; crucially, it does not improve with  $M$ , even if the underlying signal is known. This is motivation to consider the second category of MRA methods, where shifts are not estimated.

Section VI elaborates on expectation maximization (EM) which tries to compute the maximum marginalized likelihood estimator (MMLE) of the signal—marginalization is done over the shifts. This method acknowledges the difficulty of alignment by working not with estimates of the shifts themselves, but rather with estimates of the probability distributions of the shifts. As a result, EM achieves excellent numerical performance in practice. However its computational complexity is high and its performance is not understood in theory.

It has been shown recently that the sample complexity of MRA, under assumption that shifts are distributed uniformly, is proportional to  $\sigma^6$  in the low SNR regime. In other words, the number of measurements  $M$  needs to scale like  $\sigma^6$  to retain a constant estimation error [11].

In this work, we propose a framework which achieves this sample complexity by estimating the sought signal  $x$  directly using features that are invariant under translations. For instance, the mean of  $x$  is invariant under translation and can be estimated easily from the mean of all observations. We further use the power spectrum and the bispectrum of the observations—which are Fourier-transform based invariants—to estimate the magnitudes and phases of the signal’s Fourier transform, respectively.

For any fixed noise level (which may be arbitrarily large), these features can be estimated accurately provided sufficiently many measurements are available. Hence, our approach allows to deal with *any noise level*. Besides achieving the sample complexity, the computational complexity and memory requirements of the methods we describe are

relatively low. Indeed, the only operations whose computational cost grows with  $M$  are computations of averages over the data. These can be performed on-the-fly and are easily parallelizable. We mention that a recent tensor decomposition algorithm also achieves this estimation rate [29].

Given estimators for the mean and power spectrum of  $x$ , estimating the DC component and Fourier magnitudes of  $x$  is straightforward. In this paper, we thus focus on the task of recovering the Fourier phases of  $x$  from an estimator of its bispectrum. We propose two non-convex optimization algorithms on the manifold of phases for this task, which we call bispectrum inversion. We also discuss three additional algorithms which do not require initialization (and hence could be used to initialize others), based on frequency marching, SDP relaxation and integer programming. The first two methods recover the phases exactly in the absence of noise.

Beyond MRA, the bispectrum plays a central role in a variety of signal processing applications. For instance, it is a key tool to separate Gaussian and non-Gaussian processes [30], [31]. It is also used to investigate the cosmic background radiation [32], [33], seismic signal processing [34], image deblurring [35], feature extraction for radar [36], analysis of EEG signals [37], MIMO systems [38] and classification [39] (see also [40], [41], [42], [43], [44] and references therein). In Section III, we review previous works on bispectrum inversion [45], [46], [34]. Reliable algorithms to invert the bispectrum, as studied here, may prove useful in some of these applications.

The paper is organized as follows. Section II discusses the invariant feature approach for MRA. Section III presents the non-convex algorithms on the manifold of phases for bispectrum inversion. Section IV is devoted to additional algorithms that can be used to initialize the non-convex algorithms. Section V analyzes one of the proposed non-convex algorithm. Section VI elaborates on the EM approach for MRA, Section VII shows numerical experiments and Section VIII offers conclusions and perspective.

Throughout the paper we use the following notation. Vectors  $x$  in  $\mathbb{R}^N$  or  $\mathbb{C}^N$  and  $y \in \mathbb{C}^N$  denote the underlying signal and its discrete Fourier transform (DFT), respectively. In the sequel, all indices are understood modulo  $N$ , namely, in the range  $0, \dots, N-1$ . The phase of a complex scalar  $a$ , defined as  $a/|a|$  if  $a \neq 0$  and zero otherwise, is denoted by  $\text{phase}(a)$  or  $\tilde{a}$ . The conjugate-transpose of a vector  $z$  is denoted by  $z^*$ . We use ‘ $\circ$ ’ to denote the Hadamard (entry-wise) product,  $\mathbb{E}$  for expectation,  $\text{Tr}(Z)$  for the trace and  $\|Z\|_F$  for the Frobenius norm of a matrix  $Z$ . We reserve  $T(z)$  for circulant matrices determined by their first row  $z$ , i.e.,  $T(z)[k_1, k_2] = z[k_2 - k_1]$ , and  $\mathcal{H}^N$  for the set of Hermitian matrices of size  $N \times N$ .

## II. Multireference Alignment via Invariant Features

We propose to solve the MRA problem directly using features that are invariant under translations. Unlike pairwise alignment, this approach fuses information from all  $M$  observations together—not just of pairs—and it only aims to recover the signal itself—not the translations. The essence of this idea was discussed as a possible extension in [22, Appendix A]. The invariant features can be understood either as auto-correlation functions

or as their Fourier transform. In this work, we make use of the first three invariants defined as

$$\begin{aligned} c_1 = \mu_x &= \frac{1}{N} \sum_{n=0}^{N-1} x[n], \\ c_2[n_1] &= \frac{1}{N} \sum_{n=0}^{N-1} x[n] \overline{x[n-n_1]}, \\ c_3[n_1, n_2] &= \frac{1}{N} \sum_{n=0}^{N-1} x[n] \overline{x[n-n_1]} x[n+n_2], \end{aligned} \quad (\text{II.1})$$

for  $n_1, n_2 = 0, \dots, N-1$ . It is clear that  $c_1, c_2, c_3$  are invariant under circular shifts of  $x$ . For higher-order invariants based on auto-correlations, see for instance [47].

The first feature is the mean of the signal which is the auto-correlation function of order one (i.e.,  $c_1$  in (II.1)). The distribution of the mean of  $\xi_j$  is then given by  $\mu_{\xi_j} \sim \mathcal{N}\left(\mu_x, \frac{\sigma^2}{N}\right)$  and we can estimate  $\mu_x$  as

$$\hat{\mu}_x = \frac{1}{M} \sum_{j=1}^M \left( \frac{1}{N} \sum_{n=0}^{N-1} \xi_j[n] \right) \sim \mathcal{N}\left(\mu_x, \frac{\sigma^2}{NM}\right). \quad (\text{II.2})$$

Estimating the signal's mean supplies only limited information about the signal itself. Thus, we consider also the auto-correlation function of order two (i.e.,  $c_2$  in (II.1)). Its Fourier transform, the power spectrum, is explicitly defined as

$$P_x[k] = |y[k]|^2,$$

for all  $k$ , where  $y$  is the DFT of  $x$ . An alternative way to understand the invariance of the power spectrum under shifts is through the effect of shifts on the DFT of a signal:

$$\text{DFT}(R_s x)[k] = y[k] \cdot e^{-2\pi i k s / N}. \quad (\text{II.3})$$

Thus, shifts only affect the phases of the DFT, so that  $P_{R_s x} = P_x$  for any shift  $R_s$ . Furthermore, owing to independence of the noise with respect to the signal itself and to the shift,

$$\mathbb{E}\{P_{\xi_j}[k]\} = P_x[k] + N\sigma^2,$$

where the second term is the power spectrum of the noise  $\varepsilon_j$ . Therefore, we estimate the power spectrum of  $x$  as:

$$\hat{P}_x[k] = \frac{1}{M} \sum_{j=1}^M (P_{\xi_j}[k] - N\sigma^2). \quad (\text{II.4})$$

It can be shown that  $\hat{P}_x$  is unbiased and its variance is dominated by  $\frac{\sigma^4}{M}$  for large  $\sigma$ . Hence,  $\hat{P}_x \rightarrow P_x$  as  $M \rightarrow \infty$ . In particular, accurate estimation of the power spectrum requires  $M$  to scale like  $\sigma^4$ . In the sequel, we assume that  $\sigma$  is known.<sup>1</sup>

Recovering a signal from its power spectrum is commonly referred to as *phase retrieval*. This problem received considerable attention in recent years, see for instance [48], [49], [50], [51], [52], [53], [54]. It is well known that almost no one-dimensional signal can be determined uniquely from its power spectrum. Therefore, we use the power spectrum merely to estimate the signal's Fourier magnitudes. As explained next, we use the auto-correlation of third order and its Fourier transform, the bispectrum, to estimate the Fourier phases.

Since phase retrieval is in general ill posed, we use the auto-correlation function of order three (that is,  $c_3$  in (II.1)) through its Fourier transform, the bispectrum, to estimate the Fourier phases of the sought signal. The bispectrum is a function of two frequencies  $k_1, k_2 = 0, \dots, N-1$  and is defined as [55]:

$$B_x[k_1, k_2] = y[k_1] \overline{y[k_2]} y[k_2 - k_1]. \quad (\text{II.5})$$

Note that, if  $y[0] \neq 0$ , the power spectrum is explicitly included in the bispectrum since  $P_x[k] = B_x[k, k]/y[0]$ . The fact that the bispectrum is invariant under shifts can also be deduced from (II.3). Indeed, for any shift  $R_s$ ,

$$\begin{aligned} B_{R_s x}[k_1, k_2] &= \left( y[k_1] e^{-2\pi i k_1 s/N} \right) \left( \overline{y[k_2] e^{2\pi i k_2 s/N}} \right) \cdot \left( y[k_2 - k_1] e^{2\pi i (k_1 - k_2) s/N} \right) \\ &= B_x[k_1, k_2]. \end{aligned}$$

In matrix notation, we express this as

<sup>1</sup>If  $\sigma$  is not known, it can be estimated from the data as

$$\hat{\sigma}^2 = \frac{1}{N} \cdot \text{variance} \left( \sum_{n=0}^{N-1} \xi_j[n] \right)_{j=1, \dots, M}.$$

$$B_x = yy^* \circ T(y), \quad (\text{II.6})$$

where  $T(y)$  is a circulant matrix whose first row is  $y$ , that is,  $T(y)[k_1, k_2] = y[k_2 - k_1]$ . Observe that if  $x$  is real, then  $y[k] = \overline{y[-k]}$  so that  $T(y)$  and  $B_x$  are Hermitian matrices. Simple expectation calculations lead to the conclusion that

$$\mathbb{E} \left\{ B_{\xi_j} \right\} = B_x + \sigma^2 N^2 \mu_x A, \quad (\text{II.7})$$

where  $A = A_{\mathbb{R}}$  or  $A = A_{\mathbb{C}}$  depending on  $x \in \mathbb{R}^N$  or  $x \in \mathbb{C}^N$  and

$$A_{\mathbb{R}} = \begin{bmatrix} 3 & 1 & 1 & 1 & \dots & 1 \\ 1 & 1 & 0 & 0 & \dots & 0 \\ 1 & 0 & 1 & 0 & \dots & 0 \\ 1 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & \dots & 1 \end{bmatrix}, \quad A_{\mathbb{C}} = \begin{bmatrix} 2 & 1 & 1 & 1 & \dots & 1 \\ 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix}.$$

Since the bias term is proportional to  $\mu_x$ , we propose to estimate  $\hat{B}_{x-\mu_x}$  by averaging over  $B_{\xi_j-\mu_x}$  for all  $j$ . This estimator is unbiased and its variance is controlled by  $\frac{\sigma^6}{M}$  for large  $\sigma$ . Therefore,  $M$  is required to scale like  $\sigma^6$  to ensure accurate estimation. In practice,  $\mu_x$  is not known exactly. Thus, we estimate the bispectrum by

$$\hat{B}_{x-\mu_x} = \frac{1}{M} \sum_{j=1}^M B_{\xi_j-\hat{\mu}_x}, \quad (\text{II.8})$$

which is asymptotically unbiased. For finite  $M$  and large  $\sigma$ , bias induced by the approximation  $\hat{\mu}_x \approx \mu_x$  is significantly smaller than the standard deviation of (II.8).

The bispectrum contains information about the Fourier phases of  $x$  because, defining  $\tilde{y}[k] = \text{phase}(y[k])$  and  $\tilde{B}_x[k_1, k_2] = \text{phase}(B_x[k_1, k_2])$  where  $\text{phase}$  extracts the phase of a complex number (and returns 0 if that number is 0), we have

$$\tilde{B}_x[k_1, k_2] = \tilde{y}[k_1] \overline{\tilde{y}[k_2]} \tilde{y}[k_2 - k_1]. \quad (\text{II.9})$$

In matrix notation, the normalized bispectrum takes the form  $\tilde{B}_x = \tilde{y} \tilde{y}^* \circ T(\tilde{y})$ .

Contrary to the power spectrum, the bispectrum is usually invertible. Indeed, in the absence of noise, the bispectrum determines the sought signal uniquely under moderate conditions:

**Proposition II.1**—*For  $N \geq 5$ , let  $x \in \mathbb{C}^N$  be a signal whose DFT  $y$  obeys  $y[k] \neq 0$  for  $k = 1, \dots, K$ , possibly also for  $k = 0$ , and zero otherwise. Up to integer time shifts,  $x$  is determined exactly by its bispectrum provided  $K \geq \frac{N+1}{2}$ .*

*For  $N \geq 5$ , let  $x \in \mathbb{R}^N$  be a real signal whose DFT  $y$  obeys  $y[k] \neq 0$  for  $k = 1, \dots, K$  and  $k = N-1, \dots, N-K$ , possibly also for  $k = 0$ , and zero otherwise. Up to integer time shifts,  $x$  is determined exactly by its bispectrum provided  $\frac{N}{3} \leq K \leq \frac{N-1}{2}$ .*

**Proof:** This is a direct corollary of Lemmas V.1 and B.1.

We stress that the bispectrum estimator in (II.8) is not a bispectrum itself, since the set of bispectra is not a linear space:  $\hat{B}_{x-\mu_x}$  is not invertible as such [42]. Algorithms we propose aim to find a *stable* inverse, in the sense that the recovered signal will have a bispectrum which is close to the estimated bispectrum in  $\mathbb{C}^{N \times N}$ . The following propositions combined argue formally that this can be done in the MRA model. The proofs in Appendix A are constructive.

**Proposition II.2 (Stable bispectrum inversion)**—*There exists an estimator  $\hat{x}$  with the following property. For any signal  $x$  in  $\mathbb{R}^N$  or  $\mathbb{C}^N$  whose DFT is non-vanishing, there exist a precision  $\delta = \delta(x) > 0$  and a sensitivity  $L = L(x) < \infty$  such that if an estimator  $\hat{B}_x$  of  $B_x$  satisfies  $\|\hat{B}_x - B_x\|_F \leq \delta$ , then  $\hat{x} = \hat{x}(\hat{B}_x)$  satisfies  $\min_{r=0, \dots, N-1} \|x - R_r \hat{x}\|_2 \leq L \|\hat{B}_x - B_x\|_F$ .*

**Proposition II.3 (Bispectrum estimation)**—*For any signal  $x$  in  $\mathbb{R}^N$  or  $\mathbb{C}^N$  whose DFT is non-vanishing, for any required precision  $\delta > 0$  and for any probability  $p < 1$ , there exists a constant  $C = C(x, p, \delta) < \infty$  such that, for any noise level  $\sigma > 0$ , if the number of observations  $M$  exceeds  $C \cdot (\sigma^2 + \delta^6)$ , the estimator*

$$\hat{B}_x = \frac{1}{M} \sum_{j=1}^M B_{\xi_j} - \sigma^2 N^2 \hat{\mu}_x^A$$

*satisfies  $\|\hat{B}_x - B_x\|_F \leq \delta$  with probability at least  $p$ .*

We mention that uniqueness in the continuous setup was considered in [56]. The more general setting of bispectrum over compact groups was considered in [57], [58], [59], [60].

The MRA model here assumes i.i.d. Gaussian noise. However, the estimation is performed by averaging in the bispectrum domain, where noise affecting individual entries is correlated. Consequently, one may want to use a more robust estimator, such as the median. Yet, computing the median of complex matrices is computationally expensive, while computing the average can be performed efficiently and on-the-fly, that is, without requiring to store all observations. For Gaussian noise, we have noticed numerically that using the



mean or the median for bispectrum estimation leads to comparable estimation errors (experiments not shown). In other noise models, e.g., with outliers, it might be useful to consider the median or the median of means method, see for instance [61].

### Algorithm 1

Outline of the invariant approach for MRA

---

**Input:** Set of observations  $\xi_j, j=1, \dots, M$  according to (I.1) and noise level  $\sigma$

**Output:**  $\hat{x}$ : estimation of  $x$

**Estimate invariant features:**

- 1 Compute  $\hat{\mu}_x$  according to (II.2)
- 2 Compute  $\hat{P}_x$  according to (II.4)
- 3 Compute  $\hat{B}_{x-\mu_x}$  according to (II.8)

**Estimate the signal's DFT:**

- 1 Estimate  $y[0]$  from  $\hat{\mu}_x$ . For other frequencies:
- 2 Estimate the magnitudes of  $y$  from  $\hat{P}_x$
- 3 Estimate the phases of  $y$  from  $\hat{B}_{x-\mu_x}$  (e.g., Algorithm 2)

**Return:**  $\hat{x}$ : inverse DFT of the estimated  $y$

---

Figure II.1 presents the relative estimation error of the power spectrum and bispectrum as a function of the number of observations  $M$ . For the bispectrum, the relative error is computed as

$$\text{relative error} = \frac{\|B_{x-\mu_x} - \frac{1}{M} \sum_{j=1}^M B_{\xi_j - \hat{\mu}_x}\|_F}{\|B_{x-\mu_x}\|_F},$$

and similarly for the power spectrum. As expected, the slope of all curves is approximately 1/2 in logarithmic scale, implying that the estimation error decreases as  $O(1/\sqrt{M})$ . The invariant features approach for MRA is summarized in Algorithm 1.

Consider the case in which the number of samples  $M$  may be very large whereas the size of the object is fixed, namely  $N \ll M$ . This case is of interest in many applications, such as cryo-EM [14], [15]. In this regime, the invariant features approach has two important advantages over methods that rely on estimating the translations. First, in the invariant features approach, we average over the  $M$  observations (which is computationally cheap), and then apply a more complex algorithm (say, to recover a signal from its bispectrum) whose input size is a function of  $N$  but is independent of  $M$ . Hence, the overall complexity of this approach can be relatively low. Second, the alignment-based method requires storing all  $M$  observations, namely,  $MN$  samples, which is unnecessary in the invariant features approach. There, for each observation, we just need to compute its invariants, to be averaged over all observations: this can be done online (in streaming mode) and in parallel.

### III. Non-convex Algorithms for bispectrum inversion

After estimating the first Fourier coefficient  $y[0]$  as  $N\hat{\mu}_x$ , our approach for MRA by invariant features consists of two parts. We use the power spectrum to estimate the signal's Fourier magnitudes and the bispectrum for the phases. The first part is straightforward:  $|y[k]|$  can be estimated as  $\sqrt{\hat{P}_x[k]}$  if  $\hat{P}_x[k] > 0$ , and as 0 otherwise. Hereafter, we focus on estimation of the phases of the DFT,  $\tilde{y}$ .

In the literature, two main approaches were suggested to invert the discrete bispectrum. The first is based on estimating the frequencies one after the other by exploiting simple algebraic relations [45], [46]. The second approach suggests to estimate the signal by least-squares solution and phase unwrapping [46], [34]. We improve these methods and suggest a few new algorithms. The algorithms are split into two sections. This section is devoted to two new non-convex algorithms based on optimization on the manifold of phases. Both of these algorithms require initialization. While experimentally it appears that random initialization works well, for completeness, in the next section we propose three additional algorithms which do not need initialization and hence could be used to initialize the non-convex algorithms.

#### A. Local non-convex algorithm over the manifold of phases

In this section, similarly to (II.9), we let  $\tilde{B}$  denote our estimate of the phases of  $B_x$ . Since  $\tilde{B} \approx \tilde{y} \tilde{y}^* \circ T(\tilde{y})$ , one way to model recovery of the Fourier phases  $\tilde{y}$  is by means of the non-convex least-squares optimization problem

$$\min_{z \in \mathbb{C}^N} \left\| W \circ (\tilde{B} - zz^* \circ T(z)) \right\|_F^2 \text{ subject to } |z[k]| = 1, \forall k. \quad (\text{III.1})$$

The matrix  $W \in \mathbb{R}^{N \times N}$  is a weight matrix with nonnegative entries. These weights can be used to indicate our confidence in each entry of  $\tilde{B}$ . Expanding the squared Frobenius norm yields

$$\left\| W \circ (\tilde{B} - zz^* \circ T(z)) \right\|_F^2 = \|W \circ \tilde{B}\|_F^2 + \|W \circ zz^* \circ T(z)\|_F^2 - 2\langle W \circ \tilde{B}, W \circ zz^* \circ T(z) \rangle,$$

where

$$\langle U, V \rangle = \Re\{\text{Tr}(U^*V)\}, \quad (\text{III.2})$$

is the real inner product associated to the Frobenius norm. Under the constraints on  $z$ , the first two terms are constant and the inner product term is equivalent to

$$\langle W \circ \tilde{B}, W \circ z z^* \circ T(z) \rangle = \langle z, M(z)z \rangle,$$

with

$$M(z) := W^{(2)} \circ \tilde{B} \circ \overline{T(z)}, \quad (\text{III.3})$$

where we use the notation  $W^{(2)} := W \circ W$ . One possibility is to choose  $W = \sqrt{|\hat{B}_{x-\mu_x}|}$ ,

where the absolute value and the square root are taken entry-wise, so that  $M(z) = \hat{B}_{x-\mu_x} \circ \overline{T(z)}$ . Hence, optimization problem (III.1) is equivalent to

$$\max_{z \in \mathbb{C}^N} f(z) = \langle z, M(z)z \rangle \text{ subject to } |z[k]| = 1, \forall k. \quad (\text{III.4})$$

We can also impose  $z[0] = \text{phase}(\hat{\mu}_x)$ . If  $x$  is real, we have the additional symmetry constraints  $z[k] = \overline{z[-k]}$ .

### Algorithm 2

Non-convex optimization on phase manifold

---

**Input:** The normalized bispectrum  $\tilde{B}[k_1, k_2]$  and a weight matrix  $W \in \mathbb{R}^{N \times N}$

**Output:**  $\hat{y}$ : an estimation of  $\tilde{y}$

**Compute:** Using RTR [62], [63], approximately solve:

$$\hat{y} = \arg \max_{z \in \mathbb{C}^N} \Re \{z^* M(z)z\} \text{ subject to } |z[k]| = 1, \forall k, \quad (\text{if } x \text{ is real}) z[k] = \overline{z[-k]}, \forall k,$$

---

where  $M(z) := (W \circ W) \circ \tilde{B} \circ \overline{T(z)}$ .

---

Since the cost function  $f$  is continuous and the search space is compact, a solution exists. Of course, the solution is not unique, in accordance with the invariance of the bispectrum under integer time-shifts of the underlying discrete signal. This is apparent through the fact that the cost function  $f$  is invariant under the corresponding (discrete) transformations of  $z$ . This is true independently of the data  $\tilde{B}$  and  $W$ . The proof is in Appendix C.

**Lemma III.1**—The cost function  $f$  is invariant under transformations of  $z$  that correspond to integer time-shifts of the underlying signal.

To solve this non-convex program, we use the Riemannian trust-region method (RTR) [62], whose usage is simplified by the toolbox Manopt [63]. RTR enjoys global convergence to second-order critical points, that is, points which satisfy first- and second-order necessary

optimality conditions [64] and a quadratic local convergence rate. Empirically, in the noiseless case it appears that the algorithm recovers the target signal with *random initialization*, all local minima are global (with minor technicality for even  $N$  in the real case) and all second-order critical points have an escape direction, that is, saddles are “strict”. Numerical experiments demonstrate reasonable robustness in the face of noise. This algorithm is summarized in Algorithm 2 and studied in detail in Section V.

## B. Iterative phase synchronization algorithm

In this section we present an alternative heuristic to the non-convex algorithm on the manifold of phases. This algorithm is based on iteratively solving the phase synchronization problem. Suppose we get an estimation of  $\tilde{y}$ , say  $\hat{y}_{k-1}$ . If  $\hat{y}_{k-1} \approx \tilde{y}$  is non-vanishing, then this estimation should approximately satisfy the bispectrum relation:

$$\tilde{B} \circ \overline{T(\hat{y}_{k-1})} \approx \hat{y}_{k-1} \hat{y}_{k-1}^*.$$

The underlying idea is now to push the current estimation towards  $\tilde{y}$  by finding a rank-one approximation of  $\tilde{B} \circ \overline{T(\hat{y}_{k-1})}$  with unit modulus entries. This problem can be formulated as:

$$\arg \max_{z \in \mathbb{C}^N} \Re \left\{ z^* \left( \tilde{B} \circ \overline{T(\hat{y}_{k-1})} \right) z \right\} \text{ subject to } |z[\ell]| = 1, \forall \ell, \quad (\text{III.5})$$

where we treat the matrix  $\tilde{B} \circ \overline{T(\hat{y}_{k-1})}$  as a constant. This problem is called phase synchronization. Many algorithms have been suggested to solve the phase synchronization problem. Among them are the eigenvector method, SDP relaxation, projected power method, Riemannian optimization and approximate message passing [16], [20], [17], [18], [19]. Notice that the solution of (III.5) is only defined up to a global phase, namely, if  $z$  is optimal, then so is  $ze^{i\phi}$  for any angle  $\phi$ . To resolve this ambiguity, we require knowledge of the phase of the mean,  $\tilde{y}[0]$  (which is easy to estimate from the data) and we pick the global phase of  $\hat{y}_k$  such that  $\hat{y}_k[0] = \tilde{y}[0]$ .

The  $k$ th iteration of our algorithm thus (tries to) solve the phase synchronization problem with respect to the matrix  $M_{k-1} := \tilde{B} \circ \overline{T(\hat{y}_{k-1})}$ , where  $\hat{y}_{k-1}$  is the solution of the previous estimation. Assuming the signal is real, we also impose at each iteration the conjugate-reflection property of  $\hat{y}_k[\ell] = \overline{\hat{y}_k[-\ell]}$  for all  $\ell$  so that  $M_k$  is Hermitian. In the numerical experiments in Section VII, we solve (III.5) by the Riemannian trust-region method described in [17]. Empirically, the performance of this algorithm and Algorithm 2 is indistinguishable. The algorithm is summarized in Algorithm 3.

### Algorithm 3

Iterative phase synchronization algorithm

---

**Input:** The normalized bispectrum  $\tilde{B}$ , initial estimation  $\hat{y}_0$ , phase of the mean  $\tilde{y}[0]$

**Output:**  $\hat{y}$ : estimation of  $\tilde{y}$

Set  $k = 0$

**while** stopping criterion does not trigger **do**:

- $k \leftarrow k + 1$
- Compute  $\hat{y}_k$  as a solution of (III.5)
- Fix the global phase:  $\hat{y}_k \leftarrow \hat{y}_k \cdot \frac{\tilde{y}[0]}{\hat{y}_k[0]}$
- If  $x$  is real, symmetrize:  $\hat{y}_k \leftarrow \text{phase}((\hat{y}_k)_{i^*})$ , see (B.1)

**end while**

**Return:**  $\hat{y} \leftarrow \hat{y}_k$

---

## IV. Initialization-free Algorithms

The previous section was devoted to non-convex algorithms to invert the bispectrum. In this section we present three additional algorithms based on frequency marching (FM), SDP relaxation and phase unwrapping. These algorithms do not require initialization and therefore could be used to initialize the non-convex algorithms.

We prove that FM and the SDP recover the Fourier phases exactly in the absence of noise under the assumption that we can fix  $\hat{y}[1]$ . If the signal has non-vanishing DFT,  $\hat{y}[1]$  can be estimated from the bispectrum using the fact that  $\hat{y}[1]^N$  equals

$$\text{phase}(B_x[N-1, 1]B_x[1, 2] \cdot B_x[1, 2] \cdots B_x[1, N-1]).$$

(Any  $M$ th root can be used for  $\hat{y}[1]$ , corresponding to the  $N$  possible shifts of  $x$ .) In all cases, we argue that forcing  $\hat{y}[1] = 1$  is acceptable if  $N$  is large. Indeed, recall that a shift by  $\ell$  entries in the space domain is equivalent to modulating the  $k$ th Fourier coefficient by  $e^{-2\pi i \ell k / N}$ . In particular, it means that the phase  $\hat{y}[1]$  can be shifted by  $e^{-2\pi i \ell / N}$  for an arbitrary  $\ell \in \mathbb{Z}$ . Thus, for signals of length  $N \gg 1$ , the phase  $\hat{y}[1]$  can be set arbitrarily with only small error. In the numerical experiments of Section VII, we give the correct value of  $\hat{y}[1]$  to the algorithms in order to assess their best possible behavior.

We begin by discussing the FM algorithm, which is a simple propagation method: it is exact in the absence of noise. Notwithstanding, its estimation for the low-frequency coefficients is sensitive to noise. Because of its recursive nature, error in the low frequencies propagates to the high frequencies, resulting in unreliable estimation. The other two algorithms are more computationally demanding but appear more robust.

### A. Frequency marching algorithm

The FM algorithm is a simple propagation algorithm in the spirit of [45], [46] that aims to estimate  $\hat{y}$  one frequency at a time. This algorithm has computational complexity  $\mathcal{O}(N^2)$  and it recovers  $\hat{y}$  exactly for both real and complex signals in the absence of noise, assuming  $\hat{y}[1]$  is known.

Let us denote  $\tilde{B}[k_1, k_2] = e^{i\Psi[k_1, k_2]}$  and  $\tilde{y}[k] = e^{i\psi[k]}$ . Accordingly, we can reformulate (II.5) as

$$\Psi[k_1, k_2] = \psi[k_1] - \psi[k_2] + \psi[k_2 - k_1] \bmod 2\pi,$$

where the modulo is taken over the sum of all three terms. Using this relation, we can start to estimate the missing phases. The first unknown phase,  $\psi[2]$ , can be estimated by:

$$\begin{aligned} \Psi[1, 2] &= \psi[1] - \psi[2] + \psi[1] \bmod 2\pi \\ \Rightarrow \hat{\psi}[2] &= 2\psi[1] - \Psi[1, 2] \bmod 2\pi, \end{aligned}$$

where  $\hat{\psi}[2]$  refers to the estimator of  $\psi[2]$  (defined modulo  $2\pi$ ). We can estimate the next phase in the same manner:

$$\begin{aligned} \Psi[1, 3] &= \psi[1] - \psi[3] + \psi[2] \bmod 2\pi \\ \Rightarrow \hat{\psi}[3] &= \psi[1] + \hat{\psi}[2] - \Psi[1, 3] \bmod 2\pi. \end{aligned}$$

For higher frequencies, we have more measurements to rely on. For the fourth entry, we now can derive two estimators as follows:

$$\begin{aligned} \Psi[1, 4] &= \psi[1] - \psi[4] + \psi[3] \bmod 2\pi \\ \Rightarrow \hat{\psi}^{(1)}[4] &= \psi[1] + \hat{\psi}[3] - \Psi[1, 4] \bmod 2\pi, \end{aligned}$$

and

$$\begin{aligned} \Psi[2, 4] &= \psi[2] - \psi[4] + \psi[2] \bmod 2\pi \\ \Rightarrow \hat{\psi}^{(2)}[4] &= 2\hat{\psi}[2] - \Psi[2, 4] \bmod 2\pi. \end{aligned}$$

In the noiseless case, it is clear that  $\psi[4] = \hat{\psi}^{(1)}[4] = \hat{\psi}^{(2)}[4]$ . In a noisy environment, we can reduce the noise by averaging the two estimators, where averaging is done over the set of phases (namely, over the rotation group  $\text{SO}(2)$ ) as explained in Appendix D. Specifically,

$$e^{i\hat{\psi}[4]} = \text{phase} \left( e^{i\hat{\psi}^{(1)}[4]} + e^{i\hat{\psi}^{(2)}[4]} \right).$$

We can iterate this procedure. To estimate phase  $q$ , we want to consider all entries of  $\tilde{B}[k, \ell] = \tilde{y}[k]\overline{\tilde{y}[\ell]}\tilde{y}[\ell - k]$  such that exactly one of the indices  $k$ ,  $\ell$  or  $\ell - k$  is equal to  $q$  and all other indices are in  $1, \dots, q-1$ , so that all other phases involved have already been estimated. A simple verification shows that only entries  $\tilde{B}[p, q]$ ,  $p = 1, \dots, q-1$ , have that property. Furthermore, because of symmetry in the bispectrum (V.4), half of these entries are redundant so that only entries  $\tilde{B}[p, q]$ ,  $p = 1, \dots, \lfloor \frac{q}{2} \rfloor$  remain. As a result, estimation of the  $k$ th

phase relies on averaging over  $\lfloor \frac{k}{2} \rfloor$  equations, as summarized in Algorithm 4, with the following simple guarantee. The above construction yields the following proposition.

**Proposition IV.1**—*Let  $\tilde{B} = \tilde{B}_x$  be the normalized bispectrum as defined in (II.9) and assume that  $\tilde{y}[1]$  is known. If  $y[k] \neq 0$  for  $k = 1, \dots, K$ , then Algorithm 4 recovers the Fourier phases  $\tilde{y}[k]$ ,  $k = 1, \dots, K$  exactly.*

We note in closing that, if the signal  $x$  is real, symmetries in the phases  $\tilde{y}$  and  $\tilde{B}$  can be exploited easily in FM.

## B. Semidefinite programming relaxation

In this section we assume that the DFT  $y$  is non-vanishing so that the bispectrum relation can be manipulated as

$$\tilde{B} = \tilde{y}\tilde{y}^* \circ T(\tilde{y}) \equiv \tilde{B} \circ \overline{T(\tilde{y})} = \tilde{y}\tilde{y}^*,$$

where  $\overline{T(\tilde{y})}$  is its entry-wise conjugate. The developments are easily adapted if the signal has zero mean. Similarly to the FM algorithm, we assume that  $\tilde{y}[0]$  and  $\tilde{y}[1]$  are available. We aim to estimate  $\tilde{y}$  by a convex program. As a first step, we decouple the bispectrum equation and write the problem of estimating  $\tilde{y}$  as the following non-convex optimization problem:

$$\begin{aligned} \min_{Z \in \mathcal{H}^N, z \in \mathbb{C}^N} & \left\| W \circ (\tilde{B} \circ \overline{T(z)} - Z) \right\|_F^2 \quad (\text{IV.1}) \\ \text{subject to } & Z = zz^*, \\ & \text{diag}(Z) = 1, \\ & z[0] = \tilde{y}[0], z[1] = \tilde{y}[1], \\ & (\text{if } x \text{ is real}) z[k] = \overline{z[-k]}, \forall k, \end{aligned}$$

where  $\mathcal{H}^N$  is the set of Hermitian matrices of size  $N$  and  $W \in \mathbb{R}^{N \times N}$  is a real weight matrix with positive entries. In particular, in the numerical experiments we set  $W = |\tilde{B}|$ .

### Algorithm 4

Frequency marching algorithm

---

**Input:** Normalized bispectrum  $\tilde{B}[k_1, k_2] = e^{j\psi[k_1, k_2]}$ ,  $\tilde{y}[0]$  and  $\tilde{y}[1] \neq 0$

**Output:**  $\tilde{y}$ : estimation of  $\tilde{y}$

- 1 Set  $\tilde{y}[0] = \tilde{y}[0]$  and  $e^{j\psi[1]} = \tilde{y}[1]$
- 2 **For**  $k = 2, \dots, N$  **do**:
  - a. Average the phase measurements:

$$u = \text{phase} \left( \sum_{\ell=1}^{\lfloor \frac{k}{2} \rfloor} e^{i(\hat{\psi}[\ell] + \hat{\psi}[k-\ell] - \Psi[\ell, k])} \right)$$

- b. Estimate  $\hat{\psi}[k]$  through:

$$e^{i\hat{\psi}[k]} = \begin{cases} u, & u \neq 0, \\ 1, & u = 0, \end{cases}$$

**Return:**  $\hat{y} \leftarrow e^{i\hat{\psi}}$

In the absence of noise, the minimizers of (IV.1) satisfy the bispectrum equation. However, in general these cannot be computed in polynomial time. In order to make the problem tractable, we relax the non-convex coupling constraint  $Z = zz^*$  to the convex constraint  $Z \succeq zz^*$  (that is,  $Z - zz^*$  is positive semidefinite). The convex relaxation is then given by

$$\min_{Z \in \mathcal{H}^N, z \in \mathbb{C}^N} \left\| W \circ (\tilde{B} \circ \overline{T(z)} - Z) \right\|_F^2 \quad (\text{IV.2})$$

subject to  $Z \succeq zz^*$ ,

$$\text{diag}(Z) = 1,$$

$$z[0] = \tilde{y}[0], \quad z[1] = \tilde{y}[1],$$

$$(\text{if } x \text{ is real}) \quad z[k] = \overline{z[-k]}, \quad \forall k,$$

Upon solving (IV.2), which can be done in polynomial time with interior point methods, the phases  $\hat{y}$  are estimated from  $\text{phase}(z)$ . In practice, we use CVX to solve this problem [65]. The algorithm is summarized in Algorithm 5. We note that problem (IV.2) is not a standard SDP, in that its cost function is nonlinear.

In the noiseless case, the SDP relaxation (IV.2) recovers the missing phases exactly. Interestingly, the proof is not so much based on optimality conditions as it is on an algebraic property of circulant matrices. The proof of the following property is given in Appendix E.

### Algorithm 5

Semidefinite relaxation algorithm

---

**Input:** The normalized bispectrum  $\tilde{B}$ ,  $\tilde{y}[0]$  and  $\tilde{y}[1]$

**Output:**  $\hat{y}$ : estimation of  $\tilde{y}$

**Solve** the SDP with nonlinear cost function (IV.2), for example using CVX [65]

**Return:**  $\hat{y} \leftarrow \text{phase}(z)$

---



**Lemma IV.2**—Let  $\hat{u}$  be the DFT of a vector  $u \in \mathbb{C}^N$  obeying  $u[k] = \overline{u[-k]}$ , so that  $\hat{u}$  is real. If  $u[0] = u[1] = 1$  and  $\hat{u}$  is non-negative, then  $u[k] = 1$  for all  $k$ .

The following theorem is a direct corollary of Lemma IV.2. The main proof idea is as follows. Consider  $u = \tilde{y} \circ z$  where  $(Z, z)$  is optimal for the SDP; then, the constraints ensure  $u[0] = u[1] = 1$ . Furthermore, one can see via the Schur complement that the constraints force  $T(u)$  to be positive semidefinite. Since the eigenvalues of  $T(u)$  are the DFT of  $u$ , it follows that  $\hat{u}$  is non-negative, so that the lemma above applies and  $u \equiv 1$ , or, equivalently,  $z = \tilde{y}$ . Details of the proof are in Appendix F.

**Theorem IV.3**—For a real signal with non-vanishing DFT  $y$ , if all weights in  $W$  are positive,  $\tilde{y}[0]$  and  $\tilde{y}[1]$  are known and the objective value of (IV.2) attains 0 (which is the case in the absence of noise), then the SDP has a unique solution given by  $z = \tilde{y}$  and  $Z = z z^*$ .

We close with an important remark about the symmetry breaking purpose of constraint  $z[1] = \tilde{y}[1]$  in the SDP. Because the signal  $x$  can be recovered only up to integer time shifts, even in the noiseless case, without this constraint there are at least  $N$  distinct solutions  $(z, Z)$  to the SDP. Because SDP is a convex program, any point in the convex hull of these  $N$  points is also a solution. Thus, if the symmetry is not broken, the set of solutions contains many irrelevant points. Furthermore, interior point methods tend to converge to a center of the set of solutions, which in this case is never one of the desired solutions.

### C. Phase unwrapping by integer programming algorithm

The next algorithm is based on solving an over-determined system of equations involving integers. Let us denote  $\tilde{y}[k] = e^{i\psi[k]}$  and  $\tilde{B}[k_1, k_2] = e^{i\Psi[k_1, k_2]}$  so the normalized bispectrum model is given by

$$e^{i\Psi[k_1, k_2]} = e^{i(\psi[k_1] - \psi[k_2] + \psi[k_2 - k_1])}.$$

By taking the logarithm, we get the algebraic relation

$$\Psi[k_1, k_2] + 2\pi\chi[k_1, k_2] = \psi[k_1] - \psi[k_2] + \psi[k_2 - k_1], \quad (\text{IV.3})$$

where, as a result of phase wrapping,  $\chi$  takes on integer values. Let  $\Psi_{\text{vec}}$  and  $\chi_{\text{vec}}$  be the column-stacked versions of  $\Psi$  and  $\chi$ , respectively. Then, the model reads

$$\Psi_{\text{vec}} + 2\pi\chi_{\text{vec}} = A\psi, \quad (\text{IV.4})$$

where the sparse matrix  $A \in \mathbb{R}^{N^2 \times N}$  encodes the right hand side of (IV.3). It can be verified that  $A$  is of rank  $N - 1$  (see for instance [66]), with null space corresponding to the time-

shift-induced ambiguity on the phases (II.3). Note that both the integer vector  $\chi_{\text{vec}}$  and the phases  $\psi$  are unknown. Given  $\chi_{\text{vec}}$ , the phases  $\psi$  can be obtained easily by solving

$$\min_{\psi \in \mathbb{R}^N} \|\Psi_{\text{vec}} + 2\pi\chi_{\text{vec}} - A\psi\|_p, \quad (\text{IV.5})$$

for some  $\ell_p$  norm. Observe that any error in estimating  $\chi$  may cause a big estimation error of  $\psi$  in (IV.5). These errors can be thought of as outliers. Hence, we choose to use least unsquared deviations (LUD),  $p = 1$ , which is more robust to outliers. The more challenging task is to estimate the integer vector  $\chi_{\text{vec}} \in \mathbb{Z}^{N^2}$ . To this end, we first eliminate  $\psi$  from (IV.4) as follows. Let  $C \in \mathbb{R}^{(N^2 - (N-1)) \times N^2}$  be a full rank matrix such that  $CA = 0$ , that is, the columns of  $C^T$  are in the null space of  $A^T$ . Matrix  $C$  can be designed by at least two methods. One, suggested in [67], exploits the special structure of  $A$  to design a sparse matrix composed of integer values. Another, which we use here, is to take  $C$  to have orthonormal rows which form a basis of the kernel of  $A^T$ . Numerical experiments (not shown) indicate that the latter approach is more stable. Next, we multiply both sides of (IV.4) from the left by  $C$  to get

$$C\{\Psi_{\text{vec}} + 2\pi\chi_{\text{vec}}\} = CA\psi = 0.$$

Therefore, the integer recovery problem can be formulated as

$$\min_{\chi_{\text{vec}} \in \mathbb{Z}^{N^2}} \left\| \frac{1}{2\pi} C\Psi_{\text{vec}} + C\chi_{\text{vec}} \right\|_2, \quad (\text{IV.6})$$

where we minimize over all integers. Note that  $C\Psi_{\text{vec}}$  is a known vector. The problem is then equivalent to finding a lattice vector with the basis  $C$  which is as close as possible to the vector  $-C\Psi_{\text{vec}}/(2\pi)$ . While the problem is known to be NP-hard, we approximate the solution of (IV.6) with the LLL (Lenstra–Lenstra–Lovasz) algorithm, which can be run in polynomial time [68]. The LLL algorithm computes a lattice basis, called a *reduced basis*, which is approximately orthogonal. It uses the Gram–Schmidt process to determine the quality of the basis. For more details, see [69, Ch. 17].

We note that (IV.6) is under-determined as the matrix  $C$  is of rank  $N^2 - \text{rank}(A) = N^2 - (N - 1)$ . While the LLL algorithm works with under-determined systems, in our case we can solve it for a determined system since we can fix the first  $N - 1$  entries of  $\chi_{\text{vec}}$  to be zero.<sup>2</sup> Once we have estimated  $\chi_{\text{vec}}$ , we solve (IV.5) with  $p = 1$ . This approach is summarized in Algorithm 6.

<sup>2</sup>We omit the proof of this property here and only mention that it is based on the derivation in [67].

### Algorithm 6

#### Phase unwrapping by integer programming

**Input:** The normalized bispectrum  $\tilde{B}[k_1, k_2] = e^{j\Psi[k_1, k_2]}$

**Output:**  $\hat{y}$ : estimation of  $\tilde{y}$

- 1 (integer programming) Apply the LLL algorithm to estimate the integer vector  $\chi_{\text{vec}}$  from

$$\min_{\chi_{\text{vec}} \in \mathbb{Z}^{N^2}} \|C\Psi_{\text{vec}}^{j(2\pi)} + C\chi_{\text{vec}}\|_2,$$

where  $A$  is given in (IV.4),  $CA = 0$  and  $\Psi_{\text{vec}} \in \mathbb{R}^{N^2}$  is a column-stacked version of  $\Psi$ , e.g., using code from [70].

- 2 (least-unsquared minimization) Let  $\hat{\chi}_{\text{vec}}$  be the solution of stage 1. Then, solve

$$\hat{\psi} = \arg \min_{\hat{\psi} \in \mathbb{R}^N} \|\Psi_{\text{vec}} + 2\pi\hat{\chi}_{\text{vec}} - A\psi\|_1.$$

**Return:**  $\hat{y} \leftarrow e^{j\hat{\psi}}$

## V. Analysis of optimization over phases

In this section, we study the non-convex optimization problem (III.4) and give more implementation details to solve it, since numerical experiments identify this as the method of choice for MRA from invariant features among all methods compared. We start by considering the general case of a complex signal  $x \in \mathbb{C}^N$  and consider the real case in Appendix B. Recall that we aim to maximize

$$f(z) = \langle z, M(z)z \rangle, \quad M(z) := W^{(2)} \circ \tilde{B} \circ \overline{T(z)},$$

where the inner product is defined by (III.2),  $W$  is a real weighting matrix and  $W^{(2)} := W \circ W$ . The optimization problem lives on a *manifold*, that is, a smooth nonlinear space. Indeed, the smooth cost function  $f(z)$  is to be maximized over the set

$$\mathcal{M} = \left\{ z \in \mathbb{C}^N : |z[0]| = \dots = |z[N-1]| = 1 \right\},$$

which is a Cartesian product of  $N$  unit circles in the complex plane (a torus). Theory and algorithms for optimization on manifolds can be found in the monograph [71]. We follow this formalism here. Details can also be found in [17], which deals with the similar problem of phase synchronization, using similar techniques. For the numerical experiments below, we use the toolbox Manopt which provides implementations of various optimization algorithms on manifolds [63].

Under mild conditions, the global optima of (III.4) correspond exactly to  $\tilde{y}$  up to integer time shifts. This fact is proven in Appendix G.

**Lemma V.1**—For  $N \geq 3$ , let  $x \in \mathbb{C}^N$  be a signal whose DFT  $y$  is nonzero for frequencies  $k$  in  $\{1, \dots, K\}$ , possibly also for  $k=0$ , and zero otherwise. Up to integer time shifts,  $x$  is determined exactly by its bispectrum  $B$  provided  $K \geq \frac{N+1}{2}$ . Furthermore, the global optima of (III.4) correspond exactly to the relevant phases of  $y$ —up to the effects of integer time shifts—provided  $W[k, \ell]$  is positive when  $B[k, \ell] \neq 0$ .

The problem at hand is

$$\max_{z \in \mathcal{M}} f(z). \quad (\text{V.1})$$

This is smooth but non-convex, so that in general it is hard to compute the global optimum. We derive first- and second-order necessary optimality conditions. Points which satisfy these conditions are called *critical* and *second-order critical* points, respectively. Known algorithms converge to critical points (e.g., Riemannian gradient descent) and even to second-order critical points (e.g., Riemannian trust-regions) regardless of initialization [71], [62], [64]. Empirically, despite non-convexity, the global optimum appears to be computable reliably in favorable noise regimes.

As we proceed to consider optimization algorithms for (III.4), the gradient of  $f$  will come into play:

$$\nabla f(z) = M(z)z + M(z)^* z + M^{\text{adj}}(zz^*),$$

where  $M^{\text{adj}}: \mathbb{C}^{N \times N} \rightarrow \mathbb{C}^N$  is the adjoint of  $M$  with respect to the inner product  $\langle \cdot, \cdot \rangle$ . Formally, the adjoint is defined such that, for any  $z \in \mathbb{C}^N$ ,  $X \in \mathbb{C}^{N \times N}$ ,

$$\langle z, M^{\text{adj}}(X) \rangle = \langle M(z), X \rangle.$$

Specifically, in Appendix H we show that

$$M^{\text{adj}}(X)[k] = \text{Tr} \left( T_k^T \left( W^{(2)} \circ \tilde{B} \circ \bar{X} \right) \right), \quad (\text{V.2})$$

where  $T_k$  is a circulant matrix with ones in its  $k$ th (circular) diagonal and zero otherwise, namely,

$$(T_k)[\ell', \ell] = \begin{cases} 1 & \text{if } \ell' = \ell - k, \\ 0 & \text{otherwise.} \end{cases} \quad (\text{V.3})$$

As it turns out, under the symmetries of the problem at hand, there is no need to evaluate  $M^{\text{adj}}$  explicitly. Indeed,  $\tilde{B}$  obeys

$$\tilde{B}[k_2 - k_1, k_2] = \tilde{y}[k_2 - k_1] \overline{\tilde{y}[k_2]} \tilde{y}[k_1] = \tilde{B}[k_1, k_2]. \quad (\text{V.4})$$

This property is preserved when  $\tilde{B}$  is obtained by averaging bispectra of multiple observations, as in (II.8). Assuming the same symmetry for the real weights  $W$ , we find below that  $M^{\text{adj}}(zz^*) = M(z)z$ . See Appendix I.

**Lemma V.2**—*If  $\tilde{B}[k_2 - k_1, k_2] = \tilde{B}[k_1, k_2]$  and  $W[k_2 - k_1, k_2] = W[k_1, k_2]$  for all  $k_1, k_2$ , then  $M^{\text{adj}}(zz^*) = M(z)z$  for all  $z \in \mathbb{C}^N$ .*

Thus, under the symmetries assumed in Lemma V.2, the gradient of  $f$  simplifies and we get a simple expression for the Hessian as well:

$$\begin{aligned} \nabla f(z) &= 2M(z)z + M(z)^*z, \\ \nabla^2 f(z)[\dot{z}] &= 2M(\dot{z})z + 2M(z)\dot{z} + M(\dot{z})^*z + M(z)^*\dot{z}. \end{aligned} \quad (\text{V.5})$$

For unconstrained optimization, the first-order necessary optimality conditions are  $\nabla f(z) = 0$ . In the presence of the constraint  $z \in \mathcal{M}$ , the conditions are different. Namely, following [71, eq. (3.37)], since  $\mathcal{M}$  is a submanifold of  $\mathbb{C}^N$ , first-order necessary optimality conditions state that the orthogonal projection of the gradient  $\nabla f(z)$  to the tangent space to  $\mathcal{M}$  at  $z$  must vanish. The result of this projection is called the *Riemannian gradient*. Formally, the tangent space is obtained by linearizing (differentiating) the constraints  $|z[k]|^2 = \langle z[k], z[k] \rangle = 1$  for all  $k$ , yielding

$$T_z \mathcal{M} = \{ \dot{z} \in \mathbb{C}^N : \langle z[k], \dot{z}[k] \rangle = 0, \forall k \}.$$

Orthogonal projection of  $u \in \mathbb{C}^N$  to the tangent space  $T_z \mathcal{M}$  can be computed entry-wise by subtracting from each  $u[k]$  its component aligned with  $z[k]$ . Let  $\text{Proj}_z: \mathbb{C}^N \rightarrow T_z \mathcal{M}$  denote this projection. This operation admits a compact matrix notation as

$$\begin{aligned} u &\mapsto \text{Proj}_z(u) = u - \Re\{u \circ \bar{z}\} \circ z \\ &= u - \Re\{\text{ddiag}(uz^*)\}z, \end{aligned}$$

where  $\text{ddiag}: \mathbb{C}^{N \times N} \rightarrow \mathbb{C}^{N \times N}$  sets all non-diagonal entries of a matrix to zero. Equipped with this notion and the expression for  $\nabla f(z)$  (V.5), it follows that the Riemannian gradient of  $f$  at  $z$  on  $\mathcal{M}$  is

$$\text{grad}f(z) := \text{Proj}_z(\nabla f(z)) = \nabla f(z) - D(z)z,$$

with

$$D(z) := \Re\{\text{ddiag}(\nabla f(z)z^*)\} = \Re\{\text{diag}(\nabla f(z) \circ \bar{z})\}.$$

**Lemma V.3**—*If  $z \in \mathcal{M}$  is optimal for (V.1), then  $\text{grad} f(z) = 0$ ; equivalently,  $\text{diag}(\nabla f(z)z^*) = \nabla f(z) \circ \bar{z}$  is real.*

**Proof:** See [72, Rem. 4.2 and Cor. 4.2]. For the equivalence, notice that  $\text{Proj}_z(u) = 0$  if and only if  $u[k] = \Re\{u[k]\bar{z}[k]\}z[k]$  for all  $k$ , and multiply by  $\bar{z}[k]$  on both sides using  $|z[k]| = 1$ .

A point  $z$  which satisfies these conditions is called a critical point. Likewise, we can define a notion of *Riemannian Hessian* as the linear, self-adjoint operator on  $T_z\mathcal{M}$  which captures infinitesimal changes in the Riemannian gradient around  $z$ . Without getting into technical details, we follow [71, eq. (5.15)] and define (with  $D$  the directional derivative operator):

$$\begin{aligned} \text{Hess}f(z)[\dot{z}] &:= \text{Proj}_z(D(z \mapsto \text{grad}f(z))(z)[\dot{z}]) \\ &= \text{Proj}_z(\nabla^2 f(z)[\dot{z}] - D(z)\dot{z} - (DD(z)[\dot{z}]z), \end{aligned}$$

where  $DD(z)[\ ]$  is a real, diagonal matrix. Its contribution to the Hessian is zero, since  $(DD(z)[\ ])z$  vanishes under the projection  $\text{Proj}_z$ . Hence,

$$\text{Hess}f(z)[\dot{z}] = \text{Proj}_z(\nabla^2 f(z)[\dot{z}] - D(z)\dot{z}).$$

The Riemannian Hessian intervenes in the second-order necessary optimality conditions as follows.

**Lemma V.4**—*If  $z \in \mathcal{M}$  is optimal for (V.1), then  $\text{grad} f(z) = 0$  and  $\text{Hess} f(z) \leq 0$ , that is, for all  $\dot{z} \in T_z\mathcal{M}$  we have*

$$\langle \dot{z}, \text{Hess}f(z)[\dot{z}] \rangle = \langle \dot{z}, \nabla^2 f(z)[\dot{z}] \rangle - \langle \dot{z}, D(z)\dot{z} \rangle \leq 0.$$

**Proof:** See [72, Rem. 4.2 and Cor. 4.2]. In the equality, we used the fact that  $\text{Proj}_z$  is self-adjoint and  $\dot{z} \in T_z\mathcal{M}$ .

A point  $z$  which satisfies these conditions is called a *second-order critical point*. With unit weights, the following lemma shows that second-order critical points  $z$ , in the noiseless case, cannot have an arbitrarily bad objective value  $f(z)$ . This result is weak, however, since empirically it is observed that in the noiseless case local optimization methods consistently converge to global optima whose value are  $N^2$ , suggesting that all second-order critical points are global optima in this simplified scenario. While we do not have a proof for this stronger conjecture, we provide the lemma below because it is analogous to [17, Lemma 14] which, in that reference, is a key step toward proving global optimality of second-order critical points.

**Lemma V.5**—*In the absence of noise and with unit weights, a second-order critical point  $z$  of (V.1) satisfies*

$$\nabla f(z) \circ \bar{z} \geq 2(\sqrt{3} - 1) > 0.$$

In particular, this implies

$$f(z) = \frac{1}{3} \langle z, \nabla f(z) \rangle \geq \frac{2(\sqrt{3} - 1)}{3} N.$$

**Proof:** See Appendix J for the proof of the inequality. It follows from two key considerations. First, because  $z$  is a critical point, Lemma V.3 indicates that  $\nabla f(z) \circ \bar{z}$  is real. Second, because  $z$  is second-order critical, the Riemannian Hessian at  $z$  must be negative semidefinite by Lemma V.4. Applied to all tangent directions at  $z$  which perturb only one phase at a time implies the desired inequality. The fact that  $3f(z) = \langle z, \nabla f(z) \rangle$  follows from (V.5).

One final ingredient that is necessary to optimize  $f$  over  $\mathcal{M}$  is a means of moving away from a current iterate  $z \in \mathcal{M}$  to the next by following a tangent vector. A simple means of achieving this is through a *retraction* [71, Def. 4.1.1]. For  $\mathcal{M}$ , an obvious retraction is the following:

$$\text{Retr}_z(\dot{z}) = \text{phase}(z + \dot{z}) \in \mathcal{M}. \quad (\text{V.6})$$

With the formalism of (V.1) and the above derivations, we can now run a local Riemannian optimization algorithm. As an example, the Riemannian gradient ascent algorithm would iterate the following:

$$\begin{aligned} z^{(t+1)} &= \text{Retr}_{z^{(t)}}\left(\eta^{(t)} \text{grad} f(z^{(t)})\right) \\ &= \text{phase}\left(z^{(t)} + \eta^{(t)} \text{grad} f(z^{(t)})\right), \end{aligned}$$

where  $\eta^{(l)} > 0$  is an appropriately chosen step size and  $z^{(0)} \in \mathcal{M}$  is an initial guess. It is relatively easy to choose the step sizes such that the sequence  $z^{(l)}$  converges to critical points regardless of  $z^{(0)}$ , with a linear local convergence rate [71, §4]. In practice, we prefer to use the Riemannian trust-region method (RTR) [62], whose usage is simplified by the toolbox Manopt [63]. RTR enjoys global convergence to second-order critical points [64] and a quadratic local convergence rate.

In this section, the analysis focused on complex signals. For real signals, we can follow the same methodology while taking the symmetry in the Fourier domain into account. This analysis is given in Appendix B.

## VI. Expectation maximization

In this section, we detail the expectation maximization algorithm (EM) [73] applied to MRA. As the numerical experiments in Section VII demonstrate, EM achieves excellent accuracy in estimating the signal. However, compared to the invariant features approach proposed in this paper, it is significantly slower and requires many passes over the data (thus excluding online processing).

Let  $X = [\xi_1, \dots, \xi_M]$  be the data matrix of size  $N \times M$ , following the MRA model (I.1). The maximum marginalized likelihood estimator (MMLE) for the signal  $x$  given  $X$  is the maximizer of the likelihood function  $L(x; X) = p(X|x)$  (the probability density of  $X$  given  $x$ ). This density could in principle be evaluated by marginalizing the joint distribution  $p(X, r|x)$  over the unknown shifts  $r \in \{0, \dots, N-1\}^M$ . This, however, is intractable as it involves summing over  $N^M$  terms.

Alternatively, EM tries to estimate the MMLE as follows. Given a current estimate for the signal  $x_k$ , consider the expected value of the log-likelihood function, with respect to the conditional distribution of  $r$  given  $X$  and  $x_k$ :

$$Q(x | x_k) = \mathbb{E}_{r | X, x_k} \{ \log p(X, r | x_k) \}. \quad (\text{VI.1})$$

This step is called the E-step. Then, iterate by computing the M-step:

$$x_{k+1} = \arg \max_x Q(x | x_k). \quad (\text{VI.2})$$

For the MRA model, this can be done in closed form. Indeed, the log-likelihood function follows from the i.i.d. Gaussian noise model:

$$\log p(X, r | x) = -\frac{1}{2\sigma^2} \sum_{j=1}^M \|R_{r_j} x - \xi_j\|_2^2 + \text{constant}. \quad (\text{VI.3})$$



To take the expectation with respect to  $r$ , we need to compute  $w_k^{\ell,j}$ : for each observation  $j$ , this is the probability that the shift  $r_j$  is equal to  $\ell$  given  $X$  and assuming  $x = x_k$ . This also follows easily from the i.i.d. Gaussian noise model:

$$w_k^{\ell,j} \propto \exp\left(-\frac{1}{2\sigma^2}\|R_\ell x_k - \xi_j\|_2^2\right), \quad (\text{VI.4})$$

(with appropriate scale so that  $\sum_{\ell=0}^{N-1} w_k^{\ell,j} = 1$ ). This allows to write  $Q$  down explicitly:

$$Q(x | x_k) = -\frac{1}{2\sigma^2} \sum_{j=1}^M \sum_{\ell=0}^{N-1} w_k^{\ell,j} \|R_\ell x - \xi_j\|_2^2 + \text{constant}.$$

This is a convex quadratic expression in  $x$  with maximizer

$$x_{k+1} = \frac{1}{M} \sum_{j=1}^M \sum_{\ell=0}^{N-1} w_k^{\ell,j} R_\ell^{-1} \xi_j. \quad (\text{VI.5})$$

In words: given an estimator  $x_k$ , the next estimator is obtained by averaging all shifted versions of all observations, weighted by the empirical probabilities of the shifts. Considering all shifts of all observations would, in principle, induce an iteration complexity of  $\mathcal{O}(MN^2)$ , but fortunately, for each observation, the matrix of its shifted versions is circulant, which makes it possible to use FFT to reduce the overall computational cost to  $\mathcal{O}(MN \log N)$ . See the available code for details. We note that Matlab naturally parallelizes the computations over  $M$ .

In practice, we set  $x_0 \sim \mathcal{N}(0, I_N)$  to be a random guess. Furthermore, for  $M = 3000$ , we first execute 3000 batch iterations, where the EM update is computed based on a random sample of 1000 observations (fresh sample at each iteration). This inexpensively transforms the random initialization into a ballpark estimate of the signal. The algorithm then proceeds with full-data iterations until the relative change between two consecutive estimates drops below  $10^{-5}$  (in  $\ell_2$ -norm, up to shifts).

## VII. Numerical Experiments

This section is devoted to numerical experiments, examining all proposed algorithms. Code for all algorithms and to reproduce the experiments is available online.<sup>3</sup> The experiments were conducted as follows. The true signal  $x$  of length  $N = 41$  is a fixed window of height 1 and width 21. With this signal, the signal-to-noise ratio is  $\frac{\|x\|_2^2}{\|\varepsilon\|_2^2} \approx \frac{1}{2\sigma^2}$ . We generated a set of  $M$  shifted noisy versions of  $x$  as

<sup>3</sup><https://github.com/NicolasBoumal/MRA>

$$\xi_j = R_{r_j} x + \varepsilon_j,$$

where each shift was randomly drawn from a uniform distribution over  $\{0, \dots, N-1\}$  and  $\varepsilon_j \sim \mathcal{N}(0, \sigma^2 I)$  for all  $j$ . The relative recovery error for a single experiment is defined as

$$\text{relative error}(x, \hat{x}) = \min_{s \in \{0, \dots, N-1\}} \frac{\|R_s \hat{x} - x\|_2}{\|x\|_2},$$

where  $\hat{x}$  is the estimation of the signal. All results are averaged over 20 repetitions. While we present here results for a specific signal, alternative signal models (e.g., random signals) showed similar numerical behavior.

The following figures compare the recovery errors for all proposed algorithms, with random initialization for those that need initialization. The non-convex algorithm on the manifold of phases, Algorithm 2, runs the Riemannian trust-region method (RTR) [62] using the toolbox Manopt [63]. Algorithm 3 runs 15 iterations with warm-start using the same toolbox. For the phase unwrapping algorithm, Algorithm 6, we use an implementation of LLL available in the MILES package [70]. The SDP is solved with CVX [65]. The EM algorithm is implemented as explained in Section VI. We compared the algorithms with an oracle who knows the random shifts  $r_j$  and therefore simply averages out the Gaussian noise. Experiments are run on a computer with 30 CPUs available. These CPUs are used to compute the invariants in parallel (with Matlab's `parfor`), while the EM algorithm benefits from parallelism to run the many thousands of FFTs it requires efficiently (built-in Matlab). The algorithms that need  $\mathcal{Y}[0]$  and  $\mathcal{Y}[1]$  are given the correct values.

Figures VII.1 and VII.2 present the recovery error and computation time of all algorithms as a function of the number of observations  $M$  for fixed noise level  $\sigma = 1$ . Of course, the oracle who knows the shifts of the observations is unbeatable. Algorithms 2 and 3 outperform all invariant approach methods. The inferior performance of the SDP might be explained by the fact that we are minimizing a smooth non-linear objective. This is in contrast to SDPs with linear or piecewise linear objectives which tend to promote “simple” (i.e., low rank) solutions [74], [75, Remark 6.2]. Additionally, while this is not depicted on the figure, we note that for  $\sigma = 0$  all methods get exact recovery up to machine precision. EM outperforms the best invariant features approaches by a factor of 3, at the cost of being significantly slower. For large  $M$ , the best invariant features approaches are faster than EM by a factor of 25. Note, however, that for  $M$  up to about 300, EM is faster than the other algorithms. For invariant features approaches (aside from the SDP), almost all of the time is spent computing the bispectrum estimator, while inverting the bispectrum is relatively cheap.

Figures VII.3 and VII.4 show the recovery error and computation time as a function of the noise level  $\sigma$  with  $M = 10,000$  observations. Surprisingly, for high noise level  $\sigma \gtrsim 3$ , the invariant features algorithms outperform EM.

## VIII. Conclusions and perspective

The goal of this paper is twofold. First, we have suggested a new approach for the MRA problem based on features that are invariant under translations. This technique enables us to deal with any noise level as long as we have access to enough measurements and particularly it achieves the sample complexity of MRA. The invariant features approach has low computational complexity and it requires less memory with respect to alternative methods, such as EM. If one wants to have a highly accurate solution, it can therefore be used to initialize EM.

A main ingredient of the invariant features approach is estimating the signal's Fourier phases by inverting the bispectrum. Hence, the second goal of this paper was to study algorithms for bispectrum inversion. We have proposed a few algorithms for this task. In the presence of noise, the non-convex algorithms on the manifold of phases, namely, Algorithms 2 and 3, perform the best. Empirically, these algorithms have a remarkable property: despite their non-convex landscape, they appear to converge to the target signal from random initialization. We provide some analysis for Algorithm 2 but this phenomenon is not well understood.

Our chief motivation for this work comes from the more involved problem of cryo-EM. In cryo-EM, a 3D object is estimated from its 2D projections at unknown rotations in a low SNR environment. One line of research for the object recovery is based on first estimating the unknown rotations [76], [77], [78]. However, the rotation estimation is performed in a very noisy environment and therefore might be inaccurate. An interesting question is to examine whether the 3D object can be estimated directly from the acquired data using features that are invariant under the unknown viewing directions [79].

## Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

## Acknowledgments

The authors were partially supported by Award Number R01GM090200 from the NIGMS, FA9550-17-1-0291 from AFOSR, Simons Investigator Award and Simons Collaboration on Algorithms and Geometry from Simons Foundation, and the Moore Foundation Data-Driven Discovery Investigator Award. NB is partially supported by NSF grant DMS-1719558. ZZ is partially supported by National Center for Supercomputing Applications Faculty Fellowship and University of Illinois at Urbana-Champaign College of Engineering Strategic Research Initiative.

The authors are grateful to Afonso Bandeira, Roy Lederman, William Leeb, Nir Sharon and Susannah Shoemaker for many insightful discussions. We also thank the reviewers for their useful comments, and particularly the anonymous reviewer who proposed a simpler proof for Lemma IV.2.

## References

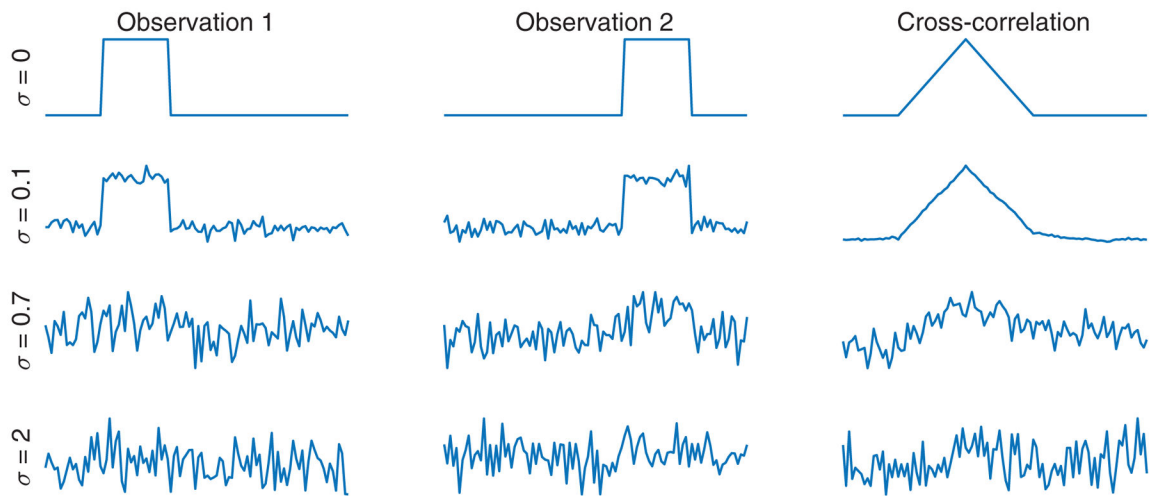
1. Diamond R. On the multiple simultaneous superposition of molecular structures by rigid body transformations. *Protein Science*. 1992; 1(10):1279–1287. [PubMed: 1303746]
2. Theobald DL, Steindel PA. Optimal simultaneous superpositioning of multiple structures with missing data. *Bioinformatics*. 2012; 28(15):1972–1979. [PubMed: 22543369]

3. Park W, Midgett CR, Madden DR, Chirikjian GS. A stochastic kinematic model of class averaging in single-particle electron microscopy. *The International journal of robotics research*. 2011; 30(6): 730–754. [PubMed: 21660125]
4. Park W, Chirikjian GS. An assembly automation approach to alignment of noncircular projections in electron microscopy. *IEEE Transactions on Automation Science and Engineering*. 2014; 11(3):668–679.
5. Zwart JP, van der Heiden R, Gelsema S, Groen F. Fast translation invariant classification of HRR range profiles in a zero phase representation. *IEE Proceedings-Radar, Sonar and Navigation*. 2003; 150(6):411–418.
6. Gil-Pita, R., Rosa-Zurera, M., Jarabo-Amores, P., López-Ferreras, F. Using multilayer perceptrons to align high range resolution radar signals. *International Conference on Artificial Neural Networks*; Springer; 2005. p. 911-916.
7. Dryden, IL., Mardia, KV. *Statistical shape analysis*. Vol. 4. J. Wiley; Chichester: 1998.
8. Foroosh H, Zerubia JB, Berthod M. Extension of phase correlation to subpixel registration. *IEEE transactions on image processing*. 2002; 11(3):188–200. [PubMed: 18244623]
9. Robinson D, Farsiu S, Milanfar P. Optimal registration of aliased images using variable projection with applications to super-resolution. *The Computer Journal*. 2009; 52(1):31–42.
10. Abbe, E., Pereira, J., Singer, A. Sample complexity of the boolean multireference alignment problem,” to appear. *The IEEE International Symposium on Information Theory (ISIT)*; 2017.
11. Bandeira A, Rigollet P, Weed J. Optimal rates of estimation for multi-reference alignment. 2017 arXiv preprint arXiv:1702.08546.
12. Bartesaghi A, Merk A, Banerjee S, Matthies D, Wu X, Milne JL, Subramaniam S. 2.2 Å resolution cryo-EM structure of  $\beta$ -galactosidase in complex with a cell-permeant inhibitor. *Science*. 2015; 348(6239):1147–1151. [PubMed: 25953817]
13. Sirohi D, Chen Z, Sun L, Klose T, Pierson TC, Rossmann MG, Kuhn RJ. The 3.8 Å resolution cryo-EM structure of Zika virus. *Science*. 2016; 352(6284):467–470. [PubMed: 27033547]
14. Frank, J. *Three-dimensional electron microscopy of macromolecular assemblies: visualization of biological molecules in their native state*. Oxford University Press; 2006.
15. van Heel M, Gowen B, Matadeen R, Orlova EV, Finn R, Pape T, Cohen D, Stark H, Schmidt R, Schatz M, et al. Single-particle electron cryo-microscopy: towards atomic resolution. *Quarterly reviews of biophysics*. 2000; 33(04):307–369. [PubMed: 11233408]
16. Singer A. Angular synchronization by eigenvectors and semidefinite programming. *Applied and computational harmonic analysis*. 2011; 30(1):20–36. [PubMed: 21179593]
17. Boumal N. Nonconvex phase synchronization. *SIAM Journal on Optimization*. 2016; 26(4):2355–2377.
18. Perry A, Wein AS, Bandeira AS, Moitra A. Message-passing algorithms for synchronization problems over compact groups. 2016 arXiv preprint arXiv:1610.04583.
19. Chen Y, Candes E. The projected power method: An efficient algorithm for joint alignment from pairwise differences. 2016 arXiv preprint arXiv:1609.05820.
20. Bandeira AS, Boumal N, Singer A. Tightness of the maximum likelihood semidefinite relaxation for angular synchronization. *Mathematical Programming*. 2017; 163(1):145–167.
21. Zhong Y, Boumal N. Near-optimal bounds for phase synchronization. 2017 arXiv preprint arXiv: 1703.06605.
22. Bandeira, AS., Charikar, M., Singer, A., Zhu, A. Multireference alignment using semidefinite programming. *Proceedings of the 5th conference on Innovations in theoretical computer science*; ACM; 2014. p. 459-470.
23. Chen, Y., Guibas, L., Huang, Q. Near-optimal joint object matching via convex relaxation. *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*; 2014. p. 100-108.
24. Kosir, P., DeWall, R., Mitchell, RA. A multiple measurement approach for feature alignment. *Aerospace and Electronics Conference, 1995. NAECON 1995., Proceedings of the IEEE 1995 National*; IEEE; 1995. p. 94-101.

25. Aguerrebere C, Delbracio M, Bartesaghi A, Sapiro G. Fundamental limits in multi-image alignment. *IEEE Transactions on Signal Processing*. 2016; 64(21):5707–5722.
26. Robinson D, Milanfar P. Fundamental performance limits in image registration. *IEEE Transactions on Image Processing*. 2004; 13(9):1185–1199. [PubMed: 15449581]
27. Weiss A, Weinstein E. Fundamental limitations in passive time delay estimation–Part I: Narrow-band systems. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. 1983; 31(2):472–486.
28. Weinstein E, Weiss A. Fundamental limitations in passive time-delay estimation–Part II: Wide-band systems. *IEEE transactions on acoustics, speech, and signal processing*. 1984; 32(5):1064–1078.
29. Perry A, Weed J, Bandeira A, Rigollet P, Singer A. The sample complexity of multi-reference alignment. 2017 arXiv preprint arXiv:1707.00943.
30. Brockett PL, Hinich MJ, Patterson D. Bispectral-based tests for the detection of gaussianity and linearity in time series. *Journal of the American Statistical Association*. 1988; 83(403):657–664.
31. Bartolo N, Komatsu E, Matarrese S, Riotto A. Non-gaussianity from inflation: theory and observations. *Physics Reports*. 2004; 402(3):103–266.
32. Luo X. The angular bispectrum of the cosmic microwave background. *The Astrophysical journal*. 1994; 427(2):L71–L71.
33. Wang L, Kamionkowski M. Cosmic microwave background bispectrum and inflation. *Physical Review D*. 2000; 61(6):063504.
34. Matsuoka T, Ulrych TJ. Phase estimation using the bispectrum. *Proceedings of the IEEE*. 1984; 72(10):1403–1411.
35. Chang MM, Tekalp AM, Erdem AT. Blur identification using the bispectrum. *IEEE transactions on signal processing*. 1991; 39(10):2323–2325.
36. Chen, T-w, Jin, W-d, Li, J. Feature extraction using surrounding-line integral bispectrum for radar emitter signal. 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence); IEEE; 2008. p. 294-298.
37. Ning T, Bronzino JD. Bispectral analysis of the rat EEG during various vigilance states. *IEEE Transactions on Biomedical Engineering*. 1989; 36(4):497–499. [PubMed: 2714832]
38. Chen B, Petropulu AP. Frequency domain blind mimo system identification based on second-and higher order statistics. *IEEE Transactions on Signal Processing*. 2001; 49(8):1677–1688.
39. Zhao Z, Singer A. Rotationally invariant image representation for viewing direction classification in cryo-EM. *Journal of structural biology*. 2014; 186(1):153–166. [PubMed: 24631969]
40. Mendel JM. Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications. *Proceedings of the IEEE*. 1991; 79(3):278–305.
41. Nikias CL, Raghuveer MR. Bispectrum estimation: A digital signal processing framework. *Proceedings of the IEEE*. 1987; 75(7):869–891.
42. Marabini R, Carazo JM. Practical issues on invariant image averaging using the bispectrum. *Signal processing*. 1994; 40(2–3):119–128.
43. Marabini R, Carazo JM. On a new computationally fast image invariant based on bispectral projections. *Pattern recognition letters*. 1996; 17(9):959–967.
44. Petropulu AP, Pozidis H. Phase reconstruction from bispectrum slices. *IEEE Transactions on Signal Processing*. 1998; 46(2):527–530.
45. Giannakis GB. Signal reconstruction from multiple correlations: frequency-and time-domain approaches. *JOSA A*. 1989; 6(5):682–697.
46. Sadler BM, Giannakis GB. Shift-and rotation-invariant object reconstruction using the bispectrum. *JOSA A*. 1992; 9(1):57–69.
47. Swami A, Giannakis G, Mendel J. Linear modeling of multidimensional non-gaussian processes using cumulants. *Multidimensional Systems and Signal Processing*. 1990; 1(1):11–37.
48. Fienup JR. Phase retrieval algorithms: a comparison. *Applied optics*. 1982; 21(15):2758–2769. [PubMed: 20396114]

49. Shechtman Y, Eldar YC, Cohen O, Chapman HN, Miao J, Segev M. Phase retrieval with application to optical imaging: a contemporary overview. *IEEE signal processing magazine*. 2015; 32(3):87–109.
50. Jaganathan, K., Oymak, S., Hassibi, B. Sparse phase retrieval: Convex algorithms and limitations. *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on; IEEE; 2013*. p. 1022-1026.
51. Bendory T, Eldar YC, Boumal N. Non-convex phase retrieval from stft measurements. *IEEE Transactions on Information Theory*. 2017
52. Beinert R, Plonka G. Ambiguities in one-dimensional discrete phase retrieval from fourier magnitudes. *Journal of Fourier Analysis and Applications*. 2015; 21(6):1169–1198.
53. Bendory T, Beinert R, Eldar YC. Fourier phase retrieval: Uniqueness and algorithms. 2017 arXiv preprint arXiv:1705.09590.
54. Bendory T, Edidin D, Eldar YC. On signal reconstruction from FROG measurements. 2017 arXiv preprint arXiv:1706.08494.
55. Tukey, JW. The spectral representation and transformation properties of the higher moments of stationary time series. In: Brillinger, DR., editor. *The Collected Works of John W. Tukey*. 1. Vol. ch 4. Wadsworth; 1984. p. 165-184.
56. Yellott JI, Iverson GJ. Uniqueness properties of higher-order autocorrelation functions. *JOSA A*. 1992; 9(3):388–404.
57. Kondor R. A novel set of rotationally and translationally invariant features for images based on the non-commutative bispectrum. 2007 arXiv preprint cs/0701127.
58. Kakarala R. Completeness of bispectrum on compact groups. 2009; 1 arXiv preprint arXiv:0902.0196.
59. Kakarala R. The bispectrum as a source of phase-sensitive invariants for fourier descriptors: a group-theoretic approach. *Journal of Mathematical Imaging and Vision*. 2012; 44(3):341–353.
60. Kakarala, R. Bispectrum on finite groups. 2009 *IEEE International Conference on Acoustics, Speech and Signal Processing; IEEE; 2009*. p. 3293-3296.
61. Devroye L, Lerasle M, Lugosi G, Oliveira RI, et al. Sub-gaussian mean estimators. *The Annals of Statistics*. 2016; 44(6):2695–2725.
62. Absil PA, Baker CG, Gallivan KA. Trust-region methods on Riemannian manifolds. *Foundations of Computational Mathematics*. 2007; 7(3):303–330.
63. Boumal N, Mishra B, Absil P-A, Sepulchre R. Manopt, a Matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*. 2014; 15:1455–1459.
64. Boumal N, Absil P-A, Cartis C. Global rates of convergence for nonconvex optimization on manifolds. 2016 arXiv preprint arXiv:1605.08101.
65. Grant M, Boyd S, Ye Y. *CVX: Matlab software for disciplined convex programming*. 2008
66. Bendory T, Sidorenko P, Eldar YC. On the uniqueness of frog methods. *IEEE Signal Processing Letters*. 2017; 24(5):722–726.
67. Marron J, Sanchez P, Sullivan R. Unwrapping algorithm for leastsquares phase recovery from the modulo  $2\pi$  bispectrum phase. *JOSA A*. 1990; 7(1):14–20.
68. Lenstra AK, Lenstra HW, Lovász L. Factoring polynomials with rational coefficients. *Mathematische Annalen*. 1982; 261(4):515–534.
69. Galbraith, SD. *Mathematics of public key cryptography*. Cambridge University Press; 2012.
70. Chang XW, Zhou T. Miles: Matlab package for solving mixed integer least squares problems. *GPS Solutions*. 2007; 11(4):289–294.
71. Absil, P-A., Mahony, R., Sepulchre, R. *Optimization algorithms on matrix manifolds*. Princeton University Press; 2009.
72. Yang WH, Zhang LH, Song R. Optimality conditions for the nonlinear programming problems on Riemannian manifolds. *Pacific Journal of Optimization*. 2014; 10(2):415–434.
73. Dempster A, Laird N, Rubin D. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society Series B (methodological)*. 1977; 39(1):1–38.
74. Pataki G. On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues. *Mathematics of operations research*. 1998; 23(2):339–358.

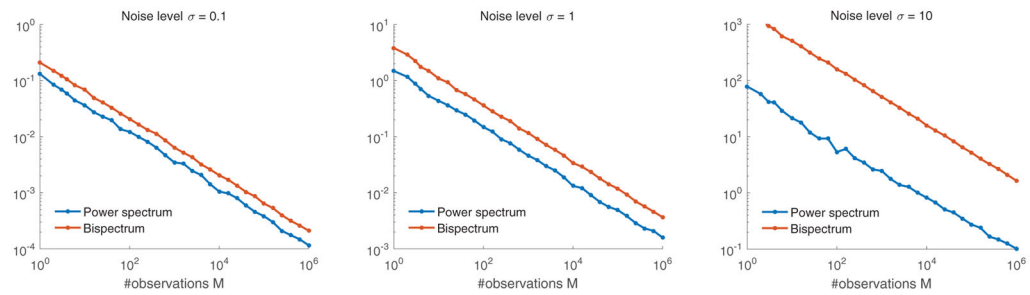
75. Boumal N. A Riemannian low-rank method for optimization over semidefinite matrices with block-diagonal constraints. 2015 arXiv preprint arXiv:1506.00575v2.
76. Singer A, Shkolnisky Y. Three-dimensional structure determination from common lines in cryo-EM by eigenvectors and semidefinite programming. *SIAM journal on imaging sciences*. 2011; 4(2):543–572. [PubMed: 22536457]
77. Shkolnisky Y, Singer A. Viewing direction estimation in cryo-EM using synchronization. *SIAM journal on imaging sciences*. 2012; 5(3):1088–1110.
78. Wang L, Singer A, Wen Z. Orientation determination of cryo-EM images using least unsquared deviations. *SIAM journal on imaging sciences*. 2013; 6(4):2450–2483. [PubMed: 24683433]
79. Kam Z. The reconstruction of structure from electron micrographs of randomly oriented particles. *Journal of Theoretical Biology*. 1980; 82(1):15–39. [PubMed: 7401655]



**Figure I.1.**

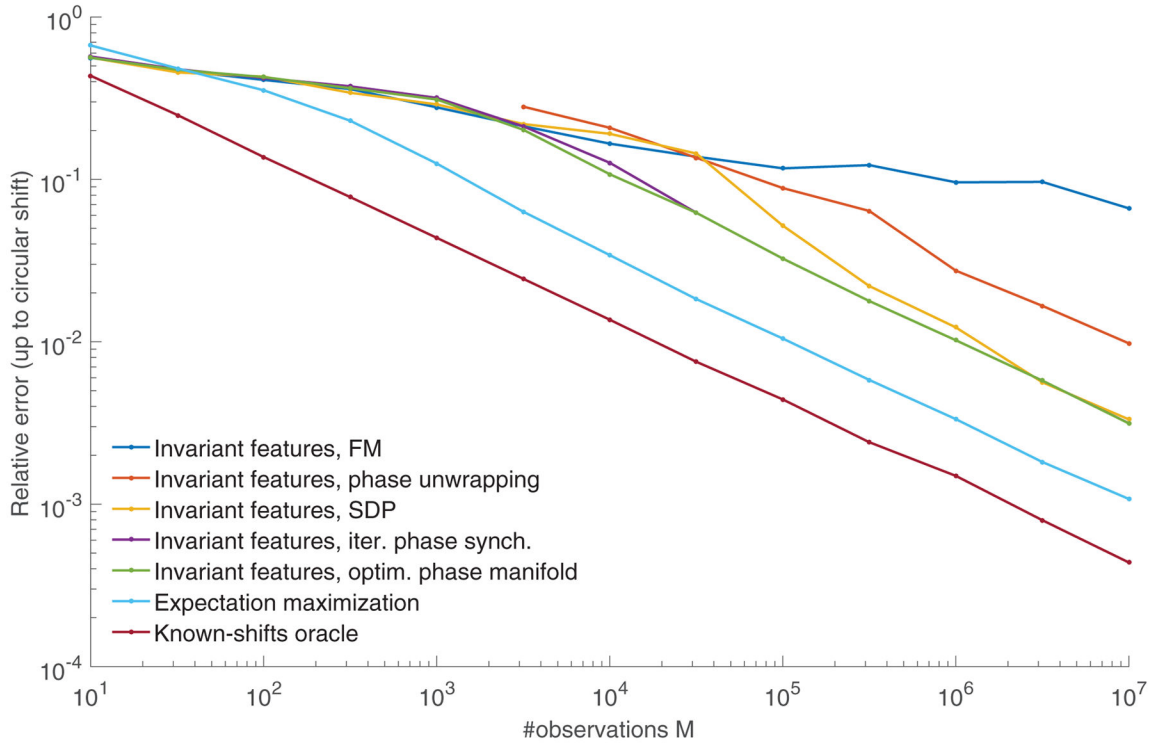
Alignment of two translated versions of the same signal in the presence of i.i.d. Gaussian noise with various standard deviations  $\sigma$ . The true signal in  $\mathbb{R}^{100}$  is a window of length 22 and height 1. Each row presents two observations and their cross-correlation. Importantly, beyond a certain threshold, noise makes pairwise alignment impossible.





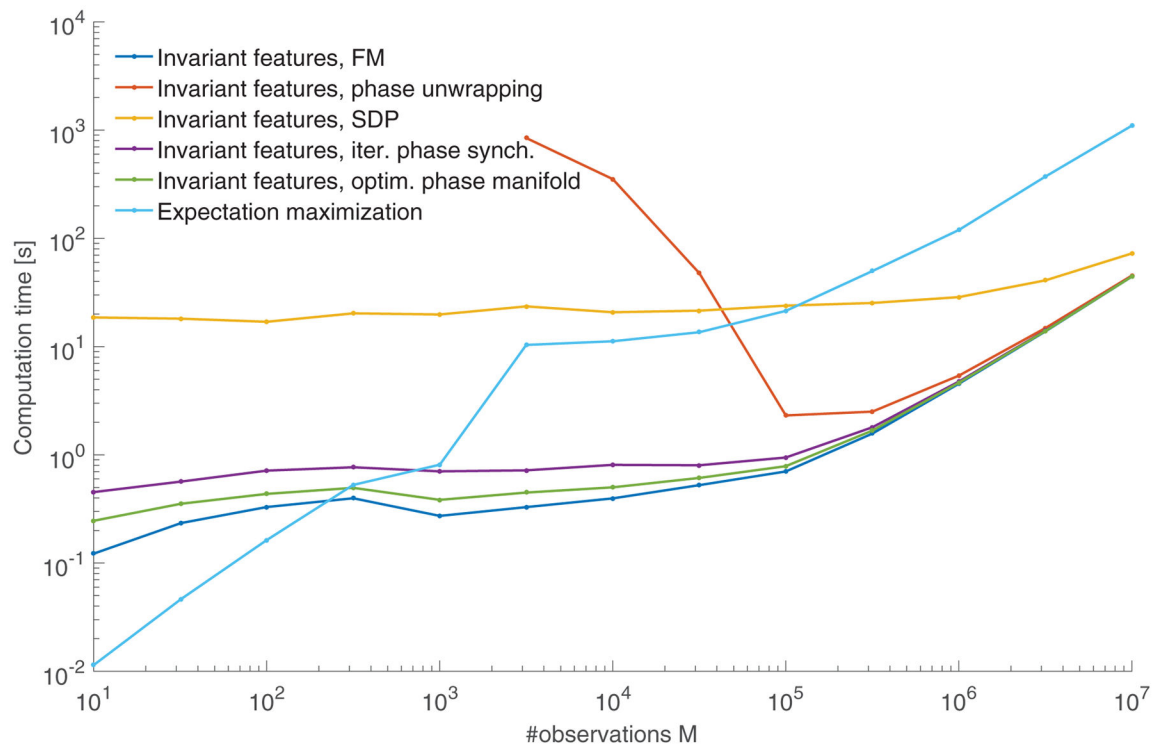
**Figure II.1.**

Relative error of estimating the power spectrum and bispectrum for different noise levels as a function of the number of observations  $M$ . Results are averaged over 10 repetitions for each value of  $M$  on a fixed real signal of length  $N=41$  with i.i.d. normal random entries. The signal-to-noise ratio is then  $1/\sigma^2$ . Importantly, the relative error decreases as  $1/\sqrt{M}$  regardless of noise level.

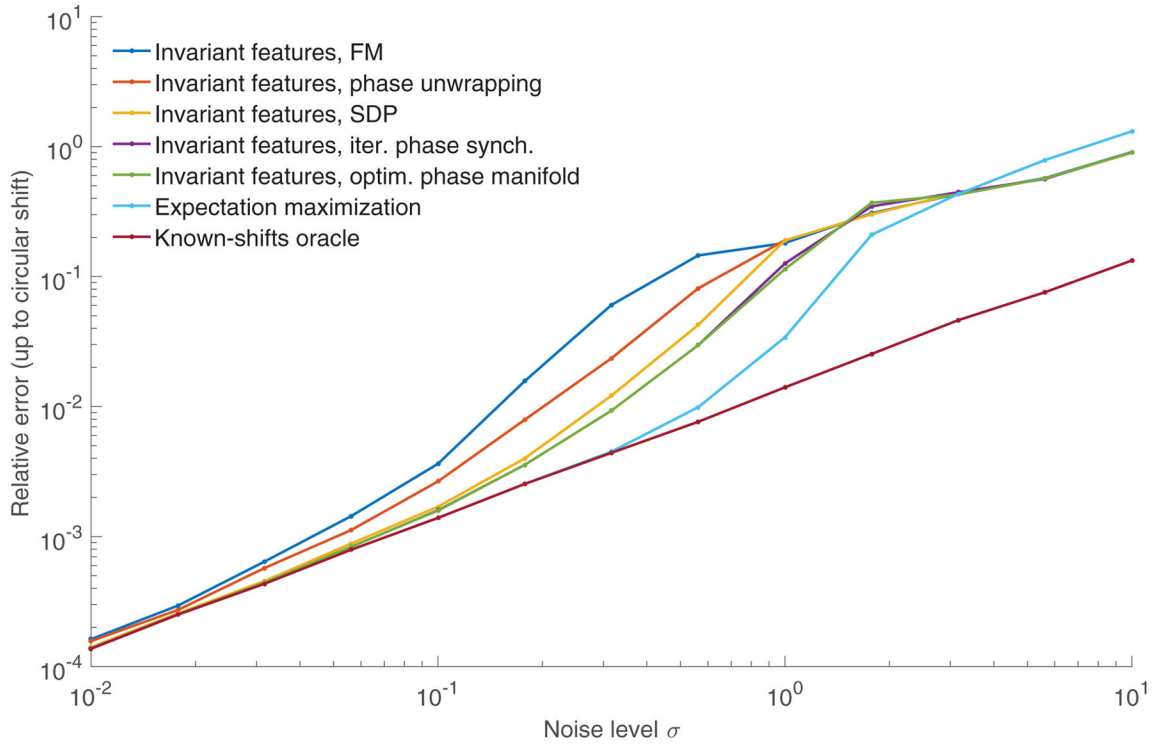


**Figure VII.1.**

Relative recovery error for the signal  $x$  as a function of the number of observations  $M$  for fixed noise level  $\sigma = 1$ . The curves corresponding to the optim. phase manifold (Algorithm 2) and the iter. phase synch. (Algorithm 3) overlap.

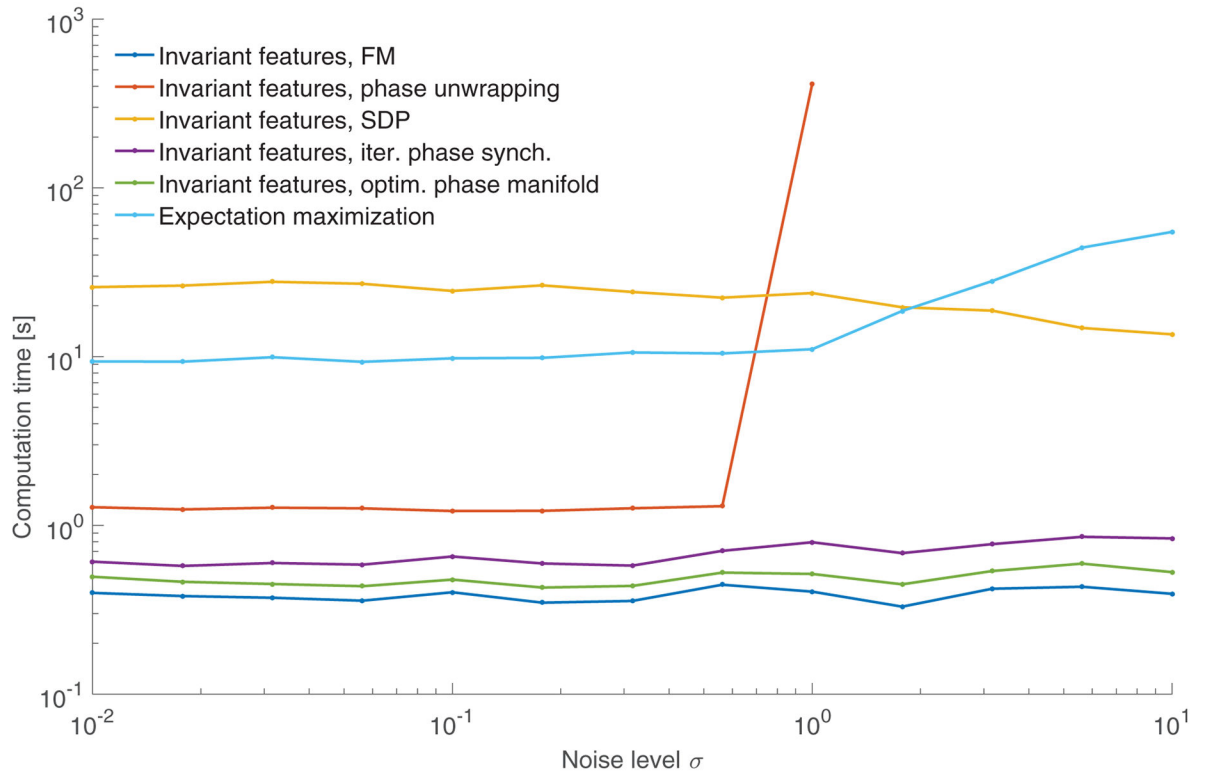


**Figure VII.2.**  
Average computation times corresponding to Figure VII.1.



**Figure VII.3.**

Relative recovery error for the signal  $x$  as a function of the noise level  $\sigma$  with  $M=10,000$  observations. The curves corresponding to the optim. phase manifold (Algorithm 2) and the iter. phase synch. (Algorithm 3) overlap.



**Figure VII.4.**  
Average computation times corresponding to Figure VII.3.

**Table I**

Comparison of main MRA approaches.  $\mathcal{F}(\mathcal{N})$  denotes the complexity of inverting the bispectrum. For instance, for the FM algorithm,  $\mathcal{F}(\mathcal{N}) = \mathcal{O}(\mathcal{N}^2)$ . Storage requirements include the possibility of streaming computations where possible.

Method	Computational complexity	Storage requirement	Comments
Template alignment	$\mathcal{O}(MN \log N)$	$\mathcal{O}(N)$	Fails at moderate SNR (see Figure I.1)
Angular synchronization	$\mathcal{O}(M^2 N \log N)$	$\mathcal{O}(M^2)$	Fails at low SNR
Expectation maximization	$\mathcal{O}(TMN \log N)$	$\mathcal{O}(MN)$	Empirically accurate; #iterations $T$ grows with noise level
Invariant features (this paper)	$\mathcal{O}(MN^2 + \mathcal{F}(\mathcal{N}))$	$\mathcal{O}(N^2)$	Under mild conditions, accurate estimation if $M$ grows as $\sigma^6$