SUPER: Sparse signals with Unknown Phases Efficiently Recovered

Sheng Cai, Mayank Bakshi, Sidharth Jaggi, Minghua Chen
{cs010,mayank,jaggi,minghua}@ie.cuhk.edu.hk
The Chinese University of Hong Kong, Hong Kong SAR China

***Abstract*—Compressive phase retrieval algorithms attempt to reconstruct a "sparse high-dimensional vector" from its "low-dimensional intensity measurements". Suppose $\mathbf{x}$ is any length-n *input vector* over $\mathbb{C}$ with exactly k non-zero entries, and A is an $m \times n$ ($k < m \ll n$) *phase measurement matrix* over $\mathbb{C}$. The decoder is handed m "intensity measurements" ($|A_1\mathbf{x}|, \ldots, |A_m\mathbf{x}|$) (corresponding to component-wise absolute values of the linear measurement $A\mathbf{x}$) – here A_i's correspond to the rows of the measurement matrix A. In this work, we present a class of measurement matrices A, and a corresponding decoding algorithm that we call SUPER, which can reconstruct $\mathbf{x}$ up to a global phase from intensity measurements. The SUPER algorithm is the first to simultaneously have the following properties: (a) it requires only $\mathcal{O}(k)$ (order-optimal) measurements, (b) the computational complexity of decoding is $\mathcal{O}(k \log k)$ (near order-optimal) arithmetic operations, (c) it succeeds with high probability over the design of A. Our results hold for all $k \in \{1, 2, \ldots, n\}$.**

I. Introduction

Phase Retrieval: In many applications, it's difficult to measure the phase information of the underlying signal. Instead, we recover the signal by its intensity measurements. For instance, in X-ray crystallography, optics [1] and image reconstruction for astronomy [2], signal/image is reconstructed from the intensity measurements of its Fourier transform.

Let $A \in \mathbb{C}^{m\times n}$ be used to denote the *phase measurement matrix*, and $\mathbf{x} \in \mathbb{C}^n$ be used to denote the unknown underlying signal. Instead of *linear* measurements of the form $y = A\mathbf{x}$ as in the *compressive sensing* literature (see, for instance, [3]) in the *phase retrieval problem* we have m *non-linear intensity measurements* of the form $b_i = |A_i\mathbf{x}|$. Here the index i is an integer in $\{1, \ldots, m\}$ (or [m] for short), A_i is the i-th row of phase measurement matrix A and $|\cdot|$ is the absolute value.

Problems of this kind have been studied over the last decades. A good survey of some of the algorithms via non-convex process can be found in [4]. Recently, two convex optimization methods, PhaseLift [5] and PhaseCut [6], have been proposed by Candès *et al.* and Waldspurger *et al.*. PhaseLift is able to reconstruct $\mathbf{x}$ with $\mathcal{O}(n \log n)$ intensity measurements by solving semidefinite programming with high probability. The A_i's are independently sampled on the unit sphere of $\mathbb{C}^n$. Later, it's shown that the number of intensity measurements can be improved to $\mathcal{O}(n)$ where A_i's are independently and identically distributed with the uniform distribution on the sphere of radius $\sqrt{n}$, or the complex normal distribution [7]. PhaseCut is inspired by solving max-cut problem via SDP. The decoding complexity for both PhaseLift and PhaseCut is $\mathcal{O}\left(n^3\right)$, which is still computationally costly when n is large.

Besides SDP-based approach, more computationally efficient algorithms are proposed such as [8], [9]. For instance, in [9], the number of intensity measurements required is $\mathcal{O}\left(n \log^3 n\right)$. However, the decoding complexity is $\mathcal{O}\left(n^2 \log^3 n\right)$ which is less than that of SDP-based approach.

Compressive Phase Retrieval: Suppose $\mathbf{x}$ is "sparse", *i.e.*, the number of non-zero components of $\mathbf{x}$ is at most k, which is much less than the length n of $\mathbf{x}$. This assumption is not uncommon in many applications like X-ray crystallography. Then, given A and b, the goal of *compressive phrase retrieval* is to reconstruct $\mathbf{x}$ as $\hat{\mathbf{x}}$, where $\hat{\mathbf{x}}$ equals $\mathbf{x}$ up to a global phase. That is, $\hat{\mathbf{x}} = \mathbf{x}e^{\iota\Theta}$ for some arbitrary fixed $\Theta \in [0, 2\pi)$. Here ι denotes the positive square root of -1. The reason we allow this degeneracy in $\hat{\mathbf{x}}$, up to a global phase factor, is that all such $\hat{\mathbf{x}}$'s result in the same measurement vector under intensity measurements. If $\hat{\mathbf{x}}$ does indeed equal $\mathbf{x}$ up to a global phase, then we denote this "equality" as $\hat{\mathbf{x}} \hat{=} \mathbf{x}$.

It is shown that $4k - 1$ intensity measurements suffice to uniquely reconstruct $\mathbf{x}$ in [10] (for $\mathbf{x} \in \mathbb{R}^n$) and [11] (for $\mathbf{x} \in \mathbb{C}^n$). However, no efficient algorithms is given. The ℓ_1-regularized PhaseLift method is introduced in the compressive phase retrieval problem in [12]. In [13], it is shown that if the number of Gaussian intensity measurements is $\mathcal{O}\left(k^2 \log n\right)$, $\mathbf{x}$ can be correctly reconstructed via ℓ_1-regularized PhaseLift.

The works in [14] and the works by Jaganathan *et al.* [15], [16], [17] study the case when the phase measurement matrix is a Fourier transform matrix. In [18], it is explained that SDP-based methods can reconstruct $\mathbf{x}$ with sparsity up to $o\left(\sqrt{n}\right)$. In [16], the algorithm based on reweighted ℓ_1-minimization with $\mathcal{O}\left(k^2 \log n\right)$ phaseless Fourier measurements is proposed to go beyond this bottleneck. When the phase measurement matrix is allowed to be designed, a combinatorial algorithm is proposed in [16] such that $\mathbf{x}$ is correctly reconstructed with $\mathcal{O}(k \log n)$ intensity measurements in $\mathcal{O}(kn \log n)$ time.

To our best knowledge, in the literature, there is no construction of a measurement matrix A and a corresponding reconstruction algorithm that correctly reconstructs $\mathbf{x}$ with an order-optimal number of measurements and with near-optimal decoding complexity simultaneously.

A. Our Contribution

In this work[1], we describe a randomized design of the phase measurement matrix A and a corresponding decoding algorithm achieving the following guarantees:

Theorem 1. *(Main theorem) There exists a measurement ensemble $\{A\}$ and a corresponding decoding algorithm for*

The work described in this paper was supported by a grant from University Grants Committee of the Hong Kong Special Administrative Region, China (Project No. AoE/E-02/08).

[1]While in this work we focus on the "sparse regime", $k = o(n)$, our techniques work for all $k \in \{1, 2, \ldots, n\}$. If $k = \omega(1)$, our algorithm has the same performance stated in Theorem 1. If k is a constant and error probability of our algorithm is P_e, then the number of measurements required is $f(P_e)k$ for some function f. We refer the reader to [19] for details.

compressive phase retrieval with the following performance:

1) For every $\mathbf{x} \in \mathbb{C}^n$, with probability $1 - o(1)$ over the randomized design of A, the algorithm exactly reconstructs $\mathbf{x}$ up to a global phase;

2) The number of measurements $m = \mathcal{O}(k)$;

3) The decoding complexity is $\mathcal{O}(k \log k)$.

II. Overview/High-level Intuition

A. Pieces of the puzzle

We first define some useful terminology.

Singletons: If a measurement b_i involves only a single non-zero component of $\mathbf{x}$, then we say that such a measurement is a *singleton*.[2] Singletons are important since they can be used to pin down the magnitude (though not the phase) of components of $\mathbf{x}$. There are several challenges, however. One lies in even identifying whether a measurement is a singleton or not. The second lies in identifying which of the $\mathbf{x}$ components being measured in b_i corresponds to the singleton. The third is to be able to do all this blindingly fast, in fact in *constant* time (independent of n and k!). Each of these challenges can be handled by using ideas from the our prior work on compressive sensing [21]. For details, see Sections IV and V below.

Doubletons: Similarly, if a measurement b_i involves exactly two non-zero components of $\mathbf{x}$, then we say that such a measurement is a *doubleton*. Doubletons, especially doubletons measuring two non-zero components of $\mathbf{x}$ which have already been measured by singletons (we call such doubletons *resolvable doubletons*), are useful since they can be used to deduce the relative phases of the two non-zero components of $\mathbf{x}$. For example, if one is given the magnitudes $|x_i|$, $|x_j|$, and $|x_i + x_j|$, then one can determine the angle θ between the phases of the complex numbers x_i and x_j (up to degeneracy of sign of θ). In fact, even this degeneracy can be resolved by an additional judiciously chosen measurement. Similar challenges to those mentioned above vis-a-vis singletons (identifying whether or not a measurement is a doubleton/resolvable doubleton, identifying which components of $\mathbf{x}$ it corresponds to, and doing so in constant time) also hold for doubletons. See Sections IV and V for details.

Mutual resolvability: We say our decoding algorithm has thus far *mutually resolved* two non-zero components x_i and x_j of $\mathbf{x}$ if the magnitudes of both x_i and x_j have been deduced, and also the relative phase between x_i and x_j has been deduced (for instance via resolvable doubleton measurements roughly described above). Note that mutual resolvability is an equivalence relation – it is reflexive, symmetric and transitive. Note therefore that if x_i and $x_{i'}$ have been mutually resolved, it is not necessary that they even are involved in the same measurement; it is sufficient that x_i and $x_{i'}$ are part of a chain of non-zero components of $\mathbf{x}$ that are pairwise mutually resolved. Finally, we note that as our decoding algorithm progresses, if it is successful, in fact *all* the non-zero components of $\mathbf{x}$ are eventually mutually resolved. Hence this property of mutual resolvability is perhaps most interesting in the intermediate stages of our decoding algorithm.

Giant component: We say that a subset of the non-zero components of $\mathbf{x}$ form a giant component if it is the largest subset satisfying the two properties: 1) The subset is of size linear in k. 2) Any pair of components in the subset have been mutually resolved (thus far) by the decoding algorithm.

[2]We borrow this terminology (of singletons, doubletons, multitons, etc) from the compressive sensing work of Pawar *et al* [20].

Essentially, our algorithm proceeds by iteratively enlarging the giant component until it engorges all the non-zero components of $\mathbf{x}$.

Resolvable multiton: We say that a measurement b_i is a resolvable multiton if it is the case that exactly one (say x_i) of the non-zero components of $\mathbf{x}$ involved in the measurement b_i is outside the giant component, and at least one of non-zero components of $\mathbf{x}$ is inside the giant component. Such measurements are useful since, in the latter parts of our algorithm, there are not enough resolvable doubletons. By carefully choosing the parameters of the algorithm, one can guarantee that a constant fraction of measurements are resolvable mutitons.

Judiciously designed measurements (see Section IV) enable one to mutually resolve the component x_i that is outside the giant component, with the components of $\mathbf{x}$ inside the giant component, by solving a quadratic equation. Care is indeed required in choosing the measurements since the amplitude measurement process is inherently non-linear, and there may not be a "clean" manner to mutually resolve x_i via arbitrary measurements – indeed the design of such a measurement process is also one of the intellectual contributions we wish to highlight in this work. We call this process "cancelling out" the already resolved components of $\mathbf{x}$.

B. Putting the pieces together

Seeding phase: In the first phase, called the *seeding phase*, there are $\mathcal{O}(k)$ "sparse" measurements (each measurement involves, in expectation, $\mathcal{O}(n/k)$ components of $\mathbf{x}$). We demonstrate that by first examining the measurements corresponding to this phase, the decoding algorithm is already able to decode a constant fraction (say $1/2$)[3] of the components of $\mathbf{x}$ up to a global phase. The algorithm is able to do this since we are able to show that a "significant" fraction of measurements are singletons and resolvable doubletons. Standard results in percolation theory [22] then lead one to conclude that the number of non-zero nodes that are mutually resolvable is linear in k, *i.e.*, that there is a giant component. Hence this phase is called the "seeding" phase, since the giant component forms the nucleus on which the remainder of the algorithm builds upon.

The key technique used in our work is to segue to a different sampling process (outlined below) than "coupon collection" process [23] (wherein one has to collect at least one copy of each of k coupons by sampling with replacement) in [16], and using resolvable multitons rather than doubletons. The challenge is to make the numbers work, not only do we require only $\mathcal{O}(k)$ measurements, but we also require our decoding complexity to be $\mathcal{O}(k \log(k))$[4].

Geometric-decay phase: This phase itself comprises of $\mathcal{O}(\log(\log(k)))$ separate stages. Each stage has half the number of measurements compared to the previous stage, but measurements in each stage are twice as "dense" as the measurements in the previous stage. So, for instance, if in the first stage of the geometric-decay phase, there are say ck measurements, with each measurement involving n/k components of $\mathbf{x}$, then in the second stage of the geometric-decay phase, there are $ck/2$ measurements, but each measurement involves $2n/k$ components of $\mathbf{x}$.

[3]Here, $1/2$ is arbitrarily chosen to simplify the presentation of intuition. The actual fraction of resolved non-zero components in the seeding phase is different from $1/2$. See Section VII for details. Here, the parameter $1/2$ for the geometric-decay phase in this section is due to the same reason.

[4]In this section, we focus on the number of measurements and decoding complexity. For the error probability, please refer to Section VII.

There are two reasons for this choice of parameters. Firstly, with such a geometric decay in the number of measurements in each stage, the overall number of measurements in the geometric-decay phase is still $\mathcal{O}(k)$. Secondly, we show that with the geometric increase in the density of measurements, a significant fraction of measurements in each stage lead to resolvable multitons, and use this to show that the number of unresolved components decays geometrically.

The reason we run the geometric-decay phase for only $\mathcal{O}(\log(\log(k)))$ stages is also two-fold. Firstly, after that many stages, with the number of unresolved components halving at every stage, the number of unresolved components of $\mathbf{x}$ is, in expectation, $\mathcal{O}(k/\log(k))$. Hence the concentration inequalities (which depend on the number of unresolved components) we use to control the probabilities of error get progressively weaker (though they still result in good concentration at the last stage of the geometric-decay phase). Secondly, and more importantly, the number of non-zero components in each resolvable multiton increases geometrically as the number of stages increases. This has implications for the time-complexity of the decoding algorithm, since the time-complexity depends directly on the number of non-zero components in each measurement that need to be "cancelled out". By terminating the geometric-decay phase after $\mathcal{O}(\log(\log(k)))$ stages ensures that, in expectation, the number of such "cancellations" is at most $\mathcal{O}(\log(k))$, and hence the overall time-complexity of the algorithm scales as $\mathcal{O}(k \cdot \log(k))$.

Cleaning-up phase: Finally, we segue to what we call the "cleaning-up" phase. As noted above, after the geometric-decay phase the number of unresolved components of $\mathbf{x}$ is, in expectation, $k' \triangleq \mathcal{O}(k/\log(k))$. To fit our budget of $\mathcal{O}(k)$ measurements, and $\mathcal{O}(k\log(k))$ decoding time, we now segue to using "coupon collection" as a primitive. This may be viewed as restarting the seeding (first) phase, but with different parameters. In particular, the problem dimension has now been significantly reduced (since there are now only k' unresolved components of $\mathbf{x}$). Therefore we can now afford to pay the coupon collection penalty that we avoided in the seeding phase by moving to the geometric-decay phase.

Specifically, in this cleaning-up phase we take $\mathcal{O}(k'\log(k'))$ measurements so as to resolve the remaining k' unresolved components of $\mathbf{x}$. Note that $\mathcal{O}(k'\log(k'))$ scales as $\mathcal{O}(k)$. Each measurement we take has the same density as the measurements in the last stage of the geometric decay phase, and hence the time-complexity of resolving measurements also scales in the same manner. However, since there are many more measurements than in the last stage of the geometric-decay phase, by standard arguments corresponding to the coupon collection problem we are able to argue that for each unresolved component of $\mathbf{x}$ there is at least one resolvable multiton that helps resolve it.

III. Graph properties

We construct a series of bipartite graphs with some desirable properties outlined in this section. We then use the structure of the bipartite graphs to generate our measurement matrix A in Section IV and design the corresponding reconstruction algorithm in Section V. Each left nodes of a bipartite graph represents a component of $\mathbf{x}$ and each right node represents a set of intensity measurements. For each edge in a bipartite graph, it is assigned different weights which are discussed in Section IV.

A. Seeding Phase

The properties of the bipartite graph, $\mathcal{G}_I$, in the first phase are as follows: 1) There are ck right nodes, where c is a constant. 2) Each edge in $\mathcal{G}_I$ appears with probability $1/k$. 3) Many singleton nodes: In expectation, a constant fraction of right nodes are singleton nodes. 4) Many resolvable doubleton nodes: In expectation, a constant fraction of right nodes are resolvable doubleton nodes. For properties 4) and 5), see Section VI for details.

Another graph $\mathcal{H}$ is implied by $\mathcal{G}_I$. Each vertex in $\mathcal{H}$ represents a non-zero component of $\mathbf{x}$ and there is an edge in $\mathcal{H}$ if and only if two left nodes involved are mutually resolved. $\mathcal{H}$ has the following property: $\mathcal{H}$ has a giant connected component: The connected component, $\mathcal{H}'$ contains a constant fraction of nodes in $\mathcal{H}$. This property is formally stated in Section VI.

B. Geometric-decay phase

There are $L = \mathcal{O}(\log\log k)$ separate bipartite graphs/stages in this phase. The properties of the l-th bipartite graph, $\mathcal{G}_{II,l}$ ($l = 1, 2, \ldots, L$), are as follows: 1) There are $cf_{II,l-1}k$ right nodes, where $f_{II,l-1}$ is the expected fraction of unresolved non-zero components of $\mathbf{x}$ after the $(l-1)$-th stage of decoding process in the second phase. $f_{II,0} = f_I$ is the expected fraction of unresolved non-zero components after seeding phase. The value of $f_{II,l}$ is discussed in Section VI. 2) Each edges in $\mathcal{G}_{II,l}$ appears with probability $1/\left(f_{II,l-1}k\right)$. 3) Many resolvable multiton nodes: In expectation, a constant fraction of right nodes are resolvable multiton nodes. See Section VI for details.

For a newly resolved non-zero component, the corresponding node in $\mathcal{H}$ is appended to the giant connected component, $\mathcal{H}'$. In expectation, there are $\left(f_{II,l-1} - f_{II,l}\right)k$ non-zero components decoded in the l-th stage of decoding. After $\mathcal{O}(\log\log k)$ stages, there are $\mathcal{O}(k/\log k)$ unresolved non-zero components of $\mathbf{x}$ left.

C. Cleaning-up phase

The properties of the bipartite graph, $\mathcal{G}_{III}$, in the last phase are as follows: 1) There are $c\left(k/\log k\right)\log\left(k/\log k\right) = \mathcal{O}(k)$ right nodes. 2) Each edges in $\mathcal{G}_{III}$ appears with probability $\log k/k$. 3) Many resolvable multiton nodes.

In this stage, all the resolved non-zero components of size $\mathcal{O}(k/\log k)$ are finally recovered using resolvable multiton nodes by "Cancelling out" process and a Coupon Collection argument.

IV. Measurement Design

For a bipartite graph $\mathcal{G}$ ($\mathcal{G}$ is one of the $\mathcal{G}_I$, $\mathcal{G}_{II,l}$'s and $\mathcal{G}_{III}$), there are n nodes on the left and $m'_{\mathcal{G}}$ nodes on the right . $A(\mathcal{G})'$ is the dimension-$m'_{\mathcal{G}} \times n$ adjacent matrix of $\mathcal{G}$ where the entry at i-th row and j-th column equals to 1 if and only if i-th right node connects to the j-th left node for $j \in [n]$ and $i \in \left[m'_{\mathcal{G}}\right]$. The dimension-$m_{\mathcal{G}} \times n$ phase measurement matrix $A(\mathcal{G})$ is designed based on $A(\mathcal{G})'$ where $m_{\mathcal{G}} = 5m'_{\mathcal{G}}$. By appending all the matrix $A(\mathcal{G})$ sequentially, we get the actual $m \times n$ measurement matrix A where $m = \Sigma_{\mathcal{G}} m_{\mathcal{G}}$. For i-th row $A(\mathcal{G})'_i$ of $A(\mathcal{G})'$, a set of rows (of size 5) of $A(\mathcal{G})$ are designed for $i \in \left[m'_{\mathcal{G}}\right]$. If the j-th entry of $A(\mathcal{G})'_i$ is zero, then corresponding set of entries of $A(\mathcal{G})$ are all zero for all $j \in [n]$. In the following measurement matrix design, we design the entries corresponding to non-zero entries in $A(\mathcal{G})'$. See Section V for how these measurements are used for decoding.

1) Trigonometric entries: The j-th entries of the $(5i-4)$-th and $(5i-3)$-th rows of $A(\mathcal{G})$ are denoted by $a_{i,j}^{(\mathcal{G},1)}$ and $a_{i,j}^{(\mathcal{G},2)}$.

The values are set to be $\cos(j\pi/(2n))$ and $\iota\sin(j\pi/(2n))$, respectively. Here ι denotes the positive square root of -1 and $\pi/(2n)$ can be treated as the unit phase of the entry design.

2) Structured unit complex entries: The j-th entry of the $(5i-2)$-th row of $A(\mathcal{G})$ is denoted by $a_{i,j}^{(\mathcal{G},3)}$. The value is set to be $\exp(\iota j\pi/(2n))$.

3) Unit entries: The j-th entry of the $(5i-1)$-th row of A is denoted by $a_{i,j}^{(\mathcal{G},4)}$. The value is set to be 1.

4) Random unit complex entries: The j-th entry of the $5i$-th row of A is denoted by $a_{i,j}^{(\mathcal{G},5)}$ used for verification and resolving degeneracy in the decoding process. The value is set to be $a_{i,(j)}^{(\mathcal{G},5)} = \exp(\iota\phi_{i,j})$, where $\phi_{i,j}$ is chosen uniformly at random from $[0,\ \pi/2]$[5].

V. Reconstruction Algorithm

Let $b_i^{(\mathcal{G},q)}$ denote the $[5(i-1)+q]$-th measurement generated by $A(\mathcal{G})$. Here, $\mathcal{G}$ is one of the $\mathcal{G}_I$, $\mathcal{G}_{II,l}$'s and $\mathcal{G}_{III}$, $i \in [m_{\mathcal{G}}]$, and $q \in [5]$.

A. Seeding phase

Initialization: We initialize by setting the signal estimate vector $\hat{\mathbf{x}}$ to all-zero vector 0^n. Each right node $i \in \left[m'_{\mathcal{G}_I}\right]$ attaches an empty neighbor list $\mathcal{N}(i)$. let $\mathcal{D}$ denote a list of the resolvable doubletons. Initially, $\mathcal{D}$ is empty. Set $i = 1$.

Singleton Identification, Verification and Magnitude Recovery: Compute $s_i = \arctan\left(b_i^{(\mathcal{G}_I,2)}/b_i^{(\mathcal{G}_I,1)}\right)/(\pi/(2n))$. Check whether s_i is an integer. If s_i is not an integer, increment i by 1 and start a new iteration. If s_i is an integer, we do the following steps:

1) Singleton Identification: We tentatively identify that i is a singleton.

2) Magnitude Estimation: Assume that s_i-th entry of $\mathbf{x}$ is non-zero and $|\hat{x}_{s_i}| = \iota b_i^{(\mathcal{G}_I,2)}/a_{i,s_i}^{(\mathcal{G}_I,2)}$.

3) Verification: If $|\hat{x}_{s_i}| \neq \left|b_i^{(\mathcal{G}_I,5)}\right|$, the verification fails and increment i by 1 and start a new iteration. If verification passes, we do the following steps:

a) Updating Neighbor List: s_i is appended to the neighbor lists of all its neighbors. For $i \in \left[m'_{\mathcal{G}_I}\right]$, it is no longer considered in the later process if $|\mathcal{N}(i)| \geq 3$ since in the next step we only care about doubleton whose neighbor list size equals 2.

b) Increment i by 1 and start a new iteration.

Doubleton Identification: For each i whose neighbor list is of size 2, it is appended to the resolvable doubleton list $\mathcal{D}$ where $\mathcal{N}(i)[1]$ and $\mathcal{N}(i)[2]$ (or i_1 and i_2 for short) are the two indices of non-zero components whose magnitudes have been recovered.

Relative Phase Recovery: To compute connected component of $\mathcal{H}$, Depth first search (DFS) [24] for adjacent list representation of $\mathcal{H}$ is applied in this step. For each $i \in \mathcal{D}$, the elements in $\mathcal{N}(i)$ tell which two vertices in $\mathcal{H}$ are connected. DFS outputs connected components of graph $\mathcal{H}$. For the first node in a connected component, its phase is set to be zero. We run the DFS, for each edge in $\mathcal{H}$, with additional steps stated below:

1) Relative Phase Estimation: We know that i's two neighbors are i_1 and i_2. The fourth measurement is used to derive the phase between i_1-th and i_2-th components of $\mathbf{x}$, $\theta = |\theta_{i_1} - \theta_{i_2}|$, by Law of Cosines[6].

2) Resolving Degeneracy and Verification: The verification measurement helps to resolve the degeneracy of sign of θ (*i.e.*, whether θ or $-\theta$ is the actual phase difference we are interested in) by checking whether $||\hat{\mathbf{x}}_{i_1}|\exp(\iota\phi_{i,i_1}) + |\hat{\mathbf{x}}_{i_2}|\exp(\iota(\phi_{i,i_2}+\theta))| = \left|b_i^{(\mathcal{G}_I,5)}\right|$, or $||\hat{\mathbf{x}}_{i_1}|\exp(\iota\phi_{i,i_1}) + |\hat{\mathbf{x}}_{i_2}|\exp(\iota(\phi_{i,i_2}-\theta))| = \left|b_i^{(\mathcal{G}_I,5)}\right|$. If neither of the above equations holds, then i is not a resolvable doubleton. Namely, there is no edge between $\mathcal{N}(i)[1]$-th and $\mathcal{N}(i)[2]$-th components of $\mathbf{x}$.

When the DFS terminates, we can find the largest connected component of $\mathcal{H}$, $\mathcal{H}'$.

B. Geometric-decay and Cleaning-up phases

Claim 2. ("Cancelling out" Process) For a bipartite graph $\mathcal{G}$ in geometric-decay phase or cleaning-up phase, if a right node i is a resolvable multiton node, it involves exactly one (unknown) undecoded non-zero component, $\mathbf{x}_j$, and at least one (known) resolved non-zero components. Then, we are able to find the location of $\mathbf{x}_j$, j, and resolve $\mathbf{x}_j$ (both magnitude and relative phase).

We omit the proof here and refer the reader to [19, Claim 2] for a detailed proof. In each stage at geometric-decay phase and cleaning-up phase, we go through all the right nodes, find resolvable multitons and use them to recover unresolved non-zero components by the "cancelling out" process. For a newly resolved component of $\mathbf{x}$, the corresponding node in $\mathcal{H}$ is appended to $\mathcal{H}'$. In the end, the size of the node set of $\mathcal{H}'$ should be k.

VI. Parameters Design

All the parameters designed in this section are calculated based on expectation. The actual performance of our algorithm will be discussed in Section VII. We refer the reader to [19, Section VI] for detailed calculation.

A. Seeding phase

1) Magnitude Recovery by singletons

The probability of a right node being a singleton node, P_S, equals $\binom{k}{1}(1/k)(1-1/k)^{k-1}$ which converges to e^{-1} as k goes to infinity. The expected number of singletons is $ck \times P_S$. Thus, the expected number of different non-zero components whose magnitudes are recovered is $k(1-\exp(-cP_S))$ by [19, Lemma 3].

2) Relative Phase Recovery by resolvable doubletons

The probability of a right node being doubleton, P_D, equals $\binom{k}{2}(1/k)^2(1-1/k)^{k-2}$. The expected number of doubletons is $= ck \times P_D$. The expected number of resolvable doubletons is $\left[\binom{k(1-\exp(-cP_S))}{2}/\binom{k}{2}\right] \cdot ckP_D$ which converges to $(1-\exp(-cP_S))^2 cke^{-1}/2$ as k goes to infinity. So, the expected number of different pairs of components whose relative phase is recovered by resolvable doubletons is $(1+\mathcal{O}(1/k))(1-\exp(-cP_S))^2 ckP_D$ by [19, Lemma 3].

3) The giant connected components

We need to find the size of giant connected component of a random graph with $k(1-\exp(-cP_S))$ nodes and $(1+\mathcal{O}(1/k))(1-\exp(-cP_S))^2 ckP_D$ distinct edges. Let's say the size is $(1-f_I)k$ where f_I is the function of c. By

[5]If $\phi_{i,j}$ is chosen with $\Omega(\log(k))$ bits of precision, the error probability of verification and resolving degeneracy (see relative phase recovery part in Section V) in a single step is at most $\mathcal{O}(1/poly(k))$. Since the total number of times one needs to verify and resolve degeneracy is $\mathcal{O}(k)$, by applying Union bound over the decoding process, the probability of incorrect decoding is upper bounded by $\mathcal{O}(1/poly(k))$.

[6]Given the lengths of two complex number A and B, we can deduce the phase between A and B, Δ, by Law of Cosines if we also know the length of $A+B$. To be more explicit, $-\cos\Delta = \left(|A|^2+|B|^2-|A+B|^2\right)/(2|A||B|)$.

[19, Theorem 4], when $2\left(1-\exp\left(-cP_S\right)\right)cP_D > 1$, the giant connected component exists (this inequality holds when constant c is large enough) and the size of the giant component is $\left(1-f_I\right)k = \beta_c k\left(1-\exp\left(-cP_S\right)\right)$ where β_c is the unique solution to $\beta+\exp\left[-\beta\cdot 2\left(1-\exp\left(-cP_S\right)\right)cP_D\right]=1$.

B. Geometric-decay phase

In the $(l+1)$-th stage $(0 \le l \le L-1)$, the probability that a right node being a resolvable multiton, $P_M^{(II,l+1)}$, equals $\left(1-1/\left(f_{II,l}k\right)\right)^{f_{II,l}k-1}-\left(1-1/\left(f_{II,l}k\right)\right)^{k-1}$ which converges to $e^{-1}-\exp\left(-1/f_{II,l}\right)$ as k goes to infinity. The expected number of resolvable multitons is $cf_{II,l}kP_M^{(II,l+1)}$. The expected number of non-zero components which are resolved (both magnitude and phase), $\left(f_{II,l}-f_{II,l+1}\right)k$, equals $f_{II,l}\left(1-e^{-cP_M^{(II,l+1)}}\right)$ by [19, Lemma 3]. Therefore, $f_{II,l+1}=\exp\left(-cP_M^{(II,l+1)}\right)f_{II,l}$. We can compute the value of $f_{II,l}$ recursively. Note that $P_M^{(II,l)}$ increases as l increases.

C. Cleaning-up phase

Recall that, in this phase, each edges appears with probability $\log k/k$ and there are $c\left(k/\log k\right)\log\left(k/\log k\right)=\mathcal{O}(k)$ right nodes in $\mathcal{G}_{III}$.

VII. Performance of algorithm

For completeness, the reader is refered to [19, Section VII] for a detailed proof.

A. Number of measurements

The number of measurements in the seeding phase and cleaning-up phase is $\mathcal{O}(k)$. For geometric-decay phase, we show that $f_{II,l+1}=\exp\left(-cP_M^{(II,l+1)}\right)f_{II,l}$, and $P_M^{(II,l)}$ increases as l increases in Section VI. So, $f_{II,l+1}\le\exp\left(-c(l+1)P_M^{(II,1)}\right)f_I$ holds for all $0\le l\le L-1$. Thus, the number of measurement in the geometric-decay phase is $\mathcal{O}(k)$. Then, the total number of measurements is $\mathcal{O}(k)$.

B. Decoding complexity

Almost all the operations take constant time except for DFS in the seeding phase and "Cancelling out" process in the geometric-decay and cleaning-up phases. For DFS, the time complexity is linear in the size of node set and edge set. Since there are k nodes and $\mathcal{O}(k)$ edges involved in the seeding phase, the time complexity is $\mathcal{O}(k)$. For "Cancelling out" process, the time complexity depends on the number of resolved non-zero components which corresponds to a resolvable multiton. In the later stage/phase, more non-zero components are associated with a measurement. Since the number of measurements is $\mathcal{O}(k)$, it suffices to show that each measurement involves at most $\mathcal{O}(\log k)$ non-zero components (even if they are unresolved) in the cleaning-up phase with probability at least $1-o(1/k)$. In fact, this can be shown by Chernoff bound. Thus, we know that the decoding complexity is at most $\mathcal{O}(k\log k)$ with probability at least $1-k\cdot o(1/k)=1-o(1)$ by Union bound.

C. Correctness

The actual number of resolved non-zero components in each phase/stage deviates from the expected value but it can be concentrated around expectation with high probability. Let g_I denote the actual fraction of unresolved non-zero components after seeding phase. Let $g_{II,l}$ denote the actual fraction of unresolved non-zero components after the l-th stage in geometric-decay phase. Let $g_{II,0}=g_I$.

After the seeding phase, we show that $\left(1-\epsilon_I\right)f_I\le g_I\le\left(1+\epsilon_I\right)f_I$ holds with probability $1-\mathcal{O}\left(k^{-1/3}\right)$, and ϵ_I scales as $\mathcal{O}\left(k^{-1/3}\right)$. In the end of geometric-decay phase, we show that $\left(1-\epsilon_{II,L}\right)f_{II,L}\le g_{II,L}\le\left(1+\epsilon_{II,L}\right)f_{II,L}$ holds with probability $1-\mathcal{O}\left(\log\log k\cdot k^{-1/3}\right)$, and $\epsilon_{II,L}$ scales as $\mathcal{O}\left(\log k\cdot k^{-1/3}\right)$. Finally, when the cleaning-up phase ends, all the remaining non-zero components are resolved with probability $1-\mathcal{O}(\log k/k)$ given the concentration results in the previous phases are correct by [19, Theorem 5]. Therefore, by Union bound, SUPER algorithm resolves all the non-zero components with probability $1-o(1)$.

References

[1] R. Millane, "Phase retrieval in crystallography and optics," *J. Opt. Soc. Am. A*, vol. 7, pp. 394–411, 1990.

[2] J. Dainty and J. Fienup, "Phase retrieval and image reconstruction for astronomy," *In: H. Stark, ed., Image Recovery: Theory and Application, Academic Press, New York*, pp. 231–275, 1987.

[3] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[4] J. Fienup, "Phase retrieval algorithms: a personal tour [invited]," *Appl. Opt.*, vol. 52, pp. 45–56, 2013.

[5] E. Candès, T. Strohmer, and V. Voroninski, "Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming," *Communications on Pure and Applied Mathematics*, vol. 66, no. 8, pp. 1241–1274, 2013.

[6] I. Waldspurger, A. d'Aspremont, and S. Mallat, "Phase recovery, maxcut and complex semidefinite programming," *Mathematical Programming*, pp. 1–35, 2013.

[7] E. Candès and X. Li, "Solving quadratic equations via phaselift when there are about as many equations as unknowns," *Foundations of Computational Mathematics*, pp. 1–10, 2013.

[8] B. Alexeev, A. S. Bandeira, M. Fickus, and D. G. Mixon, "Phase retrieval with polarization," *e-prints, arXiv:1210.7752[cs.IT]*, 2012.

[9] P. Netrapalli, P. Jain, and S. Sanghavi, "Phase retrieval using alternating minimization," *e-prints, arXiv:1306.0160v1 [stat.ML]*, 2013.

[10] H. Ohlsson and Y. C. Eldar, "On conditions for uniqueness in sparse phase retrieval," *e-prints, arXiv:1308.5447[cs.IT]*, 2013.

[11] M. Akçakaya and V. Tarokh, "New conditions for sparse phase retrieval," *e-prints, arXiv:1310.1351[cs.IT]*, 2013.

[12] H. Ohlsson, A. Yang, R. Dong, and S. Sastry, "Compressive phase retrieval from squared output measurements via semidefinite programming," *e-prints, arXiv:1111.6323v3 [math.ST]*, 2011.

[13] X. Li and V. Voroninski, "Sparse signal recovery from quadratic measurements via convex programming," *e-prints, arXiv:1209.4785[cs.IT]*, 2012.

[14] Y. Shechtman, A. Beck, and Y. C. Eldar, "Gespar: Efficient phase retrieval of sparse signals," *e-prints, arXiv:1301.1018[cs.IT]*, 2013.

[15] K. Jaganathan, S. Oymak, and B. Hassibi, "Recovery of sparse 1-d signals from the magnitudes of their fourier transform," in *2012 IEEE International Symposium on Information Theory Proceedings (ISIT)*, 2012, pp. 1473–1477.

[16] ——, "Sparse phase retrieval: Convex algorithms and limitations," in *2013 IEEE International Symposium on Information Theory Proceedings (ISIT)*, 2013, pp. 1022–1026.

[17] ——, "Phase retrieval for sparse signals using rank minimization," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 3449–3452.

[18] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi, "Simultaneously structured models with application to sparse and low-rank matrices," *e-prints, arXiv:1212.3753[cs.IT]*, 2013.

[19] S. Cai, M. Bakshi, S. Jaggi, and M. Chen, "Super: Sparse signals with unknown phases efficiently recovered," *e-prints, arXiv:1401.4269[cs.IT]*, 2014.

[20] S. Pawar and K. Ramchandran, "A hybrid dft-ldpc framework for fast, efficient and robust compressive sensing," in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2012, pp. 1943–1950.

[21] M. Bakshi, S. Jaggi, S. Cai, and M. Chen, "Sho-fa: Robust compressive sensing with order-optimal complexity, measurements, and bits," in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2012, pp. 786–793.

[22] B. Bollobas, "Random graphs," *Cambridge University Press*, 2001.

[23] R. Motwani and P. Raghavan, "Randomized algorithms," *Cambridge University Press*, 1995.

[24] R. Tarjan, "Depth first search and linear graph algorithms," *SIAM Journal on Computing*, 1972.