Phase Retrieval,  
New Results on an Old Problem.

Alex d’Aspremont, CNRS & École Normale Supérieure, Paris,

with Fajwel Fogel, Irène Waldspurger and Stéphane Mallat, ENS.

Support from ERC (project SIPA).
Introduction: diffraction imaging

Diffraction imaging

- Sensors only record the **magnitude** of diffracted rays, and lose the **phase**.
- The phase is required to invert the 2D Fourier transform and reconstruct the sample density.

[Candes et al., 2011]
Focus on the **phase retrieval** problem, i.e. solve

\[
\text{find } x \text{ such that } |\langle a_i, x \rangle|^2 = b_i^2, \quad i = 1, \ldots, n
\]

in the variable \( x \in \mathbb{C}^p \).

- Reconstruct a signal \( x \) from the **amplitude of \( n \) linear measurements** \( A \).
- Easy to write, very **hard to solve** in general.
- We seek a **tractable** procedure, i.e. a polynomial time algorithm with explicit approximation and complexity guarantees.
We want more than uniqueness of the solution.

- A tractable algorithm to solve the phasing problem in polynomial-time.
- A solution that is stable and robust to noise.

For certain measurement matrices $A$, this is indeed possible. . .
Introduction

**Greedy algorithm** [Gerchberg and Saxton, 1972], find \( y = Ax \) given \( b = |Ax| \)

---

**Input:** An initial \( y^1 \in \mathbb{C}^n \), i.e. such that \( |y^1| = b \).

1. for \( k = 1, \ldots, N - 1 \) do
2. Set \( w = AA^\dagger y^k \), (project \( y \) on \( \mathcal{R}(A) \).)
3. Set \( y_i^{k+1} = b_i \frac{w}{|w|} \), (match \( |y| \) with \( b \).)
4. end for

**Output:** \( y_N \in \mathbb{C}^n \).

---

Similar to **alternating projections**. Sometimes it works, sometimes it doesn’t…

**Can we do better?**
Given user ratings

Make **personalized** recommendations for other movies. . .
A **linear prediction** model

\[ \text{rating}_{ij} = u_i^T v_j \]

where \( u_i \) represents user characteristics and \( v_j \) movie features.

Collaborative prediction is a **matrix factorization** problem

\[ M = U^T V \]

\( U \in \mathbb{R}^{n \times k} \) user types, \( V \in \mathbb{R}^{k \times m} \) movie features, \( M \in \mathbb{R}^{n \times m} \) ratings.

Assume \( M \) is **low rank**.
Matrix completion. [Recht et al., 2007, Candes and Recht, 2008, Candes and Tao, 2010].

- The **NETFLIX** problem can be written as

\[
\begin{align*}
\text{Minimize} & \quad \text{Rank}(X) \\
\text{subject to} & \quad \text{Tr}(A_i X) = b_i, \quad i = 1, \ldots, n \\
& \quad X \succeq 0
\end{align*}
\]

- For certain matrices $A_i$, it suffices to solve

\[
\begin{align*}
\text{Minimize} & \quad \text{Tr}(X) \\
\text{subject to} & \quad \text{Tr}(A_i X) = b_i, \quad i = 1, \ldots, n \\
& \quad X \succeq 0
\end{align*}
\]

which is a **convex problem** in $X \in \mathbb{S}_n$. 
Introduction: phase retrieval as a SDP

- [Chai et al., 2011, Candes et al., 2013a], lifting technique from [Shor, 1987]

\[ |\langle a_i, x \rangle|^2 = b_i^2 \iff \text{Tr}(a_i a_i^* x x^*) = b_i^2 \]

to formulate \textit{phase recovery as a matrix completion problem}

Minimize \( \text{Rank}(X) \)

such that \( \text{Tr}(a_i a_i^* X) = b_i^2, \quad i = 1, \ldots, n \)

\( X \succeq 0 \)

- [Candes, Strohmer, and Voroninski, 2013a] show that under certain conditions on \( A \) and \( x_0 \), it suffices to solve

Minimize \( \text{Tr}(X) \)

such that \( \text{Tr}(a_i a_i^* X) = b_i^2, \quad i = 1, \ldots, n \)

\( X \succeq 0 \)

which is a (convex) \textit{semidefinite program} in \( X \in \mathbb{H}_p \).
**Introduction**

A very sparse (and incomplete) list of references...

**Algorithms**

- Greedy algorithm [Gerchberg and Saxton, 1972]
- Classical survey of early algorithms by [Fienup, 1982].
- NP-complete [Sahinoglou and Cabrera, 1991].
- Matrix completion formulation [Chai, Moscoso, and Papanicolaou, 2011] and [Candes, Strohmer, and Voroninski, 2013a]

**Applications**

- X-ray and crystallography imaging [Harrison, 1993], diffraction imaging [Bunk et al., 2007] or microscopy [Miao et al., 2008].
- Audio signal processing [Griffin and Lim, 1984].
Outline

- Introduction
- **Algorithms**
  - Exploiting structure
  - Numerical results
  - Experimental setup?
A linear program (LP) is written

\[
\begin{align*}
& \text{minimize} & & c^T x \\
& \text{subject to} & & Ax = b \\
& & & x \geq 0
\end{align*}
\]

where \( x \geq 0 \) means that the coefficients of the vector \( x \) are nonnegative.

A semidefinite program (SDP) is written

\[
\begin{align*}
& \text{minimize} & & \text{Tr}(CX) \\
& \text{subject to} & & \text{Tr}(A_i X) = b_i, \quad i = 1, \ldots, m \\
& & & X \succeq 0
\end{align*}
\]

where \( X \succeq 0 \) means that the matrix variable \( X \in S_n \) is positive semidefinite.

- Nesterov and Nemirovskii [1994] showed that the interior point algorithms used for linear programs could be extended to semidefinite programs.
- Efficient solvers, many (unexpected) applications.
Phase problem in phase

We can **decouple** the phase and magnitude reconstruction problems.

- \( Ax = \text{diag}(b)u \) where \( u \in \mathbb{C}^n \) is a **phase vector** with \( |u_i| = 1 \).

- The phase recovery problem can be written

\[
\min_{u \in \mathbb{C}^n, |u_i|=1, \ x \in \mathbb{C}^p} \| Ax - \text{diag}(b)u \|_2^2,
\]

- The inner minimization problem in \( x \) is a standard least squares, with solution \( x = A^\dagger \text{diag}(b)u \), so phase recovery becomes

\[
\text{minimize} \quad u^* Mu \\
\text{subject to} \quad |u_i| = 1, \quad i = 1, \ldots, n,
\]

in \( u \in \mathbb{C}^n \), where \( M = \text{diag}(b)(I - AA^\dagger) \text{diag}(b) \succeq 0 \).
Tightness

Exact phase reconstruction in polynomial-time.

- [Candes et al., 2013a,b] show exact recovery w.h.p. for the PhaseLift relaxation

\[
\begin{align*}
\text{Minimize} & \quad \text{Tr}(X) \\
\text{such that} & \quad \text{Tr}(a_i a_i^* X) = b_i^2, \quad i = 1, \ldots, n \\
& \quad X \succeq 0 
\end{align*}
\]

when \( n = O(p) \) observations \( a_i \) picked randomly (sphere or coded Fourier).


\[
\begin{align*}
\text{Minimize} & \quad \text{Tr}(MU) \\
\text{such that} & \quad \text{diag}(U) = 1, \quad U \succeq 0 
\end{align*}
\]

similar to MAXCUT relaxation.

- [Waldspurger et al., 2012] show PhaseCut is tight when PhaseLift is.
Which observations $A$?

[Candes et al., 2013b]: The observations $A$ are constructed from multiple coded diffraction patterns

More on this later...
Algorithms

Block Coordinate Method. PhaseCut & MAXCUT

**Input:** An initial \( U^0 = I_n \) and \( \nu > 0 \) (typically small). An integer \( N > 1 \).

1. **for** \( k = 1, \ldots, N \) **do**
2. Pick \( i \in [1, n] \).
3. Compute
   \[
   u = U_{i^c,i}^k M_{i^c,i} \quad \text{and} \quad \gamma = u^* M_{i^c,i}
   \]
4. If \( \gamma > 0 \), set
   \[
   U_{i^c,i}^{k+1} = U_{i,i^c}^{k+1*} = -\sqrt{\frac{1 - \nu}{\gamma}} x
   \]
   else
   \[
   U_{i^c,i}^{k+1} = U_{i,i^c}^{k+1*} = 0.
   \]
5. **end for**

**Output:** A matrix \( U \succeq 0 \) with \( \text{diag}(U) = 1 \).

Writing \( i^c \) the index set \( \{1, \ldots, i - 1, i + 1, \ldots, n\} \).
Complexity.

- Each iteration only requires matrix vector products $O(n^2)$.
- Cost per iteration similar to greedy algorithm [Gerchberg and Saxton, 1972].
- Signal applications: matrix vector product computed efficiently using the FFT, cost per iteration reduced to $O(n \log n)$. 
Outline

- Introduction
- Algorithms
- Exploiting structure
- Numerical results
- Experimental setup?
Sparsity: known support in 2D

Electronic density: caffeine (left), 2D FFT transform (diffraction pattern, center), reconstructed using 3% of the coefficients at the core of the FFT (right).

- Molecular imaging: data is **sparse with known support.**
- Most coefficients in $b$ close to zero, so **most coefficients in $u$ can be set to zero** in

\[
\text{minimize } u^*Mu \\
\text{subject to } |u_i| = 1, \quad i = 1, \ldots n,
\]

which means significant computational savings.
We observe the magnitude of the Fourier transform of a discrete signal \( x \in \mathbb{R}^P \)

\[ |\mathcal{F}x| = b \]

We seek to reconstruct positive signals \( x \geq 0 \).

A function \( f : \mathbb{R}^s \mapsto \mathbb{C} \) is positive semidefinite if and only if the matrix \( B \) with \( B_{ij} = f(x_i - x_j) \) is Hermitian positive semidefinite for any sequence \( x_i \in \mathbb{R}^s \).

**Theorem (Bochner)**

**Fourier on positive signals.** A function \( f : \mathbb{R}^s \mapsto \mathbb{C} \) is positive semidefinite if and only if it is the Fourier transform of a (finite) nonnegative Borel measure.
Reconstruct a phase vector $u \in \mathbb{C}^n$ such that $|u| = 1$ and

$$\mathcal{F}x = \text{diag}(b)u.$$ 

We define the Toeplitz matrix $B_{ij}(y) = y|i-j|+1$, $i, j = 1, \ldots, p$, so that

$$B(y) = \begin{pmatrix} y_1 & y_2^* & \cdots & y_n^* \\ y_2 & y_1 & y_2^* & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ y_n & \cdots & y_2 & y_1 \end{pmatrix}.$$ 

Bochner's theorem.

$$x \geq 0 \iff B(\text{diag}(b)u) \succeq 0,$$

which is a (convex) linear matrix inequality in $u$. 

A. d'Aspremont
Real signals

Real valued signal. Phase problem on real valued signal is

\[
\text{minimize } \left\| \mathcal{T}(A) \begin{pmatrix} x \\ 0 \end{pmatrix} - \text{diag} \begin{pmatrix} b \\ b \end{pmatrix} \begin{pmatrix} \Re(u) \\ \Im(u) \end{pmatrix} \right\|_2^2
\]
subject to \( u \in \mathbb{C}^n, |u_i| = 1 \)
\( x \in \mathbb{R}^p \).

Here \( x = A_2^\dagger B_2 v \), where

\[
A_2 = \begin{pmatrix} \Re(A) \\ \Im(A) \end{pmatrix}, \quad B_2 = \text{diag} \begin{pmatrix} b \\ b \end{pmatrix}, \quad \text{and} \quad v = \begin{pmatrix} \Re(u) \\ \Im(u) \end{pmatrix}
\]

the phase problem is equivalent to

\[
\text{minimize } \left\| (A_2 A_2^\dagger B_2 - B_2) v \right\|_2^2
\]
subject to \( v_i^2 + v_{n+i}^2 = 1, \quad i = 1, \ldots, n, \)

in the variable \( v \in \mathbb{R}^{2n} \).
Real signals

Real valued signal. The last problem can be relaxed as

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(VM_2) \\
\text{subject to} & \quad V_{ii} + V_{n+i,n+i} = 1, \quad i = 1, \ldots, n, \\
& \quad V \succeq 0,
\end{align*}
\]

which is a semidefinite program in the variable \( V \in \mathbf{S}_{2n} \), where

\[
M_2 = (A_2A_2^\dagger B_2 - B_2)^T(A_2A_2^\dagger B_2 - B_2) = B_2^T(I - A_2A_2^\dagger)B_2.
\]

- Explicitly constrains the solution \( x \) to be real valued.
- Small increase in complexity.
Outline

- Introduction
- Algorithms
- Exploiting structure
- **Numerical results**
- Experimental setup?
Numerical Experiments: PDB molecules

Two molecules, two resolutions: 16x16 and 128x128.

Caffeine

Cocaine
Numerical Experiments: PDB molecules

Solution of the **greedy algorithm** on caffeine molecule, for various values of the number of masks and noise level $\alpha$. 

A. d'Aspremont  
Stanford SLAC, March 2014. 26/39
Solution of the **PhaseCut SDP** followed by greedy refinements, for various values of the **number of masks** and **noise level** $\alpha$. 

A. d’Aspremont
MSE between reconstructed image and true image for 2 random illuminations without noise, using SDP then Fienup (blue), and Fienup only (red).
Numerical Experiments: comparing algorithms

16x16 caffeine image. No oversampling.

Left: MSE (relative to $b$) vs. number of random masks.
Right: Probability of recovering molecular density ($\text{MSE} < 10^{-4}$) vs. number of random masks.
Numerical Experiments: comparing algorithms

16x16 cocaine image. No oversampling.

Left: MSE (relative to $b$) vs. number of random masks. Right: Probability of recovering molecular density ($MSE < 10^{-4}$) vs. number of random masks.
16x16 caffeine image. 2x oversampling.

*Left:* MSE vs. number of random masks.
*Right:* Probability of recovering molecular density ($MSE < 10^{-4}$) vs. number of random masks.
Numerical Experiments: comparing algorithms

16x16 caffeine image. Mask resolution (1x1 to 8x8 pixels).

**Left:** MSE vs. mask resolution. (2x oversampling, no noise, 3 masks).

**Right:** Probability of recovering molecular density ($MSE < 10^{-4}$) vs. number of random masks.
Numerical Experiments: comparing algorithms

16x16 cocaine image. Mask resolution (1x1 to 8x8 pixels).

Left: MSE vs. mask resolution. (2x oversampling, no noise, 3 masks).
Right: Probability of recovering molecular density ($MSE < 10^{-4}$) vs. number of random masks.
16x16 cocaine image. **Mask resolution (1x1 to 8x8 pixels).**

*Left:* MSE vs. mask resolution. **(2x oversampling, no noise, 2 masks).**

*Right:* Probability of recovering molecular density (**$MSE < 10^{-4}$**) vs. number of random masks.

A. d'Aspremont

Stanford SLAC, March 2014. 34/39
Numerical Experiments: comparing algorithms

16x16 caffeine image. **Noise.**

**Left:** MSE vs. noise level (**2x oversampling, 2 masks**).

**Right:** Probability of recovering molecular density (**$MSE < 10^{-4}$**) vs. number of random masks.

A. d’Aspremont
Numerical Experiments: comparing algorithms

16x16 cocaine image. Noise.

Left: MSE vs. noise level (2x oversampling, 2 masks).
Right: Probability of recovering molecular density ($MSE < 10^{-4}$) vs. number of random masks.
Outline

- Introduction
- Algorithms
- Exploiting structure
- Numerical results
- Experimental setup?
Observations $\mathcal{A}$: implementation

Construct observations $\mathcal{A}$ from multiple coded diffraction patterns

- Split the beam?
- Mask before/after the sample?
Conclusion

- Tractable algorithms for phase recovery
- Exact recovery results
- Exploit structure

Open questions...

- Is the SDP relaxation optimal?
- Experimental setup?
References


