Statistical Estimation in the Presence of Group Actions

Alex Wein
MIT Mathematics
In memoriam

Amelia Perry
1991 – 2018
My research interests

- Statistical and computational limits of average-case inference problems (signal planted in random noise)
My research interests

- Statistical and computational limits of average-case inference problems (signal planted in random noise)
  - Community detection (stochastic block model)
  - Spiked matrix/tensor problems
  - Synchronization / group actions (today)
My research interests

- Statistical and computational limits of average-case inference problems (signal planted in random noise)
  - Community detection (stochastic block model)
  - Spiked matrix/tensor problems
  - Synchronization / group actions (today)

- Connections to...
My research interests

- Statistical and computational limits of average-case inference problems (signal planted in random noise)
  - Community detection (stochastic block model)
  - Spiked matrix/tensor problems
  - Synchronization / group actions (today)

- Connections to...
  - Statistical physics
    - Phase transitions: easy, hard, impossible
My research interests

- Statistical and computational limits of average-case inference problems (signal planted in random noise)
  - Community detection (stochastic block model)
  - Spiked matrix/tensor problems
  - Synchronization / group actions (today)

- Connections to...
  - Statistical physics
    - Phase transitions: easy, hard, impossible
  - Algebra
    - Group theory, representation theory, invariant theory
My research interests

▶ Statistical and computational limits of average-case inference problems (signal planted in random noise)
  ▶ Community detection (stochastic block model)
  ▶ Spiked matrix/tensor problems
  ▶ Synchronization / group actions (today)

▶ Connections to...
  ▶ Statistical physics
    ▶ Phase transitions: easy, hard, impossible
  ▶ Algebra
    ▶ Group theory, representation theory, invariant theory

▶ Today: problems involving group actions
  ▶ A meeting point of statistics, algebra, signal processing computer science, statistical physics, . . .
Motivation: cryo-electron microscopy (cryo-EM)

Image credit: [Singer, Shkolnisky '11]
Motivation: cryo-electron microscopy (cryo-EM)

- Biological imaging method: determine structure of molecule
Motivation: cryo-electron microscopy (cryo-EM)

- Biological imaging method: determine structure of molecule
- 2017 Nobel Prize in Chemistry
Motivation: cryo-electron microscopy (cryo-EM)

- Biological imaging method: determine structure of molecule
- 2017 Nobel Prize in Chemistry
- Given many noisy 2D images of a 3D molecule, taken from different unknown angles
Motivation: cryo-electron microscopy (cryo-EM)

- Biological imaging method: determine structure of molecule
- 2017 Nobel Prize in Chemistry
- Given many noisy 2D images of a 3D molecule, taken from different unknown angles
- Goal is to reconstruct the 3D structure of the molecule
Motivation: cryo-electron microscopy (cryo-EM)

- Biological imaging method: determine structure of molecule
- 2017 Nobel Prize in Chemistry
- Given many noisy 2D images of a 3D molecule, taken from different unknown angles
- Goal is to reconstruct the 3D structure of the molecule
- Group action by $SO(3)$ (rotations in 3D)
Other examples

Other problems involving random group actions:
Other examples

Other problems involving random group actions:

- Image registration

Group: SO(2) (2D rotations)
Other examples

Other problems involving random group actions:

► **Image registration**

Group: $\text{SO}(2)$ (2D rotations)

Image credit: Bandeira, PhD thesis '15

► **Multi-reference alignment**

Group: $\mathbb{Z}/p$ (cyclic shifts)

Image credit: Jonathan Weed
Other examples

Other problems involving random group actions:

▶ Image registration

Group: SO(2) (2D rotations)

Applications: computer vision, radar, structural biology, robotics, geology, paleontology, ...

▶ Multi-reference alignment

Group: $\mathbb{Z}/p$ (cyclic shifts)

Image credit: [Bandeira, PhD thesis '15]

Image credit: Jonathan Weed
Other examples

Other problems involving random group actions:

▶ **Image registration**

![Image credit: Bandeira, PhD thesis '15]

Group: $\text{SO}(2)$ (2D rotations)

▶ **Multi-reference alignment**

![Image credit: Jonathan Weed]

Group: $\mathbb{Z}/p$ (cyclic shifts)

▶ Applications: computer vision, radar, structural biology, robotics, geology, paleontology, ...

▶ Methods used in practice often lack provable guarantees...
Part I: Synchronization
Synchronization problems

The synchronization approach [1]: learn the group elements

Fix a group $G$

- e.g. $\text{SO}(3)

$g \in G$

- vector of unknown group elements

- e.g. rotation of each image

Given pairwise information: for each $i < j$

- a noisy measurement of $g_i g_j^{-1}$

- e.g. by comparing two images

Goal: recover $g$ up to global right-multiplication

- can't distinguish $(g_1, \ldots, g_n)$ from $(g_1 h, \ldots, g_n h)$

In cryo-EM: once you learn the rotations, it is possible to reconstruct a de-noised model of the molecule [2]

[1] Singer ’11
[2] Singer, Shkolnisky ’11
Synchronization problems

The synchronization approach [1]: learn the group elements

- Fix a group $G$
  - e.g. $\text{SO}(3)$

---

[1] Singer ’11
[2] Singer, Shkolnisky ’11
Synchronization problems

The synchronization approach [1]: learn the group elements

- Fix a group $G$
  - e.g. $\text{SO}(3)$
- $g \in G^n$ – vector of unknown group elements
  - e.g. rotation of each image

In cryo-EM: once you learn the rotations, it is possible to reconstruct a de-noised model of the molecule [2]

[1] Singer ’11
[2] Singer, Shkolnisky ’11
Synchronization problems

The synchronization approach [1]: learn the group elements

- Fix a group $G$
  - e.g. SO(3)

- $g \in G^n$ – vector of unknown group elements
  - e.g. rotation of each image

- Given pairwise information: for each $i < j$, a noisy measurement of $g_i g_j^{-1}$
  - e.g. by comparing two images

[1] Singer '11
[2] Singer, Shkolnisky '11
Synchronization problems

The synchronization approach [1]: learn the group elements

- Fix a group $G$
  - e.g. $\text{SO}(3)$

- $g \in G^n$ – vector of unknown group elements
  - e.g. rotation of each image

- Given pairwise information: for each $i < j$, a noisy measurement of $g_i g_j^{-1}$
  - e.g. by comparing two images

- Goal: recover $g$ up to global right-multiplication
  - can’t distinguish $(g_1, \ldots, g_n)$ from $(g_1 h, \ldots, g_n h)$

---

[1] Singer ’11
[2] Singer, Shkolnisky ’11
Synchronization problems

The synchronization approach [1]: learn the group elements

- Fix a group $G$
  - e.g. $\text{SO}(3)$

- $g \in G^n$ – vector of unknown group elements
  - e.g. rotation of each image

- Given pairwise information: for each $i < j$, a noisy measurement of $g_ig_j^{-1}$
  - e.g. by comparing two images

- Goal: recover $g$ up to global right-multiplication
  - can’t distinguish $(g_1, \ldots, g_n)$ from $(g_1h, \ldots, g_nh)$

In cryo-EM: once you learn the rotations, it is possible to reconstruct a de-noised model of the molecule [2]

[1] Singer ’11
[2] Singer, Shkolnisky ’11
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

$\mathbf{G} = \mathbb{Z}/2 = \{\pm 1\}$
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
- For each $i, j$ observe $x_i x_j + \mathcal{N}(0, \sigma^2)$
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
- For each $i, j$ observe $x_i x_j + \mathcal{N}(0, \sigma^2)$
- Specifically, observe $n \times n$ matrix $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W$
- $\lambda \geq 0$ – signal-to-noise parameter
- $W$ – random noise matrix: symmetric with entries $\mathcal{N}(0, 1)$
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
- For each $i, j$ observe $x_i x_j + \mathcal{N}(0, \sigma^2)$
- Specifically, observe $n \times n$ matrix $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} \mathcal{W}$
- $\lambda \geq 0$ – signal-to-noise parameter
- $\mathcal{W}$ – random noise matrix: symmetric with entries $\mathcal{N}(0, 1)$
- $Y_{ij}$ is a noisy measurement of $x_i x_j$ (same/diff)

Normalization: MMSE is a constant (depending on $\lambda$)

This is a spiked Wigner model: in general $x_i \sim P$ (some prior)

Statistical physics makes extremely precise (non-rigorous) predictions about this type of problem

Often later proved correct
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
- For each $i, j$ observe $x_i x_j + \mathcal{N}(0, \sigma^2)$
- Specifically, observe $n \times n$ matrix $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W$
- $\lambda \geq 0$ – signal-to-noise parameter
- $W$ – random noise matrix: symmetric with entries $\mathcal{N}(0, 1)$
- $Y_{ij}$ is a noisy measurement of $x_i x_j$ (same/diff)
- Normalization: MMSE is a constant (depending on $\lambda$)
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
- For each $i, j$ observe $x_i x_j + \mathcal{N}(0, \sigma^2)$
- Specifically, observe $n \times n$ matrix $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} \mathcal{W}$
  - $\lambda \geq 0$ – signal-to-noise parameter
  - $\mathcal{W}$ – random noise matrix: symmetric with entries $\mathcal{N}(0, 1)$
  - $Y_{ij}$ is a noisy measurement of $x_i x_j$ (same/diff)
- Normalization: MMSE is a constant (depending on $\lambda$)

This is a spiked Wigner model: in general $x_i \sim \mathcal{P}$ (some prior)
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
- For each $i, j$ observe $x_i x_j + \mathcal{N}(0, \sigma^2)$
- Specifically, observe $n \times n$ matrix $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} \mathcal{W}$
- $\lambda \geq 0$ – signal-to-noise parameter
- $\mathcal{W}$ – random noise matrix: symmetric with entries $\mathcal{N}(0, 1)$
- $Y_{ij}$ is a noisy measurement of $x_i x_j$ (same/diff)
- Normalization: MMSE is a constant (depending on $\lambda$)

This is a spiked Wigner model: in general $x_i \sim \mathcal{P}$ (some prior)

Statistical physics makes extremely precise (non-rigorous) predictions about this type of problem
- Often later proved correct
A simple model: Gaussian $\mathbb{Z}/2$ synchronization

- $G = \mathbb{Z}/2 = \{\pm 1\}$
- True signal $x \in \{\pm 1\}^n$ (vector of group elements)
- Observe $n \times n$ matrix $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W$

Image credit: [Deshpande, Abbe, Montanari '15]
What does statistical physics have to do with Bayesian inference?
Statistical physics and inference

What does statistical physics have to do with Bayesian inference?

In inference, observe $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W$ and want to infer $x$
Statistical physics and inference

What does statistical physics have to do with Bayesian inference?

In inference, observe $Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W$ and want to infer $x$

Posterior distribution: $\Pr[x|Y] \propto \exp(\lambda x^\top Y x)$
Statistical physics and inference

What does statistical physics have to do with Bayesian inference?

In inference, observe $Y = \frac{\lambda}{n}xx^\top + \frac{1}{\sqrt{n}}W$ and want to infer $x$

Posterior distribution: $\Pr[x|Y] \propto \exp(\lambda x^\top Yx)$

In physics, this is called a Boltzmann/Gibbs distribution:

\[
\Pr[x] \propto \exp(-\beta H(x))
\]

- Energy ("Hamiltonian") $H(x) = -x^\top Yx$
- Temperature $\beta = \lambda$
Statistical physics and inference

What does statistical physics have to do with Bayesian inference?

In inference, observe \( Y = \frac{\lambda}{n}xx^\top + \frac{1}{\sqrt{n}}W \) and want to infer \( x \)

Posterior distribution: \( \Pr[x|Y] \propto \exp(\lambda x^\top Yx) \)

In physics, this is called a **Boltzmann/Gibbs** distribution:

\[
\Pr[x] \propto \exp(-\beta H(x))
\]

- **Energy ("Hamiltonian")** \( H(x) = -x^\top Yx \)
- **Temperature** \( \beta = \lambda \)

So posterior distribution of Bayesian inference obeys the same equations as a disordered physical system (e.g. magnet, spin glass)
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

[1] Pearl ’82
[2] Donoho, Maleki, Montanari ’09
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

▶ Each unknown $x_i$ is a “node”

---

[1] Pearl ’82
[2] Donoho, Maleki, Montanari ’09
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

- Each unknown $x_i$ is a “node”
- Each observation (“interaction”) $Y_{ij}$ is an “edge”
  - In our case, a complete graph

[1] Pearl '82
[2] Donoho, Maleki, Montanari '09
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

- Each unknown $x_i$ is a “node”
- Each observation (“interaction”) $Y_{ij}$ is an “edge”
  - In our case, a complete graph
- Nodes iteratively pass “messages” or “beliefs” to each other along edges, and then update their own beliefs

[1] Pearl ’82
[2] Donoho, Maleki, Montanari ’09
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

- Each unknown $x_i$ is a “node”
- Each observation (“interaction”) $Y_{ij}$ is an “edge”
  - In our case, a complete graph
- Nodes iteratively pass “messages” or “beliefs” to each other along edges, and then update their own beliefs
- Hard to analyze

[1] Pearl ’82
[2] Donoho, Maleki, Montanari ’09
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

▶ Each unknown $x_i$ is a “node”
▶ Each observation (“interaction”) $Y_{ij}$ is an “edge”
  ▶ In our case, a complete graph
▶ Nodes iteratively pass “messages” or “beliefs” to each other along edges, and then update their own beliefs
▶ Hard to analyze

In our case (since interactions are “dense”), we can use a simplification of BP called AMP (approximate message passing) [2]

[1] Pearl ’82
[2] Donoho, Maleki, Montanari ’09
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

- Each unknown $x_i$ is a “node”
- Each observation (“interaction”) $Y_{ij}$ is an “edge”
  - In our case, a complete graph
- Nodes iteratively pass “messages” or “beliefs” to each other along edges, and then update their own beliefs
- Hard to analyze

In our case (since interactions are “dense”), we can use a simplification of BP called AMP (approximate message passing) [2]

- Easy/possible to analyze

---

[1] Pearl ‘82
BP and AMP

“Axiom” from statistical physics: the best algorithm for every* problem is BP (belief propagation) [1]

- Each unknown $x_i$ is a “node”
- Each observation (“interaction”) $Y_{ij}$ is an “edge”
  - In our case, a complete graph
- Nodes iteratively pass “messages” or “beliefs” to each other along edges, and then update their own beliefs
- Hard to analyze

In our case (since interactions are “dense”), we can use a simplification of BP called AMP (approximate message passing) [2]

- Easy/possible to analyze
- Provably optimal mean squared error for many problems

[1] Pearl ’82
[2] Donoho, Maleki, Montanari ’09
AMP for \( \mathbb{Z}/2 \) synchronization

\[
Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W, \quad x \in \{\pm 1\}^n
\]
AMP for \( \mathbb{Z}/2 \) synchronization

\[ Y = \frac{\lambda}{n} x x^\top + \frac{1}{\sqrt{n}} W, \quad x \in \{\pm 1\}^n \]

AMP algorithm:

- State \( v \in \mathbb{R}^n \) – estimate for \( x \)
AMP for $\mathbb{Z}/2$ synchronization

$$Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W, \quad x \in \{\pm 1\}^n$$

AMP algorithm:

- State $v \in \mathbb{R}^n$ – estimate for $x$
- Initialize $v$ to small random vector
AMP for \(\mathbb{Z}/2\) synchronization

\[
Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W, \quad x \in \{\pm 1\}^n
\]

AMP algorithm:

- State \(v \in \mathbb{R}^n\) – estimate for \(x\)
- Initialize \(v\) to small random vector
- Repeat:
  - 1. Power iteration: \(v \leftarrow Yv\) (power iteration)
amp for \( \mathbb{Z}/2 \) synchronization

\[
Y = \frac{\lambda}{n} x x^\top + \frac{1}{\sqrt{n}} W, \quad x \in \{\pm 1\}^n
\]

AMP algorithm:

- State \( v \in \mathbb{R}^n \) – estimate for \( x \)
- Initialize \( v \) to small random vector
- Repeat:
  1. Power iteration: \( v \leftarrow Yv \) (power iteration)
  2. Onsager: \( v \leftarrow v + [\text{Onsager term}] \)
AMP for $\mathbb{Z}/2$ synchronization

$$Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W, \quad x \in \{\pm 1\}^n$$

AMP algorithm:

- State $v \in \mathbb{R}^n$ – estimate for $x$
- Initialize $v$ to small random vector
- Repeat:
  1. Power iteration: $v \leftarrow Yv$ (power iteration)
  2. Onsager: $v \leftarrow v + [\text{Onsager term}]
  3. Entrywise soft projection: $v_i \leftarrow \tanh(\lambda v_i)$ (for all $i$)
  - Resulting values in $[-1, 1]$
AMP is optimal

\[ Y = \frac{\lambda}{n} xx^\top + \frac{1}{\sqrt{n}} W, \quad x \in \{\pm 1\}^n \]

For \( \mathbb{Z}/2 \) synchronization, AMP is provably optimal.

---

Deshpande, Abbe, Montanari, '15
Free energy landscapes

What do physics predictions look like?

$F(\gamma) = \lambda \left[ -\lambda^2 \frac{\gamma^2}{4} + 1 \right] + \frac{1}{2} \gamma \left( \gamma \lambda^2 + 1 \right) - E_z \sim N(0, 1) \log(2 \cosh(\gamma + \sqrt{\gamma^2 + z}))$

$x$-axis: $\gamma$: correlation with true signal (related to MSE)

$y$-axis: $f$: free energy – AMP's "objective function" (minimize)

AMP – gradient descent starting from $\gamma = 0$ (left side)

STAT (statistical) – global minimum

So yields computational and statistical MSE for each $\lambda$

Lesieur, Krzakala, Zdeborová '15
Free energy landscapes

What do physics predictions look like?

\[ f(\gamma) = \frac{1}{\lambda} \left[ \frac{-\lambda^2}{4} \left( \frac{\gamma^2}{\lambda^4} + 1 \right) + \frac{1}{2} \gamma \left( \frac{\gamma}{\lambda^2} + 1 \right) - \mathbb{E}_{z \sim \mathcal{N}(0,1)} \log(2 \cosh(\gamma + \sqrt{\gamma}z)) \right] \]
Free energy landscapes

What do physics predictions look like?

\[ f(\gamma) = \frac{1}{\lambda} \left[ -\frac{\lambda^2}{4} \left( \frac{\gamma^2}{\lambda^4} + 1 \right) + \frac{1}{2} \gamma \left( \frac{\gamma^2}{\lambda^2} + 1 \right) - \mathbb{E}_{z \sim \mathcal{N}(0,1)} \log(2 \cosh(\gamma + \sqrt{\gamma}z)) \right] \]

X-axis $\gamma$: correlation with true signal (related to MSE)
Y-axis $f$: free energy – AMP’s “objective function” (minimize)

Lesieur, Krzakala, Zdeborová ’15
Free energy landscapes

What do physics predictions look like?

\[ f(\gamma) = \frac{1}{\lambda} \left[ -\frac{\lambda^2}{4} \left( \frac{\gamma^2}{\lambda^4} + 1 \right) + \frac{1}{2} \gamma \left( \frac{\gamma}{\lambda^2} + 1 \right) - \mathbb{E}_{z \sim \mathcal{N}(0,1)} \log(2 \cosh(\gamma + \sqrt{\gamma} z)) \right] \]

x-axis \( \gamma \): correlation with true signal (related to MSE)
y-axis \( f \): free energy – AMP’s “objective function” (minimize)

AMP – gradient descent starting from \( \gamma = 0 \) (left side)

STAT (statistical) – global minimum

Lesieur, Krzakala, Zdeborová ’15
Free energy landscapes

What do physics predictions look like?

\[ f(\gamma) = \frac{1}{\lambda} \left[ -\frac{\lambda^2}{4} \left( \frac{\gamma^2}{\lambda^4} + 1 \right) + \frac{1}{2} \gamma \left( \frac{\gamma}{\lambda^2} + 1 \right) - \mathbb{E}_{z \sim \mathcal{N}(0,1)} \log(2 \cosh(\gamma + \sqrt{\gamma}z)) \right] \]

x-axis \( \gamma \): correlation with true signal (related to MSE)

y-axis \( f \): **free energy** – AMP’s “objective function” (minimize)

AMP – gradient descent starting from \( \gamma = 0 \) (left side)

STAT (statistical) – global minimum

So yields **computational** and **statistical** MSE for each \( \lambda \)

Lesieur, Krzakala, Zdeborová ’15
Our contributions

Joint work with Amelia Perry, Afonso Bandeira, Ankur Moitra

Perry, W., Bandeira, Moitra, *Message-passing algorithms for synchronization problems over compact groups*, to appear in CPAM

Our contributions

Joint work with Amelia Perry, Afonso Bandeira, Ankur Moitra

- Using representation theory we define a very general Gaussian observation model for synchronization over any compact group

---

Perry, W., Bandeira, Moitra, *Message-passing algorithms for synchronization problems over compact groups*, to appear in CPAM

Our contributions

Joint work with Amelia Perry, Afonso Bandeira, Ankur Moitra

- Using representation theory we define a very general Gaussian observation model for synchronization over any compact group
  - Significantly generalizes $\mathbb{Z}/2$ case

Perry, W., Bandeira, Moitra, *Message-passing algorithms for synchronization problems over compact groups*, to appear in CPAM

Our contributions

Joint work with Amelia Perry, Afonso Bandeira, Ankur Moitra

- Using representation theory we define a very general Gaussian observation model for synchronization over any compact group
  - Significantly generalizes $\mathbb{Z}/2$ case
- We give a precise analysis of the statistical and computational limits of this model
Our contributions

Joint work with Amelia Perry, Afonso Bandeira, Ankur Moitra

- Using representation theory we define a very general Gaussian observation model for synchronization over any compact group
  - Significantly generalizes $\mathbb{Z}/2$ case
- We give a precise analysis of the statistical and computational limits of this model
  - Uses non-rigorous (but well-established) ideas from statistical physics
    - Methods proven correct in related settings

Perry, W., Bandeira, Moitra, *Message-passing algorithms for synchronization problems over compact groups*, to appear in CPAM

Our contributions

Joint work with Amelia Perry, Afonso Bandeira, Ankur Moitra

- Using representation theory we define a very general Gaussian observation model for synchronization over any compact group
  - Significantly generalizes $\mathbb{Z}/2$ case
- We give a precise analysis of the statistical and computational limits of this model
  - Uses non-rigorous (but well-established) ideas from statistical physics
    - Methods proven correct in related settings
  - Includes an AMP algorithm which we believe is optimal among all polynomial-time algorithms

---

Perry, W., Bandeira, Moitra, *Message-passing algorithms for synchronization problems over compact groups*, to appear in CPAM

Our contributions

Joint work with Amelia Perry, Afonso Bandeira, Ankur Moitra

- Using representation theory we define a very general Gaussian observation model for synchronization over any compact group
  - Significantly generalizes $\mathbb{Z}/2$ case

- We give a precise analysis of the statistical and computational limits of this model
  - Uses non-rigorous (but well-established) ideas from statistical physics
    - Methods proven correct in related settings
  - Includes an AMP algorithm which we believe is optimal among all polynomial-time algorithms

- Also some rigorous statistical lower and upper bounds

---

Perry, W., Bandeira, Moitra, *Message-passing algorithms for synchronization problems over compact groups*, to appear in CPAM

Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{ z \in \mathbb{C} : |z| = 1 \}$ (angles)
Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{ z \in \mathbb{C} : |z| = 1 \}$ (angles)
- True signal $x \in U(1)^n$
Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{ z \in \mathbb{C} : |z| = 1 \}$ (angles)
- True signal $x \in U(1)^n$
- $W$ – complex Gaussian noise (GUE)
Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{ z \in \mathbb{C} : |z| = 1 \}$ (angles)
- True signal $x \in U(1)^n$
- $W$ – complex Gaussian noise (GUE)
- Observe
Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{ z \in \mathbb{C} : |z| = 1 \}$ (angles)
- True signal $x \in U(1)^n$
- $W$ – complex Gaussian noise (GUE)
- Observe

$$Y^{(1)} = \frac{\lambda_1}{n} xx^* + \frac{1}{\sqrt{n}} W^{(1)}$$
Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{z \in \mathbb{C} : |z| = 1\}$ (angles)
- True signal $x \in U(1)^n$  
- $W$ – complex Gaussian noise (GUE)
- Observe

$$Y^{(1)} = \frac{\lambda_1}{n} xx^* + \frac{1}{\sqrt{n}} W^{(1)}$$

$$Y^{(2)} = \frac{\lambda_2}{n} x^2 x^{*2} + \frac{1}{\sqrt{n}} W^{(2)}$$

$$\vdots$$

$$Y^{(K)} = \frac{\lambda_K}{n} x^K x^{*K} + \frac{1}{\sqrt{n}} W^{(K)}$$

where $x^k$ means entry-wise $k$th power.
Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{ z \in \mathbb{C} : |z| = 1 \}$ (angles)
- True signal $x \in U(1)^n$
- $W$ – complex Gaussian noise (GUE)
- Observe

$$Y^{(1)} = \frac{\lambda_1}{n} xx^* + \frac{1}{\sqrt{n}} W^{(1)}$$

$$Y^{(2)} = \frac{\lambda_2}{n} x^2 x^* x^2 + \frac{1}{\sqrt{n}} W^{(2)}$$

$$\vdots$$

$$Y^{(K)} = \frac{\lambda_K}{n} x^K x^* x^K + \frac{1}{\sqrt{n}} W^{(K)}$$

where $x^K$ means entry-wise $k$th power.

- This model has information on different frequencies
Multi-frequency $U(1)$ synchronization

- $G = U(1) = \{ z \in \mathbb{C} : |z| = 1 \}$ (angles)
- True signal $x \in U(1)^n$
- $W$ – complex Gaussian noise (GUE)
- Observe

$$Y^{(1)} = \frac{\lambda_1}{n} xx^* + \frac{1}{\sqrt{n}} W^{(1)}$$

$$Y^{(2)} = \frac{\lambda_2}{n} x^2 x^{*2} + \frac{1}{\sqrt{n}} W^{(2)}$$

$$\cdots$$

$$Y^{(K)} = \frac{\lambda_K}{n} x^K x^{*K} + \frac{1}{\sqrt{n}} W^{(K)}$$

where $x^k$ means entry-wise $k$th power.

- This model has information on different frequencies
- Challenge: how to synthesize information across frequencies?
AMP for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^{*k} + \frac{1}{\sqrt{n}} W^{(k)}$$ for $k = 1, \ldots, K$
AMP for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^{*k} + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

Algorithm’s state: $v^{(k)} \in \mathbb{C}^n$ for each frequency $k$

- $v^{(k)}$ is an estimate of $(x_1^k, \ldots, x_n^k)$
AMP for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^{*k} + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

Algorithm’s state: $\nu^{(k)} \in \mathbb{C}^n$ for each frequency $k$

- $\nu^{(k)}$ is an estimate of $(x_1^k, \ldots, x_n^k)$

AMP algorithm:

- Power iteration (separately on each frequency):
  $$\nu^{(k)} \leftarrow Y^{(k)} \nu^{(k)}$$
AMP for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^* + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

Algorithm’s state: $\nu^{(k)} \in \mathbb{C}^n$ for each frequency $k$

- $\nu^{(k)}$ is an estimate of $(x^k_1, \ldots, x^k_n)$

AMP algorithm:

- Power iteration (separately on each frequency): $\nu^{(k)} \leftarrow Y^{(k)} \nu^{(k)}$
- “Soft projection” (separately on each index $i$): $\nu_i^{(\cdot)} \leftarrow \mathcal{F}(\nu_i^{(\cdot)})$
  - This synthesizes the frequencies in a non-trivial way
AMP for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^*k + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

Algorithm’s state: $\nu^{(k)} \in \mathbb{C}^n$ for each frequency $k$

- $\nu^{(k)}$ is an estimate of $(x_1^k, \ldots, x_n^k)$

AMP algorithm:

- Power iteration (separately on each frequency):
  $$\nu^{(k)} \leftarrow Y^{(k)} \nu^{(k)}$$

- “Soft projection” (separately on each index $i$):
  $$\nu_i^{(\cdot)} \leftarrow \mathcal{F}(\nu_i^{(\cdot)})$$
  - This synthesizes the frequencies in a non-trivial way

- Onsager correction term
AMP for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^{*k} + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

Algorithm’s state: $\nu^{(k)} \in \mathbb{C}^n$ for each frequency $k$

- $\nu^{(k)}$ is an estimate of $(x_1^k, \ldots, x_n^k)$

AMP algorithm:

- Power iteration (separately on each frequency): $\nu^{(k)} \leftarrow Y^{(k)} \nu^{(k)}$

- “Soft projection” (separately on each index $i$): $\nu_i^{(\cdot)} \leftarrow F(\nu_i^{(\cdot)})$
  - This synthesizes the frequencies in a non-trivial way

- Onsager correction term

Analysis of AMP:

- Exact expression for AMP’s MSE (as $n \to \infty$) as a function of $\lambda_1, \ldots, \lambda_K$
AMP for $U(1)$ synchronization

$$
Y^{(k)} = \frac{\lambda_k}{n} x^k x^* k + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for} \quad k = 1, \ldots, K
$$

Algorithm’s state: $v^{(k)} \in \mathbb{C}^n$ for each frequency $k$

- $v^{(k)}$ is an estimate of $(x_1^k, \ldots, x_n^k)$

AMP algorithm:

- Power iteration (separately on each frequency):
  $$v^{(k)} \leftarrow Y^{(k)} v^{(k)}$$

- “Soft projection” (separately on each index $i$):
  $$v_i^{(\cdot)} \leftarrow F(v_i^{(\cdot)})$$
  - This synthesizes the frequencies in a non-trivial way

- Onsager correction term

Analysis of AMP:

- Exact expression for AMP’s MSE (as $n \to \infty$) as a function of $\lambda_1, \ldots, \lambda_K$
- Also, exact expression for the statistically optimal MSE
Results for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^* + \frac{1}{\sqrt{n}} W^{(k)}$$ \text{ for } k = 1, \ldots, K
Results for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^{*k} + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

- Single frequency: given $Y^{(k)}$, can non-trivially estimate $x^k$ iff $\lambda_k > 1$
Results for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^* + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

- Single frequency: given $Y^{(k)}$, can non-trivially estimate $x^k$ iff $\lambda_k > 1$
- Information-theoretically, with $\lambda_1 = \cdots = \lambda_K = \lambda$, need $\lambda \sim \sqrt{\log K / K}$ (for large $K$)
Results for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^*k + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K$$

- Single frequency: given $Y^{(k)}$, can non-trivially estimate $x^k$ iff $\lambda_k > 1$
- Information-theoretically, with $\lambda_1 = \cdots = \lambda_K = \lambda$, need $\lambda \sim \sqrt{\log K/K}$ (for large $K$)
- But AMP (and conjecturally, any poly-time algorithm) requires $\lambda_k > 1$ for some $k$
Results for $U(1)$ synchronization

$$Y^{(k)} = \frac{\lambda_k}{n} x^k x^* + \frac{1}{\sqrt{n}} \mathcal{W}^{(k)} \quad \text{for } k = 1, \ldots, K$$

- Single frequency: given $Y^{(k)}$, can non-trivially estimate $x^k$ iff $\lambda_k > 1$

- Information-theoretically, with $\lambda_1 = \cdots = \lambda_K = \lambda$, need $\lambda \sim \sqrt{\log K / K}$ (for large $K$)

- But AMP (and conjecturally, any poly-time algorithm) requires $\lambda_k > 1$ for some $k$
  - Computationally hard to synthesize sub-critical ($\lambda \leq 1$) frequencies
Results for $U(1)$ synchronization

\[ Y^{(k)} = \frac{\lambda_k}{n} x^k x^* + \frac{1}{\sqrt{n}} W^{(k)} \quad \text{for } k = 1, \ldots, K \]

- Single frequency: given $Y^{(k)}$, can non-trivially estimate $x^k$ iff $\lambda_k > 1$

- Information-theoretically, with $\lambda_1 = \cdots = \lambda_K = \lambda$, need $\lambda \sim \sqrt{\log K / K}$ (for large $K$)

- But AMP (and conjecturally, any poly-time algorithm) requires $\lambda_k > 1$ for some $k$
  - Computationally hard to synthesize sub-critical ($\lambda \leq 1$) frequencies

- But once above the $\lambda = 1$ threshold, adding frequencies helps reduce MSE of AMP
Results for $U(1)$ synchronization

Solid: AMP ($n = 100$)  
Dotted: theoretical ($n \to \infty$)  
Same $\lambda$ on each frequency

($K = \text{num freq}$)

---

Image credit: Perry, W., Bandeira, Moitra, *Message-passing algorithms for synchronization problems over compact groups*, to appear in CPAM
General groups

All of the above extends to any compact group

- E.g. Any finite group; $SO(3)$
General groups

All of the above extends to any compact group

▶ E.g. Any finite group; $SO(3)$

How to even define the model?

▶ Need to add “noise” to a group element $g_i g_j^{-1}$
General groups

All of the above extends to any compact group

- E.g. Any finite group; SO(3)

How to even define the model?

- Need to add “noise” to a group element $g_i g_j^{-1}$

**Answer:** Use representation theory to represent a group element as a matrix (and then add Gaussian noise)
General groups

All of the above extends to any compact group

▶ E.g. Any finite group; $SO(3)$

How to even define the model?

▶ Need to add “noise” to a group element $g_i g_j^{-1}$

Answer: Use representation theory to represent a group element as a matrix (and then add Gaussian noise)

▶ A representation $\rho$ of $G$ is a way to assign a matrix $\rho(g)$ to each $g \in G$

▶ Formally, a homomorphism $\rho : G \to \text{GL}(\mathbb{C}^d) = \{d \times d \text{ invertible matrices}\}$
General groups

All of the above extends to any compact group

- E.g. Any finite group; $SO(3)$

How to even define the model?

- Need to add “noise” to a group element $g_i g_j^{-1}$

**Answer:** Use representation theory to represent a group element as a matrix (and then add Gaussian noise)

- A representation $\rho$ of $G$ is a way to assign a matrix $\rho(g)$ to each $g \in G$

- Formally, a homomorphism
  \[
  \rho : G \rightarrow \text{GL}(\mathbb{C}^d) = \{d \times d \text{ invertible matrices}\}
  \]

Frequencies are replaced by irreducible representations of $G$

- Fourier theory for functions $G \rightarrow \mathbb{C}$
General groups

All of the above extends to any compact group

- E.g. Any finite group; $SO(3)$

How to even define the model?

- Need to add “noise” to a group element $g_i g_j^{-1}$

Answer: Use representation theory to represent a group element as a matrix (and then add Gaussian noise)

- A representation $\rho$ of $G$ is a way to assign a matrix $\rho(g)$ to each $g \in G$

- Formally, a homomorphism $\rho : G \rightarrow \text{GL}(\mathbb{C}^d) = \{d \times d \text{ invertible matrices}\}$

Frequencies are replaced by irreducible representations of $G$

- Fourier theory for functions $G \rightarrow \mathbb{C}$

For $U(1)$, 1D irreducible representation for each $k$: $\rho_k(g) = g^k$
Part II: Orbit Recovery
Back to cryo-EM

Image credit: [Singer, Shkolnisky ’11]
Synchronization is not the ideal model for cryo-EM
Back to cryo-EM

Synchronization is not the ideal model for cryo-EM

- The synchronization approach disregards the underlying signal (the molecule)
Synchronization is not the ideal model for cryo-EM

- The synchronization approach disregards the underlying signal (the molecule)
- Our Gaussian synchronization model assumes independent noise on each pair $i, j$ of images, whereas actually there is independent noise on each image
Synchronization is not the ideal model for cryo-EM

- The synchronization approach disregards the underlying signal (the molecule)
- Our Gaussian synchronization model assumes independent noise on each pair $i,j$ of images, whereas actually there is independent noise on each image
- For high noise, it is impossible to reliably recover the rotations
  - So we should not try to estimate the rotations!
Orbit recovery problem

Let $G$ be a **compact** group acting **linearly** and **continuously** on a finite-dimensional real vector space $V = \mathbb{R}^p$. 
Orbit recovery problem

Let $G$ be a compact group acting linearly and continuously on a finite-dimensional real vector space $V = \mathbb{R}^p$.

- Compact: e.g. any finite group, $SO(2)$, $SO(3)$
Orbit recovery problem

Let $G$ be a compact group acting linearly and continuously on a finite-dimensional real vector space $V = \mathbb{R}^p$.

- Compact: e.g. any finite group, $SO(2)$, $SO(3)$

- Linear: $\rho : G \to \text{GL}(V) = \{\text{invertible } p \times p \text{ matrices}\}$ (homomorphism)
**Orbit recovery problem**

Let $G$ be a compact group acting linearly and continuously on a finite-dimensional real vector space $V = \mathbb{R}^p$.

- **Compact**: e.g. any finite group, $SO(2)$, $SO(3)$

- **Linear**: $\rho : G \to \text{GL}(V) = \{\text{invertible } p \times p \text{ matrices}\}$ (homomorphism)

- **Action**: $g \cdot x = \rho(g)x$ for $g \in G, x \in V$
Orbit recovery problem

Let $G$ be a **compact** group acting **linearly** and **continuously** on a finite-dimensional real vector space $V = \mathbb{R}^p$.

- **Compact:** e.g. any finite group, $SO(2)$, $SO(3)$
- **Linear:** $\rho: G \to GL(V) = \{\text{invertible } p \times p \text{ matrices}\}$ (homomorphism)
- **Action:** $g \cdot x = \rho(g)x$ for $g \in G, x \in V$
- **Continuous:** $\rho$ is continuous
Orbit recovery problem

Let $G$ be a compact group acting linearly and continuously on a finite-dimensional real vector space $V = \mathbb{R}^p$. 

For $i = 1, \ldots, n$ observe $y_i = g_i \cdot x + \epsilon_i$ where $\ldots$ $g_i \sim \text{Haar}(G)$ (“uniform distribution” on $G$) $\quad \epsilon_i \sim \mathcal{N}(0, \sigma^2 I_p)$ (noise) $\quad$ Goal: Recover some $\tilde{x}$ in the orbit $\{g \cdot x : g \in G\}$ of $x$. 

24 / 39
Orbit recovery problem

Let $G$ be a compact group acting linearly and continuously on a finite-dimensional real vector space $V = \mathbb{R}^p$.

Unknown signal $x \in V$ (e.g. the molecule)
Orbit recovery problem

Let $G$ be a compact group acting linearly and continuously on a finite-dimensional real vector space $V = \mathbb{R}^p$.

Unknown signal $x \in V$ (e.g. the molecule)

For $i = 1, \ldots, n$ observe $y_i = g_i \cdot x + \epsilon_i$ where...

- $g_i \sim \text{Haar}(G)$ (“uniform distribution” on $G$)
- $\epsilon_i \sim \mathcal{N}(0, \sigma^2 I_p)$ (noise)
Orbit recovery problem

Let $G$ be a compact group acting linearly and continuously on a finite-dimensional real vector space $V = \mathbb{R}^p$.

Unknown signal $x \in V$ (e.g. the molecule)

For $i = 1, \ldots, n$ observe $y_i = g_i \cdot x + \varepsilon_i$ where...

- $g_i \sim \text{Haar}(G)$ ("uniform distribution" on $G$)
- $\varepsilon_i \sim \mathcal{N}(0, \sigma^2 I_p)$ (noise)

Goal: Recover some $\tilde{x}$ in the orbit $\{g \cdot x : g \in G\}$ of $x$
Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

Image credit: Jonathan Weed

Method of invariants \[1,2\]: measure features of the signal \( x \) that are shift-invariant

Degree-1:
\[ \sum_i x_i \text{ (mean)} \]

Degree-2:
\[ \sum_i x_i^2, x_1 x_2 + x_2 x_3 + \cdots + x_p x_1, \ldots \text{ (autocorrelation)} \]

Degree-3:
\[ x_1 x_2 x_4 + x_2 x_3 x_5 + \cdots \text{ (triple correlation)} \]

Invariant features are easy to estimate from the samples


Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

Method of invariants [1,2] : measure features of the signal \( x \) that are shift-invariant

---


Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

Method of invariants \([1,2]\) : measure features of the signal \( x \) that are shift-invariant

Degree-1:

---

Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

**Method of invariants** [1,2]: measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \)

---

Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \] acts on \( \mathbb{R}^p \) via cyclic shifts

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \epsilon_i \) with \( \epsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

Method of invariants [1,2]: measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

**Method of invariants** [1,2] : measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2:

---


Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \] acts on \( \mathbb{R}^p \) via cyclic shifts

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \epsilon_i \) with \( \epsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

**Method of invariants** [1,2] : measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2: \( \sum_i x_i^2 \)


Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \epsilon_i \) with \( \epsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

**Method of invariants** [1,2] : measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2: \( \sum_i x_i^2, \quad x_1x_2 + x_2x_3 + \cdots + x_px_1, \ldots \)

---


Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

Method of invariants \([1,2]\) : measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2: \( \sum_i x_i^2, \ x_1 x_2 + x_2 x_3 + \cdots + x_p x_1, \ldots \) (autocorrelation)

Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \] acts on \( \mathbb{R}^p \) via cyclic shifts

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

**Method of invariants** [1,2] : measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2: \( \sum_i x_i^2, \quad x_1x_2 + x_2x_3 + \cdots + x_p x_1, \ldots \) (autocorrelation)

Degree-3:

---

Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \epsilon_i \) with \( \epsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

**Method of invariants** [1,2] : measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2: \( \sum_i x_i^2, \ x_1 x_2 + x_2 x_3 + \cdots + x_p x_1, \ldots \) (autocorrelation)

Degree-3: \( x_1 x_2 x_4 + x_2 x_3 x_5 + \ldots \)

---

Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

**Method of invariants** [1,2]: measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2: \( \sum_i x_i^2, \quad x_1 x_2 + x_2 x_3 + \cdots + x_p x_1, \ldots \) (autocorrelation)

Degree-3: \( x_1 x_2 x_4 + x_2 x_3 x_5 + \ldots \) (triple correlation)

Special case: multi-reference alignment (MRA)

\[ G = \mathbb{Z}/p \text{ acts on } \mathbb{R}^p \text{ via cyclic shifts} \]

For \( i = 1, \ldots, n \) observe \( y_i = g_i \cdot x + \varepsilon_i \) with \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

Method of invariants [1,2]: measure features of the signal \( x \) that are shift-invariant

Degree-1: \( \sum_i x_i \) (mean)

Degree-2: \( \sum_i x_i^2, x_1 x_2 + x_2 x_3 + \cdots + x_p x_1, \ldots \) (autocorrelation)

Degree-3: \( x_1 x_2 x_4 + x_2 x_3 x_5 + \cdots \) (triple correlation)

Invariant features are easy to estimate from the samples

Sample complexity

**Theorem [1]:**
(Upper bound) With noise level $\sigma$, can estimate degree-$d$ invariants using $n = O(\sigma^2 d)$ samples.

Sample complexity

**Theorem [1]:**
(Upper bound) With noise level $\sigma$, can estimate degree-$d$ invariants using $n = O(\sigma^{2d})$ samples.
(Lower bound) If $x^{(1)}, x^{(2)}$ agree on all invariants of degree $\leq d - 1$ then $\Omega(\sigma^{2d})$ samples are required to distinguish them.
  ▶ Method of invariants is optimal

---

Sample complexity

Theorem [1]:
(Upper bound) With noise level $\sigma$, can estimate degree-$d$ invariants using $n = O(\sigma^{2d})$ samples.
(Lower bound) If $x^{(1)}, x^{(2)}$ agree on all invariants of degree $\leq d - 1$ then $\Omega(\sigma^{2d})$ samples are required to distinguish them.

▶ Method of invariants is optimal

Question: What degree $d^*$ of invariants do we need to learn before we can recover (the orbit of) $x$?

▶ Optimal sample complexity is $n = \Theta(\sigma^{2d^*})$

---

Sample complexity

**Theorem [1]:**
(Upper bound) With noise level $\sigma$, can estimate degree-$d$ invariants using $n = O(\sigma^{2d})$ samples.
(Lower bound) If $x^{(1)}, x^{(2)}$ agree on all invariants of degree $\leq d - 1$ then $\Omega(\sigma^{2d})$ samples are required to distinguish them.

▶ **Method of invariants is optimal**

**Question:** What degree $d^*$ of invariants do we need to learn before we can recover (the orbit of) $x$?

▶ **Optimal sample complexity is** $n = \Theta(\sigma^{2d^*})$

**Answer (for MRA) [1]:**

▶ For “generic” $x$, degree 3 is sufficient, so sample complexity $n = \Theta(\sigma^6)$

Sample complexity

**Theorem [1]:**
(Upper bound) With noise level $\sigma$, can estimate degree-$d$ invariants using $n = O(\sigma^{2d})$ samples.
(Lower bound) If $x^{(1)}, x^{(2)}$ agree on all invariants of degree $\leq d - 1$ then $\Omega(\sigma^{2d})$ samples are required to distinguish them.

- Method of invariants is optimal

**Question:** What degree $d^*$ of invariants do we need to learn before we can recover (the orbit of) $x$?

- Optimal sample complexity is $n = \Theta(\sigma^{2d^*})$

**Answer (for MRA) [1]:**

- For “generic” $x$, degree 3 is sufficient, so sample complexity $n = \Theta(\sigma^6)$

- But for a measure-zero set of “bad” signals, need much higher degree (as high as $p$)

---

Another viewpoint: mixtures of Gaussians

MRA sample: \( y = g \cdot x + \varepsilon \) with \( g \sim G, \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \)
Another viewpoint: mixtures of Gaussians

MRA sample: $y = g \cdot x + \varepsilon$ with $g \sim G$, $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$

The distribution of $y$ is a (uniform) mixture of $|G|$ Gaussians centered at \{\(g \cdot x : g \in G\)\}
Another viewpoint: mixtures of Gaussians

MRA sample: \( y = g \cdot x + \varepsilon \) with \( g \sim G, \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \)

The distribution of \( y \) is a (uniform) mixture of \(|G|\) Gaussians centered at \( \{g \cdot x : g \in G\} \)

▶ For infinite groups, a mixture of infinitely-many Gaussians
Another viewpoint: mixtures of Gaussians

MRA sample: $y = g \cdot x + \varepsilon$ with $g \sim G$, $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$

The distribution of $y$ is a (uniform) mixture of $|G|$ Gaussians centered at $\{g \cdot x : g \in G\}$

- For infinite groups, a mixture of infinitely-many Gaussians

Method of moments: Estimate moments $\mathbb{E}[y], \mathbb{E}[yy^\top], \ldots, \mathbb{E}[y^\otimes d]$
Another viewpoint: mixtures of Gaussians

MRA sample: \( y = g \cdot x + \varepsilon \) with \( g \sim G, \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \)

The distribution of \( y \) is a (uniform) mixture of \(|G|\) Gaussians centered at \( \{g \cdot x : g \in G\} \)

- For infinite groups, a mixture of infinitely-many Gaussians

Method of moments: Estimate moments \( \mathbb{E}[y], \mathbb{E}[yy^\top], \ldots, \mathbb{E}[y\otimes d] \)

\[
\mathbb{E}[y\otimes k] \sim \mathbb{E}_g [(g \cdot x)\otimes k]
\]
Another viewpoint: mixtures of Gaussians

MRA sample: \(y = g \cdot x + \varepsilon\) with \(g \sim G, \varepsilon \sim \mathcal{N}(0, \sigma^2 I)\)

The distribution of \(y\) is a (uniform) mixture of \(|G|\) Gaussians centered at \(\{g \cdot x : g \in G\}\)

- For infinite groups, a mixture of infinitely-many Gaussians

**Method of moments:** Estimate moments \(\mathbb{E}[y], \mathbb{E}[yy^\top], \ldots, \mathbb{E}[y^\otimes d]\)

\[\mathbb{E}[y^\otimes k] \sim \mathbb{E}_g[(g \cdot x)^\otimes k]\]

**Fact:** Moments are equivalent to invariants

- \(\mathbb{E}_g[(g \cdot x)^\otimes k]\) contains the same information as the degree-\(k\) invariant polynomials
Our contributions

Joint work with Ben Blum-Smith, Afonso Bandeira, Amelia Perry, Jonathan Weed

Bandeira, Blum-Smith, Perry, Weed, W., *Estimation under group actions: recovering orbits from invariants*, 2017
Our contributions

Joint work with Ben Blum-Smith, Afonso Bandeira, Amelia Perry, Jonathan Weed

▶ We generalize from MRA to any compact group

Bandeira, Blum-Smith, Perry, Weed, W., Estimation under group actions: recovering orbits from invariants, 2017
Our contributions

Joint work with Ben Blum-Smith, Afonso Bandeira, Amelia Perry, Jonathan Weed

▶ We generalize from MRA to any compact group

▶ Again, the method of invariants/moments is optimal

Bandeira, Blum-Smith, Perry, Weed, W., *Estimation under group actions: recovering orbits from invariants*, 2017
Our contributions

Joint work with Ben Blum-Smith, Afonso Bandeira, Amelia Perry, Jonathan Weed

- We generalize from MRA to any compact group
- Again, the method of invariants/moments is optimal
- We give an (inefficient) algorithm that achieves optimal sample complexity: solve polynomial system

Bandeira, Blum-Smith, Perry, Weed, W., Estimation under group actions: recovering orbits from invariants, 2017
Our contributions

Joint work with Ben Blum-Smith, Afonso Bandeira, Amelia Perry, Jonathan Weed

- We generalize from MRA to any compact group
- Again, the method of invariants/moments is optimal
- We give an (inefficient) algorithm that achieves optimal sample complexity: solve polynomial system
- To determine what degree of invariants are required, we use invariant theory and algebraic geometry
  - How to tell if polynomial equations have a unique solution

Bandeira, Blum-Smith, Perry, Weed, W., Estimation under group actions: recovering orbits from invariants, 2017
Invariant theory

Variables $x_1, \ldots, x_p$ (corresponding to the coordinates of $x$)
Invariant theory

Variables $x_1, \ldots, x_p$ (corresponding to the coordinates of $x$)

The invariant ring $\mathbb{R}[x]^G$ is the subring of $\mathbb{R}[x] := \mathbb{R}[x_1, \ldots, x_p]$ consisting of polynomials $f$ such that $f(g \cdot x) = f(x) \ \forall g \in G$. 
Invariant theory

Variables $x_1, \ldots, x_p$ (corresponding to the coordinates of $x$)

The invariant ring $\mathbb{R}[x]^G$ is the subring of $\mathbb{R}[x] := \mathbb{R}[x_1, \ldots, x_p]$ consisting of polynomials $f$ such that $f(g \cdot x) = f(x) \quad \forall g \in G$.

- Aside: A main result of invariant theory is that $\mathbb{R}[x]^G$ is finitely-generated
Invariant theory

Variables $x_1, \ldots, x_p$ (corresponding to the coordinates of $x$)

The invariant ring $\mathbb{R}[x]^G$ is the subring of $\mathbb{R}[x] := \mathbb{R}[x_1, \ldots, x_p]$ consisting of polynomials $f$ such that $f(g \cdot x) = f(x)$ $\forall g \in G$.

- Aside: A main result of invariant theory is that $\mathbb{R}[x]^G$ is finitely-generated

$\mathbb{R}[x]^G_{\leq d}$ – invariants of degree $\leq d$
Invariant theory

Variables $x_1, \ldots, x_p$ (corresponding to the coordinates of $x$)

The invariant ring $\mathbb{R}[x]^G$ is the subring of $\mathbb{R}[x] := \mathbb{R}[x_1, \ldots, x_p]$ consisting of polynomials $f$ such that $f(g \cdot x) = f(x)$ $\forall g \in G$.

- Aside: A main result of invariant theory is that $\mathbb{R}[x]^G$ is finitely-generated

$\mathbb{R}[x]^G_{\leq d}$ – invariants of degree $\leq d$

(Simple) algorithm:

- Pick $d^*$ (to be chosen later)
- Using $\Theta(\sigma^{2d^*})$ samples, estimate invariants up to degree $d^*$:
  learn value $f(x)$ for all $f \in \mathbb{R}[x]^G_{\leq d}$
- Solve for an $\hat{x}$ that is consistent with those values:
  $f(\hat{x}) = f(x)$ $\forall f \in \mathbb{R}[x]^G_{\leq d}$ (polynomial system of equations)
All invariants determine orbit

**Theorem** [1]: If $G$ is compact, for every $x \in V$, the full invariant ring $\mathbb{R}[x]^G$ determines $x$ up to orbit.

- In the sense that if $x, x'$ do not lie in the same orbit, there exists $f \in \mathbb{R}[x]^G$ that separates them: $f(x) \neq f(x')$

---


All invariants determine orbit

**Theorem [1]:** If $G$ is compact, for every $x \in V$, the full invariant ring $\mathbb{R}[x]^G$ determines $x$ up to orbit.

- In the sense that if $x, x'$ do not lie in the same orbit, there exists $f \in \mathbb{R}[x]^G$ that separates them: $f(x) \neq f(x')$.

**Corollary:** Suppose that for some $d$, $\mathbb{R}[x]_{\leq d}^G$ generates $\mathbb{R}[x]^G$ (as an $\mathbb{R}$-algebra). Then $\mathbb{R}[x]_{\leq d}^G$ determines $x$ up to orbit and so sample complexity is $O(\sigma^{2d})$.

---


All invariants determine orbit

**Theorem [1]:** If $G$ is compact, for every $x \in V$, the full invariant ring $\mathbb{R}[x]^G$ determines $x$ up to orbit.

- In the sense that if $x, x'$ do not lie in the same orbit, there exists $f \in \mathbb{R}[x]^G$ that separates them: $f(x) \neq f(x')$

**Corollary:** Suppose that for some $d$, $\mathbb{R}[x]_{\leq d}^G$ generates $\mathbb{R}[x]^G$ (as an $\mathbb{R}$-algebra). Then $\mathbb{R}[x]_{\leq d}^G$ determines $x$ up to orbit and so sample complexity is $O(\sigma^{2d})$.

**Problem:** This is for worst-case $x \in V$. For MRA (cyclic shifts) this requires $d = p$ whereas generic $x$ only requires $d = 3$ [2].

---


All invariants determine orbit

**Theorem** [1]: If $G$ is compact, for every $x \in V$, the full invariant ring $\mathbb{R}[x]^G$ determines $x$ up to orbit.

- In the sense that if $x, x'$ do not lie in the same orbit, there exists $f \in \mathbb{R}[x]^G$ that separates them: $f(x) \neq f(x')$

**Corollary**: Suppose that for some $d$, $\mathbb{R}[x]_{\leq d}^G$ generates $\mathbb{R}[x]^G$ (as an $\mathbb{R}$-algebra). Then $\mathbb{R}[x]_{\leq d}^G$ determines $x$ up to orbit and so sample complexity is $O(\sigma^{2d})$.

**Problem**: This is for worst-case $x \in V$. For MRA (cyclic shifts) this requires $d = p$ whereas generic $x$ only requires $d = 3$ [2].

Actually care about whether $\mathbb{R}[x]_{\leq d}^G$ generically determines $\mathbb{R}[x]^G$

---


Do polynomials \textit{generically} determine other polynomials?

Say we have $A \subseteq B \subseteq \mathbb{R}[x]$

- (Technically need to assume $B$ is finitely generated)
Do polynomials generically determine other polynomials?

Say we have $A \subseteq B \subseteq \mathbb{R}[x]$

- (Technically need to assume $B$ is finitely generated)

**Question:** Do the values $\{a(x) : a \in A\}$ generically determine the values $\{b(x) : b \in B\}$?
Do polynomials **generically** determine other polynomials?

Say we have $A \subseteq B \subseteq \mathbb{R}[x]$

- (Technically need to assume $B$ is finitely generated)

**Question**: Do the values $\{a(x) : a \in A\}$ generically determine the values $\{b(x) : b \in B\}$?

**Definition**: Polynomials $f_1, \ldots, f_m$ are **algebraically independent** if there is no $P \in \mathbb{R}[y_1, \ldots, y_m]$ with $P(f_1, \ldots, f_m) \equiv 0$. 

- "Generic": $x$ lies in a particular full-measure set.
Do polynomials **generically** determine other polynomials?

Say we have $A \subseteq B \subseteq \mathbb{R}[x]$

- (Technically need to assume $B$ is finitely generated)

**Question:** Do the values $\{a(x) : a \in A\}$ generically determine the values $\{b(x) : b \in B\}$?

**Definition:** Polynomials $f_1, \ldots, f_m$ are algebraically independent if there is no $P \in \mathbb{R}[y_1, \ldots, y_m]$ with $P(f_1, \ldots, f_m) \equiv 0$.

**Definition:** For $U \subseteq \mathbb{R}[x]$, the **transcendence degree** $\text{trdeg}(U)$ is the number of algebraically independent polynomials in $U$. 

---

31 / 39
Do polynomials generically determine other polynomials?

Say we have $A \subseteq B \subseteq \mathbb{R}[x]$

- (Technically need to assume $B$ is finitely generated)

**Question:** Do the values $\{a(x) : a \in A\}$ generically determine the values $\{b(x) : b \in B\}$?

**Definition:** Polynomials $f_1, \ldots, f_m$ are algebraically independent if there is no $P \in \mathbb{R}[y_1, \ldots, y_m]$ with $P(f_1, \ldots, f_m) \equiv 0$.

**Definition:** For $U \subseteq \mathbb{R}[x]$, the transcendence degree $\text{trdeg}(U)$ is the number of algebraically independent polynomials in $U$.

**Answer:** Suppose $\text{trdeg}(A) = \text{trdeg}(B)$. If $x$ is “generic” then the values $\{a(x) : a \in A\}$ determine a finite number of possibilities for the entire collection $\{b(x) : b \in B\}$.

- “Generic”: $x$ lies in a particular full-measure set
How to test algebraic independence?

This is actually easy!

Theorem (Jacobian criterion):
Polynomials $f_1, \ldots, f_m \in \mathbb{R}[x_1, \ldots, x_p]$ are algebraically independent if and only if the $m \times p$ Jacobian matrix $J_{ij} = \frac{\partial f_i}{\partial x_j}$ has full row rank. (Still true if you evaluate $J$ at a generic point $x$.)

▶ Why: Tests whether map $(x_1, \ldots, x_p) \mapsto (f_1(x), \ldots, f_m(x))$ is locally surjective.
How to test algebraic independence?

This is actually easy!
How to test algebraic independence?

This is actually easy!

**Theorem (Jacobian criterion):**
Polynomials \( f_1, \ldots, f_m \in \mathbb{R}[x_1, \ldots, x_p] \) are algebraically independent if and only if the \( m \times p \) Jacobian matrix \( J_{ij} = \frac{\partial f_i}{\partial x_j} \) has full row rank. (Still true if you evaluate \( J \) at a generic point \( x \).)
How to test algebraic independence?

This is actually easy!

**Theorem (Jacobian criterion):**

Polynomials $f_1, \ldots, f_m \in \mathbb{R}[x_1, \ldots, x_p]$ are algebraically independent if and only if the $m \times p$ Jacobian matrix $J_{ij} = \frac{\partial f_i}{\partial x_j}$ has full row rank. (Still true if you evaluate $J$ at a generic point $x$.)

- Why: Tests whether map $(x_1, \ldots, x_p) \mapsto (f_1(x), \ldots, f_m(x))$ is locally surjective
Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for generic list recovery.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$. 
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for generic list recovery.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group \( G \) and action on \( V \)) and outputs the degree \( d^* \) of invariants required for generic list recovery.

- List recovery: output a finite list \( \hat{x}^{(1)}, \hat{x}^{(2)}, \ldots \), one of which (approximately) lies in the orbit of the true \( x \)
- List recovery may be good enough in practice?

Procedure:
Generic list recovery

Our **main result** is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for **generic list recovery**.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?

**Procedure:**

- Need to test whether $\mathbb{R}[x]_G^{\leq d}$ determines $\mathbb{R}[x]^G$ (generically)
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for **generic list recovery**.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?

**Procedure:**

- Need to test whether $\mathbb{R}[x]^G_{\leq d}$ determines $\mathbb{R}[x]^G$ (generically)
- So need to check if $\text{trdeg}(\mathbb{R}[x]^G_{\leq d}) = \text{trdeg}(\mathbb{R}[x]^G)$
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for generic list recovery.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$.
- List recovery may be good enough in practice?

Procedure:

- Need to test whether $\mathbb{R}[x]^G_{\leq d}$ determines $\mathbb{R}[x]^G$ (generically).
- So need to check if $\text{trdeg}(\mathbb{R}[x]^G_{\leq d}) = \text{trdeg}(\mathbb{R}[x]^G)$.
- $\text{trdeg}(\mathbb{R}[x]^G)$ is easy: $\dim(x) - \dim(\text{orbit})$. 
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for generic list recovery.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?

Procedure:

- Need to test whether $\mathbb{R}[x]_{\leq d}^G$ determines $\mathbb{R}[x]^G$ (generically)
- So need to check if $\text{trdeg}(\mathbb{R}[x]_{\leq d}^G) = \text{trdeg}(\mathbb{R}[x]^G)$
- $\text{trdeg}(\mathbb{R}[x]^G)$ is easy: $\dim(x) - \dim(\text{orbit})$
- $\text{trdeg}(\mathbb{R}[x]_{\leq d}^G)$ via Jacobian criterion
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for generic list recovery.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?

Comments:
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for generic list recovery.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?

Comments:

- For e.g. MRA (cyclic shifts), need to test each $p$ separately on a computer
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for \textit{generic list recovery}.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?

Comments:

- For e.g. MRA (cyclic shifts), need to test each $p$ separately on a computer
- Not an efficient algorithm to solve any particular instance
Generic list recovery

Our main result is an efficient procedure that takes the problem setup as input (group $G$ and action on $V$) and outputs the degree $d^*$ of invariants required for generic list recovery.

- List recovery: output a finite list $\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots$, one of which (approximately) lies in the orbit of the true $x$
- List recovery may be good enough in practice?

Comments:

- For e.g. MRA (cyclic shifts), need to test each $p$ separately on a computer
- Not an efficient algorithm to solve any particular instance
- There is also an algorithm to bound the size of the list (or test for unique recovery), but it is not efficient (Gröbner bases)
Generalized orbit recovery problem

Extensions:

Projection (e.g. cryo-EM):

\[ y_i = \Pi(g_i \cdot x) + \epsilon_i \]

\[ \Pi : V \rightarrow W \quad \text{linear} \]

\[ \epsilon_i \sim N(0, \sigma^2 I) \]

Heterogeneity:

\[ K \text{ signals } x(1), \ldots, x(K) \]

Mixing weights \( \{w_1, \ldots, w_K\} \in \Delta K \)

\[ y_i = \Pi(g_i \cdot x(k_i)) + \epsilon_i \]

\[ k_i \sim \{1, \ldots, K\} \quad \text{according to } w \]

Same methods apply!

Order-d moments now only give access to a particular subspace of \( R^G \)

For heterogeneity, work over a bigger group \( G^K \) acting on \( (x(1), \ldots, x(K)) \in V \oplus K \).
Generalized orbit recovery problem

Extensions:

- Projection (e.g. cryo-EM):
  - Observe \( y_i = \Pi(g_i \cdot x) + \varepsilon_i \)
  - \( \Pi : V \rightarrow W \) linear
  - \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)
Generalized orbit recovery problem

Extensions:

- **Projection (e.g. cryo-EM):**
  - Observe $y_i = \Pi(g_i \cdot x) + \varepsilon_i$
  - $\Pi : V \rightarrow W$ linear
  - $\varepsilon_i \sim N(0, \sigma^2 I)$

- **Heterogeneity:**
  - $K$ signals $x^{(1)}, \ldots, x^{(K)}$
  - Mixing weights $(w_1, \ldots, w_K) \in \Delta_K$
  - Observe $y_i = \Pi(g_i \cdot x^{(k_i)}) + \varepsilon_i$
  - $k_i \sim \{1, \ldots, K\}$ according to $w$
Generalized orbit recovery problem

Extensions:

- **Projection (e.g. cryo-EM):**
  - Observe $y_i = \Pi(g_i \cdot x) + \varepsilon_i$
  - $\Pi : V \rightarrow W$ linear
  - $\varepsilon_i \sim \mathcal{N}(0, \sigma^2 I)$

- **Heterogeneity:**
  - $K$ signals $x^{(1)}, \ldots, x^{(K)}$
  - Mixing weights $(w_1, \ldots, w_K) \in \Delta_K$
  - Observe $y_i = \Pi(g_i \cdot x^{(k_i)}) + \varepsilon_i$
  - $k_i \sim \{1, \ldots, K\}$ according to $w$

Same methods apply!
Generalized orbit recovery problem

Extensions:

- **Projection (e.g. cryo-EM):**
  - Observe \( y_i = \Pi(g_i \cdot x) + \varepsilon_i \)
  - \( \Pi : V \rightarrow W \) linear
  - \( \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I) \)

- **Heterogeneity:**
  - \( K \) signals \( x^{(1)}, \ldots, x^{(K)} \)
  - Mixing weights \( (w_1, \ldots, w_K) \in \Delta_K \)
  - Observe \( y_i = \Pi(g_i \cdot x^{(k_i)}) + \varepsilon_i \)
  - \( k_i \sim \{1, \ldots, K\} \) according to \( w \)

Same methods apply!

- **Order-\( d \) moments now only give access to a particular subspace of \( \mathbb{R}[x]^G \)**
Generalized orbit recovery problem

Extensions:

- **Projection (e.g. cryo-EM):**
  - Observe $y_i = \Pi(g_i \cdot x) + \varepsilon_i$
  - $\Pi : V \rightarrow W$ linear
  - $\varepsilon_i \sim \mathcal{N}(0, \sigma^2 I)$

- **Heterogeneity:**
  - $K$ signals $x^{(1)}, \ldots, x^{(K)}$
  - Mixing weights $(w_1, \ldots, w_K) \in \Delta_K$
  - Observe $y_i = \Pi(g_i \cdot x^{(k_i)}) + \varepsilon_i$
  - $k_i \sim \{1, \ldots, K\}$ according to $w$

Same methods apply!

- Order-$d$ moments now only give access to a particular subspace of $\mathbb{R}[x]^G$

- For heterogeneity, work over a bigger group $G^K$ acting on $(x^{(1)}, \ldots, x^{(K)}) \in V^{\oplus K}$
Results: cryo-EM

Our methods show that for cryo-EM, generic list recovery is possible at degree 3.
Results: cryo-EM

Our methods show that for cryo-EM, generic list recovery is possible at degree 3

So information-theoretic sample complexity is $\Theta(\sigma^6)$
Results: cryo-EM

Our methods show that for cryo-EM, generic list recovery is possible at degree 3

So information-theoretic sample complexity is $\Theta(\sigma^6)$

Ongoing work: polynomial time algorithm for cryo-EM
Efficient recovery: tensor decomposition

Restrict to finite group

Recall: with $O(\sigma^6)$ samples, can estimate the third moment:

$$T_3(x) = \sum_{g \in G} (g \cdot x)^{\otimes 3}$$

Efficient recovery: tensor decomposition

Restrict to finite group

Recall: with $O(\sigma^6)$ samples, can estimate the third moment:

$$T_3(x) = \sum_{g \in G} (g \cdot x) \otimes^3$$

This is an instance of tensor decomposition: Given $\sum_{i=1}^m a_i \otimes^3$ for some $a_1, \ldots, a_m \in \mathbb{R}^p$, recover $\{a_i\}$

Efficient recovery: tensor decomposition

Restrict to finite group

Recall: with $O(\sigma^6)$ samples, can estimate the third moment:

$$T_3(x) = \sum_{g \in G} (g \cdot x) \otimes^3$$

This is an instance of tensor decomposition: Given $\sum_{i=1}^{m} a_i \otimes^3$ for some $a_1, \ldots, a_m \in \mathbb{R}^p$, recover $\{a_i\}$

For MRA: since $m \leq p$ ("undercomplete") can apply Jennrich’s algorithm to decompose tensor efficiently [1]

Example: heterogeneous MRA

MRA with multiple signals $x^{(1)}, \ldots, x^{(K)}$

$$T_d(x) = \sum_{k=1}^{K} \sum_{g \in G} (g \cdot x^{(k)}) \otimes d$$

[1] Perry, Weed, Bandeira, Rigollet, Singer '17
[3] Ma, Shi, Steurer '16
Example: heterogeneous MRA

MRA with multiple signals $x^{(1)}, \ldots, x^{(K)}$

$$T_d(x) = \sum_{k=1}^{K} \sum_{g \in G} (g \cdot x^{(k)}) \otimes d$$

Jennrich’s algorithm works if given 5th moment $\leadsto n = O(\sigma^{10})$ [1]

[1] Perry, Weed, Bandeira, Rigollet, Singer ’17
Example: heterogeneous MRA

MRA with multiple signals $x^{(1)}, \ldots, x^{(K)}$

$$T_d(x) = \sum_{k=1}^{K} \sum_{g \in G} (g \cdot x^{(k)}) \otimes d$$

Jennrich’s algorithm works if given 5th moment $\Leftrightarrow n = O(\sigma^{10})$ [1]

Information-theoretically, 3rd moment suffices if $K \leq p/6$

---

[1] Perry, Weed, Bandeira, Rigollet, Singer ’17
Example: heterogeneous MRA

MRA with multiple signals \(x^{(1)}, \ldots, x^{(K)}\)

\[
T_d(x) = \sum_{k=1}^{K} \sum_{g \in G} (g \cdot x^{(k)}) \otimes d
\]

Jennrich’s algorithm works if given 5th moment \(\sim n = O(\sigma^{10})\) [1]

Information-theoretically, 3rd moment suffices if \(K \leq p/6\)

If signals \(x^{(k)}\) are random (i.i.d. Gaussian), conjectured that efficient recovery is possible from 3rd moment iff \(K \leq \sqrt{p}\) [2]

---

[1] Perry, Weed, Bandeira, Rigollet, Singer ‘17
Example: heterogeneous MRA

MRA with multiple signals $x^{(1)}, \ldots, x^{(K)}$

$$T_d(x) = \sum_{k=1}^{K} \sum_{g \in G} (g \cdot x^{(k)}) \otimes d$$

Jennrich’s algorithm works if given 5th moment $\leadsto n = O(\sigma^{10})$ [1]

Information-theoretically, 3rd moment suffices if $K \leq p/6$

If signals $x^{(k)}$ are random (i.i.d. Gaussian), conjectured that efficient recovery is possible from 3rd moment iff $K \leq \sqrt{p}$ [2]

New result (with A. Moitra): if $K \leq \sqrt{p}/\text{polylog}(p)$ then for random signals, efficient recovery is possible from 3rd moment

- Based on random overcomplete 3-tensor decomposition [3]

---

[1] Perry, Weed, Bandeira, Rigollet, Singer ’17
Acknowledgements
Acknowledgements

- Ankur Moitra
Acknowledgements

- Ankur Moitra
- Michel Goemans
Acknowledgements

▶ Ankur Moitra
▶ Michel Goemans
▶ Philippe Rigollet
Acknowledgements

- Ankur Moitra
- Michel Goemans
- Philippe Rigollet
- Afonso Bandeira
Acknowledgements

- Ankur Moitra
- Michel Goemans
- Philippe Rigollet
- Afonso Bandeira
- Collaborators
Acknowledgements

- Ankur Moitra
- Michel Goemans
- Philippe Rigollet
- Afonso Bandeira
- Collaborators
- Family
Acknowledgements

- Ankur Moitra
- Michel Goemans
- Philippe Rigollet
- Afonso Bandeira
- Collaborators
- Family
- Thank you!