Statistical Image Recovery: A Message-Passing Perspective

Phil Schniter

The Ohio State University

Collaborators: Sundeep Rangan (NYU) and Alyson Fletcher (UC Santa Cruz)

Supported in part by NSF grants CCF-1018368 and NSF grant CCF-1218754.

Duke University — April 10, 2015

Image Recovery

- In image recovery, we want to
  - recover a image $\boldsymbol{x} \in \mathbb{C}^N$
  - from corrupted measurements $\boldsymbol{y} \in \mathbb{C}^M$
  - of hidden linear transform outputs $\boldsymbol{z} = \boldsymbol{\Phi x} \in \mathbb{C}^M$.
- The measurement corruption mechanism might be
  - additive noise: $y_i = z_i + w_i$
  - phase-less: $y_i = |z_i + w_i|$
  - one-bit: $y_i = \text{sgn}(z_i + w_i)$
  - photon-limited (Poisson), etc...
- The image is structured in that $\boldsymbol{\Omega x} \in \mathbb{C}^D$ is ...
  - sparse (sufficiently few nonzeros)
  - co-sparse (sufficiently many zeros),

Statistical Approach to Image Recovery

In the statistical approach to image recovery...

- measurements modeled via likelihood $p(\boldsymbol{y}|\boldsymbol{x}) \propto \exp(-g(\boldsymbol{\Phi x}))$
- image modeled via prior distribution $p(\boldsymbol{x}) \propto \exp(-f(\boldsymbol{\Omega x}))$

- The posterior

$$p(\boldsymbol{x}|\boldsymbol{y}) = p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})/p(\boldsymbol{y}),$$

  tells *all* we can learn about $\boldsymbol{x}$ from $\boldsymbol{y}$, but is expensive to compute.

- Instead, one usually settles for point estimates like the
  - MAP estimate: $\hat{\boldsymbol{x}}_{\text{MAP}} = \arg\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y})$
  - MMSE estimate: $\hat{\boldsymbol{x}}_{\text{MMSE}} = \mathrm{E}\{\boldsymbol{x}|\boldsymbol{y}\} = \int_{\mathbb{C}^N} \boldsymbol{x}\, p(\boldsymbol{x}|\boldsymbol{y}) d\boldsymbol{x}$

  and perhaps marginal uncertainty information like $\mathrm{var}\{x_j|\boldsymbol{y}\}$.

MAP Estimation

- MAP estimation can be reformulated as

$$\begin{aligned}\hat{\boldsymbol{x}}_{\mathsf{MAP}} &= \arg\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) \\ &= \arg\min_{\boldsymbol{x}}\{-\ln p(\boldsymbol{x}|\boldsymbol{y})\} = \arg\min_{\boldsymbol{x}}\{-\ln p(\boldsymbol{y}|\boldsymbol{x}) - \ln p(\boldsymbol{x})\} \\ &= \arg\min_{\boldsymbol{x}} \underbrace{g(\boldsymbol{\Phi x})}_{\text{data fidelity}} + \underbrace{f(\boldsymbol{\Omega x})}_{\text{regularization}}\end{aligned}$$

  and thus viewed from a "non-statistical" perspective.

- We often choose g and f that are convex and separable

$$\begin{aligned}g(\boldsymbol{z}) &= \textstyle\sum_i g_i(z_i) \\ f(\boldsymbol{u}) &= \textstyle\sum_d f_d(u_d)\end{aligned}$$

  to facilitate efficient algorithms (e.g., $g(\boldsymbol{z}) = \|\boldsymbol{y} - \boldsymbol{z}\|_2^2$, $f(\boldsymbol{u}) = \|\boldsymbol{u}\|_1$).

Prototypical Optimization Algorithms

Iterative soft thresholding ($g(\boldsymbol{z}) = \frac{1}{2\sigma_w^2}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2, \boldsymbol{\Omega} = \boldsymbol{I}$):

for $t = 1, 2, 3, \ldots$

$$\boldsymbol{v}_t = \boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}_t \quad \text{residual}$$

$$\boldsymbol{x}_{t+1} = \text{prox}_{\tau f}\big(\boldsymbol{x}_t + \boldsymbol{\Phi}^{\mathsf{H}}\boldsymbol{v}_t\big) \quad \text{component-wise thresholding}$$

Forward-backward primal-dual[1] ($\boldsymbol{\Omega} = \boldsymbol{I}$):

for $t = 1, 2, 3, \ldots$

$$\tilde{\boldsymbol{s}}_{t+1} = \text{prox}_{\sigma g^*}(\boldsymbol{s}_t + \sigma\boldsymbol{\Phi}\boldsymbol{x}_n) \quad \text{proximal gradient ascent}$$

$$\hat{\boldsymbol{s}}_{t+1} = \theta\tilde{\boldsymbol{s}}_{t+1} + (1-\theta)\boldsymbol{s}_t \quad \text{relaxation, } \theta > 0$$

$$\tilde{\boldsymbol{x}}_{t+1} = \text{prox}_{\tau f}\big(\boldsymbol{x}_t - \tau\boldsymbol{\Phi}^{\mathsf{H}}\hat{\boldsymbol{s}}_{t+1}\big) \quad \text{proximal gradient descent}$$

$$\begin{bmatrix}\boldsymbol{x}_{t+1}\\ \boldsymbol{s}_{t+1}\end{bmatrix} = \beta_t\begin{bmatrix}\tilde{\boldsymbol{x}}_{t+1}\\ \tilde{\boldsymbol{s}}_{t+1}\end{bmatrix} + (1-\beta_t)\begin{bmatrix}\boldsymbol{x}_t\\ \boldsymbol{s}_t\end{bmatrix} \quad \text{relaxation, } \beta_t > 0$$

- $[\text{prox}_{\tau f}(\boldsymbol{r})]_d \triangleq \arg\min_x f_d(x) + \frac{1}{2\tau}|x - r_d|^2$ often in closed-form.
- No matrix inversions. Can leverage fast $\boldsymbol{\Phi}$ & $\boldsymbol{\Phi}^{\mathsf{H}}$ (e.g., FFT).

[1] Komodakis,Pesquet–arXiv:1406.5429

Questions

- How to choose stepsizes τ, σ and relaxation parameters like β_t?
- How to "tune" g and f to the data (e.g., noise variance, sparsity)?
- Is there a sacrifice in restricting g and f to be convex?
- Is there a sacrifice in pursuing MAP rather than MMSE?
  If so, how do we *efficiently* solve the MMSE problem?

$$\hat{\boldsymbol{x}}_{\mathsf{MMSE}} = \int_{\mathbb{C}^N} \boldsymbol{x}\, p(\boldsymbol{x}|\boldsymbol{y}) d\boldsymbol{x}$$

- How do we get marginal uncertainty information like $\text{var}\{x_j|\boldsymbol{y}\}$?

Next, I will describe a *fast* method that addresses *all* of these questions.

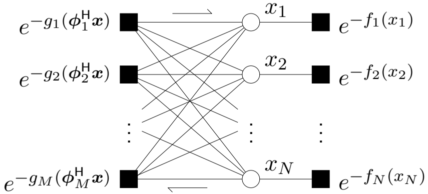
The 21st Century Approach: Crowd-Source It!

1) Factor the posterior, exposing the statistical structure of the problem:

$$p(\boldsymbol{x}|\boldsymbol{y}) \propto \prod_{i=1}^{M} e^{-g_i(\boldsymbol{\phi}_i^{\mathsf{H}}\boldsymbol{x})} \prod_{d=1}^{D} e^{-f_d(\boldsymbol{\omega}_d^{\mathsf{H}}\boldsymbol{x})},$$

Can visualize using the factor graph (drawn here for $\boldsymbol{\Omega} = \boldsymbol{I}, D = N$):

(White circles are random variables and black boxes are factors.)



2) Inference algorithm: Pass messages (pdfs) between nodes until they agree. In MMSE case, gives full marginal posteriors $p(x_j|\boldsymbol{y})$.

Next, suppose $\boldsymbol{\Omega} = \boldsymbol{I}$ (canonical sparsity) and rename $\boldsymbol{\Phi} \rightarrow \boldsymbol{A}$...

The Blessings of Dimensionality

In general, loops in the factor graph are bad!

But in the large-system limit, if $\boldsymbol{A}$ is i.i.d. sub-Gaussian then ...

- messages can be approximated as Gaussian due to CLT,
- differences between messages approximated via Taylor's expansion,[2]
  $\rightarrow$ Approximate Message Passing (AMP) algorithm
- per-iteration behavior characterized by a scalar state-evolution (SE),
- if SE has unique fixed point, it is MMSE/MAP optimal.[3]

In fact, AMP's SE can be used to characterize *fundamental* performance.

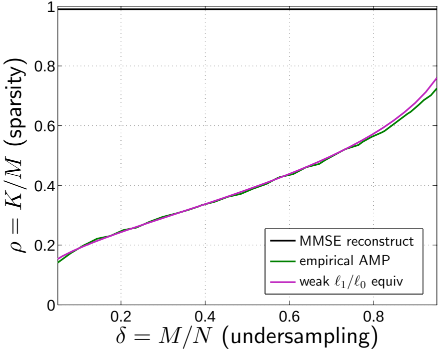
[2] Donoho,Maleki,Montanari–PNAS'09

[3] Bayati,Montanari–IT'11

Example Application of AMP State-Evolution Analysis

AMP SE yields a closed-form expression[4] for weak ℓ_1/ℓ_0 equivalence:

$$\rho(\delta) = \max_{c>0} \frac{1 - 2\delta^{-1}[(1+c^2)\Phi(-c) - c\phi(c)]}{1 + c^2 - 2[(1+c^2)\Phi(-c) - c\phi(c)]},$$



[4] Donoho,Maleki,Montanari–PNAS'09

AMP for Quadratic data-fidelity (i.e., AWGN)

MAP version of AMP ($g(\boldsymbol{z}) = \frac{1}{2\sigma_w^2}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2, \boldsymbol{\Omega} = \boldsymbol{I}$):

for $t = 1, 2, 3, \ldots$

$\boldsymbol{v}_t = \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_t + \frac{N}{M}\frac{\nu_t^x}{\tau_{t-1}}\boldsymbol{v}_{t-1}$	Onsager-corrected residual
$\tau_t = \sigma_w^2 + \frac{N}{M}\nu_t^x$ or $\frac{1}{M}\|\boldsymbol{v}_t\|_2^2$	error-variance of prox input
$\boldsymbol{x}_{t+1} = \text{prox}_{\tau_t f}(\boldsymbol{x}_t + \boldsymbol{A}^{\mathsf{H}}\boldsymbol{v}_t)$	component-wise thresholding
$\nu_{t+1}^x = \text{avg}\{\underbrace{\tau_t\, \text{prox}'_{\tau_t f}(\boldsymbol{x}_t + \boldsymbol{A}^{\mathsf{H}}\boldsymbol{v}_t)}\}$	error-variance of prox output
$\to \text{var}\{x_i \mid \boldsymbol{y}\}$	marginal uncertainty

- Onsager correction $\rightsquigarrow$ prox input an AWGN-corrupted version of true $\boldsymbol{x}$ (with error variance τ_t). Thus, prox becomes the scalar MAP denoiser!
- For MMSE-AMP, simply replace prox with scalar MMSE denoiser.

Generalized[5] AMP: Possibly non-quadratic data fidelity

Damped MAP GAMP ($\boldsymbol{\Omega} = \boldsymbol{I}$):

for $t = 1, 2, 3, \ldots$

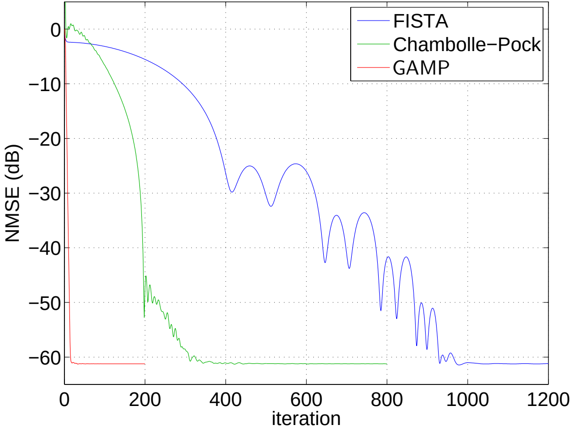
$1/\sigma_t = \nu_t^x \|\boldsymbol{A}\|_F^2 / M$	stepsize adaptation
$\tilde{\boldsymbol{s}}_{t+1} = \text{prox}_{\sigma_t g^*}(\boldsymbol{s}_t + \sigma_t \boldsymbol{A}\boldsymbol{x}_n)$	proximal gradient
$\nu_{t+1}^s = \text{avg}\{\sigma_t \, \text{prox}'_{\sigma_t g^*}(\boldsymbol{s}_t + \sigma_t \boldsymbol{A}\boldsymbol{x}_n)\}$	sensitivity
$1/\tau_t = \nu_{t+1}^s \|\boldsymbol{A}\|_F^2 / N$	stepsize adaptation
$\tilde{\boldsymbol{x}}_{t+1} = \text{prox}_{\tau_t f}(\boldsymbol{x}_t - \tau_t \boldsymbol{A}^{\mathsf{H}} \tilde{\boldsymbol{s}}_{t+1})$	proximal gradient $(\theta = 1)$
$\nu_{t+1}^x = \text{avg}\{\tau_t \, \text{prox}'_{\tau_t f}(\boldsymbol{x}_t - \tau_t \boldsymbol{A}^{\mathsf{H}} \hat{\boldsymbol{s}}_{t+1})\}$	sensitivity
$\begin{bmatrix} \boldsymbol{x}_{t+1} \\ \boldsymbol{s}_{t+1} \end{bmatrix} = \beta_t \begin{bmatrix} \tilde{\boldsymbol{x}}_{t+1} \\ \tilde{\boldsymbol{s}}_{t+1} \end{bmatrix} + (1 - \beta_t) \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{s}_t \end{bmatrix}$	damping, $\beta_t \in (0, 1]$

- Step-sizes σ_t and τ_t are adapted.
- Onsager correction term now equals $-\boldsymbol{s}_t / \sigma_t$.
- For MMSE, replace prox with scalar MMSE denoiser.

[5] Rangan—arXiv:1010:5141

How fast is (G)AMP?

Pretty fast, at least for i.i.d. Gaussian $\boldsymbol{A}$:



Above: LASSO recovery of a 40-sparse 1000-length Bernoulli-Gaussian signal from 400 AWGN-corrupted measurements.

What about generic matrices $\boldsymbol{A}$?

Here is what we know about GAMP:

- It may diverge! But...
- MAP case: if it converges, then it converges to a local minimum of the MAP cost function.[6]
- MMSE case: if it converges, then it converges to a local minimum of the large-system-limit Bethe free energy (LSL-BFE):[6]

$$J(b_x, b_z) = D(b_x \| e^{-f}) + D(b_z \| e^{-g}) + \bar{h}\big(\operatorname{var}(\boldsymbol{x}|b_x), \operatorname{var}(\boldsymbol{z}|b_z)\big)$$

$$b_x, b_z : \text{separable posteriors pdfs s.t. } \mathrm{E}\{\boldsymbol{A}\boldsymbol{x}|b_x\} = \mathrm{E}\{\boldsymbol{z}|b_z\}$$

- Gaussian case: convergence is determined by the peak-to-average ratio of the squared singular-values in $\boldsymbol{A}$. For any $\boldsymbol{A}$, possible to find fixed damping coefficient $\beta_t = \beta$ that guarantees global convergence.[7]

[6] Rangan,Schniter,Riegler,Fletcher,Cevher–arXiv:1301.6295

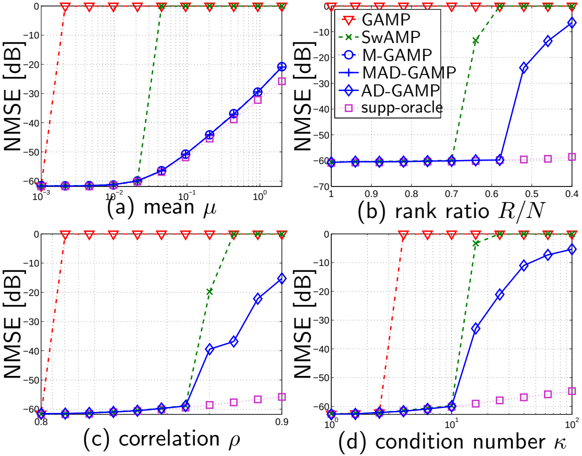
[7] Rangan,Schniter,Fletcher–arXiv:1402.3210

Improving GAMP convergence under generic $\boldsymbol{A}$

Heuristic approaches:

- mean removal[8]
- adaptive damping[8]
- serial updating[9]

On right:
Recovery of a 200-sparse 1000-length BG signal from 500 AWGN-corrupted measurements.



[8] Vila,Schniter,Rangan,Krzakala,Zdeborova–arXiv:1412.2005

[9] Manoel,Krzakala,Tramel,Zdeborova–arXiv:1406.4311

ADMM-GAMP: A Provably Convergent Alternative

- Idea: direct minimization of MMSE-GAMP cost function:

$$\underset{\text{separable pdfs } b_x, b_z}{\arg\min} \; D(b_x \| e^{-f}) + D(b_z \| e^{-g}) + \bar{h}\big(\operatorname{var}(\boldsymbol{x}|b_x), \operatorname{var}(\boldsymbol{z}|b_z)\big) \quad \text{s.t. } \operatorname{E}\{\boldsymbol{Ax}|b_x\} = \operatorname{E}\{\boldsymbol{z}|b_z\}$$

- Challenge: $\bar{h}(\operatorname{var}(b))$ is neither convex nor concave in $b \triangleq (b_x, b_z)$.
- Solution: a double loop algorithm:[10]
  - Outer loop: linearize $\bar{h}$ about current guess $\rightarrow$ convex $+$ concave

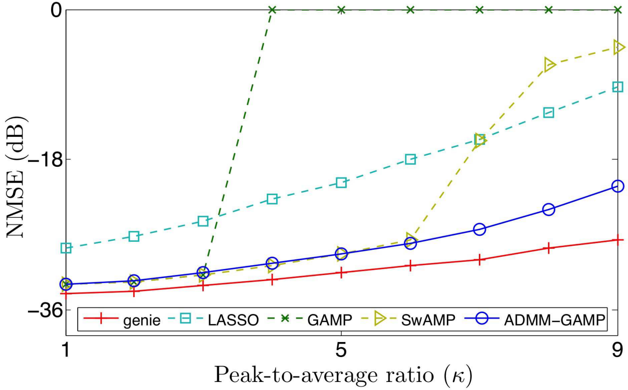
$$D(b_x \| e^{-f}) + D(b_z \| e^{-g}) + \tfrac{1}{2\boldsymbol{\tau}}^{\mathsf{T}} \operatorname{var}(\boldsymbol{x}|b_x) + \tfrac{\boldsymbol{\sigma}}{2}^{\mathsf{T}} \operatorname{var}(\boldsymbol{z}|b_z).$$

  - Inner loop: Minimize linearized LSL-BFE using ADMM under constraints $E(\boldsymbol{x}|b_x) = \boldsymbol{v}$, $\operatorname{E}(\boldsymbol{z}|b_z) = \boldsymbol{Av}$ using penalty vectors $\frac{1}{2\boldsymbol{\tau}}$ and $\frac{\boldsymbol{\sigma}}{2}$, respectively.
  - Result is basically GAMP plus one additional LS step for $\boldsymbol{v}$.
- Can prove global linear convergence under strongly convex f and g.
- MAP case obtained as "zero-temperature" limit of MMSE case.

[10] Rangan, Fletcher, Schniter, Kamilov–arXiv:1501.01797

Example of ADMM-GAMP

Recovery of 200-sparse 1000-length BG signal from $m = 600$ AWGN-corrupted measurements, versus squared-singular-value ratio.



- ADMM-GAMP does not break down like other variants of GAMP.
- ADMM-GAMP outperforms LASSO since MMSE is better than MAP.

Generalized AMP for Analysis CS (GrAMPA)

- Until now we've focused on the canonical sparsity basis $\boldsymbol{\Omega} = \boldsymbol{I}$.
- What about generic analysis operators $\boldsymbol{\Omega}$ (e.g., TV, SARA)?
- Can handle this in GAMP framework by[11] ...
  - stacking matrices: $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{\Phi} \\ \boldsymbol{\Omega} \end{bmatrix}$
  - setting penalties $\{g_i\}_{i=1}^{M}$ to observation log-likelihoods
  - setting penalties $\{g_i\}_{i=M+1}^{M+D}$ to co-sparsity log-priors.
- For the co-sparsity penalties ...
  - ℓ_0-like works better when $\boldsymbol{\Omega}$ is highly overcomplete.
  - we propose the "sparse non-informative parameter estimator (SNIPE)" $\rightsquigarrow$ MMSE denoiser for Bernoulli-$*$ prior in the limit of infinite-variance $*$.

[11] Borgerding,Schniter,Rangan–arXiv:1312.3968

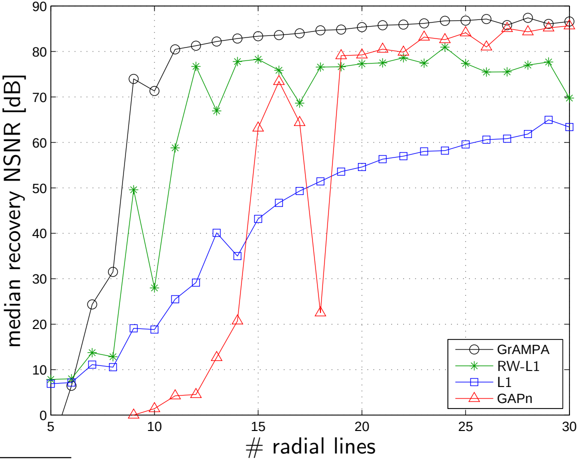
GrAMPA  meets the Phantom 

- Ω: total variation (Hor,Vert,Diag)
- Φ: radial Fourier
- $\mathrm{SNR} = 80\mathrm{dB}$

Avg Runtime:

0.3s: GrAMPA
1.8s: L1
9.7s: RW-L1[12]
30.1s: GAPn[13]

64×64 Shepp-Logan phantom



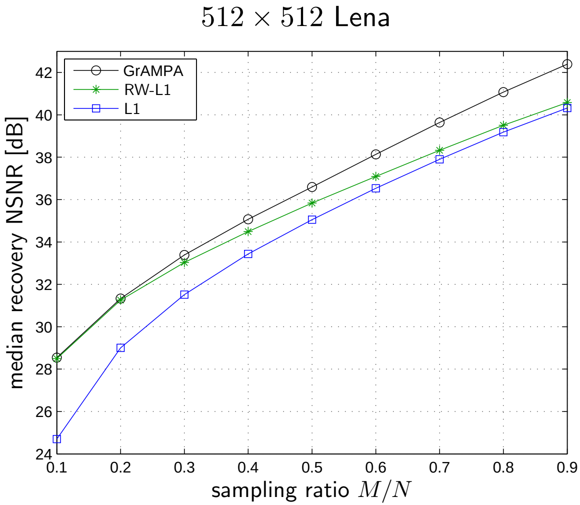
[12] Carrillo,McEwen,VanDeVille,Thiran,Wiaux–SPL'13
[13] Nam,Davies,Elad,Gribonval–CAMSAP'11

GrAMPA  meets Lena  and SARA 

- $\mathbf{\Omega}$: Db1-8 (SARA)
- $\mathbf{\Phi}$: spread spectrum
- SNR $= 40$dB

Avg Runtime:

220s: GrAMPA
225s: L1
2687s: RW-L1



Tuning the Hyperparameters

- The log-prior f often has tunable parameters (e.g., sparsity). How to choose them?
  - The input to (G)AMP's denoiser input is an AWGN corrupted version of the truth with known noise variance. Thus,
    1. learn prior via EM[14] (deconvolution of blurred pdf), or
    2. apply Stein's Unbiased Risk Estimator.[15]
  - Can learn entire f by tuning a many-term Gaussian-mixture (GM).
- The log-likelihood g also has tunable parameters (e.g., noise variance). How to choose them?
  - The LSL-BFE gives an approximate upper bound on the $-$log-likelihood. The AWGN case results in simple closed-form tuning.[16] For the non-AWGN case, we proposed a Newton-based algorithm.[17]

---

[14] Vila,Schniter–SAHD'11 & TSP'13

[15] Mousavi,Maleki,Baraniuk–arXiv:1311.0035 / Guo,Davies–arXiv:1409.0440

[16] Krzakala,Mezard,Sausset,Sun,Zdeborova–JSM'12

[17] Schniter,Rangan–arXiv:1405.5618

Compressive Phase Retrieval

- Problem: Reconstruct a sparse signal from intensity-only measurements of a complex measurement operator (e.g., Fourier transform).
- Applications: X-ray imaging, optics, microscopy, acoustics, etc.
- $M \approx 4K$ measurements are necessary & sufficient.
- "Lifting" based convex algorithms work with $M \gtrsim O(K^2 \log N)$ and complexity $O(N^3)$, which is not practical.
- We proposed to use MMSE-GAMP with Rician likelihood

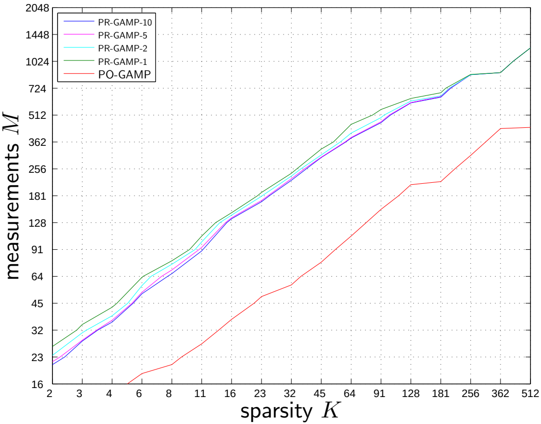
$$\exp\big(-g_i(z_i;\nu^w)\big) = \frac{2y_i}{\nu^w}\exp\Big(-\frac{y_i^2+|z_i|^2}{\nu^w}\Big) I_0\Big(\frac{2y_i|z_i|}{\nu^w}\Big) 1_{y_i \geq 0}$$

and Bernoulli-Gaussian signal prior.[18]

[18] Schniter,Rangan–arXiv:1405.5618

Phase-transition curves

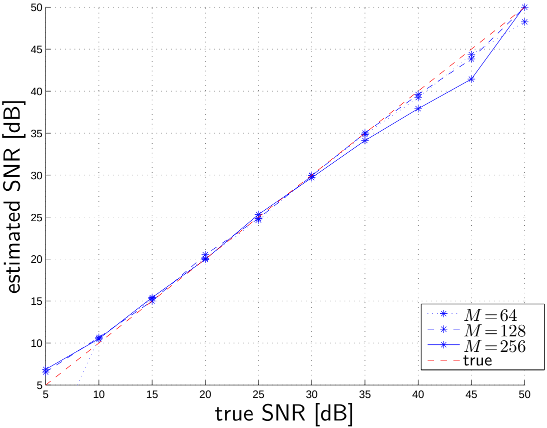
- $N = 512$, BG
- iid Gaussian $\boldsymbol{A}$
- SNR $= 100$ dB
- NMSE $< 10^{-6}$ above PTC



- For $K \ll N$, PTC suggests $M \geq 2K \log_2(N/K)$ suffices.
- Phase-retrieval GAMP requires $\approx 4\times$ the number of measurements as phase-oracle GAMP. (Very interesting!)

Accuracy of Noise-Variance Learning

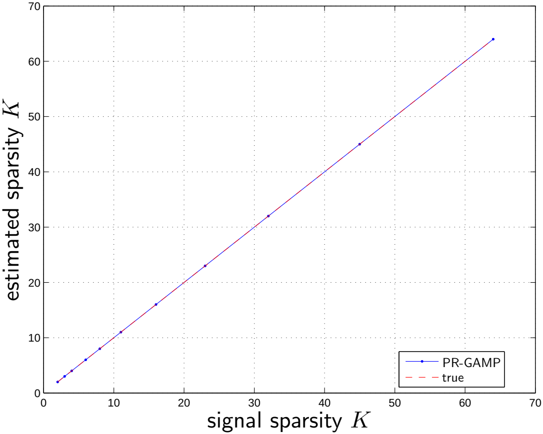
The estimated noise variance, averaged over 10 realizations, at several measurement lengths M, for signal length $N = 512$ and sparsity $K = 4$:



- The LSL-BFE-based likelihood-tuning method is accurate across a wide SNR range.

Accuracy of Sparsity-Rate Learning

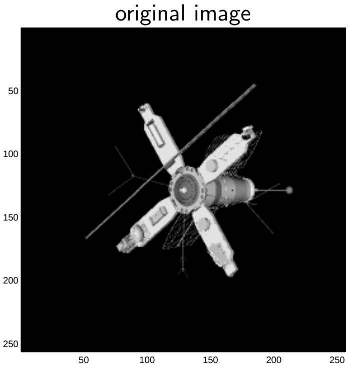
The average estimated sparsity for $M = 512$ over 10 realizations:

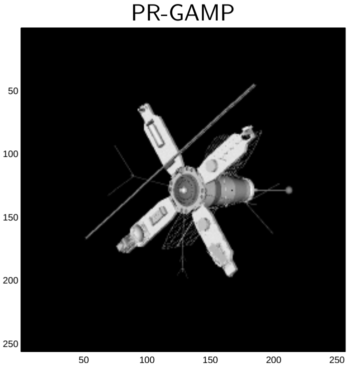


- The EM-based prior-tuning method is accurate across a wide sparsity range.

Compressive Image Recovery

65536 image pixels, 32768 measurements, 30dB SNR:





NMSE = -37.5 dB, runtime = 1.8 sec.

Conclusions

Approximate message passing . . .

- is IST / primal-dual, but with carefully adapted stepsizes,
- provides posterior uncertainty information (not just point estimates),
- is Bayes-optimal in the large-system limit with i.i.d. sub-Gaussian $\boldsymbol{A}$,
- can diverge with generic $\boldsymbol{A}$, but robustified by damping / direct-min,
- can be used in synthesis-CS or analysis-CS settings,
- leads to easy tuning of hyperparameters,
- often leads to state-of-the-art accuracy *and* runtime.

Thanks for listening!