Computational methods in applied inverse problems

Uri Ascher

Department of Computer Science
University of British Columbia
October 2017

Four lectures

- Calibration and simulation of deformable objects
- Data manipulation and completion
- Estimating the trace of a large implicit matrix and applications
- Numerical analysis in visual computing: not too little, not too much

References

- van den Doel & A., Adaptive and stochastic algorithms for EIT and DC resistivity problems with piecewise constant solutions and many measurements, SIAM J. SISC, 2012.
- Roosta-Khorasani, van den Doel & A., Stochastic algorithms for inverse problems involving PDEs and many measurements, SIAM J. SISC, 2014.
- Roosta-Khorasani & A., Improved bounds on sample size for implicit matrix trace estimators. J. FOCM, 2015.
- Roosta-Khorasani, Székely & A., Assessing stochastic algorithms for large scale nonlinear least squares problems using external probabilities of linear combinations of gamma variables. SIAM/ASA JUQ, 2015.
- A. & Roosta-Khorasani, Algorithms that satisfy a stopping criterion, probably. Vietnam J. Math, 2016.

Fred (Farbod) Roosta-Khorasani

UBC $\rightarrow$ Berkeley $\rightarrow$ University of Queensland, Brisbane

Outline

- Motivation
- Estimating trace of implicit matrix
- Probabilistic relaxation of stopping criterion
- Stochastic model reduction
- Numerical experiments: DC resistivity inverse problem
- Conclusions

Estimating matrix trace: choose n

- Let A be an $s \times s$ symmetric positive semi-definite (SPSD) matrix.
- Assume that A is not available and is only given **implicitly**: effectively, can only carry out matrix-vector products $A * \mathbf{v}$ with this matrix.
- **Goal**: estimate the trace of A, denoted $tr(A)$, effectively.
- Let $\mathbf{w}$ be drawn from a probability distribution satisfying $\mathbb{E}(\mathbf{w}\mathbf{w}^T) = I$. Then

$$tr(A) = tr(B^T B) = \|B\|_F^2 = \mathbb{E}((B\mathbf{w})^T (B\mathbf{w})) = \mathbb{E}(\mathbf{w}^T A \mathbf{w}).$$

- Approximating expectation $\Leftrightarrow$ Approximating the trace $tr(A)$
  - Monte-Carlo approximation

$$tr(A) \approx \frac{1}{n} \sum_{j=1}^{n} \mathbf{w}_j^T A \mathbf{w}_j.$$

  - Note we can obtain exact trace using $n = s$ samples with $\mathbf{w}_j$ a scaled jth column of identity; but we want $n \ll s$.

Estimating matrix trace: choose n

- Let A be an $s \times s$ symmetric positive semi-definite (SPSD) matrix.
- Assume that A is not available and is only given **implicitly**: effectively, can only carry out matrix-vector products $A * \mathbf{v}$ with this matrix.
- **Goal**: estimate the trace of A, denoted $tr(A)$, effectively.
- Let $\mathbf{w}$ be drawn from a probability distribution satisfying $\mathbb{E}(\mathbf{w}\mathbf{w}^T) = I$. Then

$$tr(A) = tr(B^T B) = \|B\|_F^2 = \mathbb{E}((B\mathbf{w})^T(B\mathbf{w})) = \mathbb{E}(\mathbf{w}^T A\mathbf{w}).$$

- Approximating expectation $\Leftrightarrow$ Approximating the trace $tr(A)$
  - Monte-Carlo approximation

$$tr(A) \approx \frac{1}{n}\sum_{j=1}^{n} \mathbf{w}_j^T A\mathbf{w}_j.$$

  - Note we can obtain exact trace using $n = s$ samples with $\mathbf{w}_j$ a scaled jth column of identity; but we want $n \ll s$.

Estimating matrix trace: choose n

- Let A be an $s \times s$ symmetric positive semi-definite (SPSD) matrix.
- Assume that A is not available and is only given **implicitly**: effectively, can only carry out matrix-vector products $A * \mathbf{v}$ with this matrix.
- **Goal**: estimate the trace of A, denoted $tr(A)$, effectively.
- Let $\mathbf{w}$ be drawn from a probability distribution satisfying $\mathbb{E}(\mathbf{w}\mathbf{w}^T) = I$. Then

$$tr(A) = tr(B^T B) = \|B\|_F^2 = \mathbb{E}((B\mathbf{w})^T(B\mathbf{w})) = \mathbb{E}(\mathbf{w}^T A\mathbf{w}).$$

- Approximating expectation $\Leftrightarrow$ Approximating the trace $tr(A)$
  - Monte-Carlo approximation

$$tr(A) \approx \frac{1}{n}\sum_{j=1}^{n} \mathbf{w}_j^T A\mathbf{w}_j.$$

  - Note we can obtain exact trace using $n = s$ samples with $\mathbf{w}_j$ a scaled jth column of identity; but we want $n \ll s$.

Which probability distribution?

Can choose from several:

- Gaussian: normal distribution
- Hutchinson: Rademacher distribution – for $\mathbf{w} = (w_1, \ldots, w_s)^T$, $Pr(w_j = 1) = Pr(w_j = -1) = 1/2$
- Random subset (RS) of scaled columns of the identity matrix.

Selling features:

- Hutchinson (1990) showed that Rademacher has smaller variance than normal distribution. This made the Hutchinson choice popular in industry, too.
- RS applies to general matrices, unlike the other two. On the other hand, for SPSD matrices, RS is somewhat slower than the other two.

Probabilistic bounds, Roosta & A.

[Roosta-Khorasani & A., 2015; Avron & Toledo, 2011] following [Achlioptas, 2001], want n smallest for a desired quality, look for probabilistic bounds.

- Given pair (ε, δ) positive and small
- Probability distribution Δ
- Want **lower bound on** n such that

$$Pr\left(|tr_\Delta^n(A) - tr(A)| \leq \varepsilon\ tr(A)\right) \geq 1 - \delta$$

$$tr_\Delta^n(A) = \frac{1}{n}\sum_{j=1}^{n} \mathbf{w}_j^T A \mathbf{w}_j$$

with $\mathbf{w}_j$ drawn from probability distribution Δ

General sufficient bounds

Given a pair (ε, δ), denote

$$c = c(\varepsilon, \delta) = \varepsilon^{-2} \ln(2/\delta).$$

Assume A is an $s \times s$ SPSD matrix.

1. **Theorem 1a.** Using Gaussian probability distribution, the probabilistic bound holds if

$$n \geq 8c(\varepsilon, \delta)$$

2. **Theorem 2a.** Using [Hutchinson, 1990], namely, for $\mathbf{w} = (w_1, \ldots, w_s)^T$, $Pr(w_i = 1) = Pr(w_i = -1) = 1/2$, the probabilistic bound holds if

$$n \geq 6c(\varepsilon, \delta)$$

Note that these bounds are independent of matrix properties! They are interesting for large size s and mild tolerance ε.

More specific sufficient bounds

1. **Theorem 1b.** Using Gaussian probability distribution, let $\mathcal{K}_G = \frac{\|A\|}{tr(A)}$. Then the probabilistic bound holds if

$$n > 8\mathcal{K}_G c(\varepsilon, \delta)$$

2. **Theorem 2b.** Using [Hutchinson, 1990], let $\mathcal{K}_H = \max_j \sum_{k \neq j} \frac{a_{k,j}^2}{a_{jj}^2}$. Then the probabilistic bound holds if

$$n > 2\mathcal{K}_H c(\varepsilon, \delta).$$

Sufficient bounds for RS (aka random unit vector)

Given a pair (ε, δ), denote

$$c = c(\varepsilon, \delta) = \varepsilon^{-2} \ln(2/\delta).$$

Let U_1 and U_2 refer to the uniform distribution of unit vectors with and without replacement, respectively.

- **Theorem 3.** Let $\mathcal{K}_U = \max_{\substack{1 \le i,j \le s \\ i \neq j}} \frac{s}{tr(A)} |a_{ii} - a_{jj}|$.

  Then, using RS distribution, the probabilistic bound holds with U_1 if

$$n > \frac{\mathcal{K}_U^2}{2} c(\varepsilon, \delta) \equiv \mathcal{F}$$

  and with U_2 if

$$n \geq \frac{s+1}{1 + \frac{s-1}{\mathcal{F}}}$$

Generally, RS shines where the other two cannot be used.

Sufficient bounds for RS (aka random unit vector)

Given a pair (ε, δ), denote

$$c = c(\varepsilon, \delta) = \varepsilon^{-2} \ln(2/\delta).$$

Let U_1 and U_2 refer to the uniform distribution of unit vectors with and without replacement, respectively.

- **Theorem 3.** Let $\mathcal{K}_U = \max_{\substack{1 \le i,j \le s \\ i \neq j}} \frac{s}{tr(A)} |a_{ii} - a_{jj}|$.

  Then, using RS distribution, the probabilistic bound holds with U_1 if

$$n > \frac{\mathcal{K}_U^2}{2} c(\varepsilon, \delta) \equiv \mathcal{F}$$

  and with U_2 if

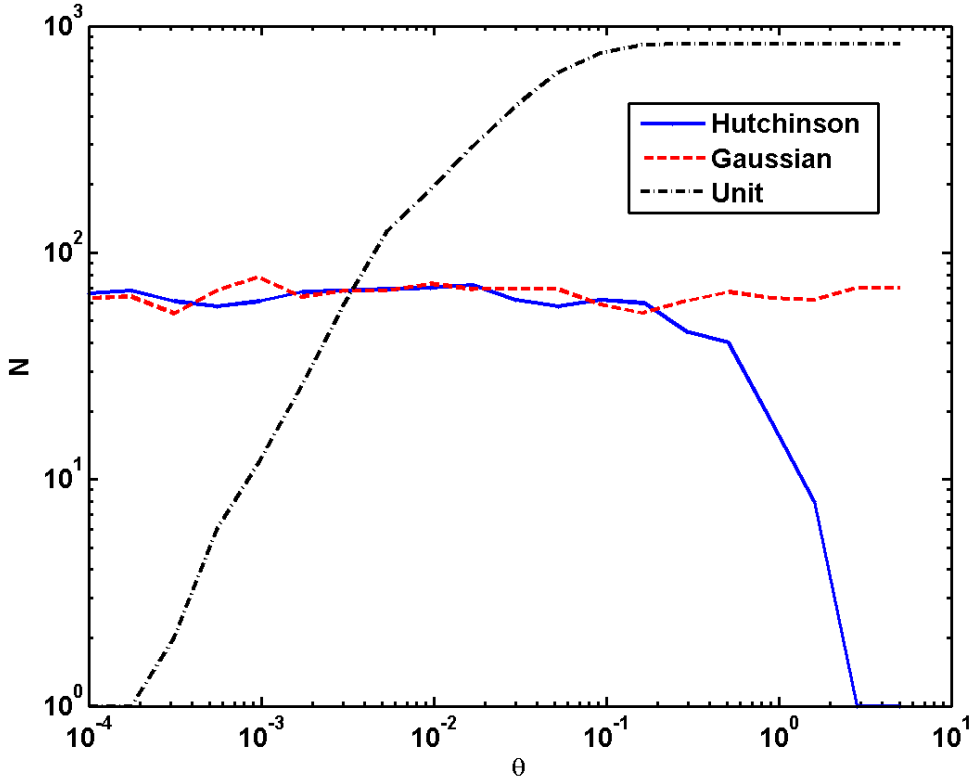
$$n \geq \frac{s+1}{1 + \frac{s-1}{\mathcal{F}}}$$

Generally, RS shines where the other two cannot be used.

Example ($s = 1000,\ \varepsilon = \delta = 0.2$)

Consider $A = \mathbf{x}\mathbf{x}^T/\|\mathbf{x}\|^2$, where $\mathbf{x} \in \mathbb{R}^s$, and for some $\theta > 0$, $x_i = \exp(-i\theta),\ 1 \le i \le s$. Then (denoting $r = rank(A)$)

$$tr(A) = 1,\ r = 1.$$



Preferring to work with...

- The Hutchinson method has a **smaller variance**.
- Our **sufficient** theoretical bounds are occasionally better for Hutchinson than for the Gaussian.
- However, we also want **necessary** bounds, and have been able to obtain the latter only for the Gaussian distribution.
- Fortunately, for the inverse problem applications we have considered, experience indicates that the Gaussian distribution is not worse. Moreover, $\mathrm{rank}(A) = l \ll s$ is relatively low.
- **So, we'll work with the Gaussian distribution (without forgetting RS for tough situations).**

Preferring to work with...

- The Hutchinson method has a **smaller variance**.
- Our **sufficient** theoretical bounds are occasionally better for Hutchinson than for the Gaussian.
- However, we also want **necessary** bounds, and have been able to obtain the latter only for the Gaussian distribution.
- Fortunately, for the inverse problem applications we have considered, experience indicates that the Gaussian distribution is not worse. Moreover, $\mathrm{rank}(A) = l \ll s$ is relatively low.
- So, we'll work with the Gaussian distribution (without forgetting RS for tough situations).

Preferring to work with...

- The Hutchinson method has a **smaller variance**.
- Our **sufficient** theoretical bounds are occasionally better for Hutchinson than for the Gaussian.
- However, we also want **necessary** bounds, and have been able to obtain the latter only for the Gaussian distribution.
- Fortunately, for the inverse problem applications we have considered, experience indicates that the Gaussian distribution is not worse. Moreover, $\mathrm{rank}(A) = l \ll s$ is relatively low.
- **So, we'll work with the Gaussian distribution (without forgetting RS for tough situations).**

Outline

- Motivation
- Estimating trace of implicit matrix
- Probabilistic relaxation of stopping criterion
  - Importance of stopping criterion
  - Inverse problem setting and regularization
  - Many data sets
  - How large should n be?
- Stochastic model reduction
- Numerical experiments: DC resistivity inverse problem
- Conclusions

Probabilistic relaxation of stopping criterion

[A. & Roosta, 2016]

- Importance of stopping criterion
- Inverse problem setting and regularization
- Many data sets
- How large should n be?

Stopping criterion and stopping tolerance

- A typical iterative algorithm in scientific computing requires a stopping criterion:
- Given a tolerance ρ and some mechanism for estimating error, stop algorithm execution at first iteration k such that

$$\text{error\_estimate}(k) \leq \rho.$$

- But, *is ρ really given?!*
- (Relatedly: *to what extent is the stopping criterion adequate?*)
  - The **numerical analyst** designs algorithm and writes general-purpose software, and thus expects ρ or invents it as needed.
  - The **practitioner**, or the **customer** of numerical algorithms and math software, often would not know such a precise value.
- This gap can lead to surprises, and occasionally, advantages.

Stopping criterion and stopping tolerance

- A typical iterative algorithm in scientific computing requires a stopping criterion:
- Given a tolerance ρ and some mechanism for estimating error, stop algorithm execution at first iteration k such that

$$\text{error\_estimate}(k) \leq \rho.$$

- But, *is ρ really given?!*
- (Relatedly: *to what extent is the stopping criterion adequate?*)
  - The **numerical analyst** designs algorithm and writes general-purpose software, and thus expects ρ or invents it as needed.
  - The **practitioner**, or the **customer** of numerical algorithms and math software, often would not know such a precise value.
- This gap can lead to surprises, and occasionally, advantages.

Stopping criterion and stopping tolerance

- A typical iterative algorithm in scientific computing requires a stopping criterion:
- Given a tolerance ρ and some mechanism for estimating error, stop algorithm execution at first iteration k such that

$$\text{error\_estimate}(k) \leq \rho.$$

- But, *is ρ really given?!*
- (Relatedly: *to what extent is the stopping criterion adequate?*)
  - The **numerical analyst** designs algorithm and writes general-purpose software, and thus expects ρ or invents it as needed.
  - The **practitioner**, or the **customer** of numerical algorithms and math software, often would not know such a precise value.
- This gap can lead to surprises, and occasionally, advantages.

Probabilistic relaxation of stopping criterion

- Importance of stopping criterion
- Inverse problem setting and regularization
- Many data sets
- How large should n be?

Inverse problem setting

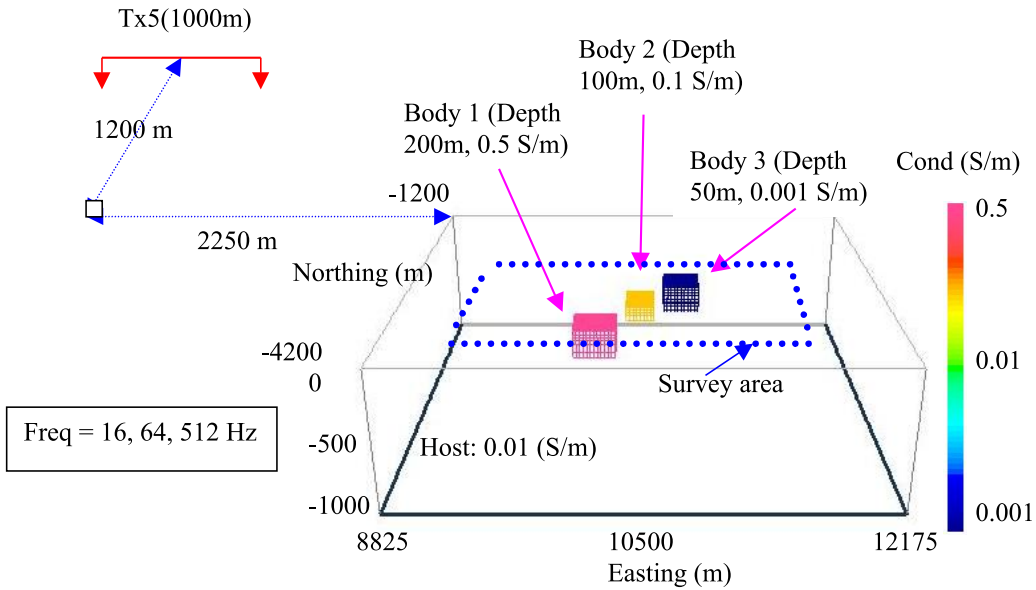
- Given observed data $\mathbf{d} \in \mathbb{R}^I$ and a forward operator $\mathbf{f}(m)$ which provides predicted data for each instance of distributed parameter function m, find m (discretized and reshaped into $\mathbf{m}$) such that the predicted and observed data agree to within noise:

$$\mathbf{d} \quad = \quad \mathbf{f}(\mathbf{m}) + \eta.$$

- Consider a case where a PDE must be solved to evaluate the forward operator, i.e., $\mathbf{f}(\mathbf{m}) = P\mathbf{u} = PG(\mathbf{m})\mathbf{q}$, where G is a discrete Green's function.

Example: electromagnetic data inversion in mining exploration

G is Green's function for Maxwell's equations in time or frequency domain, $\mathbf{m}$ is conductivity or resistivity. [Haber, A. & Oldenburg, 2004]



Discrepancy principle

- Given observed data $\mathbf{d} \in \mathbb{R}^l$ and a forward operator $\mathbf{f}(m)$ which provides predicted data for each instance of distributed parameter function m, find m (discretized and reshaped into $\mathbf{m}$) such that the predicted and observed data agree to within noise:

$$\mathbf{d} \quad = \quad \mathbf{f}(\mathbf{m}) + \eta.$$

- Consider a case where a PDE must be solved to evaluate the forward operator, i.e., $\mathbf{f}(\mathbf{m}) = P\mathbf{u} = PG(\mathbf{m})\mathbf{q}$, where G is a discrete Green's function.
- Assuming $\eta \sim \mathcal{N}(0, \sigma^2 I)$, the maximum likelihood (ML) data misfit function is

$$\phi(\mathbf{m}) = \|\mathbf{f}(\mathbf{m}) - \mathbf{d}\|_2^2$$

- The discrepancy principle yields the stopping criterion

$$\phi(\mathbf{m}) \leq \rho, \quad \text{where} \quad \rho = \sigma^2 l$$

Stopping criterion assessment

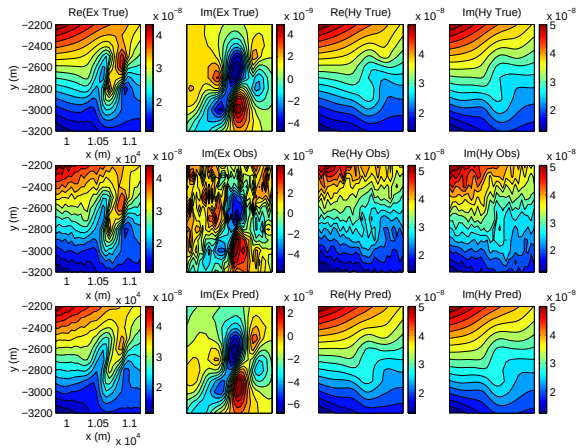
- Here we have a meaningful tolerance for an interesting class of problems!

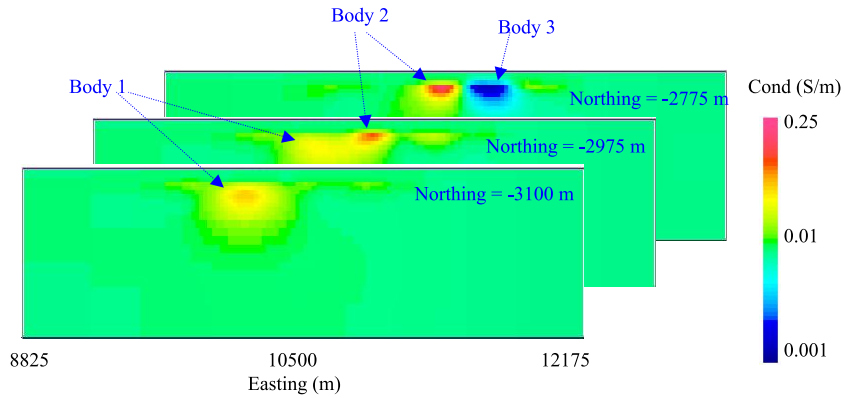
  However:

- Idealized statistical assumptions often do not quite hold in practice.
- In computations, under these possibly unrealistic assumptions, wanting to fit data but not noise, we typically reduce $\phi(\mathbf{m})$ to about $\approx 1.5\rho$.
- **Importantly**, we are really after minimizing $\|\mathbf{m}^* - \mathbf{m}\|$, not $\phi(\mathbf{m})$. It's just that the latter is what we have to work with.
- So, assuming that a known tolerance ρ *must* be satisfied could be *too rigid*.

For the EM data inversion example

G is Green's function for Maxwell's equations in frequency domain,
$\mathbf{m}$ is conductivity or resistivity.





Regularization and constrained formulations

- Typically, solve iteratively and approximately a regularized problem

$$\min_{\mathbf{m}} \ \phi(\mathbf{m}) + \beta R(\mathbf{m}),$$

  where $R(\mathbf{m}) \geq 0$ is a **prior**, $\beta > 0$ a parameter.
- Equivalent constrained formulation:

$$\min_{\mathbf{m}} \ \phi(\mathbf{m}), \quad \text{s.t. } R(\mathbf{m}) \leq \tau.$$

- Another equivalent constrained formulation:

$$\min_{\mathbf{m}} \ R(\mathbf{m}), \quad \text{s.t. } \phi(\mathbf{m}) \leq \rho.$$

  The non-negative parameters ρ, τ, β are related to one another in a nontrivial manner.
- In the above context we may know ρ to a higher certainty level than the other two!

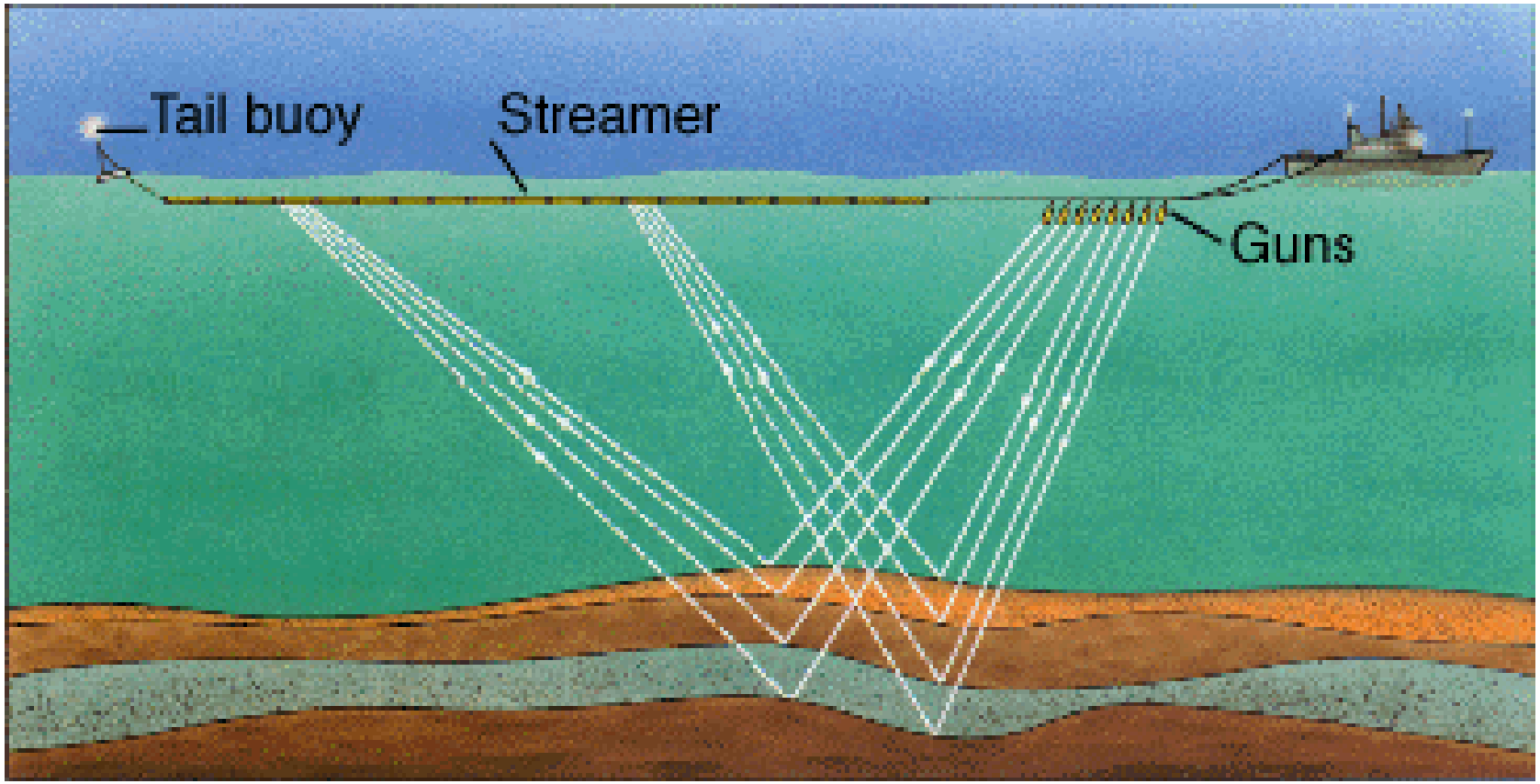
Probabilistic relaxation of stopping criterion

- Importance of stopping criterion
- Inverse problem setting and regularization
- Many data sets
- How large should n be?

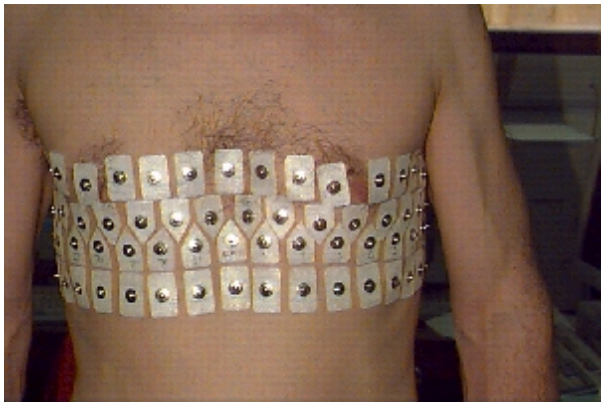
PDE inverse problems & many data sets

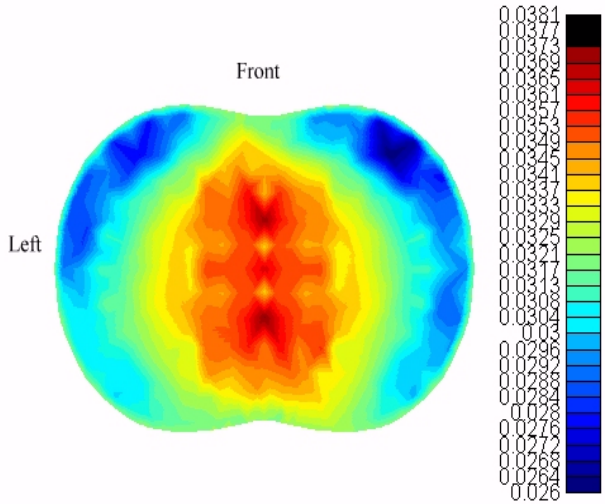
- Nonlinear parameter function estimation problems involving partial differential equation (PDE) constraints
- Many measurements for obtaining credible reconstructions
- Applications
  - Electromagnetic data inversion in mining exploration
  - Seismic data inversion in oil exploration
  - Direct current (DC) resistivity
  - Electrical impedance tomography (EIT)
  - Diffuse optical tomography (DOT)
  - Quantitative photoacoustic tomography (QPAT)

Example: full waveform inversion



Example: EIT




Front
Left
0.0381
0.0377
0.0373
0.0369
0.0365
0.0361
0.0357
0.0353
0.0349
0.0345
0.0341
0.0337
0.0333
0.0329
0.0325
0.0321
0.0317
0.0313
0.0308
0.0304
0.03
0.0296
0.0292
0.0288
0.0284
0.028
0.0276
0.0272
0.0268
0.0264
0.026

Inverse problem with s data sets

After discretization and for our problems of interest:

$$\begin{aligned} \mathbf{d}_i &= \mathbf{f}_i(\mathbf{m}) + \boldsymbol{\eta}_i, \quad i = 1, 2, \ldots, s \\ \mathbf{f}_i(\mathbf{m}) &= P\mathbf{u}_i = PG(\mathbf{m})\mathbf{q}_i \end{aligned}$$

Calculating "$G(\mathbf{m})\mathbf{q}_i$" for each i is costly!

- $\mathbf{d}_i \in \mathbb{R}^l$ is the measurement obtained in the i^{th} experiment
- $\mathbf{f}_i$ is the known forward operator for the i^{th} experiment
- $\mathbf{m} \in \mathbb{R}^{l_m}$ is the sought-after model
- $\boldsymbol{\eta}_i$ is the noise incurred in the i^{th} experiment
- s is the total number of experiments
- $\mathbf{u}_i \in \mathbb{R}^{l_u}$ is the ith field
- $\mathbf{q}_i \in \mathbb{R}^{l_u}$ is the ith source
- G^{-1} is a square matrix discretizing the PDE with the BC
- $P = P_i$ is the projection matrix for the i^{th} experiment

Merit function

- Assume $\boldsymbol{\eta}_i \sim \mathcal{N}(0, \sigma^2 I)$
- Maximum Likelihood (ML) data misfit:

$$\phi(\mathbf{m}) = \sum_{i=1}^{s} \|\mathbf{f}_i(\mathbf{m}) - \mathbf{d}_i\|_2^2 = \|F(\mathbf{m}) - D\|_F^2$$

  - $F \in \mathbb{R}^{l \times s}$ with $\mathbf{f}_i(\mathbf{m})$ for i^{th} column
  - $D \in \mathbb{R}^{l \times s}$ with $\mathbf{d}_i$ for i^{th} column
- Maximum a Posteriori (MAP) merit function:

$$\phi_{R,\beta}(\mathbf{m}) = \phi(\mathbf{m}) + \beta R(\mathbf{m})$$

  - $R(\mathbf{m})$ can be implicit (e.g., dynamic regularization)
- Seek to reduce merit function to the noise level.
- The discrepancy principle gives error criterion

$$\phi(\mathbf{m}) \leq \rho, \quad \text{where } \rho = \sigma^2 ls.$$

So, again a reasonably solid error criterion; but **how reasonable?**

Merit function

- Assume $\boldsymbol{\eta}_i \sim \mathcal{N}(0, \sigma^2 I)$
- Maximum Likelihood (ML) data misfit:

$$\phi(\mathbf{m}) = \sum_{i=1}^{s} \|\mathbf{f}_i(\mathbf{m}) - \mathbf{d}_i\|_2^2 = \|F(\mathbf{m}) - D\|_F^2$$

  - $F \in \mathbb{R}^{l \times s}$ with $\mathbf{f}_i(\mathbf{m})$ for i^{th} column
  - $D \in \mathbb{R}^{l \times s}$ with $\mathbf{d}_i$ for i^{th} column
- Maximum a Posteriori (MAP) merit function:

$$\phi_{R,\beta}(\mathbf{m}) = \phi(\mathbf{m}) + \beta R(\mathbf{m})$$

  - $R(\mathbf{m})$ can be implicit (e.g., dynamic regularization)
- Seek to reduce merit function to the noise level.
- The discrepancy principle gives error criterion

$$\phi(\mathbf{m}) \leq \rho, \quad \text{where } \rho = \sigma^2 ls.$$

So, again a reasonably solid error criterion; but **how reasonable**?

Example: DC resistivity

- PDE with multiple sources

$$\begin{aligned} \nabla \cdot (\mu(\mathbf{x}) \nabla u_i) &= q_i, \quad i = 1, \ldots, s, \\ \frac{\partial u_i}{\partial \nu}|_{\partial\Omega} &= 0. \end{aligned}$$

- Conductivity $\mu(\mathbf{x})$ is expressed as a point-wise function of $m(\mathbf{x})$.
- The operator $G(\mathbf{m})$ is the inverse of the above PDE discretized on a staggered grid.
- Use different selections of sources q_i, yielding corresponding fields u_i.
- Data is measured only on part of the domain's boundary.
- Use any prior we may have for this very difficult problem!

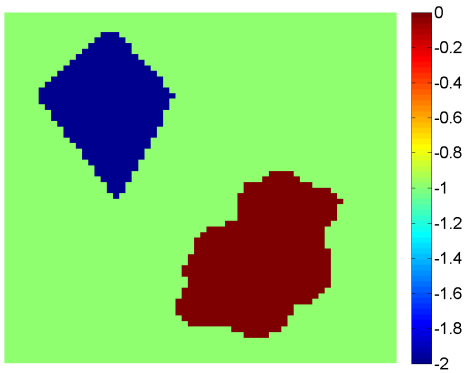
DC RESISTIVITY EXPERIMENT SETUP

- Domain Ω is the unit square. Sources are of the form

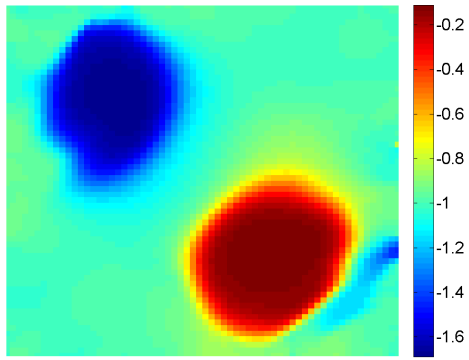
$$q_i(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_1^i) - \delta(\mathbf{x} - \mathbf{x}_2^i)$$

  with $\mathbf{x}_1$ positive unit point source on west boundary, $\mathbf{x}_2$ negative unit point source on east boundary. Vary p boundary wall locations to get $s = p^2$ data sets.
- Receivers are all grid points on north and south walls. No sources or receivers at corners.
- Uniform 64×64 mesh
- For bounds set $\mu_{\max} = 1.2 \max \mu(\mathbf{x})$, $\mu_{\min} = 1.2^{-1} \min \mu(\mathbf{x})$
- PCG inner iteration limit $\tilde{r} = 20$; cgtol $=$ 1.e-3.

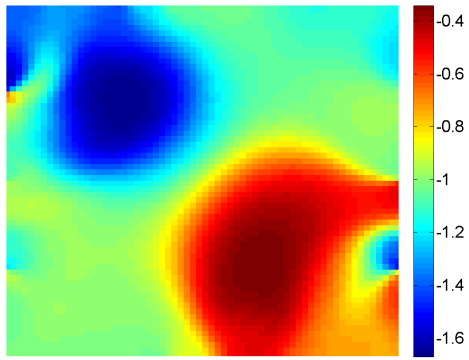
Example: $\mu_I = .1$, $\mu_{II} = 1$, $\mu_{III} = .01$, noise 2%



(a) True model



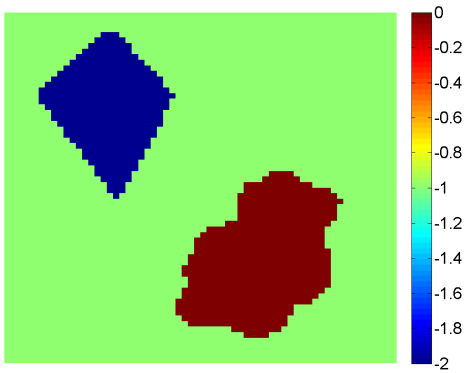
(b) s=3,969



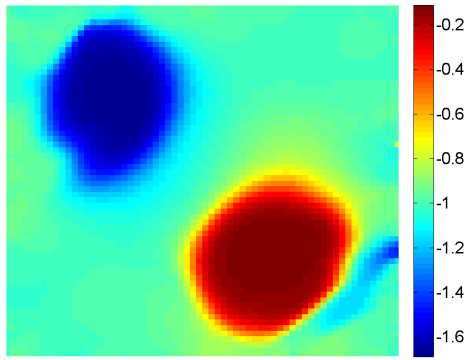
(c) s=49

- Thus, we want s larger for better reconstruction quality.
- But the cost of solving the problem grows very fast! (at least linearly with s). Need to find more efficient approximations for evaluating misfit function $\phi(\mathbf{m})$.

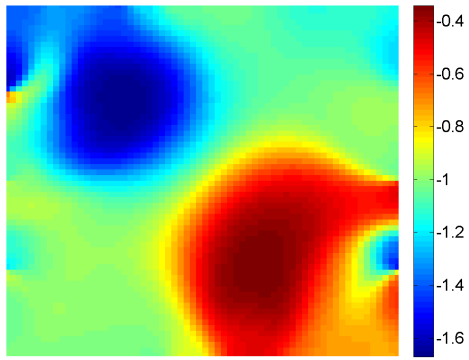
EXAMPLE: $\mu_I = .1$, $\mu_{II} = 1$, $\mu_{III} = .01$, NOISE 2%



(d) True model



(e) s=3,969



(f) s=49

- Thus, we want s larger for better reconstruction quality.
- But the cost of solving the problem grows very fast! (at least linearly with s). Need to find more efficient approximations for evaluating misfit function $\phi(\mathbf{m})$.

Approximating misfit function efficiently

- *Look for inexpensive approximations for* $\phi(\mathbf{m})$
- Define $B = F(\mathbf{m}) - D$ and $A = B^T B$ SPSD. Then

$$\phi(\mathbf{m}) = \|B\|_F^2 = tr(A) = \mathbb{E}(\|B\mathbf{w}\|_2^2),$$

where $tr(A)$ is the trace of A, $\mathbb{E}$ is expectation, and $\mathbf{w}$ is random vector drawn from any distribution satisfying $\mathbb{E}(\mathbf{w}\mathbf{w}^T) = I$.

- So, apply Monte Carlo sampling to the expectation:

$$\widehat{\phi}(\mathbf{m}, n) := \frac{1}{n} \sum_{j=1}^{n} \|B(\mathbf{m})\mathbf{w}_j\|_2^2 \approx \phi(\mathbf{m}).$$

- Can achieve this in only n forward operator evaluations because $\sum_{i=1}^{s} \mathbf{f}(\mathbf{m}, \mathbf{q}_i) w_i^{(j)} = \mathbf{f}(\mathbf{m}, \sum_{i=1}^{s} w_i^{(j)} \mathbf{q}_i)$ for each random vector realization $\mathbf{w}_j = (w_1^{(j)}, \ldots, w_s^{(j)})^T$.

Probabilistic relaxation of stopping criterion

- Importance of stopping criterion
- Inverse problem setting and regularization
- Many data sets
- How large should n be?

How large should n be?

- Obviously, probabilistically the larger n, the closer is $\widehat{\phi}(\mathbf{m}, n)$ to $\phi(\mathbf{m})$.
- This brings again the question how important it is to evaluate $\phi(\mathbf{m})$ exactly, which in turn depends on the certainty we have in the error model and the corresponding stopping criterion.
- **Here it is natural to satisfy our original stopping criterion only within some probability range**.
- We simplify discussion by assuming that ρ is given and the error model is approximately valid.

Outline

- Motivation
- Estimating trace of implicit matrix
- Probabilistic relaxation of stopping criterion
- Stochastic model reduction
  - Model reduction and implicit matrix trace estimation
  - Refined theory, including both sufficient and necessary bounds
  - Algorithm from inverse problem with many right-hand sides
  - Uncertainty quantification
- Numerical experiments: DC resistivity inverse problem
- Conclusions

Stochastic model reduction

- Model reduction and implicit matrix trace estimation
- Refined theory, including both sufficient and necessary bounds
- Algorithm from inverse problem with many right-hand sides
- Uncertainty quantification

Model reduction and stochastic approximation

- Recall, calculating $\phi(\mathbf{m})$ requires s PDE solves!
- In the kth iteration for $\mathbf{m}$, approximate $\phi(\mathbf{m}) = \|F(\mathbf{m}) - D\|_F^2$ at $\mathbf{m} = \mathbf{m}_k$ by an unbiased estimator

$$\widehat{\phi}(\mathbf{m}, s_k) = \hat{\phi}(\mathbf{m}, W) = \frac{1}{s_k}\|(F(\mathbf{m}) - D)W\|_F^2,$$

where W is an $s \times s_k$ matrix (hopefully $s_k \ll s$)

$$W = W_k = \left[\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{s_k}\right],$$

with $\mathbf{w}_j$ drawn from a distribution satisfying $\mathbb{E}(\mathbf{w}\mathbf{w}^T) = I$.

Model reduction and stochastic approximation

- Recall, calculating $\phi(\mathbf{m})$ requires s PDE solves!
- In the kth iteration for $\mathbf{m}$, approximate $\phi(\mathbf{m}) = \|F(\mathbf{m}) - D\|_F^2$ at $\mathbf{m} = \mathbf{m}_k$ by an unbiased estimator

$$\widehat{\phi}(\mathbf{m}, s_k) = \hat{\phi}(\mathbf{m}, W) = \frac{1}{s_k}\|(F(\mathbf{m}) - D)W\|_F^2,$$

where W is an $s \times s_k$ matrix (hopefully $s_k \ll s$)

$$W = W_k = \left[\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{s_k}\right],$$

with $\mathbf{w}_j$ drawn from a distribution satisfying $\mathbb{E}(\mathbf{w}\mathbf{w}^T) = I$.

The misfit trace

- Let $B(\mathbf{m}) = F(\mathbf{m}) - D \in \mathbb{R}^{l \times s}$
- Then $A = B^T B$ is implicit symmetric positive semi-definite (SPSD); effectively, can only carry out matrix-vector products $A * \mathbf{v}$ with this $s \times s$ matrix.

$$\phi(\mathbf{m}) = \|B(\mathbf{m})\|_F^2 = tr(B^T B) = \mathbb{E}(\mathbf{w}^T A \mathbf{w}).$$

- Approximating expectation $\Leftrightarrow$ Approximating the trace $\phi(\mathbf{m}) = tr(A)$
  - Monte-Carlo approximation

$$tr(A) \approx \frac{1}{n} \sum_{j=1}^{n} \mathbf{w}_j^T A \mathbf{w}_j = \frac{1}{n} \sum_{j=1}^{n} \|B\mathbf{w}_j\|_2^2 \ .$$

  - Note we can obtain exact trace using $n = s$ samples with $\mathbf{w}_j$ a scaled jth column of identity; but we want $n \ll s$.
    Link notation with our application: $n = s_k$.

Stochastic model reduction

- Model reduction and implicit matrix trace estimation
- Refined theory, including both sufficient and necessary bounds
- Algorithm from inverse problem with many right-hand sides
- Uncertainty quantification

Refined necessary & sufficient bounds for Gaussian probability distribution I

- Define $Q(n) := \frac{1}{n} Q_n$, where $Q_n \sim \chi^2_n$ denotes a chi-squared r.v of degree n. Note that $Q(n) \sim Gamma(n/2, n/2)$.
- Consider satisfying

$$\Pr\Big(tr_n(A) \geq (1-\varepsilon) tr(A)\Big) \geq 1-\delta. \tag{1}$$

- **Theorem 4.**
  - (I) **Sufficient condition:** there exists some integer $n_0 \geq 1$ such that

$$\Pr\big(Q(n_0) < (1-\varepsilon)\big) \leq \delta.$$

    Furthermore, (1) holds for all $n \geq n_0$.
  - (II) **Necessary condition:** if (1) holds for some $n_0 \geq 1$, then for all $n \geq n_0$

$$P^-_{\varepsilon,r}(n) := \Pr\big(Q(nr) < (1-\varepsilon)\big) \leq \delta, \quad r = \operatorname{rank}(A).$$

Refined necessary & sufficient bounds for Gaussian probability distribution II

- Consider satisfying

$$\Pr\Big(tr_n(A) \le (1+\varepsilon)tr(A)\Big) \ge 1-\delta. \tag{2}$$

- **Theorem 5.**
  - (I) **Sufficient condition:** if the inequality

$$\Pr\big(Q(n_0) \le (1+\varepsilon)\big) \ge 1-\delta$$

    is satisfied for some $n_0 > \varepsilon^{-1}$, then (2) holds with $n = n_0$. Furthermore, there is always an $n_0 > \varepsilon^{-2}$ such that this inequality is satisfied and, for such n_0, it follows that (2) holds for all $n \ge n_0$.
  - (II) **Necessary condition:** if (2) holds for some $n_0 > \varepsilon^{-1}$, then

$$P^+_{\varepsilon,r}(n) := \Pr\big(Q(nr) \le (1+\varepsilon)\big) \ge 1-\delta, \quad r = \operatorname{rank}(A)$$

    with $n = n_0$. Furthermore, if $n_0 > \varepsilon^{-2}r^{-2}$, then this inequality holds for all $n \ge n_0$.

Stochastic model reduction

- Model reduction and implicit matrix trace estimation
- Refined theory, including both sufficient and necessary bounds
- Algorithm from inverse problem with many right-hand sides
- Uncertainty quantification

Algorithm for the many-datasets IP

- At iteration k, $k = 0, 1, 2, \ldots,$ we update $\mathbf{m} = \mathbf{m}_k$ to

$$\mathbf{m}_{k+1} = \mathbf{m}_k + \gamma_k \delta \mathbf{m}_k$$

- The search direction $\delta \mathbf{m}_k$ is obtained using a stabilized Gauss-Newton (GN) method (although *gradient descent* and *L-BFGS* methods may also be considered), and $0 < \gamma_k \leq 1$ is a step size.
- Reduce the misfit, $\phi(\mathbf{m})$, using implicit regularization by applying a limited number of inner preconditioned conjugate gradient (PCG) iterations to carry out the kth outer GN iteration.

Algorithm for the many-datasets IP

- At iteration k, $k = 0, 1, 2, \ldots$, we update $\mathbf{m} = \mathbf{m}_k$ to

$$\mathbf{m}_{k+1} = \mathbf{m}_k + \gamma_k \delta \mathbf{m}_k$$

- The search direction $\delta \mathbf{m}_k$ is obtained using a stabilized Gauss-Newton (GN) method (although *gradient descent* and *L-BFGS* methods may also be considered), and $0 < \gamma_k \leq 1$ is a step size.
- Reduce the misfit, $\phi(\mathbf{m})$, using implicit regularization by applying a limited number of inner preconditioned conjugate gradient (PCG) iterations to carry out the kth outer GN iteration.

Algorithm for the many-datasets IP

- At iteration k, $k = 0, 1, 2, \ldots$, we update $\mathbf{m} = \mathbf{m}_k$ to

$$\mathbf{m}_{k+1} = \mathbf{m}_k + \gamma_k \delta \mathbf{m}_k$$

- The search direction $\delta \mathbf{m}_k$ is obtained using a stabilized Gauss-Newton (GN) method (although *gradient descent* and *L-BFGS* methods may also be considered), and $0 < \gamma_k \leq 1$ is a step size.
- Reduce the misfit, $\phi(\mathbf{m})$, using implicit regularization by applying a limited number of inner preconditioned conjugate gradient (PCG) iterations to carry out the kth outer GN iteration.

Gauss-Newton method

- For one data set:

$$\min_{\mathbf{m}} \ \phi(\mathbf{m}) = \|\mathbf{f}(\mathbf{m}) - \mathbf{d}\|_2^2, \quad \text{sensitivity } J = \frac{\partial \mathbf{f}}{\partial \mathbf{m}}.$$

- GN solves at each iteration k a linearized least squares problem. Normal equations:

$$\begin{aligned} (J^T J)\delta\mathbf{m} &= -\nabla\phi(\mathbf{m}) \\ \mathbf{m}_{k+1} &\leftarrow \mathbf{m}_k + \delta\mathbf{m}. \end{aligned}$$

- But the linear system is singular!
  Moreover, even for one experiment, $\mathbf{f}(\mathbf{m}) = PG(\mathbf{m})\mathbf{q}$, the Jacobian is very expensive to calculate and store.

Gauss-Newton method

- For one data set:

$$\min_{\mathbf{m}} \ \phi(\mathbf{m}) = \|\mathbf{f}(\mathbf{m}) - \mathbf{d}\|_2^2, \quad \text{sensitivity } J = \frac{\partial \mathbf{f}}{\partial \mathbf{m}}.$$

- GN solves at each iteration k a linearized least squares problem. Normal equations:

$$\begin{aligned} \left(J^T J\right)\delta\mathbf{m} &= -\nabla\phi(\mathbf{m}) \\ \mathbf{m}_{k+1} &\leftarrow \mathbf{m}_k + \delta\mathbf{m}. \end{aligned}$$

- But the linear system is singular!
  Moreover, even for one experiment, $\mathbf{f}(\mathbf{m}) = PG(\mathbf{m})\mathbf{q}$, the Jacobian is very expensive to calculate and store.

Stabilized Gauss-Newton method

- So, apply a few conjugate gradient (CG) inner iterations, preconditioned with a Laplacian.
- Stabilized GN:

$$\begin{aligned} J^T(J\delta\mathbf{m}_k) &\approx -\nabla\phi(\mathbf{m}_k) \\ \gamma_k &\approx \operatorname{argmin}_\gamma \phi(\mathbf{m}_k + \gamma\delta\mathbf{m}_k) \\ \mathbf{m}_{k+1} &\leftarrow \mathbf{m}_k + \gamma_k\delta\mathbf{m}_k. \end{aligned}$$

- Obtain inexpensive regularization by **iterative regularization, dynamical regularization** [Hansen, 1999; van den Doel & A., 2007].

Algorithm for the many-datasets IP

- At iteration k, $k = 0, 1, 2, \ldots$, we update $\mathbf{m} = \mathbf{m}_k$ to

$$\mathbf{m}_{k+1} = \mathbf{m}_k + \gamma_k \delta \mathbf{m}_k$$

- The search direction $\delta \mathbf{m}_k$ is obtained using a stabilized Gauss-Newton (GN) method (although *gradient descent* and *L-BFGS* methods may also be considered), and $0 < \gamma_k \leq 1$ is a step size.
- Reduce the misfit, $\phi(\mathbf{m})$, using implicit regularization by applying a limited number of inner preconditioned conjugate gradient (PCG) iterations to carry out the kth outer GN iteration.
- In our model reduction approach the GN method is further modified.

Modified stabilized Gauss-Newton method

- For the multi-experiment case, rewrite the objective as

$$\phi(\mathbf{m}) = \|F(\mathbf{m}) - D\|_F^2,$$
$$F = \left[\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_s\right] \in \mathbb{R}^{l\times s}, \quad D = \left[\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_s\right] \in \mathbb{R}^{l\times s}.$$

- At kth GN iteration, use $s \times s_k$ matrix $W = W_k = \left[\mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^{s_k}\right]$ to estimate ϕ by

$$\hat{\phi}(\mathbf{m}, W) = \frac{1}{s_k}\|(F(m) - D)W\|_F^2.$$

Note $FW = PG(m)(QW), \quad Q = \left[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_s\right]$.

- The essential GN step now reads

$$\left(\sum_{i=1}^{s_k} \hat{J}_i^T \hat{J}_i\right) \delta\mathbf{m}_k = -\nabla_{\mathbf{m}_k}\hat{\phi}.$$

- Next, apply dynamic regularization (stabilized GN) as before.

Modified stabilized Gauss-Newton method

- For the multi-experiment case, rewrite the objective as

$$\phi(\mathbf{m}) = \|F(\mathbf{m}) - D\|_F^2,$$
$$F = \left[\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_s\right] \in \mathbb{R}^{l\times s}, \quad D = \left[\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_s\right] \in \mathbb{R}^{l\times s}.$$

- At kth GN iteration, use $s \times s_k$ matrix $W = W_k = \left[\mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^{s_k}\right]$ to estimate ϕ by

$$\hat{\phi}(\mathbf{m}, W) = \frac{1}{s_k}\|(F(m) - D)W\|_F^2.$$

  Note $FW = PG(m)(QW), \quad Q = \left[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_s\right]$.

- The essential GN step now reads

$$\left(\sum_{i=1}^{s_k} \hat{J}_i^T \hat{J}_i\right) \delta\mathbf{m}_k = -\nabla_{\mathbf{m}_k}\hat{\phi}.$$

- Next, apply dynamic regularization (stabilized GN) as before.

Algorithm for the many-datasets inverse problem

- At iteration k, $k = 0, 1, 2, \ldots$, we update $\mathbf{m} = \mathbf{m}_k$ to

$$\mathbf{m}_{k+1} = \mathbf{m}_k + \gamma_k \delta \mathbf{m}_k$$

- The search direction $\delta \mathbf{m}_k$ is obtained using a stabilized Gauss-Newton (GN) method (although *gradient descent* and *L-BFGS* methods may also be considered), and $0 < \gamma_k \leq 1$ is a step size.
- Reduce the misfit, $\phi(\mathbf{m})$, using implicit regularization by applying a limited number of inner preconditioned conjugate gradient (PCG) iterations to carry out the kth outer GN iteration.
- In our model reduction approach the GN method is further modified.
- Monte-Carlo approximations of ϕ arise when (i) calculating $\delta \mathbf{m}_k$, (ii) attempting to verify that the current sample is representative, and (iii) assessing the stopping rule. [Roosta, Doel & A., 2014]

Algorithm

- **Given:** Q, D, ρ, m_0, sufficient decrease indicator $\kappa \leq 1$.
- **Initialize:** $\mathbf{m} = \mathbf{m}_0$, $s_0 = 1$.
- **for** $k = 0, 1, 2, \ldots$ until termination
  1. Draw $W_k^f \in \mathbb{R}^{s \times s_k}$.
  2. **Fitting**: Perform one nonlinear iteration, with $W = W_k^f$.
  3. Draw $W_k^c \in \mathbb{R}^{s \times n_c}$, $W_k^u \in \mathbb{R}^{s \times n_u}$, $W_k^t \in \mathbb{R}^{s \times n_t}$ as needed.
  4. **Cross validation**: **if** $\hat{\phi}(\mathbf{m}_{k+1}, W_k^c) \leq \kappa \hat{\phi}(\mathbf{m}_k, W_k^c)$, then
     - **Uncertainty Check**: **if** $\hat{\phi}(\mathbf{m}_{k+1}, W_k^u) \leq \rho$, then
     - **Stopping Criterion**: **terminate if** $\hat{\phi}(\mathbf{m}_{k+1}, W_k^t) \leq \rho$; otherwise set $s_{k+1} = s_k$.
  5. **else**
     - **Sample size Increase**: For example set $s_{k+1} = \min(2s_k, s)$.

Stochastic model reduction

- Model reduction and implicit matrix trace estimation
- Refined theory, including both sufficient and necessary bounds
- Algorithm from inverse problem with many right-hand sides
- Uncertainty quantification

Uncertainty quantification

- But how are the (crucial) matrix widths n_c, n_u and n_t determined?
- Here, Theorems 4 & 5 help.
- Specifically, given three pairs of small numbers $(\varepsilon_c, \delta_c)$, $(\varepsilon_u, \delta_u)$, and $(\varepsilon_t, \delta_t)$, we can determine n_c, n_u and n_t, respectively, so that the probabilistic bounds (1) or (2) hold.
- This allows treating the stopping criteria *probabilistically in a quantifiable way*! Larger values of (ε, δ) correspond to treating the tolerance with less rigidity, which translates in turn to a cheaper computational process.

Uncertainty quantification

- But how are the (crucial) matrix widths n_c, n_u and n_t determined?
- Here, Theorems 4 & 5 help.
- Specifically, given three pairs of small numbers $(\varepsilon_c, \delta_c)$, $(\varepsilon_u, \delta_u)$, and $(\varepsilon_t, \delta_t)$, we can determine n_c, n_u and n_t, respectively, so that the probabilistic bounds (1) or (2) hold.
- This allows treating the stopping criteria *probabilistically in a quantifiable way*! Larger values of (ε, δ) correspond to treating the tolerance with less rigidity, which translates in turn to a cheaper computational process.

Uncertainty quantification cont.

- For instance, consider replacing the termination condition

$$\begin{aligned} \phi(\mathbf{m}_{k+1}) &\leq \rho, \quad \text{by either} && (3a) \\ \widehat{\phi}(\mathbf{m}_{k+1}, n_t) &\leq (1-\varepsilon)\rho, \quad \text{or} && (3b) \\ \widehat{\phi}(\mathbf{m}_{k+1}, n_t) &\leq (1+\varepsilon)\rho, && (3c) \end{aligned}$$

  for a suitable $n = n_t$, given $(\varepsilon = \varepsilon_t, \delta = \delta_t)$.

- If (3b) holds, then (3a) holds with probability of at least $(1-\delta)$.
  If (3c) does *not* hold, then (3a) does not hold with a probability of at least $(1-\delta)$.
- Higher uncertainty in value of ρ is reflected by choosing larger δ.
- The probability of false positive/negative is governed by ε.

Outline

- Motivation
- Estimating trace of implicit matrix
- Probabilistic relaxation of stopping criterion
- Stochastic model reduction
- Numerical experiments: DC resistivity inverse problem
- Conclusions

DC RESISTIVITY EXPERIMENTS

- PDE with multiple sources

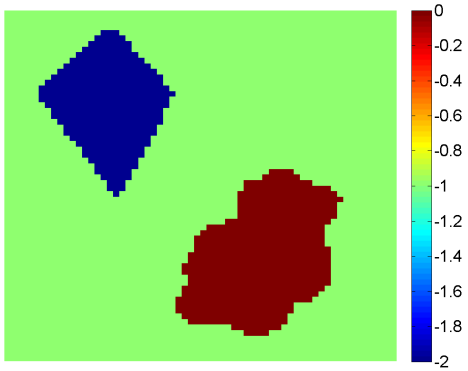
$$\begin{aligned} \nabla \cdot (\mu(\mathbf{x}) \nabla u_i) &= q_i, \quad i = 1, \ldots, s, \\ \frac{\partial u_i}{\partial \nu}|_{\partial\Omega} &= 0. \end{aligned}$$

- Conductivity $\mu(\mathbf{x})$ is expressed as a pointwise function of $m(\mathbf{x})$.
- Domain Ω is unit square. Sources are of the form $q_i(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_1^i) - \delta(\mathbf{x} - \mathbf{x}_2^i)$ with $\mathbf{x}_1$ positive unit point source on west boundary, $\mathbf{x}_2$ negative unit point source on east boundary. Vary p boundary wall locations to get $s = p^2$ data sets.
- Receivers are all grid points on north and south walls. No sources or receivers at corners.
- Uniform 64×64 grid with $s = 3,969$.
- For bounds set $\mu_{\max} = 1.2 \max \mu(\mathbf{x})$, $\mu_{\min} = 1.2^{-1} \min \mu(\mathbf{x})$.
- PCG inner iteration limit $\tilde{r} = 20$; cgtol $=$ 1.e-3.

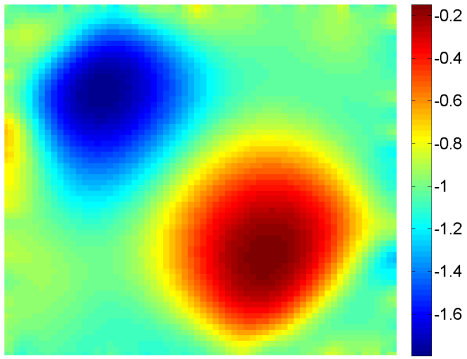
EXAMPLE: $\mu_I = .1$, $\mu_{II} = 1$, $\mu_{III} = .01$, NOISE 2%

Method	Vanilla (3,969)	Gaussian (3,969)	Vanilla (49)
Work	527,877	5,142	5,733

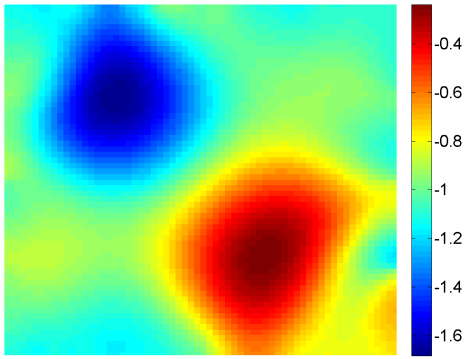
Our algorithm delivers much better reconstructions for the price!



(g) True model



(h) Gaussian, s=3,969

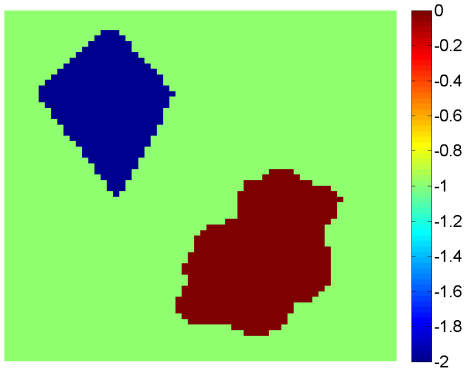


(i) Vanilla, s=49

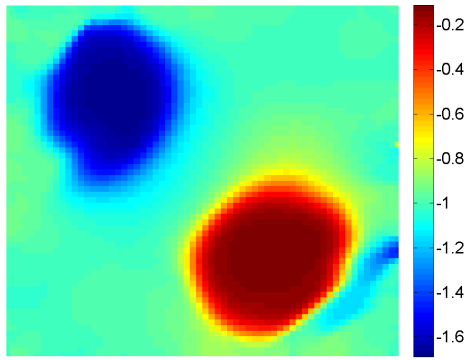
Same with total variation (TV) added

Method	Vanilla (3,969)	Gaussian (3,969)	Vanilla (49)
Work	476,280	4,618	5,978

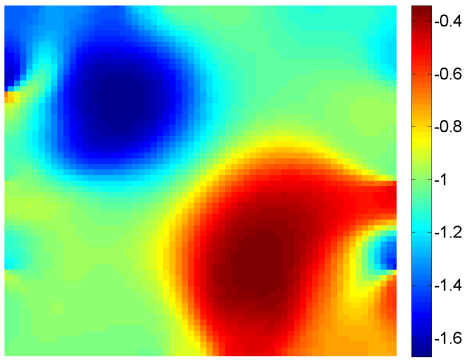
Sharper reconstruction with randomized $s = 3,969$, poorer with vanilla $s = 49$! Thus, even with added info that true solution is piecewise constant, TV can be worse without "sufficient data".



(j) True model



(k) Gaussian, s=3,969



(l) Vanilla, s=49

Conclusions

- A rigid, deterministic convergence error tolerance is hardly ever known in practice. Relaxing requirements accordingly can lead to significant efficiency improvement.
- We have provided necessary and sufficient bounds for the probabilistic convergence of Monte Carlo methods for trace estimation. This allows relaxing the notion of an error criterion with a tolerance in a disciplined manner.
- A general *algorithm* for nonlinear inverse problems with many data sets has been proposed; efficiency gains up to 2 orders of magnitude have been demonstrated.
- Mais *applications*, por favor.