

Compressed Sensing for Nuclear Engineering

NC State Nuclear Engineering Seminar

Ryan G. McClarren

Texas A&M University

Acknowledgements

- Collaborators:

- Vincent Laboure



- Weixiong Zheng



- Pablo Vaquer



- Yuriy Ayzman



- and Jonathan Madsen



- (TAMU

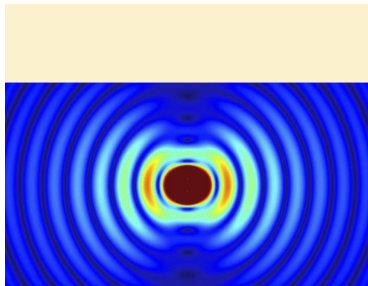
- PhD. Students)

- Cory Hauck (Oak Ridge National Lab)
- Martin Frank (RWTH Aachen)

- Sponsors:

- US National Science Foundation, Division of Mathematical Sciences
- US DOE NEUP
- US DOE NNSA
- Los Alamos National Laboratory
- Lawrence Livermore National Laboratory

First, a plug



Radiation and You

a guide to the radiation in the world around us all

Ryan G. McClarren, PhD.

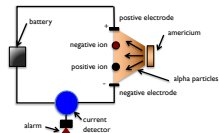
7

Some common uses of radiation

Radiation has many uses in your life. Smoke detectors use radiation to tell us if there is a fire. Inside a smoke detector there is a radioactive atom called americium. Americium shoots out protons and neutrons as alpha radiation. When there is smoke, the protons and neutrons are blocked, and the smoke detector makes a loud sound to let us know there is a fire.



Outside of a household smoke detector.



When smoke blocks radiation, the alarm goes off in a smoke detector.



Here is a picture of the americium from inside a smoke detector. The colorful part in the middle is the americium: the americium is smaller than a grain of sand.

Section 1

- 1 Introduction
 - Background
- 2 The Single Detector Neutron Camera
- 3 Disjoint Monte Carlo Tallies
- 4 Sparse Estimation of Sensitivities
 - Problem settings
 - Coefficient Estimation
- 5 Improving Numerical Methods Using Optimization
- 6 Summary

What is this talk about?

- In engineering and science we often use approximations, reconstructions, and representations that have nice properties such as
 - Linearity
 - Minimizing the error in the L_2 norm
- This talk covers the benefits of relaxing these nice properties, to enable new technologies and improve simulations.
- Much of the technology we need to get these improvements can be explained in terms of compressed sensing.

What is compressed sensing?

- Compressed sensing is an approach to obtaining, reconstructing, and reducing signals.
- Usually requires solving an underdetermined linear system augmented with a cost function to be optimized.
- To understand how compressed sensing works we will need to understand the following concepts:
 - **Representation:** How do we express a signal in terms of a discrete set of functions?
 - **Measurement:** How do we acquire the signal?
- We will use the example of a grayscale image to demonstrate these concepts.

Signal Representation

- The image at right can be stored as $1600 \times 1200 \approx 2 \times 10^6$ numbers between 0 and 1.
- We could think of this as a long vector of 2 million numbers.
- Other representations are possible, and some of them could exactly match the image data using fewer numbers.
- This is possible because images have structure in them: they are not random noise.
- For instance, if one pixel is dark, the pixels near it are likely to be dark as well.

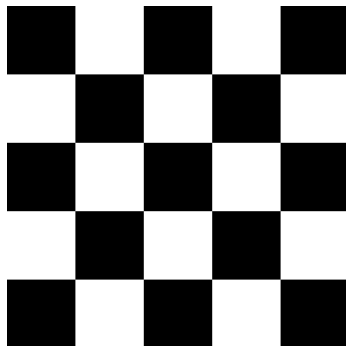


Signal Representation

- This image of a checkerboard has millions of pixels, but we can represent it using only 13 numbers.
- We say that we can represent this image in a basis set where the basis is the 25 different functions that are 1 in a single square and 0 everywhere else:

$$z = \sum_{i=1}^{25} w_i b(x, y).$$

- Because only 13 of these functions will have nonzero weights, w_i , we say that the image is **sparse** in this basis.



Measuring a Signal

- In general we can an image \mathbf{x} to the measurement \mathbf{b} via the linear system

$$\mathbf{Ax} = \mathbf{b}.$$

- The measurement matrix \mathbf{A} represents how the image maps onto the measurement.
- In your phone there is a CCD element for every pixel in the image. In such an instance $\mathbf{x} = \mathbf{b}$, and \mathbf{A} is the identity matrix.
- There are other possible measurement matrices: consider a single, large CCD where we block all but a single pixel, and record the value from the single CCD, and then repeatedly move the unblocked pixel N times.
- This measurement matrix \mathbf{A} is also an identity matrix.

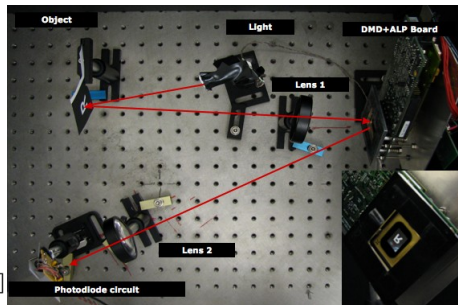
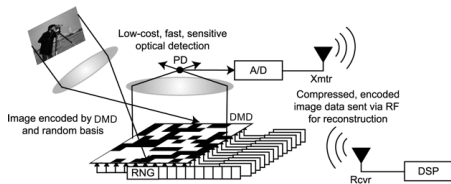
Measuring a Signal

- What we would like to do is take M measurements of an image of N pixels with $M \ll N$.
- This will make the number of entries in \mathbf{b} equal to M , and make \mathbf{A} rectangular with size $M \times N$.
- The system $\mathbf{Ax} = \mathbf{b}$ no longer has a unique solution.
- To come up with a unique solution we have to constrain the problem somehow.
- Before we talk about constraining the problem, we will show how this has been done in practice.

Single Pixel Camera

- This is the idea behind the single-pixel camera: sample the image projected onto random linear combinations of the pixels in the image. Each linear combination only requires the measurement of a single scalar value, i.e., a single CCD.

Single Pixel Camera



from <http://dsp.rice.edu/cscamera>

Single Pixel Camera

- In the case of the single pixel camera, each row of the measurement matrix \mathbf{A} is a random string of zeros and ones of length N .
- The vector \mathbf{b} is the sum of the pixels that are reflected to the detector by the mirror.
- In this case we find the solution to the system $\mathbf{Ax} = \mathbf{b}$ through the optimization problem

$$\text{minimize TV}(\mathbf{x}) \quad \text{subject to } \|\mathbf{Ax} - \mathbf{b}\|_2 \leq \epsilon,$$

where the total variation (TV) of an image is the sum of the squares of the forward difference in the horizontal and vertical directions for each pixel (i.e., how much does the image change from pixel to pixel).

- This problem will have a unique solution, that minimizes the reconstruction error (the second term) and the total variation in the image.
- Other optimization problems are possible, but for images, minimizing the TV norm is reasonable.

Single Pixel Camera Experimental Results



Left: Original 256 x 256 image, Right: Reconstruction from 1500 single-pixel samples (1/50)

Single Pixel Camera

- The actual matrix \mathbf{A} does not need to be stored (just the random number seed used to generate it).
- The signal is encrypted because one needs to know \mathbf{b} and \mathbf{A} to determine the image.
- Trades computation for memory: the \mathbf{b} vector is much smaller than the full image, but we must solve an optimization problem to recover the image.

Section 2

- 1 Introduction
 - Background
- 2 The Single Detector Neutron Camera
- 3 Disjoint Monte Carlo Tallies
- 4 Sparse Estimation of Sensitivities
 - Problem settings
 - Coefficient Estimation
- 5 Improving Numerical Methods Using Optimization
- 6 Summary

Neutron Detection is Hard; Imaging is Harder

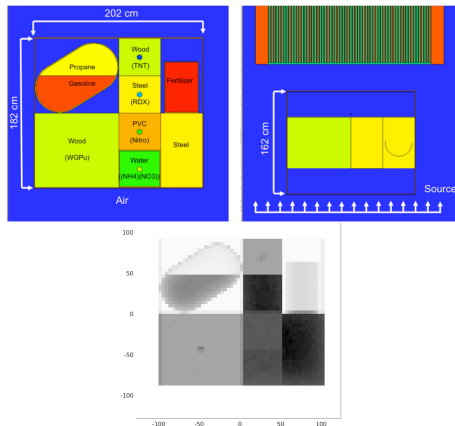
- Neutrons, due to their lack of charge, are difficult to detect, especially at high energies.
- Typically to detect a neutron it must be slowed to low energy through a series of scattering collisions.
- For imaging purposes, this scattering process removes any information about where the neutrons came from.
- Moreover, due to the difficulty in capturing neutrons we would like our detector to be as big as possible.
- There is not a mirror for neutrons, so we have to be a bit more clever.

Collimator plus Mask Equals Single Detector Camera

- Consider a fast neutron collimator, (e.g., a borated polyethylene block with holes drilled through it, densalloy, etc.),
- With a means to block channels in the collimator, either by inserting borated poly, or closing a flap or shutter, etc.
- Behind the collimator we place a thermalizing medium, then a thermal neutron detector.
- By blocking a number of channels at random, recording the signal in the detector and repeating, we can mimic the single pixel camera.
- The upshot is that we can image with neutrons of any energy, using only a single detector.

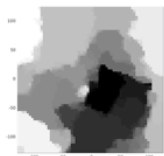
Simulation results

- We consider an active interrogation problem^a of a cargo container with 14.1 MeV neutrons.
- We have a collimator that is a 3 m square and has $64 \times 64 = 4,096$ openings. The collimator is made of densalloy and each has a closable flap of thickness 2 cm.

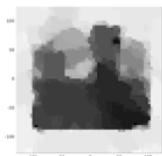


^aEberhardt, J., S. Rainey, R. Stevens, B. Sowerby, J. Tickner, Fast neutron radiography scanner for the detection of contraband in air cargo containers, Applied Radiation and Isotopes 63 (2) (2005) 179188.

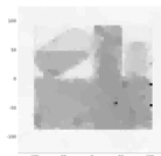
Simulation results



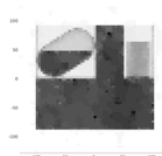
(a)



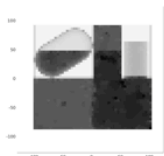
(b)



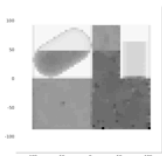
(c)



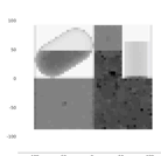
(d)



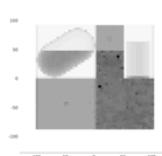
(e)



(f)



(g)



(h)

The logarithmic view of the reconstructions for the active interrogation problem using (a) 1%, (b) 5%, (c) 10%, (d) 20%, (e) 30%, (f) 40%, (g) 50%, and (h) 70% of the pixel count

Single-Detector Neutron Camera

- Simulation results are promising, but building a bench-top prototype is an important next step.
- The time needed to acquire the images is on open question and will depend on several factors.
- Can this technology be combined with position-sensitive neutron detectors?

Section 3

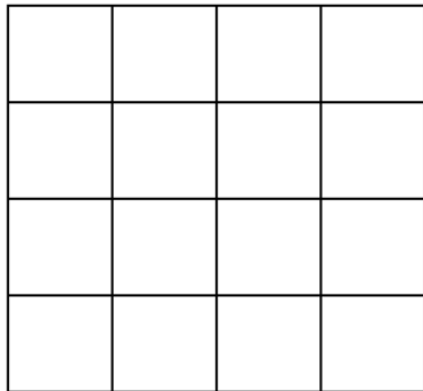
- 1 Introduction
 - Background
- 2 The Single Detector Neutron Camera
- 3 Disjoint Monte Carlo Tallies**
- 4 Sparse Estimation of Sensitivities
 - Problem settings
 - Coefficient Estimation
- 5 Improving Numerical Methods Using Optimization
- 6 Summary

Disjoint Monte Carlo Tallies

- One of the benefits of using the Monte Carlo method for particle transport simulations is that it uses simulated particles as analogs of the real particles in a system.
- As such we can create an analog of the single pixel camera inside of a Monte Carlo simulation to
 - Reduce the memory required to tally quantities, and
 - Reduce statistical noise in the simulation.
- The idea is to store random linear combinations of quantities estimated via Monte Carlo rather than a single value for each spatial location.

Demonstration of Disjoint Monte Carlo Tallies

- If we want the solution on the grid on the right, we need a tally for each voxel.
- Instead, we could define a series of disjoint tallies that take linear combinations of the tallies for each voxel.
- If we need fewer disjoint tallies than the number of voxels, we can reduce the memory footprint for the calculation.

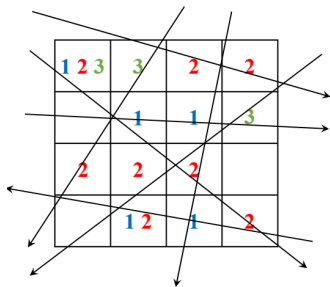


Demonstration of Disjoint Monte Carlo Tallies

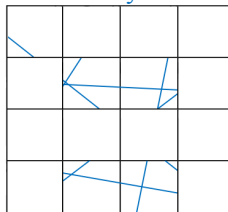
- If we want the solution on the grid on the right, we need a tally for each voxel.
- Instead, we could define a series of disjoint tallies that take linear combinations of the tallies for each voxel.
- If we need fewer disjoint tallies than the number of voxels, we can reduce the memory footprint for the calculation.

1 2 3	3	2	2
	1	1	3
2	2	2	
	1 2	1	2

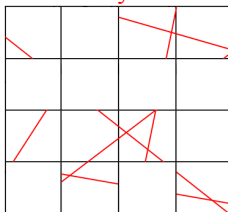
Demonstration of Disjoint Monte Carlo Tallies



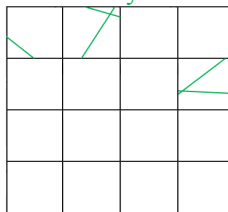
Tally 1



Tally 2

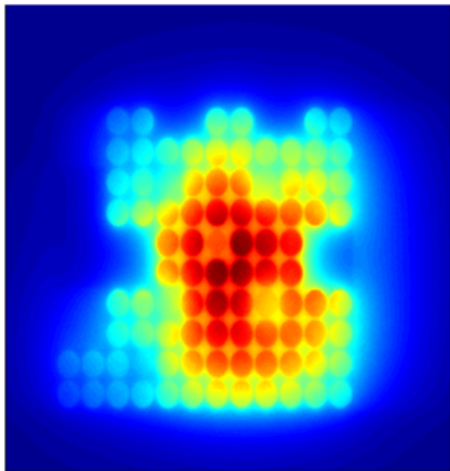


Tally 3



Reactor Calculation with Disjoint Tallies

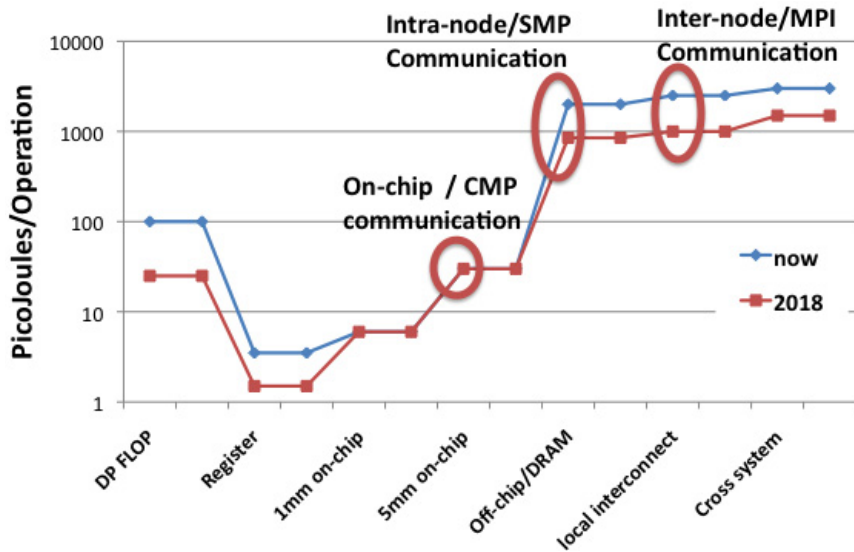
- As a test we look at a 2-D grid of the fast scalar flux in the Texas A&M TRIGA reactor.
- The grid has 1024×1024 voxels.
- Criticality calculation with 2,500 fission cycles and 200,000 neutrons per cycle.



Reactor Calculation with Disjoint Tallies: Details

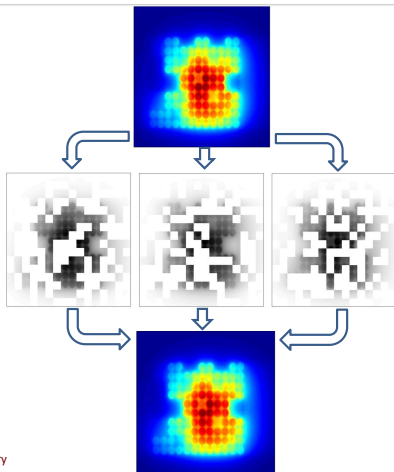
- To make the reconstruction more simple, we divide the problem into blocks, and constrict the disjoint tallies to belong to only a single block.
- In the results that follow, the blocks are of size 64×64 .
- This makes the optimization problem we have to solve smaller, but we have to solve more of them.
- Gives the user the ability to reconstruct the solution only in places of interest.
- We are trading calculation time for memory.

Memory means Power



Reactor Calculation with Disjoint Tallies

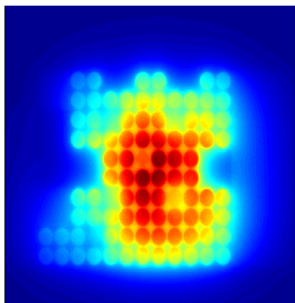
1. Run MC simulation
2. Tally the neutron flux using disjoint tallies
 - Store each tally as a single number!
3. Reconstruct a 2D map of the neutron flux



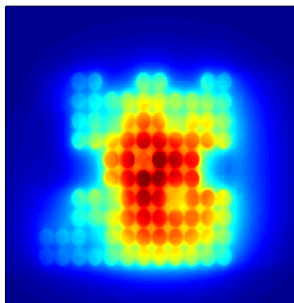
ICTT 2015 - 24th International Conference on Transport Theory

Reactor Calculation with Disjoint Tallies

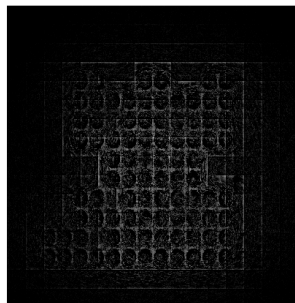
Original



10% of Memory



Difference



Reactor Calculation with Disjoint Tallies: Noise Reduction

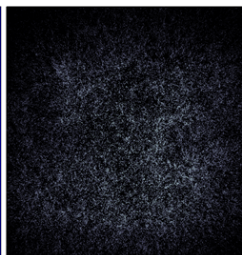
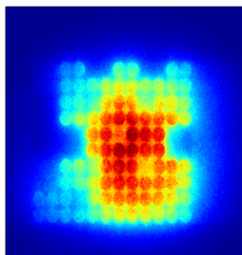
- We found that using disjoint tallies plus reconstruction can give lower statistical noise than standard tallies.
- This is likely due to the form of the reconstruction problem:

$$\text{minimize TV}(\mathbf{x}) \quad \text{subject to } \|\mathbf{Ax} - \mathbf{b}\|_2 \leq \varepsilon.$$

- If the measurement vector is noisy, the reconstruction problem only tries to match it up to ε . Therefore, it can smooth out the noise.
- To show this we run the problem again but with 100x fewer histories and compare the results with the fine resolution calculation.

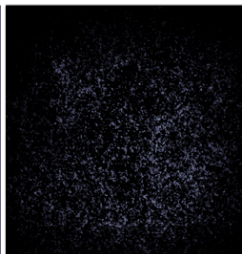
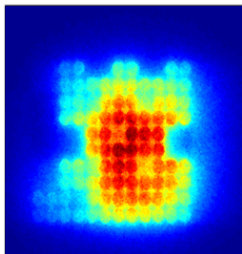
Reactor Calculation with Disjoint Tallies

Original 2D mesh tally of flux in NSCR (left) and statistical error (right)



$$\|e\|_1 = 0.0078$$
$$\|e\|_2 = 0.0014$$
$$\|e\|_\infty = 0.0071$$

Compressed to 10% of computer memory (left) and statistical error (right)



$$\|e\|_1 = 0.0069$$
$$\|e\|_2 = 0.0012$$
$$\|e\|_\infty = 0.0063$$

Section 4

- 1 Introduction
 - Background
- 2 The Single Detector Neutron Camera
- 3 Disjoint Monte Carlo Tallies
- 4 Sparse Estimation of Sensitivities**
 - Problem settings
 - Coefficient Estimation
- 5 Improving Numerical Methods Using Optimization
- 6 Summary

Sensitivity Estimation

- When we perform simulations, not all the inputs to the simulation are known exactly, (e.g., cross-sections).
- These cases we are often interested in the sensitivity of our calculation to the input parameters.
- For instance, if we compute the reactivity ρ , we can express this quantity as a first-order Taylor series

$$\rho(\mathbf{x}) = \rho(\bar{\mathbf{x}}) + \frac{\partial \rho}{\partial x_1} \delta x_1 + \dots + \frac{\partial \rho}{\partial x_p} \delta x_p + \varepsilon.$$

- The sensitivity to a given parameter is the partial derivative with respect to that parameter. Using finite differences this takes $p + 1$ runs of the code to compute.
- In practice, many of these sensitivities have a small magnitude, even zero.
- **Goal:** Estimate the sensitivities with fewer code evaluations than finite differences by exploiting the sparsity of the sensitivities.

Idea: Treat Sensitivity Estimation as a Regression Problem

- If we run the code n times we will have a data set of the form

$$\rho_1 = \rho(\bar{\mathbf{x}}) + \frac{\partial \rho}{\partial x_1} \delta_{x_{11}} + \cdots + \frac{\partial \rho}{\partial x_p} \delta_{x_{p1}} + \varepsilon,$$

$$\rho_2 = \rho(\bar{\mathbf{x}}) + \frac{\partial \rho}{\partial x_1} \delta_{x_{12}} + \cdots + \frac{\partial \rho}{\partial x_p} \delta_{x_{p2}}, + \varepsilon$$

\vdots

$$\rho_n = \rho(\bar{\mathbf{x}}) + \frac{\partial \rho}{\partial x_1} \delta_{x_{1n}} + \cdots + \frac{\partial \rho}{\partial x_p} \delta_{x_{pn}} + \varepsilon.$$

- We could estimate the sensitivities using ordinary least squares.

General Regression Problem

- The general regression problem is written as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

- \mathbf{Y} : data (outcomes), \mathbf{X} : input matrix, $\boldsymbol{\beta}$: regression coefficients, $\boldsymbol{\varepsilon}$: errors

- $$\mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} 1 & X_{11} & X_{12} & \cdots & X_{1p} \\ 1 & X_{21} & X_{22} & \cdots & X_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & X_{n2} & \cdots & X_{np} \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \quad \text{and} \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

Problems with Ordinary Least Squares

- The direct “solve” by ordinary least squares (OLS)

$$\beta \approx (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}.$$

- Several common situations can make OLS ill-conditioned or ill-posed:
 - $n < p$: Number of samples is smaller than number of parameters
 - \mathbf{X} contains interdependencies, i.e., multi-collinearity, if high order terms are included
 - In either case, $\mathbf{X}^T \mathbf{X}$ is rank deficient and not invertible
 - Alternative approaches like the pseudo-inverse can give unreasonable results as has been demonstrated in previous work.
- OLS is the solution to the minimization problem

$$\beta = \underset{\beta}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2.$$

- However, due to sparsity in the sensitivities (many are near zero), we can solve this using an optimization problem (just like in the image reconstruction).
- This is known as regularized regression.

Modified Minimization Problem

- What we want to do is change the functional we minimize to include a penalty based on the magnitude of the β 's.
- This makes sense because we want to find a fit that has many small values for β .
- Consider the problem of estimating the coefficients in the problem

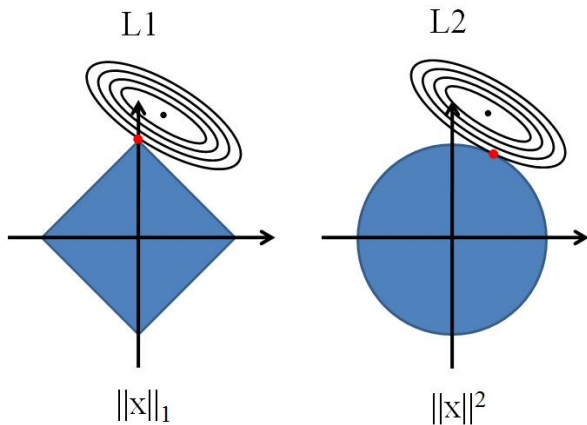
$$y = a + bx + \varepsilon,$$

by minimizing

$$\sum_i \varepsilon_i^2 + (|a|^p + |b|^p)^{1/p}.$$

- The curve of equal value of $(|a|^p + |b|^p)^{1/p}$ is a circle for $p = 2$ and a diamond for $p = 1$.
- The curves of equal value for $|\varepsilon|$ are ellipses.
- Using an ℓ_1 penalty will favor making one of the coefficients smaller.

The magic of the ℓ_1 norm



from <https://tianyizhou.wordpress.com/2010/08/23/compressed-sensing-review-1-reconstruction-algorithms/>

Previous UQ Work

- In the nuclear field, Watanabe et al. used L1 minimization to estimate first-order sensitivity coefficients for a pincell burnup problem with 5000 parameters. They needed 500 simulations to estimate the parameters efficiently. These results did not leverage a regression framework, which could lead to improvement.
- For climate uncertainty analysis, LLNL researchers have used lasso-type approaches to estimate polynomial chaos expansion coefficients.
- In this presentation I'll present the results of a bake-off to compare different approaches to estimate second-order sensitivity coefficients, i.e., the quadratic and interaction terms neglected in a first-order sensitivity analysis.

Regularized Regression Approaches

In these methods we explicitly change the minimization problem.

- Lasso regression (OLS plus an ℓ_1 penalty based on size of β 's):

$$\beta = \underset{\beta}{\operatorname{argmin}} \{ \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda_1 \|\beta\|_1 \} \quad (1)$$

- Ridge regression (OLS plus an ℓ_2 penalty based on size of β 's):

$$\beta = \underset{\beta}{\operatorname{argmin}} \{ \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda_2 \|\beta\|_2^2 \} \quad (2)$$

- Elastic net regression (Combination of Lasso and Ridge):

$$\beta = \underset{\beta}{\operatorname{argmin}} \{ \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \alpha \lambda_1 \|\beta\|_1 + (1 - \alpha) \lambda_2 \|\beta\|_2^2 \} \quad (3)$$

- Dantzig selector (Minimize ℓ_∞ error in fit with ℓ_1 penalty on β 's):

$$\beta = \underset{\beta}{\operatorname{argmin}} \{ \|\beta^T (\mathbf{Y} - \mathbf{X}\beta)\|_\infty + \lambda_1 \|\beta\|_1 \} \quad (4)$$

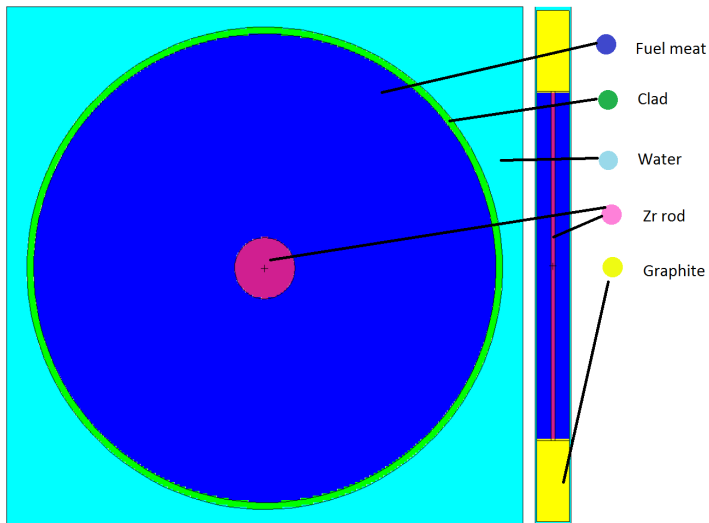
Non-Bayesian Regularization Regression Approaches (cont'd)

- Non-Bayesian L-2 norm constraint put too much strength on limiting parameters with higher magnitudes: over-penalization

Problem settings

Lattice of TRIGA fuels pin modeled with MCNP

- QoI: k_{eff}



Problem descriptions

There are 299 sensitivity coefficients taken into account in this problem:

- 23 input parameters:
 - 6 geometric parameters: e.g. r -fuel (fuel radius)
 - 17 material parameters: e.g. ρ -Zr (Zr rod mass density)
- 253 pairwise interactions (23 choose 2)
- 23 quadratic terms

The aim is to investigate the sensitivity of the criticality to the parameters, especially the second order terms. The model is:

$$\frac{\delta k}{k} \approx \sum_{i=1}^{23} c_i \left(\frac{\delta x_i}{x_i} \right) + \sum_{i=1}^{22} \sum_{j=i+1}^{23} c_{ij} \left(\frac{\delta x_i}{x_i} \right) \left(\frac{\delta x_j}{x_j} \right) + \sum_{i=1}^{23} c_{ii} \left(\frac{\delta x_i}{x_i} \right)^2 \quad (5)$$

where c_i , c_{ij} and c_{ii} , $i = 1, \dots, 23, j \neq i$, are the first order, interactive and quadratic sensitivity coefficients, respectively.

Reference data

- We are going to compare reference sensitivity coefficients to the coefficients computed by various regularized regression techniques using many few code runs (cases).
- The reference coefficients are computed using 1058 cases.
 - We need 46 total simulations for the linear and quadratic parameters
 - 1012 simulations are needed for the 253 interactions (4 simulations for each)
- The goal of this research is to see if regularized regression techniques can give coefficient estimates close to the references using many fewer simulation runs than the 1058 cases.

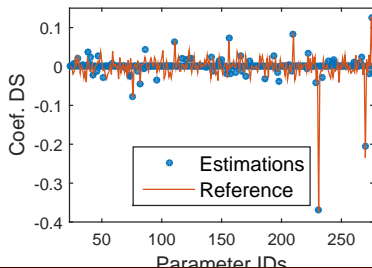
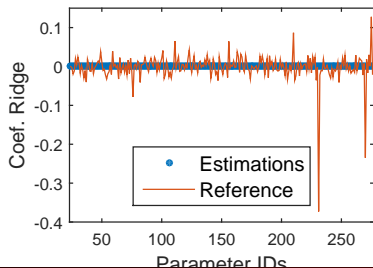
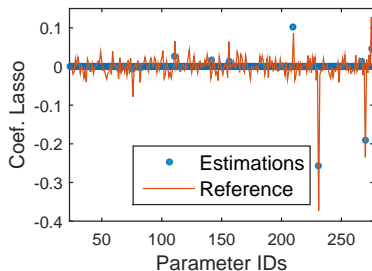
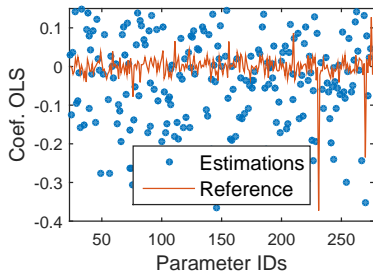
Coefficient estimation

We wish to estimate the numeric value of the coefficients and compare with the reference result.

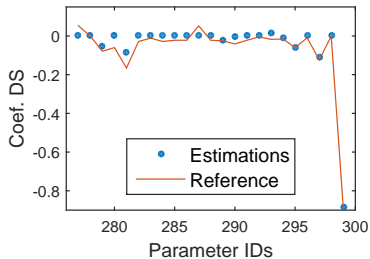
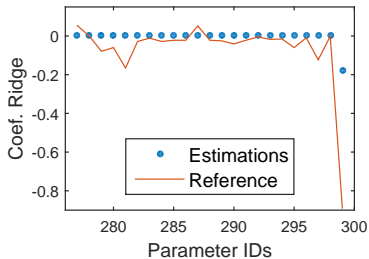
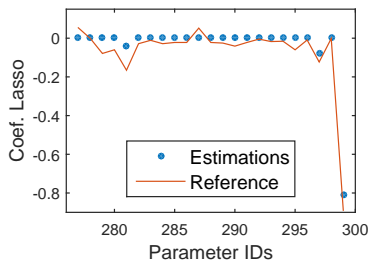
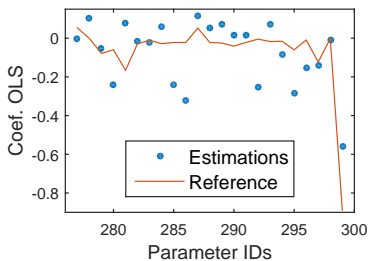
- Each parameter is assigned an ID
- IDs from 24 to 276: interactive coefficients
- IDs from 277 to 299: quadratic coefficients

The results that follow all use 299 samples, about 28% of those used in the reference calculation.

Coefficient Estimation: Interactions



Coefficient Estimation: Quadratic



Section 5

- 1 Introduction
 - Background
- 2 The Single Detector Neutron Camera
- 3 Disjoint Monte Carlo Tallies
- 4 Sparse Estimation of Sensitivities
 - Problem settings
 - Coefficient Estimation
- 5 Improving Numerical Methods Using Optimization**
- 6 Summary

Problems with Standard Numerical Representations

- In numerical methods we often represent a solution as an expansion in basis functions:

$$u(x) \approx \sum_{\ell=1}^N c_{\ell} b_{\ell}(x).$$

- The standard way of estimating coefficients is by solving the least-squares optimization problem:

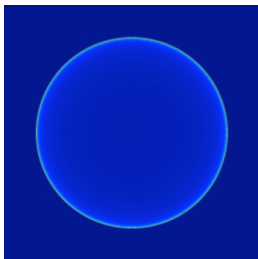
$$\text{minimize } \int \left(u(x) - \sum_{\ell=1}^N c_{\ell} b_{\ell}(x) \right)^2 dx.$$

- The solution to this problem is

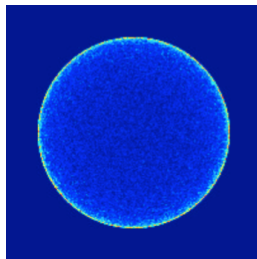
$$c_{\ell} = \int u(x) b_{\ell}(x) dx.$$

- These types of approximations arise in finite element methods, and the P_n method for radiation transport.
- These can have the same shortcomings as least-squares.

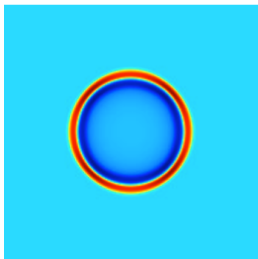
Example: The Line Source Problem ²



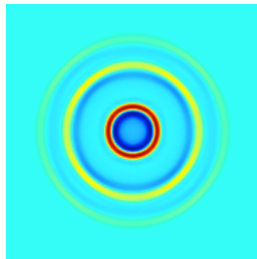
(a) analytic



(b) Monte-Carlo

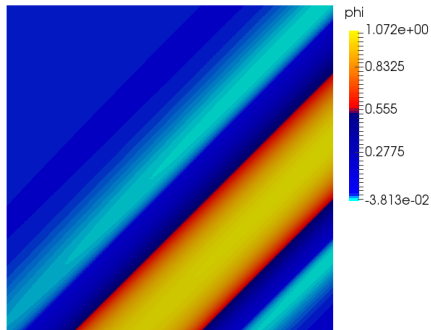


(c) P_1

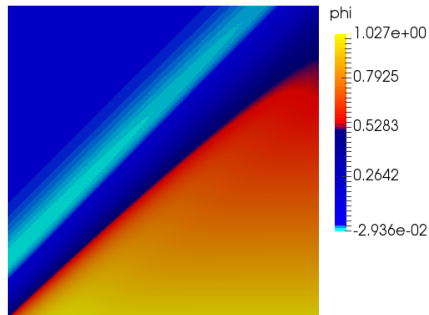


(d) P_5

Transport of a beam



25: L_2 in void



26: L_2 in absorber

Change the Optimization Problem

- Instead of solving the least-squares optimization problem we could regularize it with a penalty:

$$\text{minimize } \int \left(u(x) - \sum_{\ell=1}^N c_{\ell} b_{\ell}(x) \right)^2 dx + \sum_{\ell=1}^N c_{\ell}^2.$$

- This solution filters out high frequency oscillations and leads to a filtered P_n method.
- Or, we could solve an ℓ_1 minimization problem:

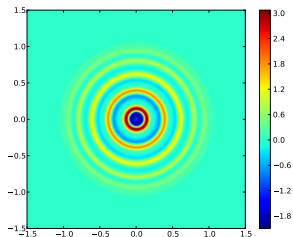
$$\text{minimize } \int \left| u(x) - \sum_{\ell=1}^N c_{\ell} b_{\ell}(x) \right| dx.$$

- The absolute value is easy to work with, but we can approximate it as

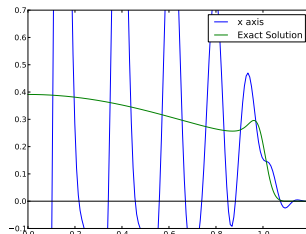
$$|a| \approx \sqrt{a^2 + \theta}, \quad 0 < \theta \ll 1.$$

- This leads to a regularized L_1 finite element method.

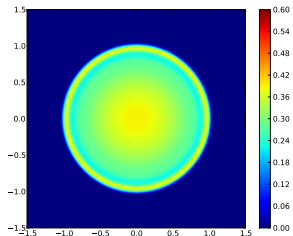
The Line Source Problem Again



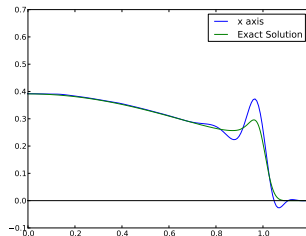
(a) P_{11}



(b) P_{11} -Lineout

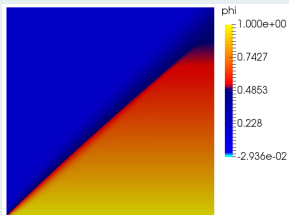


(c) FP_{11}

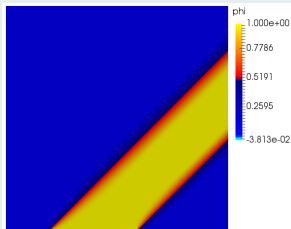


(d) FP_{11} -Lineout

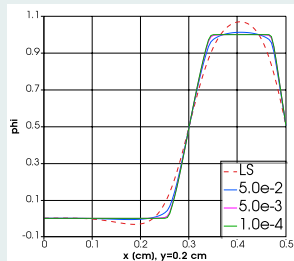
Transport of a beam



27: Absorber problem



28: Void



29: Line-outs with dif. θ in void

Section 6

- 1 Introduction
 - Background
- 2 The Single Detector Neutron Camera
- 3 Disjoint Monte Carlo Tallies
- 4 Sparse Estimation of Sensitivities
 - Problem settings
 - Coefficient Estimation
- 5 Improving Numerical Methods Using Optimization
- 6 Summary**

- The unifying theme in this talk is that we can use optimization to change the way that we approach problems of detection, simulation, and analysis.
- Can enable a single detector to be a camera: results exist for optical images and simulations suggest that it is possible for neutrons.
- Can improve simulation memory footprint and accuracy.

Thank you!

Compressed Sensing for Nuclear Engineering

NC State Nuclear Engineering Seminar

Ryan G. McClarren

Texas A&M University