Due Date: 17:00, Wed, March 4th, 2015
TA: Yi Shen (yishen@sep.stanford.edu)

Lab 6 - Helical coordinates and preconditioning

Your name[1]

ABSTRACT

We will look at helical coordinates and the steps required to set up and utilize them. You will start by creating different helical filters of different sizes and dimensions and test convolving and deconvolving by these filters. Then, you will factorize the some filters and test the accuracy of the factorization. Finally, you will precondition an optimization problem using helical deconvolution.

INTRODUCTION

In the textbook we saw how helical filters can handle any number of dimensions efficiently and easily. Moreover, they allow several operations, such as deconvolution and factorization, in any dimensions using corresponding 1D operators. This makes helical filters and operators very useful for many applications, especially preconditioning.

Preconditioning can provide great speed ups and efficiency improvements for the optimization. However, it requires knowing the inverse of the regularization operator. In many cases, it is very difficult to estimate the inverse operator. However, the helical deconvolution allows us to divide by the filter response of the operator, assuming stationarity, without having to know the inverse.

YOUR ASSIGNMENT

In this lab, you will explore and utilize the SEP library of codes that exist in (`/opt/SEP/seplib/seplib_base/lib/class/gee/`) in your programs. This library has many solvers, stepper, operators and tools. Moreover, you can use any dataset in (`/net/server/homes/sep/prof/gee/Data`) to do your tests. When importing a dataset, make sure to follow the same steps done in Lab1. First, you will create 2D helix filters and test their convolution and deconvolution operators. Second, you will factorize the Laplacian operator as well as another operator of your choice and test

[1]**e-mail:** youremail@stanford.edu

the accuracy of the factorization. Finally, you will run and compare a regularized optimization to a preconditioned optimization. You are not provided with anything but this PDF. You will create the whole project directory and code all the programs needed to answer the questions. Make sure you check and utilize the modules in the SEP library before you code your own. Modify the `Makefile` to add rules that generate the required tests, results and figures for each question. Also be sure to include the correct and complete set of prerequisites for each `Makefile` rule (including the executable programs). Also, do not forget to add the PDF figures to the default target and include the figures in this paper.

Questions

Helix filters

1. Create and plot a 2D model with some sparse spikes and make sure some spikes are close to the edges and corners. Create a helix filter of your choice (be creative) for the spiky model. Apply and plot the forward convolution of the filter on the model to see the filter response. Is this response what you expect? How does the filter behave around the edges?

YOUR ANSWER.

2. Apply and plot the adjoint convolution on the result of the previous question. Is the adjoint a good approximation to the inverse? Why are the results symmetric at each spike location?

YOUR ANSWER.

3. Apply and plot the forward deconvolution of the filter on the model to see the inverse filter response. How does the inverse response compare to the filter itself (be elaborate)?

YOUR ANSWER.

4. Apply and plot the adjoint deconvolution on the result of the previous question. How does this result compare to the results in the second question (be elaborate)?

YOUR ANSWER.

5. Use the deconvolution operator on the results of the second question to remove the effect of the adjoint convolution operator. Did the deconvolution operator correctly remove the adjoint convolution effects?

YOUR ANSWER.

6. Use the deconvolution operator on the results of the previous question to remove the effect of the forward convolution operator. Did the deconvolution operator recover the original spiky model? Plot your results.

YOUR ANSWER.

Factorization

7. Compute the helix derivative by factorizing the Laplacian operator. Pick a dataset from the data directory and apply first derivative, the Laplacian operator and the helix derivative. Plot each of your results. How do the three operators compare? Is the helix derivative closer to the first derivative or Laplacian operator?

YOUR ANSWER.

8. Apply and plot the adjoint of helix derivative on the results of its forward? Is this result identical to applying the Laplacian operator? Why?

YOUR ANSWER.

9. Create any filter of your choice, called filter A. Now create a second filter, called filter B, which is the autocorrelation of the filter A. Apply the forward and adjoint of filter A and compare it to applying the forward of filter B. Plot all your results.

YOUR ANSWER.

10. Factorize filter B. Does the factorization recovers filter A? Why or why not? Plot your results.

YOUR ANSWER.

Preconditioning

11. Set up a regularized optimization of your choice. You can either pick a problem done in previous lab or make a new one given the datasets and operators available to you. The problem you pick should require regularization to give proper results where the regularization operator is stationary but not only diagonal. Clearly define the objective function and all its components.

YOUR ANSWER.

12. Why does this problem need to be regularized?

YOUR ANSWER.

13. Run the regularized solver and plot the original data, the inverted model, the reconstructed data and the curve of residual norm as a function of iterations. How well did the optimization solve the problem.

YOUR ANSWER.

14. Rewrite the objective function after preconditioning and clearly define all its components.

YOUR ANSWER.

15. Convert your regularization operator into a helix filter and run the preconditioned solver with the deconvolution operator as the inverse of the regularization operator. Plot the original data, the inverted model, the reconstructed data and the curve of residual norm as a function of iterations. How does the preconditioned solver compare to the regularized solver in both accuracy and convergence rate?

YOUR ANSWER.

Extra Credit

1. What are the underlying assumptions of the factorization you applied?

YOUR ANSWER.

2. If the regularization operator is non-stationary or non-invertible, can we do an approximate preconditioning? What would the objective function be in that case?

YOUR ANSWER.

DONE

When you are all finished modifying the source files, `Makefile` and latex file, make sure that will paper will compile correctly from a cleaned directory using the default targets only. In other words, the following sequence of commands should produce your a PDF version of your paper without any problems: `make burn`; `scons -c`; `make`; `scons`.

Once you make sure everything is working properly, clean up your directory by typing `make burn`; `scons -c`. Then, compress the lab directory using the command `tar cvzf Lab6.gzip Lab6` and submit the compressed file to your TA.