

## *Overview of Multi-grid Project*

This project example on this OCW site is a slight refinement of my final project in Prof. Strang's 18.086 class. The project implements a 2D multi-grid solver for Laplace's and Poisson's equation. Multi-grid support is included for Dirichlet boundary conditions, as well as for sources within the grid. Internally pinned Dirichlet boundaries, while possible to specify, do not function properly if multi-grid schemes are used in my implementation. I wanted to add this feature, but ran out of time at the end of the project.

I have attempted to give an explanation of how to use the solver by including the file "ProjectOCW.m," a simple numerical experiment conducted with the solver. Once all project files are placed in the same directory, one should simply be able to open the mentioned file and run it in MatLab. The file will plot the error and the guess at the solution at various steps in a V-cycle approach to multi-grid.

Additionally, I have extensively commented this file to give the user enough knowledge to modify the file to conduct experiments of one's own. The solver tracks the norm of the residual as the solver progresses, which is a key statistic to keep an eye on. Upon altering the MatLab script, the user can decide parameters like how many steps to stay on a grid, what Jacobi damping factor to use, or a number of other parameters.

The user can also import images to use as a two-dimensional source term in the problem, or import other images to define the boundary conditions or initial guess. Images can be imported into MatLab. In order to be incorporated into the solver, however, the images must become appropriately sized matrices with values corresponding to some interpretation in Laplace's equation. The simplest example is importing a monochrome bitmap for source terms, where black pixels could map to zero entries in the source terms, and white to some determined value.

I attempted to thoroughly explain "ProjectOCW.m," but left explanation of the helper functions more vague. This project was not intended originally to teach multi-grid, but rather act as an example of implementation of a multi-grid solver. Furthermore, there are definitely more efficient and elegant ways to implement a number of the operations of the solver in MatLab. For these reasons, I found a thorough explanation of the inner workings of the solver to be outside the scope of this posting, which is meant to show multi-grid in action, rather than thoroughly explain every detail of design decision in implementing the system.

I hope you enjoy running the pre-designed experiment and will alter the example file to run experiments of your own. The system here is capable of a good number of experiments. I learned a great deal about the method by simply changing parameters easily alterable in this example.

Finally, I'd like to thank Prof. Strang for all of his help and advice, Dave Willis and Jaydeep Bardhan for their help, and would like to acknowledge the immense help of *A Multigrid Tutorial* by Briggs et al., a must-read for anyone interested in learning multi-grid.

Joseph Kovač, S.B. MIT '05  
9/12/05  
Graduate Student, MIT EECS