A GEOMETRIC MULTIGRID APPROACH TO SOLVING THE 2D INHOMOGENEOUS LAPLACE EQUATION WITH INTERNAL DIRICHLET BOUNDARY CONDITIONS

Leo Grady

Siemens Corporate Research
Department of Imaging and Visualization
755 College Road East
Princeton, NJ 08540

Tolga Tasdizen and Ross Whitaker

Scientific Computing and Imaging Institute
3490 Merrill Engineering Building
Salt Lake City, UT 84105

ABSTRACT

The inhomogeneous Laplace (Poisson) equation with internal Dirichlet boundary conditions has recently appeared in several applications to image processing and analysis. Although these approaches have demonstrated quality results, the computational burden of solution demands an efficient solver. Design of an efficient multigrid solver is difficult for these problems due to unpredictable inhomogeneity in the equation coefficients and internal Dirichlet conditions with arbitrary location and value. We present a geometric multigrid approach to solving these systems designed around weighted prolongation/restriction operators and an appropriate system coarsening. This approach is compared against a modified incomplete Cholesky conjugate gradient solver for a range of image sizes. We note that this approach applies equally well to the anisotropic diffusion problem and offers an alternative method to the classic multigrid approach of Acton [1].

1. INTRODUCTION

The solution of the inhomogeneous Laplace (Poisson) equation with internal Dirichlet boundary conditions has recently appeared in several applications, ranging from image segmentation [2, 3] to image filtering [2] and image colorization [4]. Although these algorithms are framed in a discrete (graph) setting, they have been almost exclusively employed in a rectilinear coordinate system, resulting in the use of a (widely) banded Laplacian matrix. Unfortunately, traditional fast Laplace/Poisson solvers are inappropriate due to the inhomogeneity of the PDE coefficients. This paper addresses the problem of an efficient solution to the inhomogeneous problem by introducing a weighted multigrid approach [5].

A multigrid approach was applied by Acton [1] to the similar problem of anisotropic diffusion. His approach employed a simple injection restriction operator and a traditional interpolation prolongation method to prevent smoothing/restriction across edge boundaries. The formal relationship and relative advantages of our approach will be discussed in Section 4.

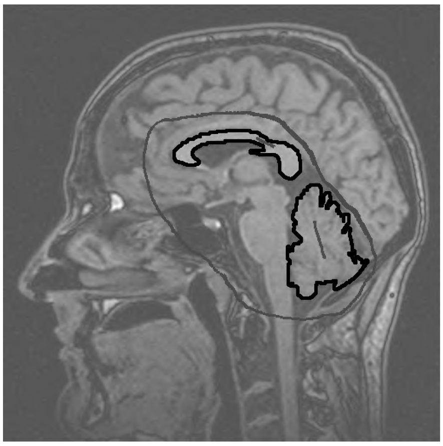


Fig. 1. Example of the multilabel segmentation obtained from the algorithm of [3]. Gray marks represent the result of user interaction (*i.e.*, seeds) to indicate three objects (corpus callosum, cerebellum and background). Thick black lines indicate the computed segment boundaries. Even though no prior knowledge is built into the algorithm to find a particular object, the algorithm correctly segments both objects, despite unusual shapes and textures.

The general application is this: Given knowledge at certain pixels, termed **seeds**, of a certain quantity of interest (*e.g.*, object labels in [3], grayscale intensities in [2] or colors in [4]), assign appropriate quantities to the unlabeled pixels that reflect the spatial structure of the image. The approach is then to set the seed pixels as Dirichlet boundary conditions (despite being internal to the domain), allow the pixel intensities to define the coefficients (weights) and then solve the Laplace equation (or Poisson equation, in the case of [2]) to find the quantities at the unlabeled nodes. As shown in [3], this approach respects weak (or absent) object boundaries, has provable robustness to noise and admits interpretation in the context of a random walk in the domain. Figure 1 shows segmentation results on a medical image obtained through application of the segmentation technique described in [3].

0-7803-9134-9/05/$20.00 ©2005 IEEE

2. PROBLEM

Since [3, 2, 4] were formulated on a discrete space (due to a finite set of pixels and corresponding "diffusion constants") we find the language of graph theory most suitable for exposition. A **graph** consists of a pair $G = (V, E)$ with **vertices (nodes)** $v \in V$ and **edges** $e \in E \subseteq V \times V$. An edge, e, spanning two vertices, v_i and v_j, is denoted by e_{ij}. A **weighted graph** assigns a nonnegative value to each edge called a **weight**. The weight of an edge, e_{ij}, is denoted by $w(e_{ij})$ or w_{ij}. The **degree** of a vertex is $d_i = \sum w(e_{ij})$ for all edges e_{ij} incident on v_i. The following will assume that our graph is connected. In our context of 2D image processing, the graph nodes are taken as the image pixels, which lie on a rectangular, 4-connected, grid. Image intensities may by converted into edge weights (*i.e.*, diffusion constants) through many different methods [6, 4, 1, 3]. Although these values may be interpreted as diffusion constants, we will refer to them as *weights* throughout this manuscript.

Define the combinatorial Laplacian matrix [7] as

$$L_{v_iv_j} = \begin{cases} d_{v_i} & \text{if } i = j, \\ -w_{ij} & \text{if } v_i \text{ and } v_j \text{ are adjacent nodes}, \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

where $L_{v_iv_j}$ is used to indicate that the matrix L is indexed by vertices v_i and v_j.

Given a set of marked pixels (obtained interactively or automatically) to be set with Dirichlet boundary conditions, we may partition the vertices into two sets, V_M (marked/seed pixels) and V_U (unmarked pixels) such that $V_M \cup V_U = V$ and $V_M \cap V_U = \emptyset$. The Laplacian may be decomposed into the form

$$L = \begin{bmatrix} L_M & B \\ B^T & L_U \end{bmatrix}, \tag{2}$$

corresponding to the marked/unmarked sets. Fixing some of the nodes as "boundary" nodes results in their removal from the Laplacian matrix and incorporation into the right hand side (see [2]). Therefore, our purpose is to develop a multigrid technique that is addressed to solving the problem

$$L_U x_U = f, \tag{3}$$

for x_U in the presence of some right hand side, f, that depends on the application (*cf.*, [2, 4, 3]). Given the context of potential theory for these applications, we refer to the quantities x_U as the **potentials** for the unmarked set. The potentials of the marked set are assumed known and fixed (depending on the application).

We note that the anisotropic diffusion problem has a similar formulation with $V_M = \emptyset$ and $V_U = V$. In this context, the solution to the combinatorial diffusion (heat) equation [7] requires solution to

$$\frac{dx}{dt} = -Lx, \tag{4}$$

given some time, t, and initial distribution, x_0. By employing a backward Euler approach to solving the diffusion equation, a linear system may be established with the form

$$\left(\frac{1}{t}I + L\right)x = \frac{1}{t}x_0, \tag{5}$$

which is equivalent to our system (3) with $f = \frac{1}{t}x_0$ and a constant addition to the diagonal of L. Since there are typically no Dirichlet boundary conditions in the formulation of diffusion, a multigrid approach to this problem is often less complicated. Although we will ignore the addition of a constant to the diagonal of our Laplacian operator, the multigrid methods developed in the present work have straightforward application to the problem of anisotropic diffusion.

3. MULTIGRID REVIEW

Multigrid methods have proven extremely successful at solving the systems of equations that arise in the solution of PDEs with linear complexity [5]. In general, there are two branches of the method — geometric and algebraic. Algebraic multigrid approaches aspire to a "black box" method that can apply the technique to an arbitrary linear system. In contrast, geometric methods evolved out of attempts to solve PDEs on a rectilinear domain (especially elliptic and parabolic systems) where it may be assumed that coarsened versions of the operator also represent rectilinear grids. Although discrete in nature, and therefore equally defined on arbitrary graphs, application of the methods in [6, 2, 4, 3] to standard images results in a Laplacian operator with a sparsity structure that represents a grid. For this reason, the method we develop is a geometric multigrid method.

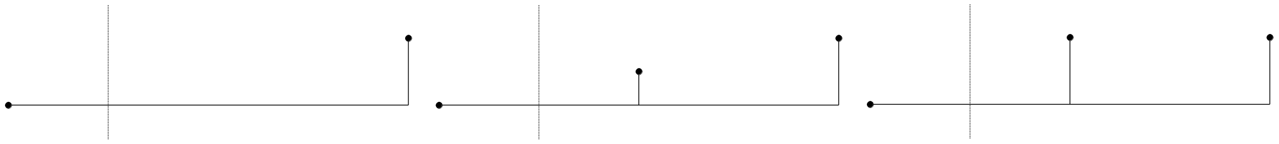
A review of the steps involved in the multigrid method is referred to [5]. The main issues that must be addressed in order to design a multigrid method are:

1. Specifying the restriction operator
2. Specifying the prolongation operator
3. Producing a coarsened operator

Section 4 outlines the design of these operators in the context of inhomogeneous Laplacian operators with internal boundary conditions. We note that design of these operators also specifies the nested dissection method and therefore allows the "full multigrid method".

4. MULTIGRID DESIGN

Acton [1] specifies the restriction operator as simple injection and the prolongation operator as "traditional interpolation". These operators may work well enough for calculating a few iterations of diffusion, but do not offer rapid convergence for the steady-state (*i.e.*, elliptical) equations



(a) Coarse-level solution

(b) Unweighted projection

(c) Weighted projection

Fig. 2. Example of a 1D projection over an object boundary (represented by the vertical dashed line). (a) Current solution at coarse level. (b) Unweighted projection operator. Note that projected values do not respect object boundary. (c) Weighted projection operator. The weighted restriction operator is defined naturally as the adjoint of this projection operator.

addressed in the present work. One reason for this, we believe, is that the weight structure of the image is not taken into account in the operator design. Secondly, there is solid theoretical foundation [5] for requiring that the prolongation and restriction operators be formally *adjoint* to each other. To review, two operators, A and A^*, are adjoint if

$$\langle Ax, y\rangle = \langle x, A^*y\rangle\,, \tag{6}$$

is satisfied. For a finite, linear, operator represented by matrix A, the adjoint is given by $A^* = A^T$ [8]. We note that Acton's prolongation/restriction operators are not adjoint to each other.

4.1. Prolongation

Our approach to designing the prolongation/restriction operators will be to define the prolongation operator (since this is intuitive) and then specify the restriction operator as its adjoint.

Bilinear interpolation is a standard prolongation operator, often termed the "full weighting" operator. Unfortunately, in our case, this operator does not respect the edge weights and therefore may interpolate over object boundaries. Therefore, we propose to use a weighted bilinear interpolation, with weights given by the edge weights. Specifically, the prolongation operator proceeds in three steps for values at the fine level, x^0, and the coarse level, x^1. For ease of exposition, we use north/south/east/west notation to indicate the neighbors of a node and the corresponding weight between them.

$$\begin{aligned}
x^0(2i,2j) &= x^1(i,j)\ \forall i,j,\\
x^0(2i+1,2j) &= \frac{(w_{\mathrm{E}}x^0_{\mathrm{E}} + w_{\mathrm{W}}x^0_{\mathrm{W}})}{w_{\mathrm{E}}+w_{\mathrm{W}}},\\
x^0(2i+1,2j+1) &= \frac{w_{\mathrm{N}}x^0_{\mathrm{N}} + w_{\mathrm{E}}x^0_{\mathrm{E}} + w_{\mathrm{S}}x^0_{\mathrm{S}} + w_{\mathrm{W}}x^0_{\mathrm{W}}}{w_{\mathrm{N}}+w_{\mathrm{E}}+w_{\mathrm{S}}+w_{\mathrm{W}}},
\end{aligned}\tag{7}$$

Effectively, the first step injects the coarse-level solution to the fine grid. The subsequent equations describe a weighted bilinear interpolation over the remaining fine-level pixels. We note that for a unity-weighted (*i.e.*, homogeneous) domain, this operator becomes standard bilinear interpolation. In Figure 2 we show 1D projection over an object boundary (represented by the vertical, dashed line) and the effects of applying standard and weighted projection.

4.2. Restriction

The restriction operator is defined as the adjoint of the weighted prolongation operator defined above. This adjoint may be written in four steps. For ease of exposition (and implementation) we first modify the fine-level vector. In practice this would be done with a temporary vector.

$$\begin{aligned}
x^0(2i+1,2j) &= x^0(2i+1,2j) + \frac{w_E x^0_E}{d_E} + \frac{w_W x^0_W}{d_W},\\
x^0(2i,2j+1) &= x^0(2i,2j+1) + \frac{w_N x^0_N}{d_N} + \frac{w_S x^0_S}{d_S},\\
x^0(2i,2j) &= x^0(2i,2j) + \frac{w_N x^0_N}{d_N} + \frac{w_E x^0_E}{d_E} +\\
&\quad \frac{w_S x^0_S}{d_S} + \frac{w_W x^0_W}{d_W},\\
x^1(i,j) &= x^0(2i,2j).
\end{aligned}\tag{8}$$

The factor d_N denotes the degree of the node to the north, etc. Effectively, one may think of this operator as reversing the projection. We note that a unity weighted lattice (*i.e.*, homogeneous domain) would cause the weighted restriction operator above to be the standard "full-weighting restriction" given in [5].

4.3. Operator coarsening

The recommended operator coarsening, given an adjoint prolongation/restriction operator is

$$L^{k+1} = RL^kP. \tag{9}$$

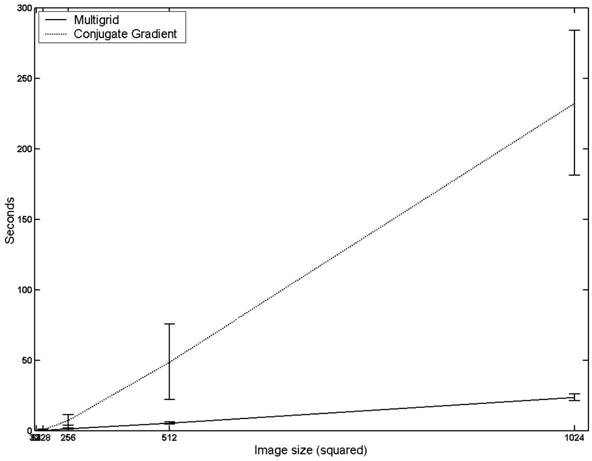


Fig. 3. Comparison of the conjugate gradient method (with incomplete Cholesky preconditioner) to the proposed multigrid method for solving (3) when segmenting images [3]. All images were square, with the length of one side given by the x-axis.

Unfortunately, such a construction of the coarsened operator yields a matrix with a sparsity pattern that does not correspond to a lattice representation, resulting in a lower efficiency implementation of coarse operations and storage. Therefore, an effective heuristic to coarsen a fine-level L while maintaining the sparsity structure of a lattice is to treat the vertical/horizontal weights as two $(N-1) \times (N-1)$ images, and applying the restriction operator to determine the weights of the coarse lattice. We employ this procedure, using the weighted restriction/prolongation operators described above.

In the methods of [3, 4] the marked nodes (*i.e.*, removed nodes) must also be incorporated into the higher-level operator. This is done by considering a coarse-level node to be marked if any of its eight fine-level neighbors are also marked[1].

5. VALIDATION AND CONCLUSION

We compared our multigrid approach for solving (3) (obtained from the segmentation problem of [3]) to a conjugate gradients method with a modified incomplete Cholesky preconditioner [9] on images of increasing size. Figure 3 plots the relative speed for the two methods and demonstrates that the proposed multigrid approach has a linear relationship to data size, and outperforms the conjugate gradient approach by roughly an order of magnitude. Furthermore, the variability of computation time for the multigrid method was also much lower.

We have presented a multigrid method for solving problems of the form (3) that have recently become important in image processing applications. Furthermore, our approach offers an alternative to Acton's [1] multigrid method for solving the anisotropic diffusion problem that employs adjoint restriction and prolongation operators which respect the diffusion constants (edge weights). The main feature of these weighted operators is that they do not smooth or restrict over object boundaries (represented by edge weights).

Future work includes extension of the multigrid method to a 3D lattice and use of an algebraic multigrid technique to design even more effective prolongation, restriction and coarsened operators.

[1] We thank Anat Levin for this suggestion.

6. REFERENCES

[1] S. T. Acton, "Multigrid anisotropic diffusion," *IEEE Trans. on Image Processing*, vol. 7, no. 3, pp. 280–291, March 1998.

[2] L. Grady, "Space-variant computer vision: A graph-theoretic approach," Ph.D. dissertation, Boston University, Boston, MA, 2004.

[3] L. Grady and G. Funka-Lea, "Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials," in *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis, ECCV 2004 Workshops CVAMIA and MMBIA*, ser. Lecture Notes in Computer Science, M. Šonka, I. A. Kakadiaris, and J. Kybic, Eds., no. LNCS3117. Prague, Czech Republic: Springer, May 2004, pp. 230–245.

[4] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in *Proceedings od ACM SIGGRAPH 2004*, ser. ACM Transaction on Graphics, vol. 23, no. 3, ACM. New York: ACM, August 2004, pp. 689–694.

[5] W. L. Briggs, V. E. Hensen, and S. F. McCormick, *A Multigrid Tutorial*, 2nd ed. SIAM, 2000.

[6] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, July 1990.

[7] F. R. K. Chung, *Spectral Graph Theory*, ser. Regional conference series in mathematics. Providence, R.I.: American Mathematical Society, 1997, no. 92.

[8] E. Kreyszig, *Introductory Functional Analysis with Applications*, ser. Wiley Classics Series. John Wiley and Sons, 1989.

[9] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed. The Johns Hopkins University Press, 1996.