Modern Preconditioned Eigensolvers for Spectral Image Segmentation and Graph Bisection
Workshop on Clustering Large Data Sets
Third IEEE International Conference on Data Mining (ICDM 2003)

Andrew V. Knyazev
Department of Mathematics
Center for Computational Biology
Center for Computational Mathematics
University of Colorado at Denver, P.O. Box 173364,
Campus Box 170, Denver, CO 80217-3364.
WWW URL: http://www-math.cudenver.edu/~aknyazev
e-mail: andrew.knyazev@cudenver.edu

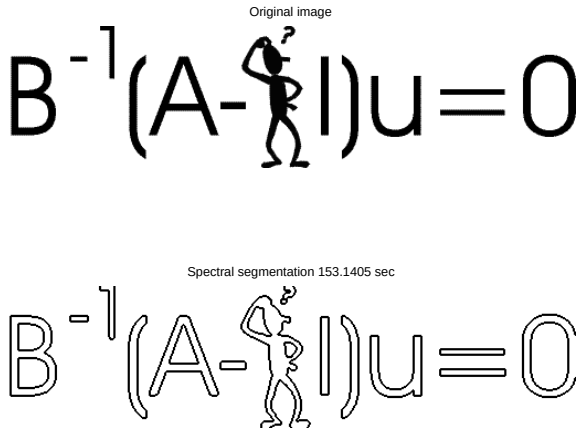


Figure 1. Segmentation by LOBPCG.

Abstract

Known spectral methods for graph bipartition and image segmentation require numerical solution of eigenvalue problems with the graph Laplacian. We discuss several modern preconditioned eigenvalue solvers for computing the Fiedler vectors of large scale eigenvalue problems. The ultimate goal is to find a method with a linear complexity, i.e. a method with computational costs that scale linearly with the problem size. A locally optimal block preconditioned conjugate gradient method,be a promising candidate (if matched with a high quality preconditioner), compared against the Lanczos method.

This material is based upon work supported by the National Science Foundation and the Intelligence Technology Innovation Center through the joint "Approaches to Combat Terrorism" Program Solicitation NSF 03-569.

1. Introduction

Several computationally efficient approaches to clustering large sparse and high-dimensional data are based on finding several extreme eigenvalues and corresponding eigenvectors of large sparse symmetric matrices. For example, a well-known problem of graph bisection and a corresponding problem of image segmentation can be mathematically formulated as a problem of minimization of normalized cuts (Ncuts). One practical approach for approximate solution of the Ncuts problem involves numerical solution of the following eigenvalue problem

$$(D-W)y = \lambda Dy, \tag{1}$$

where W is the graph association matrix with the zero diagonal and D is a diagonal matrix defined as a row sum of W. The eigenvector, corresponding to the second smallest eigenvalue, is called the Fiedler vector and it determines the graph bipartition and the corresponding image segmentation.

In a similar spectral partitioning algorithm, the following problem,

$$(D-W)y = \lambda y, \tag{2}$$

evidently analogous to (1), needs to be solved.

Replacing the original problem of minimization of a normalized cut with an eigenvalue problem (1) or (2) is a major step in reducing the computational costs of the graph bisection and image segmentation problems. However, numerical solution of the eigenvalue problem still poses major computational challenges for large size images. As even low-end digital cameras produce mega-pixel images nowadays, it is typical to face image segmentation problems that lead to eigenvalue problems with millions unknowns. Eigenvalue problems for matrices so large are difficult to solve numerically at present. Moreover, it might be expected that the demands for the increased resolution and the needs to treat live feed dynamic images will continue to overgrow the increase in the computational power. An efficient choice of a method for numerical solution of the eigenvalue problem becomes crucial. The ultimate goal could be to find a method with a linear complexity, in other words, a method with computational costs that scale linearly with the problem size.

2. Traditional eigenvalue solvers for large scale problems

The classical Lanczos method is an evident candidate for an eigenvalue solver of choice to compute the second smallest eigenpair of (1) or (2). Several software implementations of the Lanczos method are publicly available and are well maintained, e.g., ARPACK. The traditional Lanczos method requires the generalized eigenproblem (1) be reduced, at least implicitly, to a regular eigenvalue problem for a symmetric matrix. Since the matrix D is diagonal, an obvious inexpensive possibility is to apply the Lanczos method to matrix

$$(\sqrt{D})^{-1}(D-W)(\sqrt{D})^{-1}. \tag{3}$$

In the spectral partitioning algorithm, the Lanczos method can be applied directly to matrix

$$D-W. \tag{4}$$

The only, but significant, difficulty in this approach is that the smallest eigenvalues may not well relatively separated; therefore, the convergence of the Lanczos method may be slow. Even worse, by analogy with the case of the finite difference approximation of the Laplacian operator, we might expect that the separation gets worse with the increase in the problem size; thus, we do not likely achieve the linear complexity in this way.

Another well-known possibility is to apply the Lanczos method to the so-called shift-and-invert matrix, e.g., in case of eigenproblem (1) to the matrix

$$\sqrt{D}(D(1-\alpha)-W)^{-1}\sqrt{D}, \tag{5}$$

or, in case of eigenproblem (2) to the matrix

$$(D-\alpha I-W)^{-1}, \tag{6}$$

where α is a properly chosen small positive shift.

These matrices are not, of course, formed explicitly. Instead, each time the Lanczos method requires a multiplication of a vector f by matrix (5) or (6), a linear solver subroutine is called to solve the corresponding linear systems. If these linear systems are solved sufficiently accurately, the convergence of the Lanczos method is typically much faster compared to that when the matrix (3) or (4) is used in the Lanczos method. The difficulty now is that accurate numerical solution of linear systems, needed on each iteration of the Lanczos method, can be costly.

3. Preconditioned eigenvalue solvers for large scale problems

In the present talk, we discuss an alternative class of methods, namely, the preconditioned eigensolvers, e.g., Knyazev [2, 4]. With an appropriate selection of a preconditioning technique, such methods can converge as fast as the Lanczos method, applied to the matrix (6), at the computational costs per iteration comparable with that of the Lanczos method, applied to the matrix (4), thus, allowing to find a path between Scylla of the slow convergence and Charybdis of expensive linear solves.

One of the main simplifying ideas of preconditioned eigensolvers is to separate the choice of the eigensolver from the choice of the preconditioner, thus, treating the preconditioner as a "black box." E.g., if a multilevel preconditioner is used, it is called as an external subroutine outside of the eigensolver, in contrast to earlier known multilevel spectral graph bisection methods, where the levels are tightly embedded into the eigensolver. At the costs of loosing some efficiency, it allows for a great flexibility in choosing a preconditioner.

We concentrate in this talk on one of the most promising preconditioned eigensolvers, namely, on the Locally Optimal Block Preconditioned Conjugate Gradient (LOBPCG) method suggested and analyzed in Knyazev [1, 2, 3, 4]. In LOBPCG for computing a single eigenpair of matrix (3) or (4), the new iterate is determined by the Rayleigh–Ritz method on a three-dimensional subspace, which includes the previous iterate in addition to the current iterate and the preconditioned residual of the two-dimensional trial subspace of the steepest descent method. To compute the second eigenpair of (3) or (4), the iterations are performed in the orthogonal complement to the known null space. At the talk, we describe the algorithm of the LOBPCG and demonstrate results of numerical experiments.

4 Preliminary numerical results

We report here only limited and preliminary numerical results. This project is a work in progress, and we expect more results by the time of the conference.

A recently publicly released MATLAB Graph Analysis Toolbox by Leo Grady available for download at http://www.mathworks.com provides the image-matrix interface for the spectral partitioning. MATLAB's version 6.5 built-in eigensolver EIGS revision 1.45 is tested against the author code of LOBPCG. Figure 1 simply demonstrates that LOBPCG can be used for segmentation of artificial images.

Figure 2 shows segmentations of a natural 291x433 image using LOBPCG and EIGS, both applied to matrix (6) with $\alpha = 10^{-7}$ with no preconditioning. The MATLAB's built-in direct sparse *LU*-based linear solve is employed by both codes. The number of iterations in both codes is capped by three, however, we still observe good quality segmentations on both figures. The tests were performed on a Linux box with 4GB of RAM, the memory use topped slightly above 2GB. The timing is not, unfortunately, too reliable in these tests, since there were codes by other users running at the same time.

In our forthcoming tests, we plan to investigate incomplete factorization and algebraic multigrid preconditioning. We also expect to be able to run some tests on massively parallel computers, using our LOBPCG C code with MPI and HYPRE libraries, see below.

5 Availability of software for the preconditioned eigensolvers

The LOBPCG is publicly available in MATLAB, see `http://math.cudenver.edu/~aknyazev/software/CG/` and in C using MPI and HYPRE libraries for massively parallel computers, see http://www.llnl.gov/CASC/hypre/

Acknowledgements

The author is grateful to the workshop organizers for the opportunity to give a talk.

Bibliography

[1] A. V. Knyazev. A preconditioned conjugate gradient method for eigenvalue problems and its implementation in a subspace. In *International Ser. Numerical Mathematics, v. 96, Eigenwertaufgaben in Natur- und Ingenieurwissenschaften und ihre numerische Behandlung, Oberwolfach, 1990.*, pages 143–154, Basel, 1991. Birkhauser.

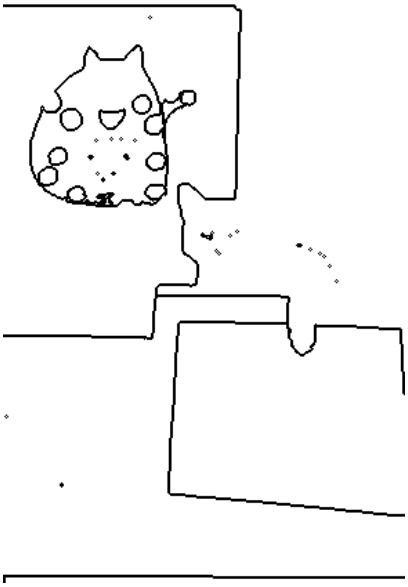
[2] A. V. Knyazev. Preconditioned eigensolvers—an oxymoron? *Electron. Trans. Numer. Anal.*, 7:104–123 (electronic), 1998. ISSN 1068-9613. Large scale eigenvalue problems (Argonne, IL, 1997).

[3] A. V. Knyazev. Preconditioned eigensolvers: practical algorithms. In Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, editors, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, pages 352–368. SIAM, Philadelphia, 2000. Section 11.3.

[4] A. V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM J. Sci. Comput.*, 23(2): 517–541, 2001.

Original image

Spectral segmentation 331.6193 sec



Original image

Spectral segmentation 486.0937 sec

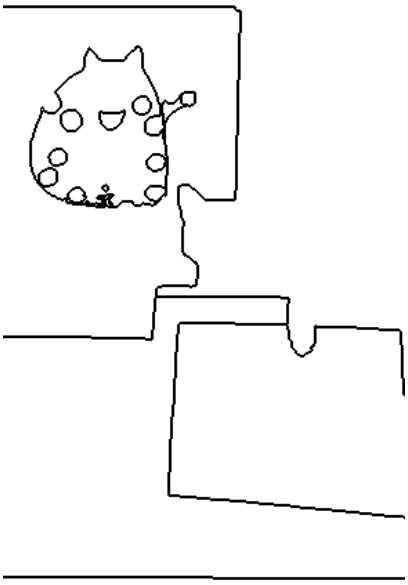


Figure 2. Spectral segmentation using LOBPCG (top) and EIGS (bottom).