Title: FEAST Eigensolver for Nonlinear Eigenvalue Problems

Author: Brendan Gavin Agnieszka Międlar Eric Polizzi

PII: S1877-7503(18)30209-6
DOI: https://doi.org/doi:10.1016/j.jocs.2018.05.006
Reference: JOCS 876

To appear in:

Received date: 26-2-2018
Revised date: 1-5-2018
Accepted date: 8-5-2018

Please cite this article as: Brendan Gavin, Agnieszka Międlar, Eric Polizzi, FEAST Eigensolver for Nonlinear Eigenvalue Problems, *<![CDATA[Journal of Computational Science]]>* (2018), https://doi.org/10.1016/j.jocs.2018.05.006

FEAST Eigensolver for Nonlinear Eigenvalue Problems

Brendan Gavin[a], Agnieszka Międlar[b], Eric Polizzi[c]

[a]*Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA 01003, USA,* bgavin@ecs.umass.edu
[b]*Department of Mathematics, University of Kansas, 1460 Jayhawk Blvd., Lawrence, KS 66045-7594, USA,* amiedlar@ku.edu
[c]*Department of Electrical and Computer Engineering, Department of Mathematics and Statistics, University of Massachusetts, Amherst, MA 01003, USA,* polizzi@ecs.umass.edu

Abstract

The linear FEAST algorithm is a method for solving linear eigenvalue problems. It uses complex contour integration to calculate the eigenvectors associated with eigenvalues that are located inside some user-defined region in the complex plane. This makes it possible to parallelize the process of solving eigenvalue problems by simply dividing the complex plane into a collection of disjoint regions and calculating the eigenpairs in each region independently of the eigenpairs in the other regions. In this paper we present a generalization of the linear FEAST algorithm that can be used to solve nonlinear eigenvalue problems. Like its linear progenitor, the nonlinear FEAST algorithm can be used to solve nonlinear eigenvalue problems for the eigenpairs corresponding to eigenvalues that lie in a user-defined region in the complex plane, thereby allowing for the calculation of large numbers of eigenpairs in parallel. We describe the nonlinear FEAST algorithm, and use several physically-motivated examples to demonstrate its properties.

Keywords: nonlinear eigenvalue problem, polynomial eigenvalue problem, quadratic eigenvalue problem, FEAST, contour integration, residual inverse iteration

1. Introduction

The nonlinear eigenvalue problem (NLEVP) consists of finding vectors $x \in \mathbb{C}^n$ and scalars $\lambda \in \mathbb{C}$ that satisfy

$$T(\lambda)x = 0, \tag{1}$$

where $T(\lambda) \in \mathbb{C}^{n\times n}$ is some matrix-valued function of λ, which we call the eigenvector residual function. The linear generalized eigenvalue problem (GEP) is a special case of (1), with

$$T(\lambda) = \lambda B - A, \quad A, B \in \mathbb{C}^{n\times n}. \tag{2}$$

Another well-known example of a NLEVP is a quadratic eigenvalue problem

$$T(\lambda) = \lambda^2 A_2 + \lambda A_1 + A_0, \quad A_2, A_1, A_0 \in \mathbb{C}^{n\times n}. \tag{3}$$

Quadratic eigenvalue problems can arise, for example, in models of physical systems undergoing damped oscillations, e.g., [1, 2]. In general $T(\lambda)$ can take a variety of different forms, depending on the physical model. Higher-degree polynomials are possible, as are non-polynomial functions of λ, e.g. [3, 4, 5, 6].

The usefulness of any physical model that produces the problem (1) is limited by the ease with which that problem can be solved. General nonlinear eigenvalue problems carry with them some unique challenges, e.g. their eigenvectors, in general, do not form a basis for $\mathbb{C}^n$, and the particular form of $T(\lambda)$ can have an impact on which solution methods are most effective. However, one of the core challenges for nonlinear eigenvalue problems is the same as for linear eigenvalue problems: solution methods that are effective for small values of n do not necessarily scale well to very large-dimensional problems, where one is usually interested in finding a small number of specific, physically-important eigenvalue-eigenvector pairs.

Most methods for solving such large-dimensional problems require generating an initial guess that is close enough to the desired eigenpairs to ensure convergence, as well as a method for calculating successive eigenpairs whose eigenvalues are close to each other, but without converging repeatedly to the same eigenpair [7]. These challenges are exacerbated in the situation that large numbers of eigenpairs are desired. The difficulty of calculating large number of eigenpairs is compounded further for NLEVP algorithms that work by approximating the desired eigenvectors in some subspace of dimension $m \ll n$; in this case, calculating large numbers of eigenpairs requires a large value of m, leading to bad scaling for the solution of the reduced-dimension eigenvalue problems that are solved inside that subspace. For linear eigenvalue problems the algorithmic complexity of solving the problem of dimension m is $O(m^3)$, and solving nonlinear eigenvalue problems always requires an amount of work that is at least equal to solving a single linear eigenvalue problem; keeping the dimension m small is thus of fundamental importance to ensure good scaling to larger problem sizes.