A Large-Grained Parallel Algorithm for Nonlinear Eigenvalue Problems Using Complex Contour Integration

Takeshi Amako, Yusaku Yamamoto, and Shao-Liang Zhang

Nagoya University, Nagoya, Aichi, 464-8603, Japan,
yamamoto@na.cse.nagoya-u.ac.jp,
WWW home page: http://www.na.cse.nagoya-u.ac.jp/~yamamoto

Abstract. We propose an algorithm for finding the roots of the nonlinear eigenvalue problem $A(\lambda)\mathbf{x} = \mathbf{0}$ that lie within a specified region of the complex plane. To this end, we construct a complex function that has simple poles at these roots and is analytical elsewhere. Then, by computing the complex moments through contour integration along the boundary of the specified region, we can locate the poles. Our algorithm has the advantage that it can find all the eigenvalues within the region. Moreover, the computationally dominant part, the evaluation of the contour integral by numerical integration, has large-grain parallelism since function evaluation at each sample point can be done independently. Numerical experiments on the Fujitsu HPC2500 supercomputer shows that our method can actually find the roots with high accuracy and can achieve nearly linear speedup using up to 16 processors.

1 Introduction

Given an $n \times n$ matrix $A(\lambda)$ whose elements are analytical functions of a complex parameter λ, we consider the problem of finding the values of λ for which a nonzero vector $\mathbf{x}$ that satisfies $A(\lambda)\mathbf{x} = \mathbf{0}$ exists. This is known as the nonlinear eigenvalue problem and the λ's are called eigenvalues. The nonlinear eigenvalue problem has applications in various fields such as nonlinear elasticity, electronic structure calculation and theoretical fluid dynamics. In this paper, we focus on finding all the eigenvalues that lie in a specified region on the complex plane.

Conventional approaches for the nonlinear eigenvalue problem include multivariate Newton's method and its modifications [1], Arnoldi method [2] and Jacobi-Davidson method [3]. However, Newton's method requires a good initial estimate for stable convergence. Arnoldi and Jacobi-Davidson methods are efficient for large sparse matrices, but in general, they cannot guarantee that all the eigenvalues in a specified region are obtained.

In our approach, we construct a complex function that has simple poles at the roots of the nonlinear eigenvalue problem and is analytical elsewhere. Then, by computing the complex moments through contour integration along the boundary of the specified region, we can locate the poles [4]. This method

has the advantage that it can find all the eigenvalues in the region. Moreover, the computationally dominant part, the evaluation of the contour integral by numerical integration, has large-grain parallelism since function evaluation at each sample point can be done independently. Thus our algorithm is expected to achieve excellent speedup in virtually any parallel environments.

We implemented our method on the Fujitsu HPC2500 supercomputer. Numerical experiments using a model problem show that our method can actually find the eigenvalues with high accuracy. Also, our method achieved nearly linear speedup using up to 16 processors for matrices of order 500 to 2000.

2 The algorithm

Assume that $A(z)$ is an $n \times n$ matrix whose elements are analytical functions of a complex parameter z and that we are interested in computing the (nonlinear) eigenvalues within a closed curve Γ on the complex plane. In the following, we restrict ourselves to the case where Γ is a circle. Now, let $f(z) = \det(A(z))$. Then $f(z)$ is an analytical function of z and the eigenvalues are characterized as the zeroes of $f(z)$.

To solve $f(z) = 0$, we use the method by Sakurai et al. based on complex contour integral [4]. Assume that there are m distinct roots of $f(z)$ within Γ and define the complex moments by

$$\mu_p = \frac{1}{2\pi i} \oint_\Gamma z^p \frac{f'(z)}{f(z)} \quad (p = 0, 1, \ldots, 2m-1). \tag{1}$$

Then μ_p can be rewritten as

$$\mu_p = \sum_{k=1}^{m} \lambda_k^p \nu_k, \tag{2}$$

where $\lambda_1, \lambda_2, \ldots, \lambda_m$ are the distinct roots of $f(z) = 0$ within Γ and $\nu_1, \nu_2, \ldots, \nu_m$ are their multiplicities [4]. To extract the information on $\{\lambda_j\}$ from $\{\mu_p\}$, we define the following matrices:

$$H_m = \begin{pmatrix} \mu_0 & \mu_1 & \cdots & \mu_{m-1} \\ \mu_1 & \mu_2 & \cdots & \mu_m \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{m-1} & \mu_m & \cdots & \mu_{2m-2} \end{pmatrix}, \quad H_m^< = \begin{pmatrix} \mu_1 & \mu_2 & \cdots & \mu_m \\ \mu_2 & \mu_3 & \cdots & \mu_{m+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_m & \mu_{m+1} & \cdots & \mu_{2m-1} \end{pmatrix}, \tag{3}$$

$$V_m = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \lambda_1 & \lambda_2 & \cdots & \lambda_m \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_1^{m-1} & \lambda_2^{m-1} & \cdots & \lambda_m^{m-1} \end{pmatrix}, \tag{4}$$

$$D_m = \mathrm{diag}(\nu_1, \nu_2, \cdots, \nu_m), \quad \Lambda_m = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_m). \tag{5}$$

Then it is easy to see that $H_m = V_m D_m V_m^T$ and $H_m^< = V_m D_m \Lambda_m V_m^T$. Noting that D_m and V_m are nonsingular, we can conclude that $\lambda = \lambda_j$ for some $1 \le j \le m$ if

and only if λ is an eigenvalue of a matrix pencil $H_m^< - \lambda H_m$. Hence, by computing the complex moments by eq. (1), forming the two Hankel matrices H_m and $H_m^<$ by eq. (3) and computing the eigenvalues of $H_m^< - \lambda H_m$, we can find the roots of $f(z)$ within Γ.

In our problem, $f(z) = \det(A(z))$ and it can be shown that

$$\frac{f'(z)}{f(z)} = \mathrm{Tr}\left[(A(z))^{-1} \frac{dA(z)}{dz} \right]. \tag{6}$$

Note that Sakurai and Sugiura propose a complex contour integral method for the linear eigenvalue problem $A\mathbf{x} = \lambda\mathbf{x}$ [5]. They use $\mathbf{u}(A - zI)^{-1}\mathbf{v}$, where $\mathbf{u}$ and $\mathbf{v}$ are random vectors, instead of $f'(z)/f(z)$. While this choice is computationally more efficient, we didn't adopt this since the Kronecker canonical form, which is the basis of their approach, is not directly applicable in the nonlinear case.

In the numerical procedure, we use the trapezoidal rule with K points to compute the complex moments $\{\mu_p\}$. Also, since we do not know m a priori, we replace it with some estimate M, which hopefully satisfies $M \geq m$. In that case, the eigenvalues of $H_M^< - \lambda H_M$ contain spurious solutions of the nonlinear eigenvalue problem. To get rid of them, we compute the eigenvector $\mathbf{x}_j$ corresponding to the computed λ_j by inverse iteration, evaluate the relative residual $\| A(\lambda_j)\mathbf{x}_j \|_\infty \,/\, (\| A(\lambda_j) \|_\infty \| \mathbf{x}_j \|_\infty)$, and discard λ_j if the residual is large or λ_j is outside Γ. This works well as we will see in the next section.

The computationally dominant part of our algorithm is the computation of $\mathrm{Tr}\left[(A(z))^{-1} \frac{dA(z)}{dz} \right]$ for the K integration points. Since the computation at each point can be done completely independently, this algorithm has large-grained parallelism. Hence, it should be able to achieve excellent speedup in virtually any parallel environments.

3 Numerical results

We implemented our algorithm using C and MPI and evaluated its performance on the Fujitsu HPC2500 supercomputer. In parallelizing the algorithm, we distributed the K integration points evenly among P processors and performed the computation of $\mathrm{Tr}\left[(A(z))^{-1} \frac{dA(z)}{dz} \right]$ at these points in parallel. Other parts of the algorithm, such as the computation of the eigenvalues of $H_M^< - \lambda H_M$, were executed on one processor. We used vendor-supplied optimized LAPACK routines to compute $(A(z))^{-1} \frac{dA(z)}{dz}$ and to find the eigenvalues of $H_M^< - \lambda H_M$. The test matrices used are of the form $A(z) = A - zI + \epsilon B(z)$, where A is a real nonsymmetric matrix whose elements follow uniform random numbers in $[0, 1]$, $B(z)$ is an anti-diagonal matrix with anti-diagonal elements e^z, and ϵ is a parameter that determines the strength of nonlinearity. In our experiments, we set $K = 128$ and $M = 10$ and varied n from 500 to 2000, P from 1 to 16, and ϵ from 0 to 0.1. The center of Γ is origin in all cases and the radius of Γ is 0.7, 0.7 and 0.5 for the $n = 500$, 100 and 2000 cases, respectively.

We show the results for the $n = 1000$ and $\epsilon = 0.01$ case in Fig. 1. In this case, it is known that the number of (nonlinear) eigenvalues in Γ is 9. Our algorithm computed 10 candidates from the eigenvalues of $H_M^< - \lambda H_M$, discarded one of them (the open diamond) as the spurious solution since the residual was around 10^{-2}, and output 9 solutions (the solid diamonds). The residuals for these nine solutions were all under 10^{-11}. Hence, our algorithm succeeded in locating all the eigenvalues in Γ with high accuracy. The situation was similar in other cases and our algorithm always found the correct number of eigenvalues in Γ with high accuracy. We also show the execution time as a function of n and P in Fig. 2. It is clear that almost linear speedup is achieved for all the cases.

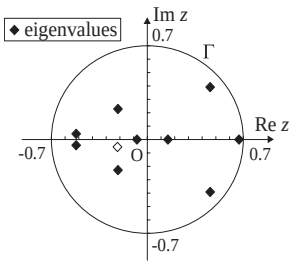

Fig. 1. Computed eigenvalues for the $n = 1000$ and $\epsilon = 0.01$ case.

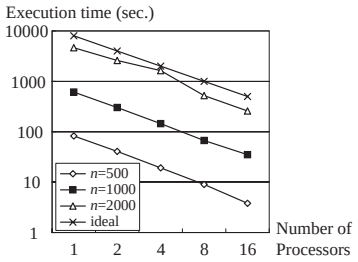

Fig. 2. Execution time of our algorithm on Fujitsu HPC2500.

4 Conclusion

We proposed a new algorithm for the nonlinear eigenvalue problem $A(\lambda)\mathbf{x} = \mathbf{0}$. Our algorithm has the ability to find all eigenvalues within a specified region on the complex plane. Also, it is expected to achieve excellent parallel speedup thanks to the large-grained parallelism. These advantages have been confirmed by numerical experiments. More numerical results will be given at the conference.

References

1. Ruhe, A.: Algorithms for the nonlinear eigenvalue problem. SIAM J. Numer. Anal. **10** (1973) 674–689
2. Betcke, T., Voss, H.: A jacobi-davidson-type projection method for nonlinear eigenvalue problems. Future Generation Comput. Syst. **20** (2004) 363–372
3. Voss, H.: An arnoldi method for nonlinear eigenvalue problems. BIT Numerical Mathematics **44** (2004) 387–401
4. Sakurai, T., Kravanja, P., Sugiura, H., Barel, M.V.: An error analysis of two related quadrature methods for computing zeros of analytic functions. J. Comput. Appl. Math. **152** (2003) 467–480
5. Sakurai, T., Sugiura, H.: A projection method for generalized eigenvalue problems. J. Comput. Appl. Math. **159** (2003) 119–128