

# ON MODIFICATIONS TO LAGUERRE’S METHOD AND THE POLYNOMIAL EIGENVALUE PROBLEM

Thomas R. Cameron\* & Nikolas I. Steckley\*\*

Davidson College\* & DiscoverOrg LLC.\*\*

## Abstract

Laguerre’s method has long been recognized for its strong virtues when computing the roots of a polynomial. Over the years, many modifications to Laguerre’s method have been suggested in an attempt to improve its convergence rate and to avoid multiple convergence to a simple root. Now, we present a modification to Laguerre’s method for the simultaneous convergence of all roots of a polynomial. Multiple numerical experiments verify both the accuracy and efficiency of this algorithm, and we provide comparisons to other root finding methods, such as POLZEROS and AMVW. Furthermore, we are able to apply this method to the polynomial eigenvalue problem, where it is effective for large degree problems, and the Tridiagonal problem.

## Introduction

Let  $p(\lambda)$  be a polynomial of degree  $m$ . Denote by  $(z_1, \dots, z_m)$  current approximations to the roots,  $r_1, \dots, r_m$  of  $p(\lambda)$ . Then, for  $j \in \{1, \dots, m\}$  define

$$f_j(\lambda) = \frac{p(\lambda)}{\prod_{\substack{i=1 \\ i \neq j}}^m (\lambda - z_i)},$$

and note that  $f_j(\lambda)$  has the same roots as  $p(\lambda)$ . Further define

$$G_j(\lambda) = \frac{f'_j(\lambda)}{f_j(\lambda)} = \frac{p'(\lambda)}{p(\lambda)} - \sum_{\substack{i=1 \\ i \neq j}}^m \frac{1}{\lambda - z_i},$$

$$H_j(\lambda) = - \left( \frac{f'_j(\lambda)}{f_j(\lambda)} \right)' = - \left( \frac{p'(\lambda)}{p(\lambda)} \right)' - \sum_{\substack{i=1 \\ i \neq j}}^m \frac{1}{(\lambda - z_i)^2}.$$

Then the  $j$ th root approximation is updated via the expression

$$\hat{z}_j = z_j - L_m(z_j),$$

where the modified Laguerre updated is defined by

$$L_m(z_j) = \frac{m}{G_j(z_j) \pm \sqrt{(m-1)(mH_j(z_j) - G_j^2(z_j))}},$$

and the sign is chosen to maximize the magnitude of the denominator.

The removal of previously computed roots is a well known modification of Laguerre’s method and has been employed in [3, 4]. Here, we create poles at  $z_1, \dots, z_{j-1}, z_{j+1}, \dots, z_m$  in order to direct the Laguerre iteration away from other root approximations. A similar idea was used by Aberth to modify Newton’s method [1].

## Iteration Style and Order of Convergence

For  $j = 1, \dots, m$ , the  $j$ th root approximation is updated via (2), provided that it is not already close enough to the root  $r_j$ . Locally, if the root is simple, convergence is *quartic*; otherwise it is linear. The iteration can be implemented in either a Jacobi style or a Gauss-Seidel style.

**Jacobi Iteration.** The modified Laguerre update (3) is computed for all  $j$  first, and the root approximations are updated afterwards. This iteration style can be parallelized.

**Gauss-Seidel Iteration.** The  $j$ th root approximation is updated as soon as possible. Therefore, the computation of (1) always makes use of the most recent approximations. For this iteration style, convergence is slightly faster.

Our current algorithm (DSLM) uses a Gauss-Seidel iteration style. For future research, we are interested in writing a parallelized version of our algorithm that would employ a Jacobi iteration style.

## Initial Estimates

Let  $p(\lambda) = \sum_{i=0}^m a_i \lambda^i$ , where  $a_0 a_m \neq 0$ . The *Newton polygon* associated with this polynomial is the upper convex hull of the discrete set  $\{(i, \log |a_i|) : i = 0, 1, \dots, m\}$ . Let  $0 = k_0 < k_1 < \dots < k_q = m$  denote the abscissas of the vertices of the Newton polygon, and define the radii

$$\mu_i = \left| \frac{a_{k_{i-1}}}{a_{k_i}} \right|^{\frac{1}{k_i - k_{i-1}}}, \quad (i = 1, \dots, q).$$

Then,  $(k_i - k_{i-1})$  evenly distributed points are placed on circles centered at 0 with radius  $\mu_i$ . These points constitute our initial estimates to the roots of  $p(\lambda)$ . Furthermore, these points are guaranteed to lie within the Pellet bounds for the polynomial  $p(\lambda)$ , and the cost complexity is  $O(m \log m)$  [2].

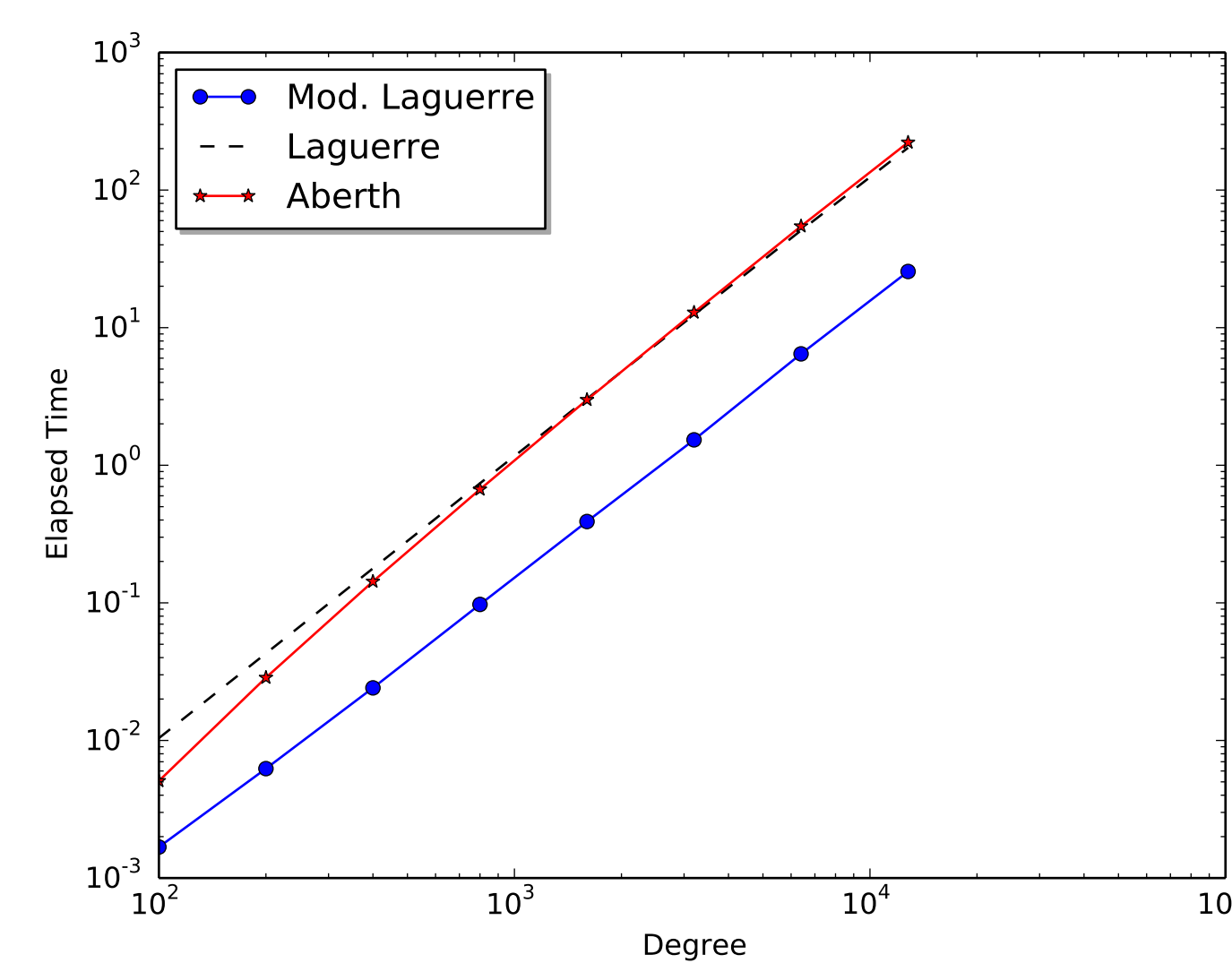
## Stopping Criteria

It follows from [5][Lemma 3] that the backward error in the root approximation  $z_j$  is bounded above by

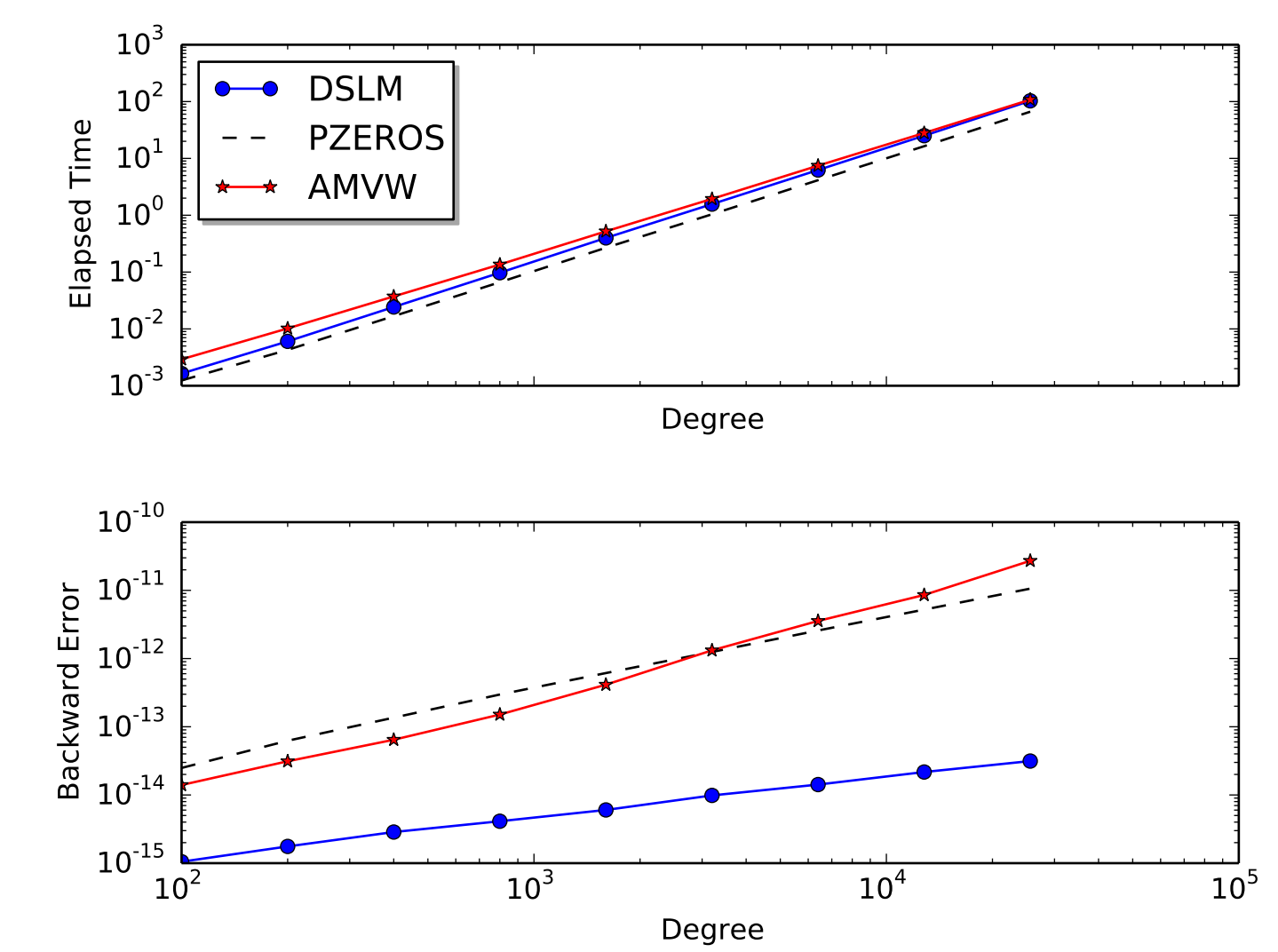
$$b(z_j) = \frac{|p(z_j)|}{\sum_{i=0}^m |a_i| |z_j|^i}.$$

If  $b(z_j) < \epsilon$ , where  $\epsilon$  denotes double precision unit roundoff, then we say that the approximation  $z_j$  has converged. This constitutes our first stopping criterion. Otherwise, we compute the modified Laguerre update (3). If  $|L_m(z_j)| < \epsilon |z_j|$ , then no relatively significant contribution will be made by  $L_m(z_j)$  and we say that the approximation  $z_j$  has converges. This constitutes our second stopping criterion. Otherwise, we update  $z_j$  via (2).

## Comparison to Aberth and Laguerre



## Comparison to POLZEROS and AMVW



	DSLM	POLZEROS	AMVW
Test 1	2.49E-11	1.05E-09	9.27E-11
Test 2	1.05E-07	1.57E-05	1.19E-05
Test 3	2.85E-03	3.49E-02	7.40E-02
Test 4	4.89E-13	5.52E-13	1.73E-11
Test 5	7.93E-12	9.65E-11	4.72E-07
Test 6	5.00E-08	2.41E-07	5.14E-02
Test 7	2.76E-04	6.26E-03	3.47E-01
Test 8	9.77E-04	9.77E-04	9.77E-04
Test 9	1.12E-03	1.30E-03	1.30E-03
Test 10	2.19E-12	1.46E-10	6.52E-11

## Conclusion

Future research includes applying this modified Laguerre method to the polynomial eigenvalue problem.

## References

- [1] O. ABERTH, *Iteration methods for finding all zeros of a polynomial simultaneously*, Mathematics of Computation, 27 (1973), pp. 339–344.
- [2] D. BINI, *Numerical computation of polynomial zeros by means of aberth’s method*, Numer Algor, 13 (1996), pp. 179–200.
- [3] P. LANCASTER, *Lambda-Matrices and Vibrating Systems*, vol. 94 of International Series of Monographs on Pure and Applied Mathematics, Pergamon, 1966.
- [4] B. PARLETT, *Laguerre’s method applied to the matrix eigenvalue problem*, Mathematics of Computation, 18 (1964), pp. 464–485.
- [5] F. TISSEUR, *Backward error and condition of polynomial eigenvalue problem*, Linear Algebra Appl., 309 (2000), pp. 339–361.