THE AAA ALGORITHM FOR RATIONAL APPROXIMATION

YUJI NAKATSUKASA , OLIVIER SÈTE , AND LLOYD N. TREFETHEN

1 Introduction and Problem Statement

Rational approximations of real or complex numbers generally come in two types. Traditional polynomial approximations produce false poles (also known as Frosat duality) when dealing with functions that have poles or singularities, as these poles nearly overlap with zero, leading to numerical instability and precision loss. And this is the problem that the authorities wanted to solve.

According to this paper, the Adaptive Antoulas – Anderson (referred to as AAA)algorithm is a numerically stable and adaptive method for rational approximation. The AAA algorithm is different from the other algorithms; it automatically selects support points and constructs a barycentric rational representation, ensuring robustness and high accuracy. It achieves excellent speed and stability even for functions with poles or singularities—situations where polynomial approximation typically fails.
It combines two ideas: first, using barycentric forms to represent and interpolate pivot points from selected sets; second, it can greedily choose pivot points to avoid instability, thereby gradually increasing the approximation degree. This algorithm also possesses an important feature: it can discretize data without relying on a specific domain, combining adaptive processing.

2 Method Explanation

Before delving into the core content of the AAA algorithm, we need to introduce rational barycentric representations first.

Consider $z_j := \{z_1, \cdots z_m\}$ which is the set of the support point, $f_j := \{f(z_j), j = 1 \cdots m\}$, and $w_j := \{w_1, \cdots w_m\}$ is the set of scalar weights.
Then the barycentric formula is defined as the following :

$$r(z) = \frac{n(z)}{d(z)} = \sum_{j=1}^{m} \frac{w_j f_j}{z - z_j} \Big/ \sum_{j=1}^{m} \frac{w_j}{z - z_j}, \tag{1}$$

where $m \geq 1$ is an integer.

But we need to be careful about the following of properties：

1. Interpolation：for each support point z_k, the formula gives $r(z_k) = f(z_k)$ provided the denominator is nonzero.

2. Degree：r is a rational function of type $(m-1, m-1)$

3. Numerical conditioning：barycentric form avoids constructing high-degree polynomials explicitly and is far better conditioned numerically than working with $p(z)/q(z)$ coefficient vectors.

Now, we suppose there exist a node polynomial ℓ, then

$$\ell(z) = \Pi_{j=1}^{m}(z - z_j). \tag{2}$$

Define $p(z) = \ell(z)n(z), q(z) = \ell(z)d(z)$, then the equation (1) can be rewritten as

$$r(z) = \frac{p(z)/\ell(z)}{d(z)/\ell(z)} = \frac{p(z)}{q(z)}. \tag{3}$$

Note the function f has $(2m-1)$ degrees offreedom and the weights $\{w_1, \cdots w_m\}$ have to satisfy

$$\text{minimize}||fd - n||_z^{(m)}, ||w||_m = 1.$$

Our goal is to get $f(z) \approx \frac{n(z)}{d(z)}$.
After constructing the center of gravity and initial conditions, we need to start selecting the support points.

step1：(In m=1)
Let $r_{m-1}(z)$ denote the current approximant. Evaluate the current residual on all sample points

$$e_{m-1}(z_i) = f(z_i) - r_{m-1}(z_i), \text{where } i = 1, \cdots, M. \tag{4}$$

The next support point z_m is choose a point $Z \in Z^{m-1}$ with the largest magnitude residual

$$\max\Big|f(z) - n(z)/d(z)\Big|,$$

which implies

$$z_m = argmax\Big|e_{m-1}(z)\Big|,$$

where $z \in Z\backslash\{z_1, \cdots, z_{m-1}\}$.
Append z_m to the support set and $f(z_m)$ to the list of support values.
This greedy choice concentrates approximation power where the current error is largest and adapts automatically to the function's features.

Assume therer is a finite set $Z \subseteq C$ of $M \gg 1$.
Let $Z^{(m)} = (Z_1^{(m)}, \cdots, Z_{M-m}^{(m)})^T$, $F^{(m)} = (F_1^{(m)}, \cdots, F_{M-m}^{(m)})^T$, and $w = (w_1, \cdots, w_m)^T$ where $||w||_m = 1$.
By solving a homogeneous linear least-squares problem derived from enforcing approximate interpolation conditions on the non-support sample points. Then define the Cauchy matrix,

$$C_{i,j} = \frac{1}{z_i - z_j}. \tag{5}$$

and let F be the vector of sampled function values and f_s the vector of support function values. Let $S_F = diag(F)$. For rows corresponding to sample indices that are not support points, consider the Loewner-like matrix

$$A = S_F C - C \text{ diag}(f_s),$$

restricted to the rows of non-support points. The key observation is that if the barycentric form (1) were exact on all sample points, then for every non-support z_i we would have

$$\sum_{j=1}^{m} \frac{w_j f(z_j)}{z_i - z_j} - f(z_i) \sum_{j=1}^{m} \frac{w_j}{z_i - z_j} = 0, \tag{6}$$

which can be written as a homogeneous linear system $Aw = 0$.
For get the unknown weights w, we need to solve the SVD of the matrix A. This is imples

$$\text{minimize} ||A^{(m)} w||_{M-m}, ||w||_m = 1.$$

After finding w, then reassignment the values which are newly defined

$$N_i = \sum_{j=1}^{m} \frac{w_j f(z_j)}{z_i - z_j}, D_i = \sum_{j=1}^{m} \frac{w_j}{z_i - z_j} \tag{7}$$

and launch a new round of calculations, until the error $< 10^{-13}$, the algorithm stops.

3 Experiment

4 Conclusion

Strengths and when AAA is effective

- Adaptivity: AAA automatically places support points where they are most needed (largest residual), which is effective for functions with localized features such as boundary layers, peaks, or nearby singularities.
- Numerical stability: representing rational functions in barycentric form and computing weights via SVD avoids ill-conditioned polynomial bases and yields reliable double-precision results in many cases.
- Generality: the algorithm works for arbitrary sample sets Z in the real or complex plane, not only intervals or circles.
- High accuracy: for smooth or analytic functions, AAA frequently attains machine precision on the sample set with modest m.
- Simplicity of use: AAA requires minimal user tuning (default tolerances usually work) and outputs rational approximants directly usable for evaluation.

Limitations and failure modes

- Cost for extremely large samples: if M is extremely large (e.g. millions of sample points) or if the required m is large, repeated SVDs can become computationally expensive. Randomized linear algebra or subsampling can mitigate this.
- Noisy data: AAA is designed for deterministic function samples. With noisy data, the greedy strategy may overfit; regularization or weighted least-squares variants are needed to obtain robust fits.
- Froissart doublets: while relatively rare, spurious pole–zero pairs can occur, especially if a very tight tolerance is enforced. A robust cleanup procedure is required to detect and remove them.
- Non-analytic or discontinuous functions: rational approximation targets analytic (or at least sufficiently smooth) structure; for highly discontinuous functions, rational fits will require many support points and lose efficiency.