J. Approx. Softw., Vol. ?, No. ? (2024) Z

Journal of Approximation Software

CPOLYMESH: Matlab and Python codes for complex polynomial approximation by Chebyshev admissible meshes

Dimitri J. Kenne[1], Alvise Sommariva[2], and Marco Vianello[3]

[1] *Doctoral School of Exact and Natural Sciences, Jagiellonian University, Krakow, Poland*
[2,3] *Department of Mathematics, University of Padova, Italy*

Received: DD/MM/YYYY – Published: DD/MM/YYYY
Communicated by: ???

Abstract

We provide Matlab and Python codes for polynomial approximation on complex compact sets with connected complement, by Chebyshev-like admissible polynomial meshes on boundaries with piecewise (trigonometric) polynomial parametrization. Such meshes have lower cardinality with respect to those previously known. They are used for polynomial least-squares, for the extraction of extremal interpolation sets of Fekete and Leja type, as well as for the computation of the uniform norms (Lebesgue constants) of polynomial projection operators.

Keywords: complex polynomial approximation, interpolation, least-squares, admissible polynomial meshes, discrete extremal sets, Approximate Fekete Points, Discrete Leja Points, Pseudo Leja Points, Lebesgue constant. (MSC2020: 65D05,65E05)

1 Introduction

In this paper we are concerned with (admissible) *polynomial meshes* $\{Z_n\}_{n\geq 1}$ and polynomial approximation on a complex compact set $K \subset \mathbb{C}$ with connected complement. By the famous Mergelyan Theorem [15], these are sets where any continuous function $f : K \to \mathbb{C}$, with holomorphic restriction to $int(K)$, can be uniformly approximated by polynomials.

Polynomial meshes are sequences of finite subsets $Z_n \subset K$ such that

$$\|p\|_K \leq c\|p\|_{Z_n}\ ,\ \ \forall p \in \mathbb{P}_n(\mathbb{C})\ , \tag{1}$$

where $\|\cdot\|$ is the uniform norm on a continuous or discrete bounded subset, and p is any polynomial with complex coefficients with degree not exceeding n (we recall that c is usually termed

* Corresponding author: marcov@math.unipd.it

the "constant" of the polynomial mesh). Since Z_n is $\mathbb{P}_n(\mathbb{C})$-determining, i.e. polynomials in $\mathbb{P}_n(\mathbb{C})$ vanishing on Z_n vanish everywhere in $\mathbb{C}$, clearly $card(Z_n) \geq n+1 = dim(\mathbb{P}_n(\mathbb{C}))$. The polynomial mesh is then called *optimal* when $card(Z_n) = O(n)$.

Starting from the seminal paper of 2008 by Calvi and Levenberg [9], polynomial meshes (that can be defined also in multivariate instances) have begun to play a relevant role in polynomial approximation. Among their numerous properties, we may recall that they are preserved by affine transformations, finite union and small perturbations, are well-suited for least-squares approximation and contain extremal subsets of Fekete and Leja type for polynomial interpolation with slowly increasing Lebesgue constant. Moreover, polynomial meshes can be conveniently used for polynomial optimization and Lebesgue constant computation with rigorous interval error bounds. Without any pretence of exhaustivity, we may quote e.g. [2, 3, 4, 5, 6, 16, 17] with the references therein.

Focalizing on the complex univariate case, it has been recently proved in [3] that optimal admissible meshes of Chebyshev type can be constructed on the boundary ∂K, provided that it lies on a union of curves in K having a piecewise polynomial or trigonometric polynomial parametrization. The construction uses the fact that $\|p\|_K = \|p\|_{\partial K}$ by the maximum principle for holomorphic functions (cf. e.g. [14]), and basic polynomial inequalities concerning Chebyshev points on the parameter real interval.

More precisely, let C_N be the set of N Chebyshev zeros in $(-1,1)$, namely $\cos((2j-1)\pi/(2N))$, $1 \leq j \leq N$, or the set of $N+1$ Chebyshev extrema in $[-1,1]$, namely $\cos(j\pi/N)$, $0 \leq j \leq N$. Consider the points

$$\mathscr{C}_\nu^m = \tau(C_N) \subset [a,b] \tag{2}$$

where

$$N = m\nu \ , \ \ \tau(u) = \frac{b-a}{2}u + \frac{b+a}{2} \ , \ \ u \in [-1,1] \ , \tag{3}$$

in the algebraic case, and

$$N = 2m\nu \ , \ \ \tau(u) = 2\arcsin\left(u \sin\left(\frac{b-a}{4}\right)\right) + \frac{b+a}{2} \ , \ \ u \in [-1,1] \ , \tag{4}$$

in the trigonometric case. Then, the following estimate holds [3]:

Proposition 1. Let $\partial K \subseteq \Gamma = \bigcup_{j=1}^{s} \Gamma_j \subseteq K$ with parametric algebraic or trigonometric arcs $\Gamma_j = \gamma_j([a_j,b_j])$ of degree $d_j = \max\{deg\,Re(\gamma_j), deg\,Im(\gamma_j)\}$, $1 \leq j \leq s$ (where the angle intervals possibly are sub-intervals of the period in the trigonometric case, namely $b_j - a_j \leq 2\pi$). Then for every $p \in \mathbb{P}_n(\mathbb{C})$, $n \geq 1$, $m > 1$

$$\|p\|_K = \|p\|_\Gamma \leq c_m \|p\|_{Z_n^m} \ , \ \ Z_n^m = \bigcup_{j=1}^{s} \gamma_j(\mathscr{C}_{nd_j}^m) \ , \ \ c_m = \frac{1}{\cos(\pi/(2m))} \ , \tag{5}$$

i.e. $\{Z_n^m\}_{n \geq 1}$ is an admissible polynomial mesh for K with constant c_m.

Estimate (5) is a cornerstone of our code for complex polynomial approximation. Notice that the class of domains with connected complement and such boundaries is very wide: it includes linear polygons, as well as curvilinear polygons with boundary tracked by splines, or by polar arcs like $\gamma_j(t) = z_0 + r_j(t)(\cos(t) + i\sin(t))$ with $r_j(t)$ a trigonometric polynomial. See the Figures below for some illustrative examples. The corresponding meshes have $O(mn)$ cardinality, that asymptotically improves the $O(n^2)$ cardinality of previously known meshes on any connected compact set of $\mathbb{C}$ whose boundary is a C^1 parametric curve with bounded tangent vectors, cf. [1].

Observe that $c_m \to 1$ as $m \to \infty$. This fact is at the base of a computable interval estimate of the Lebesgue constant (uniform operator norm) of any linear projection operator $L_n : C(K) \to \mathbb{P}_n(\mathbb{C})$ of the form

$$L_n f(z) = \sum_{j=1}^{M} f(\xi_j)\, \phi_j(z) , \tag{6}$$

where $\{\xi_j\} \subset K$ and $\{\phi_j\}$ is a set of generators of $\mathbb{P}_n(\mathbb{C})$. We recall that such an operator structure holds for polynomial interpolation at $M = n+1$ distinct nodes, where the $\phi_j(z)$ are the corresponding cardinal Lagrange polynomials, but also by polynomial least-squares at $M > n+1$ sampling nodes (cf. [3]). In both cases we simply have

$$\phi_j(z) = K_n(z, \xi_j) = \sum_{k=1}^{n+1} q_k(z)\overline{q_k(\xi_j)} , \tag{7}$$

where K_n is the reproducing kernel of the discrete scalar product with unit weights supported at the sampling nodes $\{\xi_j\}$ and $\{q_k\}$ a discrete orthogonal polynomial basis. Given any polynomial basis of $\mathbb{P}_n(\mathbb{C})$, say $\{p_k\}$, we recall that a discrete orthogonal basis can be computed in principle by a QR factorization of the corresponding Vandermonde-like interpolation matrix, as

$$[q_1(z), \ldots, q_{n+1}(z)] = [p_1(z), \cdots, p_{n+1}(z)]R^{-1} . \tag{8}$$

The following result concerning Lebesgue constants has been proved in [3]:

Proposition 2. Let $\lambda_n(z) = \sum_{j=1}^{M} |\phi_j(z)|$, $z \in K$, be the "Lebesgue function" of L_n in (6) and $\{Z_n^m\}$ the polynomial mesh of Proposition 1. Then for every $n \geq 1$, $m > 1$, the following inequalities hold

$$\|\lambda_n\|_{Z_n^m} \leq \|L_n\| \leq c_m \|\lambda_n\|_{Z_n^m} , \tag{9}$$

$$0 \leq \|L_n\| - \|\lambda_n\|_{Z_n^m} \leq (c_m - 1)\|L_n\| , \tag{10}$$

for the Lebesgue constant $\|L_n\| = \|\lambda_n\|_K = \|\lambda_n\|_\Gamma$.

We observe that

$$c_m - 1 = \frac{1 - \cos(\pi/(2m))}{\cos(\pi/(2m))} \sim \frac{\pi^2}{8m^2} \approx \frac{1.23}{m^2} , \tag{11}$$

that is $\|\lambda_n\|_{Z_n^m}$ by (10) is a $O(1/m^2)$ relative approximation of the Lebesgue constant: for $m = 4$ we already get the Lebesgue constant with an error less than 10%, i.e. we can substantially evaluate its actual order of magnitude.

2 Description of the code

After the above summary of the main theoretical results and estimates underlying the complex polynomial approximation algorithms, we can now briefly describe the code implemented in Matlab and Python and available at https://github.com/alvisesommariva/CPOLYMESH. All the main computations are performed by basic numerical linear algebra subroutines. The main functions are:

- **Polynomial Mesh Constructor**

  Function `Cpom`

This function, given the parametrization intervals $[a_j, b_j]$ and the corresponding curve components, namely the complex algebraic or trigonometric polynomials $\gamma_j(t)$ of degree d_j, computes the Chebyshev submeshes $\gamma_j(\mathscr{C}^m_{nd_j})$ and their union

$$Z^m_n = \bigcup_{j=1}^{s} \gamma_j(\mathscr{C}^m_{nd_j})$$

as in Proposition 1. Linear and curvilinear polygons defined by spline arcs are included, as well as trigonometric polar arcs like $\gamma_j(t) = z_0 + r_j(t)(\cos(t) + i\sin(t))$, with $r_j(t)$ a real trigonometric polynomial on a subinterval of the period.

- **Stabilized Vandermonde Matrix Constructor**

  Function `Cvand`

  Constructs a rectangular Vandermonde-like matrix

  $$V_n(X) = (p_j(z_i))\ ,\ \ 1 \le i \le card(X)\ ,\ 1 \le j \le n+1\ ,$$

  on a complex set $X = \{z_i\}$. In order to cope with the extreme ill-conditioning of the Vandermonde matrices with the standard monomial basis, we have chosen to work with a shifted and scaled basis

  $$p_j(z) = ((z - z_b)/\delta)^{j-1}\ ,\ \ 1 \le j \le n+1\ ,$$

  where $z_b = \frac{1}{card(X)} \sum_{z_i \in X} z_i$ is the barycenter of the points and $\delta = \max_{z_i \in X} |z_i - z_b|$ the radius of an enclosing disk. If an enclosing disk is already known, its center z_b and radius δ can be directly passed as input parameters.

- **Discrete Orthogonal Polynomials Constructor and Evaluator**

  Functions `Cdop` and `Cdopeval`

  `Cdop` computes a discrete orthogonal polynomial basis on a finite complex set X with $card(X) \ge n+1$, and `Cdopeval` evaluates the orthogonal basis on a target complex set Y. Orthogonalization is performed by applying twice a QR factorization with unitary Q and square triangular factor R, namely

  $$V_n(X) = Q_1 R_1\ ,\ \ V_n(X)/R_1 = Q_2 R_2$$

  following the well-known “twice is enough” orthogonalization rule in finite precision arithmetic [10]. The target matrix is

  $$W_n(Y) = (V_n(Y)/R_1)/R_2$$

  where the matrix operator / is preferred to `inv` in order to automatically cope with possible ill-conditioning of the triangular factors.

- **Discrete Extremal Sets Constructor**

  Function `Cdes`

  Computes three interpolation pointsets corresponding to a greedy maximization of the Vandermonde determinant modulus on the polynomial mesh Z^m_n. We do not discuss their features in detail here and refer to the quoted literature for the underlying theoretical and computational issues.

- AFP (Approximate Fekete Points): after a call to `Cdop` with $X = Z_n^m$ to get a better conditioned matrix, AFP are obtained by a QR factorization with column pivoting of the adjoint Q_2^H, taking the points in Z_n^m corresponding to the first $n+1$ elements of the column permutation vector. They do not form a sequence, but typically have the lowest Lebesgue constant among the three sets; cf. [5, 6, 8, 19].

- DLP (Discrete Leja Points): again after a call to `Cdop` with $X = Z_n^m$, DLP are obtained by a LU factorization with row pivoting of Q_2, taking the points in Z_n^m corresponding to the first $n+1$ elements of the row permutation vector. They form a sequence and in the present univariate complex instance are substantially equivalent to the iteration

$$\xi_j = argmax_{z \in Z_n^m} \prod_{k=1}^{j-1} |z - \xi_k| \ , \quad j = 2, \ldots, n+1 \ ,$$

after choosing ξ_1 as the point that maximizes the element modulus in the first column of Q_2; cf. [6, 7].

- PLP (Pseudo Leja Points): these are a sequence obtained by the iteration

$$\xi_j = argmax_{z \in Z_{j-1}^m} \prod_{k=1}^{j-1} |z - \xi_k| \ , \quad j = 2, \ldots, n+1 \ ,$$

after choosing the first point ξ_1 arbitrarily, e.g. ξ_1 is one of the points in Z_1^m with largest imaginary component; cf. [1] (and [11] for a multivariate extension).

- **Polynomial Projectors** (either Interpolation or Least-Squares)

  Function `Cfit`

  Given a sample column array $\mathbf{f} = f(X)$ of a function at a finite complex set X with $card(X) \geq n+1$, computes the polynomial projector coefficients in an orthogonal polynomial basis at X and evaluates the projector $L_n f$ at a target complex set Y. In view of (7)-(8) the computation is simply

$$L_n f(Y) = W_n(Y) Q_2^H \mathbf{f}$$

  after a call to `Cdop` on X and `Cdopeval` on Y.

- **Lebesgue Constant Evaluator**

  Function `Cleb`

  Computes on a control set Z the maximum of the Lebesgue function of interpolation on a set X with $card(X) = n+1$ or least-squares with $card(X) > n+1$, as

$$\|\lambda_n\|_Z = \|W_n(Z) Q_2^H\|_\infty = \|((V_n(Z)/R_1)/R_2) Q_2^H\|_\infty$$

  by a call to `Cdop` on X and to `Cdopeval` with $Y = Z$. In view of Proposition 2, choosing $Z = Z_n^m$ produced by a call to `Cpom` one gets the certified interval estimate (9) for the Lebesgue constant of X.

3 Numerical tests and demos

In this section we present several numerical tests for the code CPOLYMESH. All the corresponding demos are available at [12, 13], along with a number of other examples.

The compact sets considered, say $K_1,\dots,K_9$, with boundary defined by algebraic or trigonometric polynomial arcs, appear in Figure 2. For the sake of clarity and brevity we do not plot Leja-like interpolation points, whose structure however is not much different from that of Fekete-like points. The figures below have been obtained by the Matlab package [12].

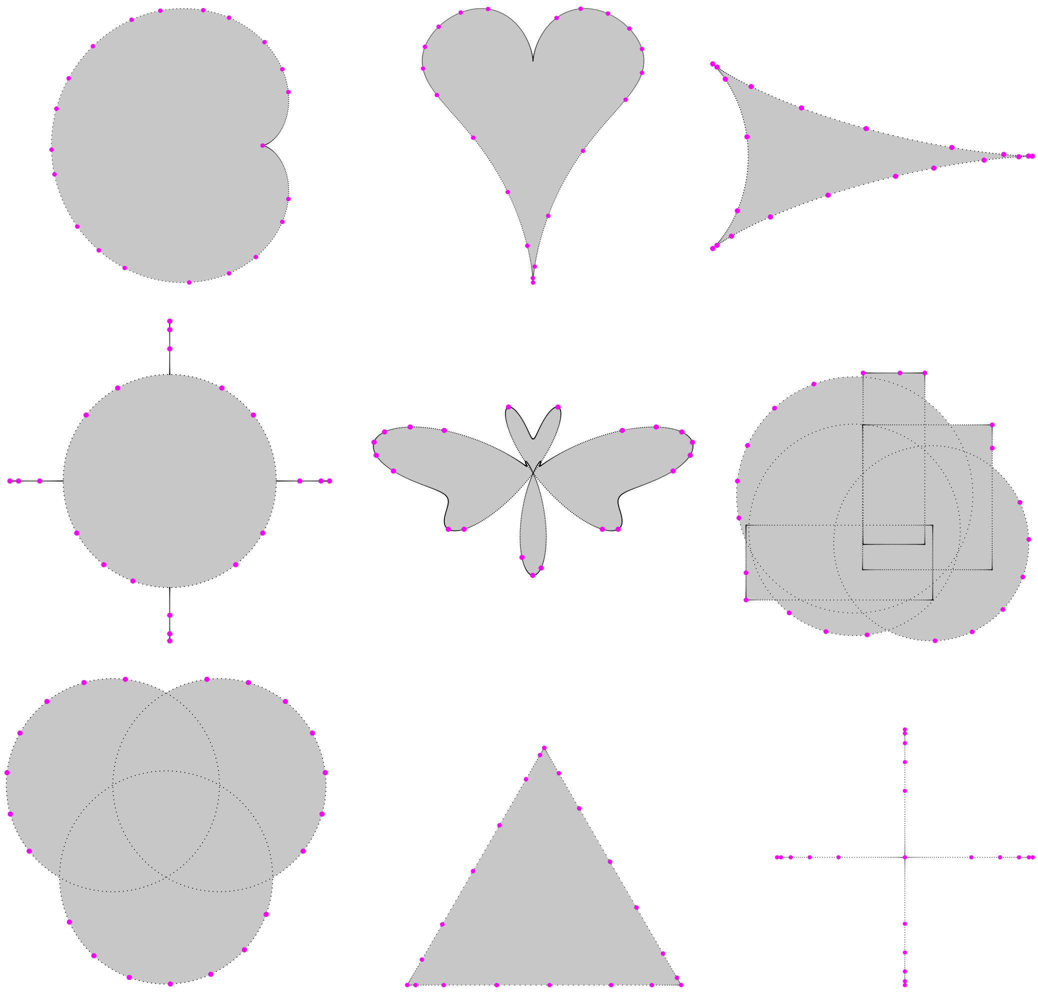

Figure 1: Exemplifying the variety of feasible (curvi)linear polygons: Approximate Fekete Points (magenta dots) for polynomial interpolation at degree $n = 20$ via Chebyshev admissible meshes (black dots) of the piecewise polynomial or trigonometric boundary, with $m = 2$ (cf. Proposition 1).

In particular, the compact sets are:

- a cardioid K_1 where ∂K_1 is defined parametrically as $z(t) = \cos(t)(1-\cos(t)+i(\sin(t)(1-\cos(t))))$, with $t \in [0,2\pi]$;
- the "Laporte heart" K_2 where ∂K_2 is defined parametrically as $z(t) = \sin^3(t)+i(\cos(t)-\cos^4(t))$, with $t \in [0,2\pi]$;
- the deltoid K_3 where ∂K_3 is defined parametrically as $z(t) = 10\exp(it)+5\exp(-2it)$, with $t \in [0,2\pi]$;

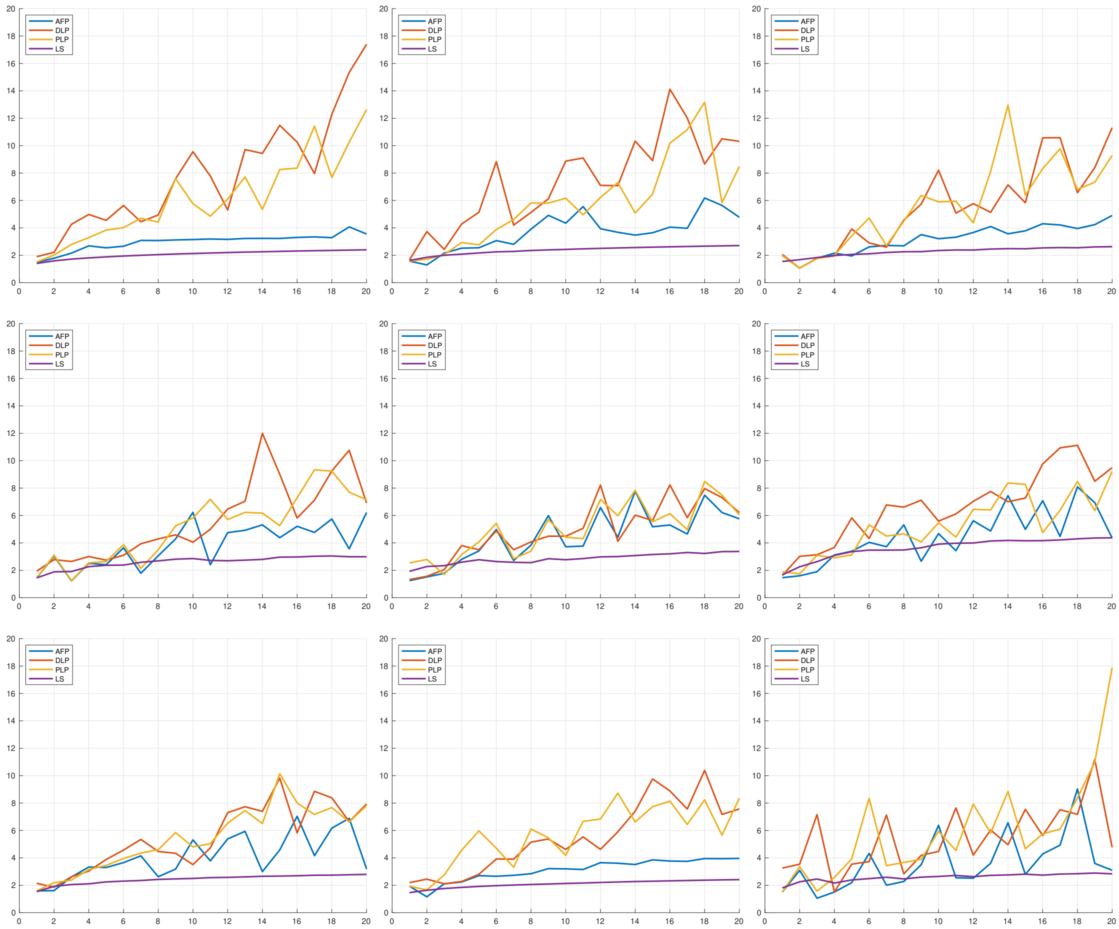

Figure 2: Lebesgue constants for the extremal interpolation sets AFP, DLP, PLP, and for LS approximation, with degrees $n = 1, 2, \ldots, 20$, on the domains of Fig. 1 in the same order.

- a sun shaped domain K_4 obtained as union of the unit-disk with 4 segments of length 0.5;
- the "Sautereau butterfly" K_5 where ∂K_5 is defined parametrically as $z(t) = (-3\cos(2t) + \sin(7t) - 1)\exp(it)$, with $t \in [0, 2\pi]$;
- the domain K_6 obtained as union of 3 random rectangles and 3 disks;
- the "Borromean-circles" domain K_7, that is the union of three disks with radii equal to $\sqrt{3}$ and centers $\exp(it_k)$ with $t_k = \frac{(4k-3)\pi}{6}$, $k = 1, 2, 3$;
- an equilateral triangle K_8 with vertices $P_k = \exp(it_k)$ with $t_k = \frac{(4k-1)\pi}{6}$, $k = 1, 2, 3$;
- a symmetric cross K_9 given as the union of 4 orthogonal unit segments with a common extremum.

We stress the variety of "(curvi)linear polygons" appearing in these examples: some domains are closure of open connected sets, other have components with no internal points. Some have a boundary corresponding to a single parametric curve, other have a complicated boundary being the union of simpler elements. In the latter case, there is no need to track accurately the boundary of the union (that in some situations could be a difficult task), since we can simply take a union admissible mesh for the union of the element boundaries. Notice also that the extremal interpolation points tend to concentrate on outward tips, cusps, angles or convex portions of the boundary and to avoid inward and concave ones (a well-known electrostatic charges-like behavior, connected with their potential theoretic background, cf. [18]).

In Figure 2 we report the Lebesgue constants of interpolation at discrete extremal sets of Fekete and Leja type, and of Least Squares approximation. We can observe that all the Lebesgue constants exhibit an apparently sub-exponential average increase in the present degree range, in line with the corresponding continuous extremal sets, cf. e.g. [9, 20]. However, Leja-like points have tendentially a more erratic behavior with larger oscillations and tendentially higher values with respect to approximate Fekete points (a phenomenon already observed for example in the multivariate framework of [7]). On the contrary, Lebesgue constants of Least Squares approximation on the whole polynomial mesh have the lowest values, with an apparently logarithmic-like behavior.

3.1 Demos summary

3.1.1 Matlab package

The Matlab package includes two demos, that we briefly comment.

1. `demo_cdes_1`: by this routine we show how to
   - define the complex domain (several ways),
   - compute an admissible mesh (AM) of a fixed degree,
   - extract extremal sets,
   - compute a certified Lebesgue constant.
2. `demo_cdes_2`: by this routine we perform all batteries of numerical tests that are described above.

   In particular, varying the degrees, we
   - compute an admissible mesh (AM) of a fixed degree,
   - extract the AFP, DLP, PLP extremal sets,
   - compute for each of them a certified Lebesgue constant,
   - plot domain, extremal points and Lebesgue contants.

   Changing the value of the variable `domain_type`, from 1 to 22, one can test our routines also on other complex geometries, like hypocycloids, epicycloids, epitrochoids, limacons, rhodoneas, eggs, bifoliums, Talbot curves, tricuspoids, torpedos, ellipses and an alternative heart-shaped domain.

3.1.2 Python package

The Python package features a primary demo as well as four additional ones that are contained within subpackages. These subpackages are named according to the points outlined in Section 2. We briefly comment these demos.

1. `demo`: This is the primary routine of the Python package, performing identical tasks to `demo_cdes_2` available from the Matlab version.

2. `demo_cleb`: This demo is included in *"lebesgue_constant_evaluator"* subpackage and performs the same functions as `demo_cdes_1` from the Matlab package.

3. `demo_cfit`: This demo is included in *"polynomial_projectors"* subpackage and performs the following tasks by varying the degrees:

   - compute an admissible mesh (AM) for each degree,
   - extract the AFP, DLP, PLP extremal sets,
   - compute the polynomial of interpolation associated with each extremal set and compute the discrete least squares polynomial fitting the whole admissible mesh,
   - compute the errors of approximation in the supremum norm by varying the polynomial degrees,
   - plot the extremal points for the highest degree and the errors of approximation.

4. `demo_cdes`: This demo is included in *"discrete_extremal_sets_constructor"* subpackage and performs the following tasks:

   - compute an admissible mesh (AM) of a fixed degree,
   - extract the AFP, DLP, PLP extremal sets,
   - plot the extremal points on separate figures.

5. `demo_cpom`: This demo is included in the *"polynomial_mesh_constructor"* subpackage. It computes the admissible mesh for a fixed degree and plots the points on a figure.

When testing our routines, changing the values inside the python function `define_domain()`, from 0 to 31, will lead to the creation of complex polynomial curves such as: 0. Unit circle, 1. Segment [-1,1], 2. Polygon M, 3. Sun, 4. Ellipse, 5. Union of circles, 6. Lune, 7. Cardioid, 8. 4 lenses, 9. Curve polygon, 10. Limacon, 11. Lissajous, 12. Egg, 13. Rhodonea, 14. Habenicht clover, 15. Bifolium, 16. Torpedo, 17. Double egg, 18. Sautereau butterfly 1, 19. Sautereau Butterfly 2, 20. Borromean circles, 23. Laporte heart, 24. Epicycloid, 25. Epitrochoid, 26. Hypocycloid, 27. Nephroid, 28. Talbot curve, 29. Tricuspoid, 30. Rectangles+circles, 31. Equilateral triangle.

Acknowledgements

Work partially supported by the DOR funds of the University of Padova, by the INdAM-GNCS 2024 Project “Kernel and polynomial methods for approximation and integration: theory and application software” (A. Sommariva and M. Vianello) and by the National Science Center - Poland, grant Preludium Bis 1, N. 2019/35/O/ST1/02245 (D.J. Kenne). The research cooperation was funded by the program Excellence Initiative – Research University at the Jagiellonian University in Krakòw (A. Sommariva).

This research has been accomplished within the RITA “Research ITalian network on Approximation", the UMI Group TAA “Approximation Theory and Applications", and the SIMAI Activity Group ANA&A.

We thank Leokadia Białas-Cież for her precious help in developing this work.

References

[1] L. Białas-Cież, J.P. Calvi, Pseudo Leja sequences, Ann. Mat. Pura Appl. 191 (2012), 53–75.

[2] L. Białas-Cież, J.P. Calvi, Invariance of polynomial inequalities under polynomial maps, J. Math. Anal. Appl. 439 (2016), 449–464.

[3] L. Białas-Cież, D. Kenne, A. Sommariva, M. Vianello, Chebyshev admissible meshes and Lebesgue constants of complex polynomial projections, J. Comput. Appl. Math. 443 (2024).

[4] L. Białas-Cież, D. Kenne, A. Sommariva, M. Vianello, Evaluating Lebesgue constants by Chebyshev polynomial meshes on cube, simplex and ball, arXiv:2311.18656, Electron. Trans. Numer. Anal., revision required.

[5] L. Bos, J.P. Calvi, N. Levenberg, A. Sommariva, M. Vianello, Geometric Weakly Admissible Meshes, Discrete Least-Squares Approximations and Approximate Fekete Points, Math. Comp. 80 (2011), 1601–1621.

[6] L. Bos, S. De Marchi, A. Sommariva, M. Vianello, Computing multivariate Fekete and Leja points by numerical linear algebra, SIAM J. Numer. Anal. 48 (2010), 1984–1999.

[7] L. Bos, S. De Marchi, A. Sommariva, M. Vianello, Weakly Admissible Meshes and Discrete Extremal Sets, Numer. Math. Theory Methods Appl. 4 (2011), 1–12.

[8] L. Bos, N. Levenberg, On the Approximate Calculation of Fekete Points: the Univariate Case, Electron. Trans. Numer. Anal. 30 (2008), 377–397.

[9] J.P. Calvi, N. Levenberg, Uniform approximation by discrete least-squares polynomials, J. Approx. Theory 152 (2008), 82–100.

[10] L. Giraud, J. Langou, M. Rozloznik, J. van den Eshof, Rounding error analysis of the classical Gram-Schmidt orthogonalization process, Numer. Math. 101 (2005), 87–100.

[11] D.J. Kenne, Multidimensional pseudo-Leja sequences, arXiv:2303.11871.

[12] D.J. Kenne, A. Sommariva, M. Vianello, CPOLYMESH, Matlab codes for complex polynomial approximation by Chebyshev admissible meshes, https://github.com/alvisesommariva/CPOLYMESH.

[13] D.J. Kenne, A. Sommariva, M. Vianello, CPOLYMESH, Python codes for complex polynomial approximation by Chebyshev admissible meshes, https://github.com/DimitriKenne/CPOLYMESH.

[14] S.G. Krantz, A Guide to Complex Variables, Mathematical Association of America, 2008.

[15] S.N. Mergelyan, Uniform approximation to functions of a complex variable, Transl. Amer. Math. Soc. 3 (1962), 294—391, Uspekhi Mat. Nauk 7:2 (1952), 31—122.

[16] F. Piazzon, M. Vianello, Small perturbations of polynomial meshes, Appl. Anal. 92 (2013), 1063–1073 .

[17] F. Piazzon, M. Vianello, A note on total degree polynomial optimization by Chebyshev grids, Optim. Lett. 12 (2018), 63–71.

[18] E.B. Saff, V. Totik, Logarithmic Potentials with External Fields, Springer, 1997.

[19] A. Sommariva, M. Vianello, Computing approximate Fekete points by QR factorizations of Vandermonde matrices, Comput. Math. Appl. 57 (2009), 1324–1336.

[20] V. Totik, The Lebesgue constants for Leja points are subexponential, J. Approx. Theory 287 (2023), 105863.