SAND2015-6579C

Multi-Variate Weighted Leja Sequences for Polynomial Approximation and UQ

John D. Jakeman

Senior Member of Techincal Staff
Optimization and Uncertainty Quantification Department
Sandia National Laboratories
Albuquerque, NM USA

Joint work with Akil Narayan

Sandia National Laboratories is a multi program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

One-dimensional un-weighted interpolation

Given the best N-th order polynomial approximation $p^\star$ of a function f the error of the interpolant f_N based upon a set of $N+1$ random variable realizations $\boldsymbol{\Xi}_{N+1}$ can be bounded by

$$\|f - f_N(\boldsymbol{\Xi})\|_\infty \leq (\Lambda_N(\boldsymbol{\Xi}) + 1)\|f - p^\star\|_\infty$$

where the Lebesgue constant for the grid $\boldsymbol{\Xi}$ is given by

$$\Lambda_N(\boldsymbol{\Xi}) = \max_{\xi\in[a,b]} \lambda_N(\xi)$$

The Lebesgue constant is the maximum value of the the Lebesgue function

$$\lambda_N(\xi) = \sum_{n=1}^{N} |l_n(\xi)|$$

where $l_n(\xi)$ are the Lagrange polynomials

$$l_n = \prod_{\substack{i=1\\n\neq i}}^{N} \frac{\xi - \xi_i}{\xi_n - \xi_i}$$

Multivariate weighted interpolation

The 1D interpolation error bound generalizes to a d-dimensional ω-weighted polynomial subspace V with N terms.

$$\|f - f_N(\boldsymbol{\Xi})\|_{L_\infty(\omega)} \leq (\Lambda_N(\boldsymbol{\Xi}) + 1)\|f - p^\star\|_{L_\infty(\omega)}$$

$$\Lambda_V(\boldsymbol{\Xi}) = \max_{\boldsymbol{\xi} \in I_{\boldsymbol{\xi}}} \sqrt{\omega(\boldsymbol{\xi})} \sum_{n=1}^{N} \left| \frac{l_n(\boldsymbol{\xi})}{\sqrt{\omega(\boldsymbol{\xi}_n)}} \right|$$

V is typically the least degree space of degree at most p, such that $N = \binom{d+p}{p}$

If we can build a multivariate interpolant and generate a point set that minimizes the Lebesgue constant we can generate very efficient PCE interpolants.

The Lebesgue constant is analogous to the condition number of least squares systems. Let p be the polynomial obtained using the coefficients c. Then letting $\hat{p}$ be the polynomial obtained by slightly perturbing the coefficients c to $\hat{c}$ we can write

$$\frac{\|p - \hat{p}\|}{\|p\|} \leq \kappa \frac{\|c - \hat{c}\|}{\|c\|}$$

Multivariate weighted interpolation

The 1D interpolation error bound generalizes to a d-dimensional ω-weighted polynomial subspace V with N terms.

$$\|f - f_N(\Xi)\|_{L_\infty(\omega)} \leq (\Lambda_N(\Xi) + 1)\|f - p^\star\|_{L_\infty(\omega)}$$

$$\Lambda_V(\Xi) = \max_{\xi \in I_\xi} \sqrt{\omega(\xi)} \sum_{n=1}^{N} \left| \frac{l_n(\xi)}{\sqrt{\omega(\xi_n)}} \right|$$

V is typically the least degree space of degree at most p, such that $N = \binom{d+p}{p}$

If we can build a multivariate interpolant and generate a point set that minimizes the Lebesgue constant we can generate very efficient PCE interpolants.

The Lebesgue constant is analogous to the condition number of least squares systems. Let p be the polynomial obtained using the coefficients c. Then letting $\hat{p}$ be the polynomial obtained by slightly perturbing the coefficients c to $\hat{c}$ we can write

$$\frac{\|p - \hat{p}\|}{\|p\|} \leq \kappa \frac{\|c - \hat{c}\|}{\|c\|}$$

We use the sqrt of the weight function because Error involving Lebesgue constant is in terms of L-infty but we want to deal with L-2. We have $\|f\|_\infty = \max(f\tilde{\omega})$ and $\|f\|_2 = \int(f^2\omega)$ so to make consistent we want $\tilde{\omega} = \sqrt{\omega}$.

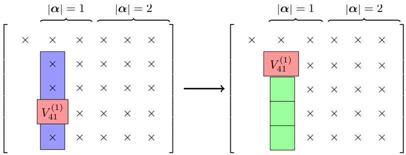
Zeros of Chebyshev polynomials have Lebesgue constant that grows $\sim \log N$

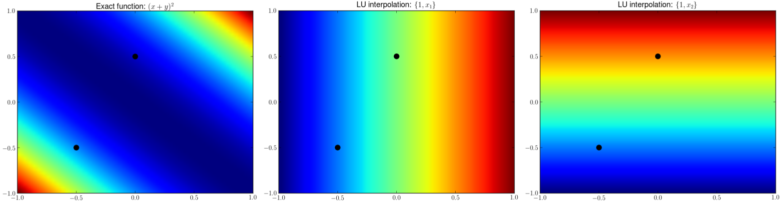
Zeros of Hermite polynomials have similar growth is sqrt of weight function is used but not if just weight function is used.

Multivariate interpolation: Pivoted LU factorization

- Set degree p such that $N \geq M$
- Specifying an ordering of the basis ϕ
- Compute $[\boldsymbol{L}, \boldsymbol{U}, \boldsymbol{P}] = \mathtt{LU}(\boldsymbol{W}\boldsymbol{V})$
- Solve $\boldsymbol{c} = (\boldsymbol{L}\boldsymbol{U})^{-1}\boldsymbol{P}\boldsymbol{W}\boldsymbol{f}$

LU factorization requires a square matrix, if $M \neq N$ then a subset of the basis must be (arbitrarily) chosen.

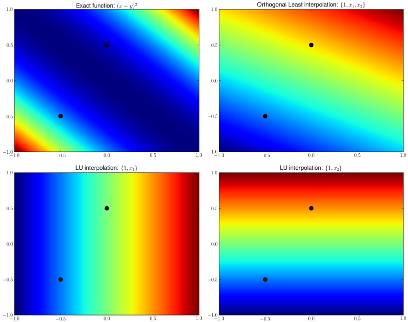


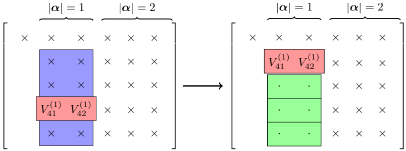


Multivariate interpolation: Least Orthogonal Interpolation

- Compute a pivoted degree-block LU factorization $\boldsymbol{PV} = \boldsymbol{LUH}$
- Solve $\boldsymbol{c} = \boldsymbol{H}^T \boldsymbol{U}^{-1} \boldsymbol{L}^{-1} \boldsymbol{f}$

- Degree p is lowest degree that interpolates the data (allows for degenerate points).
- OLI is designed for $M \leq N$.
- OLI uses a linear combination of all terms $|\alpha| \leq p$.





Un-weighted LU-Leja interpolation sequences: construction

Given Ξ_N, choose $\boldsymbol{\xi}_{N+1}$ such that

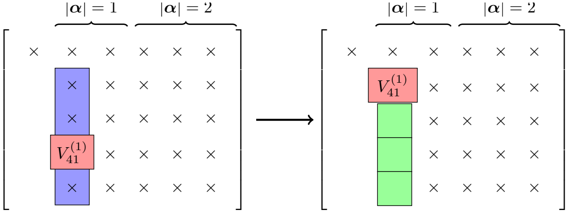
$$\boldsymbol{\xi}_{N+1} = \underset{y \in \mathcal{Y} \subset \mathbb{R}^d}{\arg\max} \left|\det V(\Xi_N \cup y)\right|$$

Row pivoted LU factorization can be used to perform a greedy search for Leja sequences over a finite candidate set proceeds as follows:

- Define large candidate set Ξ^{cand}.
- Form Vandermonde matrix $\boldsymbol{V}$, $V_m n = \phi_n(\boldsymbol{\xi}_m^{\text{cand}})$
- Factorize $[\boldsymbol{L}, \boldsymbol{U}, \boldsymbol{P}] = \texttt{LU}(\boldsymbol{V})$
- Select 'best' points. $\Xi^{\text{cand}}[P[1:M]]$. The first M pivoted rows define the discrete Leja sequence.

Pivoted LU attempts to maximize the determinant of V

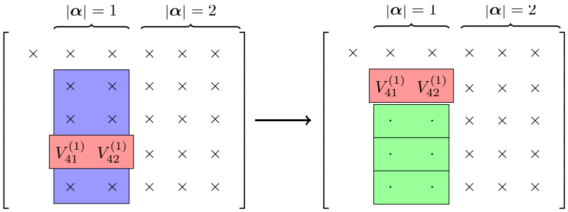
- Sequence is dependent on the ordering of ϕ_n
- Leja sequences have only been built for un-weighted spaces $w(\boldsymbol{\xi}) = 1$.



Un-weighted OLI-Leja interpolation sequences: construction

Row pivoted LU factorization can be used to peform a greedy search for Leja sequences over a finite candidate set proceeds as follows:

- Use pivots from $[\boldsymbol{L}, \boldsymbol{U}, \boldsymbol{H}, \boldsymbol{P}] = \texttt{OLI}(\boldsymbol{V})$ to select best points
- This again attempts to maximize the determinant of V through pivoting
- Sequence is not dependent on the ordering of ϕ_n
- A linear combination of basis functions is used to select pivot points (rows)
- Sequences will be more stable than LU based sequences



Weighted Leja interpolation sequences

To apply Leja sequences to weighted spaces we introduce the matrix

$$\boldsymbol{W}, \quad w_{m,m} = \sqrt{\omega(\boldsymbol{\xi}_m)}$$

Weighted LU-Leja sequences can be obtained via

$$[\boldsymbol{L}, \boldsymbol{U}, \boldsymbol{P}] = \texttt{LU}(\boldsymbol{W}\boldsymbol{V})$$

and OLI-Leja sequences via

$$[\boldsymbol{L}, \boldsymbol{U}, \boldsymbol{H}, \boldsymbol{P}] = \texttt{OLI}(\boldsymbol{W}\boldsymbol{V})$$

OLI Leja sequences: Continuous-greedy optimization

Using current sequence $\boldsymbol{\Xi}_N$ build interpolants of all polynomial basis functions of degree $|\alpha| = k$

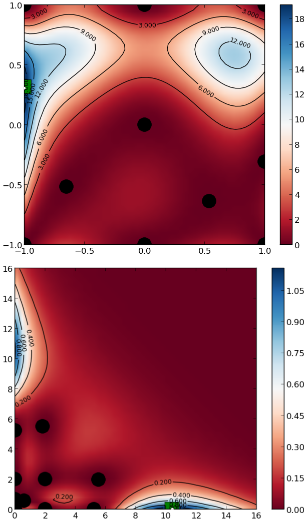
$$p_\alpha(\boldsymbol{\xi}) = I[\phi_\alpha(\boldsymbol{\xi}); \boldsymbol{\Xi}_N]$$

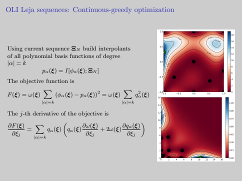
The objective function is

$$F(\boldsymbol{\xi}) = \omega(\boldsymbol{\xi}) \sum_{|\alpha|=k} \left(\phi_\alpha(\boldsymbol{\xi}) - p_\alpha(\boldsymbol{\xi})\right)^2 = \omega(\boldsymbol{\xi}) \sum_{|\alpha|=k} q_\alpha^2(\boldsymbol{\xi})$$

The j-th derivative of the objective is

$$\frac{\partial F(\boldsymbol{\xi})}{\partial \xi_j} = \sum_{|\alpha|=k} q_\alpha(\boldsymbol{\xi}) \left(q_\alpha(\boldsymbol{\xi}) \frac{\partial \omega(\boldsymbol{\xi})}{\partial \xi_j} + 2\omega(\boldsymbol{\xi}) \frac{\partial q_\alpha(\boldsymbol{\xi})}{\partial \xi_j} \right)$$

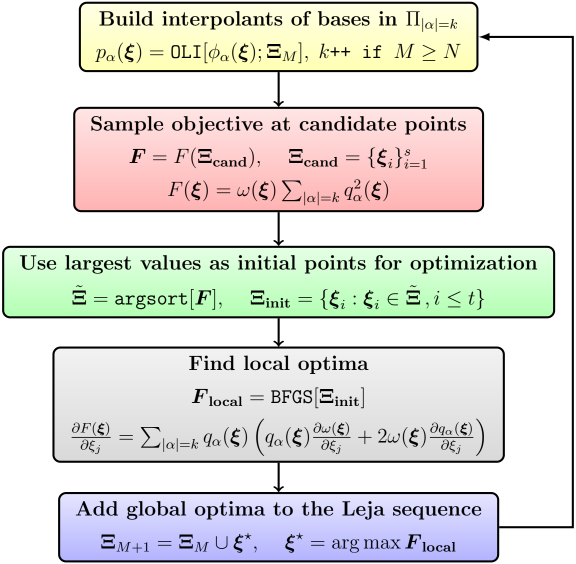




Scalar LU at iteration i maximizes the determinant by row pivoting (based upon column i). It orthogonalizes (subtracting a scalar from column i) each row to all factorized rows. OLI maximizes the determinant by row pivoting, but instead of subtracting scalar it must orthogonalize all columns corresponding to basis functions of degree k. The objective function is the row norm (column norm if we are using QR on V_k^T, where V_k is Vandermonde formed by all bases of degree k. Like scalar LU which chooses the largest scalar pivot, OLI chooses the largest row-norm.

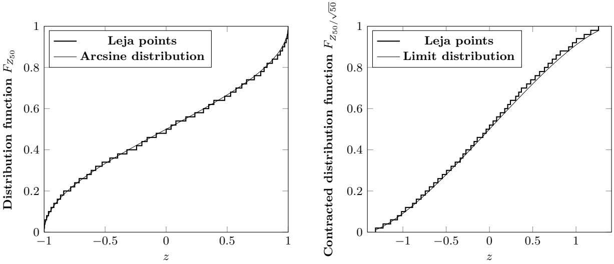
Continuous OLI Leja sequences: Algorithm

OLI continuous Leja sequence

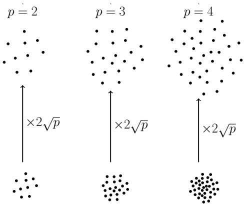


Properties of Leja sequences

1D Leja sequences distribute like the correspond like the corresponding Gauss quadrature nodes



Leja sequences has compact support even if $I_{\boldsymbol{\xi}}$ does not



Lebesgue constants of Leja sequences

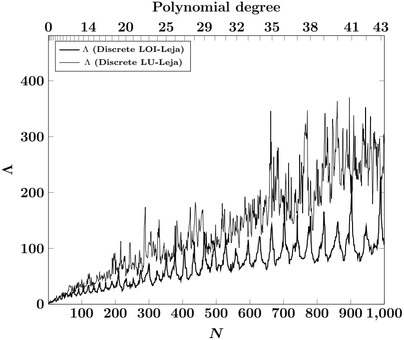
The least orthogonal Lagrange basis functions are

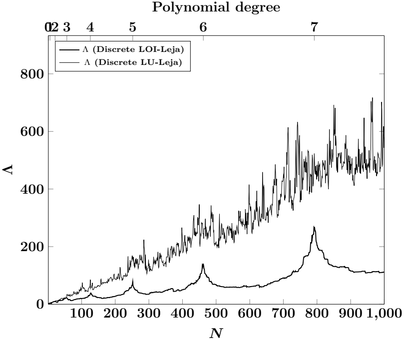
$$l_j(\boldsymbol{\xi}) = \sum_{n=1}^{N} c_n^{(j)} \phi_n(\boldsymbol{\xi}), \quad \boldsymbol{C} = \boldsymbol{H}^T \boldsymbol{U}^{-1} \boldsymbol{L}^{-1}, \quad \boldsymbol{C} = [\boldsymbol{c}^{(1)}, \cdots, \boldsymbol{c}^{(n)}]$$

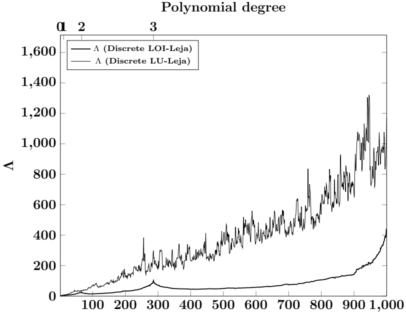
We compute the Lebesgue constant via multi-start local optimization (BFGS)

$$\Lambda_V(\boldsymbol{\Xi}) = \max_{\boldsymbol{\xi} \in I_{\boldsymbol{\xi}}} \sqrt{\omega(\boldsymbol{\xi})} \sum_{n=1}^{N} \left| \frac{l_n(\boldsymbol{\xi})}{\sqrt{\omega(\boldsymbol{\xi}_n)}} \right|$$

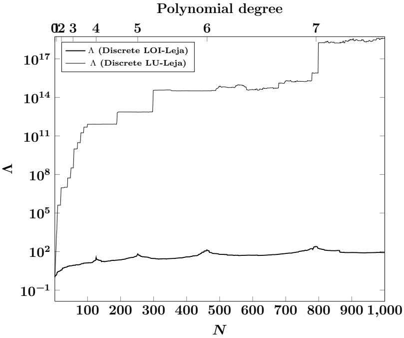
Lebesgue constants of un-weighted Leja sequences

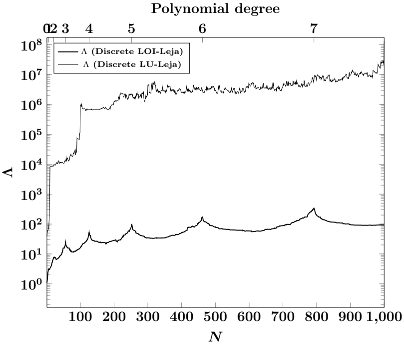






Lebesgue constants of weighted Leja sequences





Interpolation on Leja sequences: Example

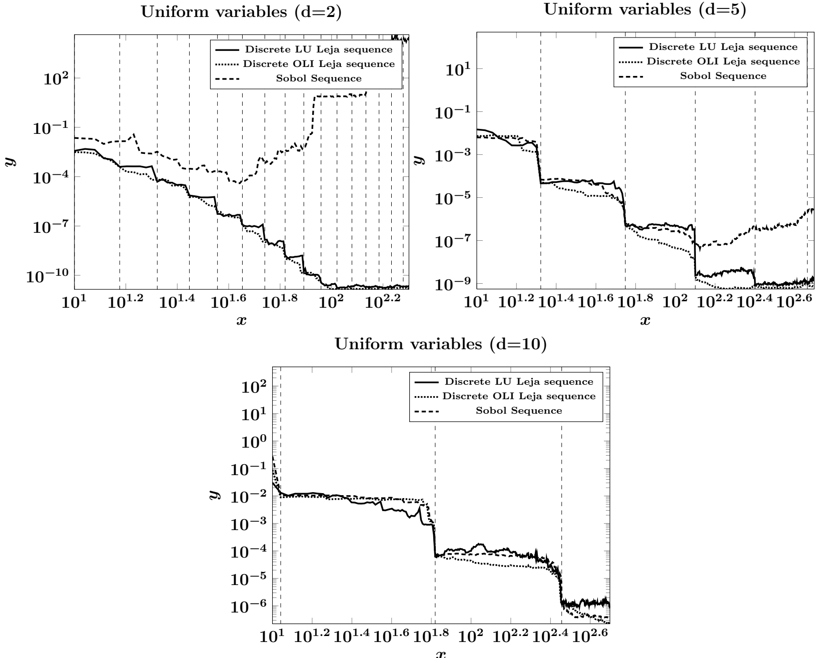
We want to approximate $q(\boldsymbol{\xi}) = u(1/3, \boldsymbol{\xi})$ where

$$-\frac{d}{dx}\left[a(x,\boldsymbol{\xi})\frac{du}{dx}(x,\boldsymbol{\xi})\right] = 1 \quad (x,\boldsymbol{\xi}) \in (0,1)\times I_{\boldsymbol{\xi}}$$
$$u(0,\boldsymbol{\xi}) = u(1,\boldsymbol{\xi}) = 0$$

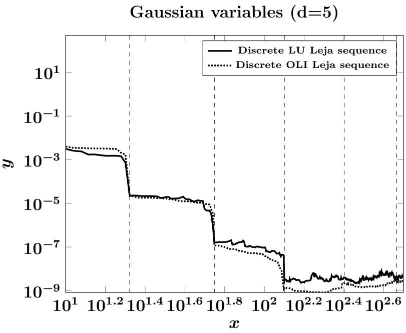
with diffusivity $\log(a(x,\boldsymbol{\xi})) = \bar{a} + \sigma_a \sum_{k=1}^{d} \sqrt{\lambda_k}\varphi_k(x)\xi_k$, where $\{\lambda_k\}_{k=1}^{d}$ and $\{\varphi_k(x)\}_{k=1}^{d}$ are determined by $C_a(x_1,x_2) = \exp\left[-\frac{(x_1-x_2)^2}{l_c^2}\right]$

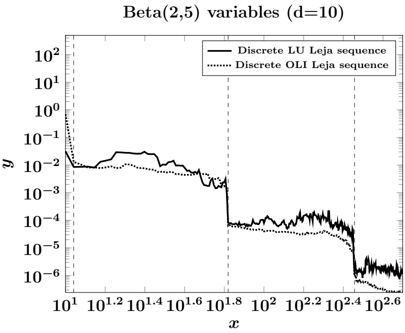
- Compute PCE using LU or OLI interpolation
- Measure accuracy in PCE approximation q_Λ by computing $M_{\text{test}}^{-1/2}\left\|q - q_\Lambda\right\|_{\ell_2(w)}$ using $M_{\text{test}} = 10000$ samples from $w(\boldsymbol{\xi})$.

Approximation accuracy of un-weighted Leja sequences



Approximation accuracy of weighted Leja sequences





Quadrature with Leja sequences

Let $\{p_n\}_{n=1}^{N}$, denote family of polynomials orthonormal under ω.

Given data f_n we wish to interpolate at the sites $\boldsymbol{\xi}_n$, so we seek the coefficients c_n solving the linear problem

$$\boldsymbol{V}\boldsymbol{c} = \boldsymbol{f}, \qquad V_{m,n} = \phi_n(\boldsymbol{\xi}_m).$$

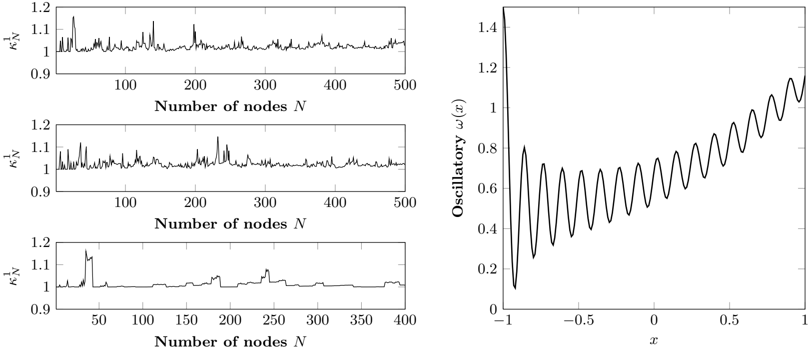
The first row of the matrix $\boldsymbol{V}^{-1}$ gives us quadrature weights v_n defining the Leja polynomial quadrature rule

$$Q_N f \simeq \sum_{n=1}^{N} v_n f(\boldsymbol{\xi}_n).$$

Stability

The condition number of the quadrature rule Q_N is given by

$$\kappa_N = \sum_{n=1}^{N} |v_n|.$$



Let μ be an integral using some quadrature weights w. Then letting $\hat{\mu}$ be the integral obtained by slightly perturbing the quadrature weights w to $\hat{w}$ we can write

$$\frac{\|\mu - \hat{\mu}\|}{\|\mu\|} \leq \kappa \frac{\|w - \hat{w}\|}{\|w\|}$$