AEROSPACE ENGINEERING
TEXAS A&M UNIVERSITY

Lecture 1
ORTHOGONAL APPROXIMATION

JOHN L. JUNKINS & ROBYN M. WOOLLANDS

Five Part Lecture Series

Picard-Chebyshev Numerical Integration Applications in Astrodynamics

Texas A&M University
Department of Aerospace Engineering
College Station, TX 77840

Spring 2017

FIVE PART LECTURE SERIES

Lecture	Title	Presenter
1	Orthogonal Approximation	Junkins
2	Numerical Quadrature	Junkins
3	Picard Chebyshev Methods & Theoretical Convergence	Woollands
4	Accelerated Picard Iteration & Adaptive Segmentation	Woollands
5	Gravity Approximations	Junkins

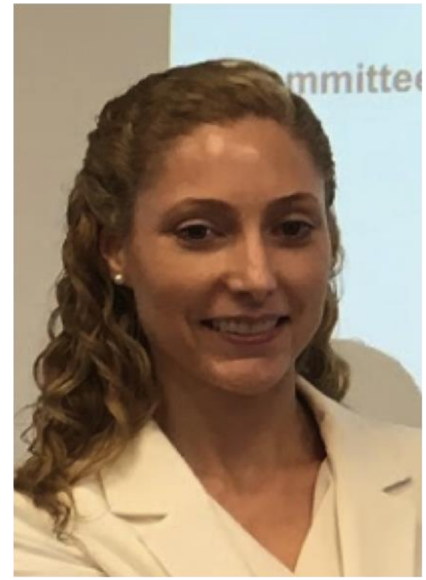
CONTACT INFORMATION

JOHN L. JUNKINS

Distinguished Professor of Aerospace Engineering

Texas A&M University
Department of Aerospace Engineering
College Station, TX 77840

junkins@tamu.edu



ROBYN M. WOOLLANDS

Postdoctoral Research Associate

Texas A&M University
Department of Aerospace Engineering
College Station, TX 77840

robyn.woollands@gmail.com

CONTENTS

FUNCTION APPROXIMATION

What is function approximation?

• Function approximation can be approached by sampling the ***true function*** at a discrete number of points (nodes) and introducing a linear combination of basis functions to approximate the given function within a specified tolerance.

Why is this useful?

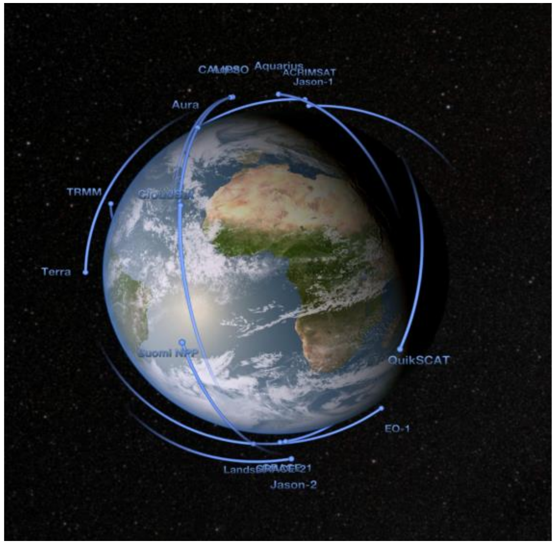
• As one example, when a function is ***not analytically integrable***, but an accurate combination of ***basis functions is integrable***, then the advantage is obvious.

• Consider the second order, nonlinear, ordinary differential equation:

$$\boxed{\ddot{\boldsymbol{r}} = -\frac{\mu}{r^3}\boldsymbol{r} + \boldsymbol{a}_d(\boldsymbol{r},\boldsymbol{v},t), \qquad \boldsymbol{v} = \dot{\boldsymbol{r}} \qquad (1)}$$

• The term $\boldsymbol{a}_d(\boldsymbol{r},\boldsymbol{v},t)$ represents perturbations: e.g., spherical harmonic gravity, drag, etc. (***More in Lecture 5***)

• Given an initial position $\boldsymbol{r}(t_0)$ and velocity $\boldsymbol{v}(t_0)$ this eqn may be solved numerically to compute $\boldsymbol{r}(t)$ and $\boldsymbol{v}(t)$.



NASA/Goddard Space Flight Center
Scientific Visualization Studio

• A remarkably stable and efficient approach (compared to existing methods), a recursive, accelerated Picard method will be introduced that allows, on the k^{th} iteration, analytical integration of the RHS of **Eq. (1)** along the $(k\text{-}1)^{th}$ path approximation.

APPROXIMATION AND RESIDUAL ERRORS

How to approximate a function?

• Approximating a function is done in a way the ***minimizes SOME MEASURE of the error*** (or residual) between the true and approximate functions.

Approximation

• The function, $f(\xi)$, is approximated by a sum of $N+1$ polynomials basis functions $\{\phi_0(\xi), \phi_1(\xi), \ldots, \phi_M(\xi)\}$ as shown:

$$f(\xi) \approx \sum_{\alpha=0}^{N} a_\alpha \phi_\alpha(\xi)$$

Basis Function
(e.g., **Chebyshev Polynomials**)

Coefficients
(**Chebyshev Coefficients**)

Residuals

• The residual errors are simply the difference between the function computed at each of the $M+1$ nodes and the approximated values (for specified N and basis function the residuals are a function only of the coefficients $\{a_0, a_1, \ldots, a_N\}$):

$$r_j = f(\xi_j) - \sum_{\alpha=0}^{N} a_\alpha \phi_\alpha(\xi_j); \; j = 0, 1, \ldots, M.$$

Residual

Function Evaluated at Sample Points

$\xi_0, \xi_1, \ldots, \xi_M$
Discrete Sample Points (Nodes)
(e.g., **Cosine Sampling or Uniform Sampling**)

VECTOR MATRIX & LEAST SQUARES

Vector Matrix Form

• Approximating a function is frequently done in a way that ***minimizes a chosen norm of the residual error r*** between the true and approximate functions.

$$\boxed{\boldsymbol{r} = \boldsymbol{f} - \Phi \boldsymbol{a}} \qquad (2)$$

$$\boldsymbol{f} = \begin{bmatrix} f(\xi_0) \\ f(\xi_1) \\ \vdots \\ f(\xi_M) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \phi_0(\xi_0) & \phi_1(\xi_0) & \cdots & \phi_N(\xi_0) \\ \phi_0(\xi_1) & \phi_1(\xi_1) & \cdots & \phi_N(\xi_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\xi_M) & \phi_1(\xi_M) & \cdots & \phi_N(\xi_M) \end{bmatrix}, \quad \boldsymbol{a} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix}$$

Least Squares Approximation

• Consider the usual linear equation $\boldsymbol{b} = A\boldsymbol{x}$. Solving this equation for $\boldsymbol{x}$ is a standard problem in linear algebra; there may be a ***unique solution*** for $\boldsymbol{x}$, an ***infinity of solutions***, or ***no exact solution***, depending on the dimensions and rank of A.

• For the over determined case (the number of eqs exceeds the number of unknowns), we expect no exact solution, in which case, the classical approach is to find $\boldsymbol{x}$ to minimize $J = \frac{1}{2}\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}$; the sum of square of the residual error $\boldsymbol{e} = \boldsymbol{b} - A\boldsymbol{x}$. By analogy, **Eq. (2)** invites us to minimize $\frac{1}{2}\boldsymbol{r}^{\mathrm{T}}\boldsymbol{r}$ to estimate $\boldsymbol{a}$.

LEAST SQUARES APPROXIMATION

Error / Residuals

- Starting with $A\boldsymbol{x} = \boldsymbol{b}$, the ***error*** ($\boldsymbol{e}$) that we seek to minimize is given $\boldsymbol{e} = \boldsymbol{b} - A\boldsymbol{x}$.

Cost Function

- The cost function (J) measures the fit quality is $J = \frac{1}{2}\boldsymbol{e}^T\boldsymbol{e} = \frac{1}{2}(\boldsymbol{b} - A\boldsymbol{x})^T(\boldsymbol{b} - A\boldsymbol{x})$.

Minimization of $J = \frac{1}{2}\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}$: To find the judicious $\boldsymbol{x}$, we expand J and simplify to

$$J = \frac{1}{2}\boldsymbol{b}^T\boldsymbol{b} - \boldsymbol{b}^T A\boldsymbol{x} + \frac{1}{2}\boldsymbol{x}^T A^T A\boldsymbol{x}.$$

The necessary conditions are: $\frac{\partial J}{\partial \mathbf{x}} = 0 - A^T\boldsymbol{b} + A^T A\mathbf{x} = 0 \Rightarrow \boxed{A^T A\mathbf{x} = A^T\boldsymbol{b}}$ (3)

The sufficient condition is: The matrix $A^T A$ must be positive definite.

- The ***normal equations***, **Eqs (3)** can be solved for $\boldsymbol{x}$ via a variety of ways, including "brute fore inversion" (or we can use the QR, SVD, or related algorithms) as

$$\boldsymbol{x} = \left(A^T A\right)^{-1} A^T\boldsymbol{b}$$ **or, with a weight matrix:** $$\boldsymbol{x} = \left(A^T W A\right)^{-1} A^T W\boldsymbol{b}.$$

MATRIX INVERSE

From the previous slide… $$\boldsymbol{x} = \left(A^T W A\right)^{-1} A^T W \boldsymbol{b}$$

• Consistent with the above, minimization of $\frac{1}{2}\boldsymbol{r}^{\mathrm{T}}\boldsymbol{r}$ gives: $\boldsymbol{a} = \left(\Phi^T W \Phi\right)^{-1} \Phi^T W \boldsymbol{f}$

Matrix Inverse

• Note that a ***matrix inverse (or some other more robust but expensive and potentially troublesome method)*** is required for least squares approximation. The requirement to numerically invert the normal equations is computationally expensive, especially as the polynomial order (N) increases.

Diagonal Matrix ⇔ Orthogonal Basis Function Approximation

• It turns out that if make judicious choices (consequence of orthogonality):…
  - the basis functions judiciously (e.g., ***Chebyshev Polynomials***),
  - the function to be approximated is sampled using a ***cosine distribution*** of nodes,
  - the ***weight matrix*** is chosen to be $W = diag[\frac{1}{2}, 1, 1, \ldots, 1, 1, \frac{1}{2}]$.

Then the matrix $\Phi^{\mathrm{T}} W \Phi$ to be inverted is ***diagonal*** and the inverse is trivial (and machine precision solution for the minimizing coefficient matrix $\boldsymbol{a}$ is possible). Furthermore, for a ***complete set*** *of orthogonal basis functions ,* N & M can be made sufficiently large to allow the ***residual error norm to be zero, to machine precision***.

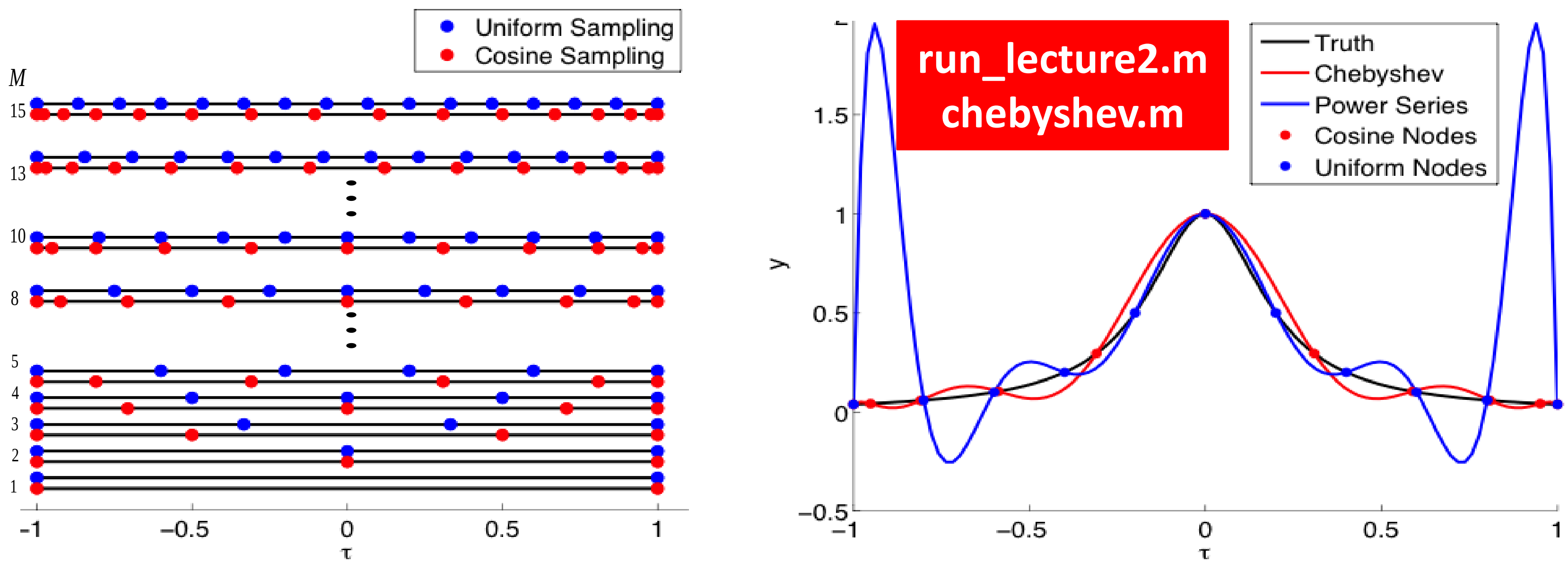
COSINE SAMPLING

Sample Points

• Sample points (of the forcing function to be fitted) are computed using a cosine node distribution as follows: $\xi_j = -\cos(j\pi/M),\ j = 0, 1, 2, ..., M.$

Runge Effect

• Note ***clustering*** of cosine nodes near the ±1 boundaries. ***Uniform sampling*** leads to the "***Runge effect***" (large oscillatory errors between nodes near the boundaries due to ***lack of support*** outside the interval). ***Denser interior*** nodes helps ***compensate*** for absence of support to the right and left of the approximation interval. Qualitatively, cosine nodes lead to nearly uniform approximation errors.



CHEBYSHEV POLYNOMIALS

Definition

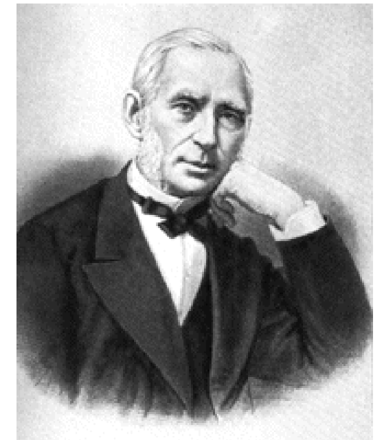
- Chebyshev polynomials are defined on the range $T(\tau) \in [-1, 1]$
- Two ways to generate Chebyshev polynomials

1. ***Recursively*** $T_0(\tau) = 1, \quad T_1(\tau) = \tau,$

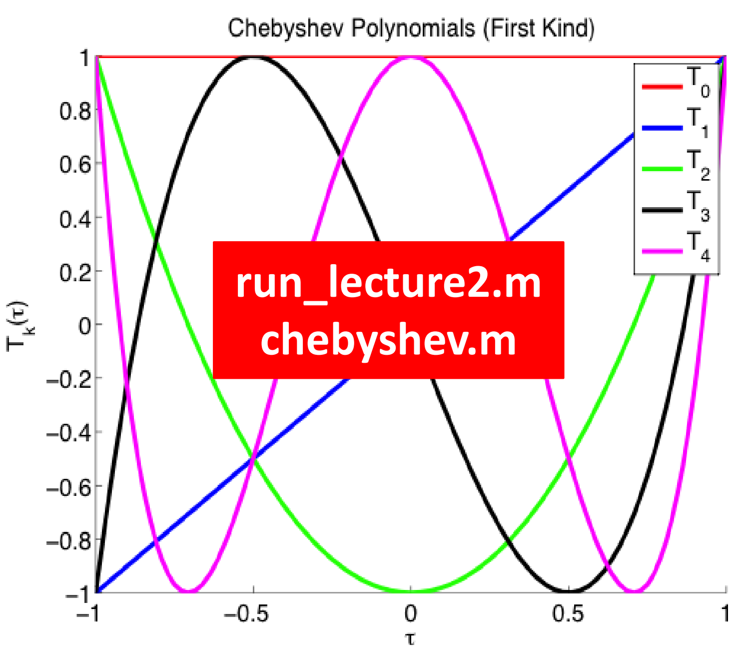
$$T_{k+1}(\tau) = 2\tau T_k(\tau) - T_{k-1}(\tau). \quad \tau \in [-1, 1]$$

2. ***Trigonometrically***

$$T_k(\tau) = \cos(k \arccos(\tau)). \qquad \tau \in [-1, 1]$$


Picture Credit: Wikipedia

Rafnuty Chebyshev (1821-1894)



Independent Variable Transformation

- The independent variable (t) is transformed into (τ: $-1 \le \tau \le +1$) as follows

$$\frac{df}{d\tau} = \frac{df}{dt}\frac{dt}{d\tau}$$

- One suggestion is: $t = t_0 + (\tau + 1)(t_f - t_0)/2$

$$\tau = -1 + 2(t - t_0)/(t_f - t_0)$$

so $\quad \dfrac{dt}{d\tau} = (t_f - t_0)/2$

WEIGHT MATRIX EXAMPLE

Example for $M = 2$

- Consider the following

$$m_{\alpha\beta} = \begin{bmatrix} m_{00} & m_{01} & m_{02} \\ m_{10} & m_{11} & m_{12} \\ m_{20} & m_{21} & m_{22} \end{bmatrix} = \Phi^{\mathrm{T}} W \Phi,$$

where $\Phi^{\mathrm{T}} W \Phi$ is given by

$$\Phi^{\mathrm{T}} W \Phi = \begin{bmatrix} \phi_0(\xi_0) & \phi_0(\xi_1) & \phi_0(\xi_2) \\ \phi_1(\xi_0) & \phi_1(\xi_1) & \phi_1(\xi_2) \\ \phi_2(\xi_0) & \phi_2(\xi_1) & \phi_2(\xi_2) \end{bmatrix} \begin{bmatrix} W_0 & 0 & 0 \\ 0 & W_1 & 0 \\ 0 & 0 & W_2 \end{bmatrix} \begin{bmatrix} \phi_0(\xi_0) & \phi_1(\xi_0) & \phi_2(\xi_0) \\ \phi_0(\xi_1) & \phi_1(\xi_1) & \phi_2(\xi_1) \\ \phi_0(\xi_2) & \phi_1(\xi_2) & \phi_2(\xi_2) \end{bmatrix}.$$

- The matrix multiplication computation leads to the following:

$$\begin{gathered} m_{00} = W_0 \phi_0^2(\xi_0) + W_1 \phi_0^2(\xi_1) + W_2 \phi_0^2(\xi_2), \\ m_{11} = W_0 \phi_1^2(\xi_0) + W_1 \phi_1^2(\xi_1) + W_2 \phi_1^2(\xi_2), \\ m_{22} = W_0 \phi_2^2(\xi_0) + W_1 \phi_2^2(\xi_1) + W_2 \phi_2^2(\xi_2), \\ m_{01} = m_{10} = W_0 \phi_0(\xi_0)\phi_1(\xi_0) + W_0 \phi_0(\xi_0)\phi_1(\xi_0) + W_2 \phi_0(\xi_2)\phi_1(\xi_2), \\ m_{02} = m_{20} = W_0 \phi_0(\xi_0)\phi_2(\xi_0) + W_1 \phi_0(\xi_1)\phi_2(\xi_1) + W_2 \phi_0(\xi_2)\phi_2(\xi_2), \\ m_{12} = m_{21} = W_0 \phi_1(\xi_0)\phi_2(\xi_0) + W_1 \phi_1(\xi_1)\phi_2(\xi_1) + W_2 \phi_1(\xi_2)\phi_2(\xi_2). \end{gathered}$$

- If $\{\phi_0(\xi), \phi_1(\xi), \phi_2(\xi)\}$ are an orthogonal set, $m_{01} = m_{10} = 0$, $m_{02} = m_{20} = 0$ and $m_{12} = m_{21} = 0$. All that remains are the diagonal terms of matrix m.

WEIGHT MATRIX

Coefficients

- Recall the Chebyshev fit coefficients: $\boldsymbol{a} = \left(\Phi^{\mathrm{T}} W \Phi\right)^{-1} \Phi^{\mathrm{T}} W \boldsymbol{f}$

Diagonal Matrix

- If the matrix $\Phi^{\mathrm{T}} W \Phi$ is diagonal then the inverse is simply the reciprocal of the diagonal elements.

$$\left(\Phi^{\mathrm{T}} W \Phi\right)^{-1} = diag\left\{1/\left(\Phi^{\mathrm{T}} W \Phi\right)_{\mathrm{ii}}\right\} = diag\left\{1/m_{00} \quad 1/m_{11} \quad \dots \quad 1/m_{NN}\right\}$$

Inner Products

- The typical element of $\Phi^{\mathrm{T}} W \Phi$ is a ***discrete inner product*** denoted $m_{\alpha\beta} = m_{\beta\alpha}$, and invoking the condition that $\Phi^{\mathrm{T}} W \Phi$ be diagonal gives rise to the orthogonality conditions, requiring the typical pair of inner products to obey:

$$m_{\alpha\beta} = m_{\beta\alpha} \equiv \langle \phi_\alpha(\xi), \phi_\beta(\xi) \rangle \equiv \sum_{j=0}^{M} W_j \phi_\alpha(\xi_j) \phi_\beta(\xi_j) = \left\{ \begin{array}{c} 0, \ for\ \alpha \neq \beta \\ m_{\alpha\alpha} = c_\alpha > 0, \ for\ \alpha = \beta \end{array} \right\},$$

Note that the orthogonality conditions depend jointly on (i) the particular basis functions, (ii) the node locations, and (iii) the weight matrix.

WHY HALF?

Preserve Orthogonality

- At a fundamental level, once the basis functions and nodes have been chosen, the weights are ***what ever they have to be*** to ensure orthogonality conditions are | satisfied.

Proof

- Chebyshev polynomials may be computed as $\phi_k(\xi_j) = \cos(k \arccos(\xi_j))$, $\xi \in [-1,1]$
- As part of the ***least squares*** analysis we showed that the ***inner product*** of two Chebyshev basis functions must be computed, i.e. $\phi_\alpha(\xi) = \cos(\alpha x)$ and $\phi_\beta(\xi) = \cos(\beta x)$, where $x_j = arccos(\xi_j)$.
- Since $\xi_j = -cos\left(\frac{j\pi}{M}\right)$, $x = \arccos(\xi_j) = \arccos\left(-cos\left(\frac{j\pi}{M}\right)\right) = \frac{(M\pi - j\pi)}{M}$.

  we can make use use of the ***products-to-sums cosine trigonometric identity*** leads to the following:

$$\cos(\alpha x)\cos(\beta x) = \frac{1}{2}\left(\cos((\alpha+\beta)x) + \cos((\alpha-\beta)x)\right)$$

- This implies that the typical product of the two basis functions within the summation may be written purely as a sum of cosine terms. We proceed with an example.

EXAMPLE

Example: $M = 3, \alpha = 2, \beta = 3$

- From the previous slide: $\cos(\alpha x)\cos(\beta x) = \frac{1}{2}\left(\cos\left((\alpha+\beta)x\right) + \cos\left((\alpha-\beta)x\right)\right)$
- The summation of the inner products, using $M = 3, \alpha = 2, \beta = 3$, results in

$$\sum_{j=0}^{M} \phi_\alpha(\xi_j)\phi_\beta(\xi_j) = \sum_{j=0}^{M} \frac{1}{2}\left(\cos\left((\alpha+\beta)x_j\right) + \cos\left((\alpha-\beta)x_j\right)\right),$$

which becomes

$$\sum_{j=0}^{3} \phi_2(\xi_j)\phi_3(\xi_j) = \frac{1}{2}\Big(\cos\left((2+3)x_0\right) + \cos\left((2-3)x_0\right) + \cos\left((2+3)x_1\right) + \cos\left((2-3)x_1\right) + \ldots$$
$$\ldots + \cos\left((2+3)x_2\right) + \cos\left((2-3)x_2\right) + \cos\left((2+3)x_3\right) + \cos\left((2-3)x_3\right)\Big).$$

- More specifically,

$$\sum_{j=0}^{3} \phi_2(\xi_j)\phi_3(\xi_j) = \frac{1}{2}\left\{ \cos\underbrace{\left((2+3)\frac{(3\pi-0\pi)}{3}\right)}_{=0} + \cos\underbrace{\left((2-3)\frac{(3\pi-0\pi)}{3}\right)}_{=0} + \cos\underbrace{\left((2+3)\frac{(3\pi-\pi)}{3}\right)}_{=\frac{10\pi}{3}\equiv\theta_1} + \cos\underbrace{\left((2-3)\frac{(3\pi-\pi)}{3}\right)}_{=-\frac{2\pi}{3}\equiv\theta_2} + \ldots\right.$$
$$\left. \ldots + \cos\underbrace{\left((2+3)\frac{(3\pi-2\pi)}{3}\right)}_{=\frac{5\pi}{3}=2\theta_1} + \cos\underbrace{\left((2-3)\frac{(3\pi-2\pi)}{3}\right)}_{=-\frac{\pi}{3}=2\theta_2} + \ldots + \cos\underbrace{\left((2+3)\frac{(3\pi-3\pi)}{3}\right)}_{=5\pi=3\theta_1} + \cos\underbrace{\left((2-3)\frac{(3\pi-3\pi)}{3}\right)}_{=-\pi=3\theta_2} \right\}.$$

- For orthogonality, the inner product of the two basis functions (ϕ_2 and ϕ_3) must be zero. Thus the above equation must equal zero. The purpose of the lower braces will soon become apparent.

EXAMPLE CONTINUED

Example: $M = 3, \alpha = 2, \beta = 3$

- Consider another trigonometric identity that is the sum of cosine terms:

$$\frac{1}{2} + \cos(\theta) + \cos(2\theta) + \ldots + \cos((M-1)\theta) + \frac{1}{2}\cos(M\theta) \equiv \frac{1}{2}\sin(M\theta)\cot\left(\frac{1}{2}\theta\right)$$

- It is clear that when $\theta = M\pi$ the RHS become zero.
- This trigonometric identity is similar to the cosine sum that results from the inner product in the last equation on the previous slide, if θ is set equal to $(\alpha + \beta)\frac{\pi}{3}$. For $\theta = (\alpha - \beta)\frac{\pi}{3}$ we have another form of the identify above.
- So in fact the last equation on the previous slide is the sum of twice the above identity, one with $\theta = (\alpha + \beta)\frac{\pi}{3}$ and one with $\theta = (\alpha - \beta)\frac{\pi}{3}$. This is shown below with the θ_1 colored red to aid in identifying the two series.

$$\frac{1}{2} + \frac{1}{2} + \cos\theta_1 + \cos\theta_2 + \cos 2\theta_1 + \cos 2\theta_2 + \frac{1}{2}\cos 3\theta_1 + \frac{1}{2}\cos 3\theta_2 = 0$$

- Independently comparing the red parts of the above equation with the red parts in the last equation on the previous slide, it is clear that the only difference is that the first and last terms are $^1/_2$ the size of those on the previous slide. Thus a weight matrix of $W = diag[½, 1, 1, \ldots, 1, 1, ½]$ is required to achieve orthogonality.

CHEBYSHEV COEFFICIENTS

Since we know the weight matrix for orthogonality, now we obtain the needed inner products:

- Recall the least squares coefficients: $\boldsymbol{a} = \left(\Phi^{\mathrm{T}} W \Phi\right)^{-1} \Phi^{\mathrm{T}} W \boldsymbol{f}$.
- Substituting in Chebyshev polynomials, utilizing ***cosine sample points*** and the ***correct weight matrix***, satisfies the orthogonality conditions allows the coefficients to be computed as:

$$a_\alpha = \langle T_\alpha(\xi), f(\xi)\rangle \equiv \frac{1}{c_\alpha}\left\{\sum_{j=0}^{M} W_j T_\alpha(\xi_j) f(\xi_j)\right\} = \frac{1}{c_\alpha}\left\{\frac{1}{2} T_\alpha(\xi_0) f(\xi_0) + \ldots + T_\alpha(\xi_{M-1}) f(\xi_{M-1}) + \frac{1}{2} T_\alpha(\xi_M) f(\xi_M)\right\},$$

where the denominators c_α are ***positive constants***.

$$c_\alpha = \langle T_\alpha(\xi), f(\xi)\rangle \equiv \sum_{j=0}^{M} W_j T_\alpha^2(\xi_j) = \left\{\frac{1}{2} T_\alpha^2(\xi_0) + T_\alpha^2(\xi_1) + \ldots + T_\alpha^2(\xi_{M-1}) + \frac{1}{2} T_\alpha^2(\xi_M)\right\}, \alpha = 0, 1, \ldots, N$$

- It can be verified (next slide) that the ***denominator inner products*** reduce to simply:

$$\begin{aligned}
c_0 &= \langle T_0(\xi), T_0(\xi)\rangle = M \\
c_\alpha &= \langle T_\alpha(\xi), T_\alpha(\xi)\rangle = M/2, && \alpha = 1, 2, \ldots, N-1 \\
c_N &= \langle T_N(\xi), T_N(\xi)\rangle = M, && \text{if } M = N \text{ (interpolation case)} \\
c_N &= \langle T_N(\xi), T_N(\xi)\rangle = M/2, && \text{if } M > N \text{ (least squares case)}
\end{aligned}$$

- The final coefficients are computed directly from the discrete inner products as:

$$a_0 = \frac{\langle T_0(\xi), f(\xi)\rangle}{\langle T_0(\xi), T_0(\xi)\rangle} = \frac{1}{M}\left\{\frac{1}{2} T_0(\xi_0) f(\xi_0) + \ldots + T_0(\xi_{M-1}) f(\xi_{M-1}) + \frac{1}{2} T_0(\xi_M) f(\xi_M)\right\}$$

$$a_\alpha = \frac{\langle T_\alpha(\xi), f(\xi)\rangle}{\langle T_\alpha(\xi), T_\alpha(\xi)\rangle} = \frac{2}{M}\left\{\frac{1}{2} T_\alpha(\xi_0) f(\xi_0) + \ldots + T_\alpha(\xi_{M-1}) f(\xi_{M-1}) + \frac{1}{2} T_\alpha(\xi_M) f(\xi_M)\right\}, \qquad \alpha = 1, 2, \ldots, N-1$$

$$a_N = \frac{\langle T_N(\xi), f(\xi)\rangle}{\langle T_N(\xi), T_N(\xi)\rangle} = \frac{1}{c_N}\left\{\frac{1}{2} T_N(\xi_0) f(\xi_0) + \ldots + T_N(\xi_{M-1}) f(\xi_{M-1}) + \frac{1}{2} T_N(\xi_M) f(\xi_M)\right\}, \qquad \begin{cases} c_N = M, & M = N \\ c_N = \dfrac{M}{2}, & M > N \end{cases}$$

VERIFICATION OF c_0, c_α, c_N

$c_0 = \langle T_0(\xi), T_0(\xi)\rangle$ $(M > \mathrm{N}, N = 0)$

- Consider the case for $N = 0$, $M = 4$. Performing the inner product and including the correct weight matrix leads to

$$c_0 = \frac{1}{2}T_0^2(\xi_0) + T_0^2(\xi_1) + T_0^2(\xi_2) + T_0^2(\xi_3) + \frac{1}{2}T_0^2(\xi_4).$$

- The zeroth order Chebyshev polynomial is $T_0 = 1$, which leads to $c_0 = \left\{\frac{1}{2} + 1 + 1 + 1 + \frac{1}{2} = 4\right\} = M.$

$c_\alpha = c_1 = \langle T_1(\xi), T_1(\xi)\rangle$ $(M > \mathrm{N})$

- Consider the case for $N = 1$, $M = 4$. Carrying out the inner product as above gives

$$c_1 = \frac{1}{2}T_1^2(\xi_0) + T_1^2(\xi_1) + T_1^2(\xi_2) + T_1^2(\xi_3) + \frac{1}{2}T_1^2(\xi_4).$$

- The cosine sample points are calculated as follows: $\xi_0 = -\cos\left(\frac{0\pi}{4}\right)$, $\xi_1 = -\cos\left(\frac{\pi}{4}\right)$, $\xi_2 = -\cos\left(\frac{2\pi}{4}\right)$, $\xi_3 = -\cos\left(\frac{3\pi}{4}\right)$, $\xi_4 = -\cos\left(\frac{4\pi}{4}\right)$. Substituting these into the first Chebyshev polynomial, $T_1 = \xi$, results in

$$c_\alpha = c_1 = \left\{\frac{1}{2} + \frac{1}{2} + 0 + \frac{1}{2} + \frac{1}{2} = 2\right\} = \frac{M}{2}.$$

$c_\alpha = c_2 = \langle T_2(\xi), T_2(\xi)\rangle$ $(M > \mathrm{N})$

- Similarly for $N = 2$, $M = 4$, and the second Chebyshev polynomial, $T_2 = 2\xi^2 - 1$, with the above sample points, we can quickly verify

$$c_\alpha = c_2 = \left\{\frac{1}{2} + 0 + 1 + 0 + \frac{1}{2} = 2\right\} = \frac{M}{2}.$$

VERIFICATION OF c_0, c_α, c_N CONTINUED

$c_\alpha = c_3 = \langle T_3(\xi), T_3(\xi) \rangle$ $(M > \mathrm{N})$

- Similarly for $N = 3$, $M = 4$, and the third Chebyshev polynomial, $T_3 = 4\xi^3 - 3\xi$, with the sample points shown on the previous slide, we have

$$c_\alpha = c_3 = \left\{ \frac{1}{2} + \frac{1}{2} + 0 + \frac{1}{2} + \frac{1}{2} = 2 \right\} = \frac{M}{2}.$$

$c_N = c_4 = \langle T_4(\xi), T_4(\xi) \rangle$ $(M = \mathrm{N})$

- Similarly for $N = 4$, $M = 4$, and the fourth Chebyshev polynomial, $T_3 = 8\xi^4 - 8\xi^2$, with the sample points shown on the previous slide, we have

$$c_N = c_4 = \left\{ \frac{1}{2} + 1 + 1 + 1 + \frac{1}{2} = 4 \right\} = M.$$

$c_N = c_4 = \langle T_4(\xi), T_4(\xi) \rangle$ $(M > \mathrm{N})$

- For $N = 4$, $M = 5$, the sample points are given by:

$$\xi_0 = -\cos\left(\frac{0\pi}{5}\right),\ \xi_1 = -\cos\left(\frac{\pi}{4}\right),\ \xi_2 = -\cos\left(\frac{2\pi}{4}\right),\ \xi_3 = -\cos\left(\frac{3\pi}{4}\right),\ \xi_4 = -\cos\left(\frac{4\pi}{4}\right),\ \xi_5 = -\cos\left(\frac{5\pi}{4}\right).$$

- Substituting these sample points into the fourth Chebyshev polynomial gives

$$c_N = c_4 \approx \left\{ \frac{1}{2} + 0.6545 + 0.0995 + 0.6545 + \frac{1}{2} = \frac{5}{2} \right\} = \frac{M}{2}.$$

VECTOR MATRIX NOTATION

Coefficients

- The least squares solution may be expressed in vector matrix form:

where $\boxed{\boldsymbol{a} = \left(T^{\mathrm{T}} W T\right)^{-1} T^{\mathrm{T}} W \boldsymbol{f} = V T^{\mathrm{T}} W \boldsymbol{f} = A \boldsymbol{f}}$, $\quad \left(T^{\mathrm{T}} W T\right)^{-1} = V = diag\left\{\frac{1}{c_0}, \frac{1}{c_\alpha}, \frac{1}{c_\alpha}, \ldots, \frac{1}{c_\alpha}, \frac{1}{c_\alpha}, \frac{1}{c_N}\right\}.$

- The least squares operator (A) is simply given by

$M=N$

$$A = \begin{bmatrix} \frac{1}{2}\frac{1}{M}T_0(\xi_0) & \frac{1}{M}T_0(\xi_1) & \cdots & \frac{1}{M}T_0(\xi_{M-1}) & \frac{1}{2}\frac{1}{M}T_0(\xi_M) \\ \frac{1}{2}\frac{2}{M}T_1(\xi_0) & \frac{2}{M}T_1(\xi_1) & \cdots & \frac{2}{M}T_1(\xi_{M-1}) & \frac{1}{2}\frac{2}{M}T_1(\xi_M) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{1}{2}\frac{2}{M}T_{N-1}(\xi_0) & \frac{2}{M}T_{N-1}(\xi_1) & \cdots & \frac{2}{M}T_{N-1}(\xi_{M-1}) & \frac{1}{2}\frac{2}{M}T_{N-1}(\xi_M) \\ \frac{1}{2}\frac{1}{M}T_N(\xi_0) & \frac{1}{M}T_N(\xi_1) & \cdots & \frac{1}{M}T_N(\xi_{M-1}) & \frac{1}{2}\frac{1}{M}T_N(\xi_M) \end{bmatrix}.$$

$M>N$

$$A = \begin{bmatrix} \frac{1}{2}\frac{1}{M}T_0(\xi_0) & \frac{1}{M}T_0(\xi_1) & \cdots & \frac{1}{M}T_0(\xi_{M-1}) & \frac{1}{2}\frac{1}{M}T_0(\xi_M) \\ \frac{1}{2}\frac{2}{M}T_1(\xi_0) & \frac{2}{M}T_1(\xi_1) & \cdots & \frac{2}{M}T_1(\xi_{M-1}) & \frac{1}{2}\frac{2}{M}T_1(\xi_M) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{1}{2}\frac{2}{M}T_{N-1}(\xi_0) & \frac{2}{M}T_{N-1}(\xi_1) & \cdots & \frac{2}{M}T_{N-1}(\xi_{M-1}) & \frac{1}{2}\frac{2}{M}T_{N-1}(\xi_M) \\ \frac{1}{2}\frac{2}{M}T_N(\xi_0) & \frac{2}{M}T_N(\xi_1) & \cdots & \frac{2}{M}T_N(\xi_{M-1}) & \frac{1}{2}\frac{2}{M}T_N(\xi_M) \end{bmatrix}.$$

In general, matrix A is an $[N+1]$x$[M+1]$ matrix. The number of ***rows*** is determined by the ***degree*** of the polynomial and the number of ***columns*** is determined by the number of ***sample points***.

FUNCTION APPROXIMATION

Return to Function Approximation:

- Using summation notation the function is approximated in terms of the matrices T, V, W and $\boldsymbol{f}$ as follows:

$$\boldsymbol{f}(\xi) \approx \sum_{\alpha=0}^{N} a_\alpha \phi_\alpha(\xi) = \sum_{\alpha=0}^{N} \underbrace{\left\{ \sum_{j=0}^{M} V_{\alpha j} W_{\alpha j} T_\alpha(\xi_j) f(\xi_j) \right\}}_{\text{Least squares fit to discrete acceleration samples to obtain coefficients for acceleration}} T_\alpha(\xi).$$

Function approximation

Chebyshev basis functions

We have now approximated the function. We can show via examples and arguments based on completeness, that almost any smooth function can be approximated, in an at worst a piecewise continuous fashion, with near machine precision …

APPROXIMATION EXAMPLE

Perturbed Two-body Acceleration

$$\ddot{\boldsymbol{r}} = -\frac{\mu}{r^3}\boldsymbol{r} + \underbrace{\boldsymbol{a}_d(\boldsymbol{r}, \boldsymbol{v}, t)}, \qquad \boldsymbol{v} = \dot{\boldsymbol{r}}$$

Gravity is modeled by infinite spherical harmonic series as fct ($\boldsymbol{r}$, t).

- Given orbit initial conditions:

$$\boldsymbol{r}_0 = \{7000 \quad 0 \quad 0\}^T \text{ km}$$
$$\boldsymbol{v}_0 = \{0 \quad 5.335 \quad 5.335\}^T \text{ km/s.}$$

- Fit perturbed acceleration along an ideal elliptical orbit, using the spherical harmonic gravity model for degree = 40 and 70. Note that more nodes are required to get a machine precision fit for the higher degree gravity model.