(guest lecture by Jon Wilkening) Math 224A, Fall 2019

# Application of O.P.'s: pseudo-spectral methods

recall recurrences:

<u>monic</u>:
$$\begin{cases} p_0 = 1 \\ p_1 = (x-a_0)p_0 \\ p_2 = (x-a_1)p_1 - b_1 p_0 \\ \vdots \end{cases}$$

<u>normalized</u>:

$$\left(\begin{array}{l} \varphi_j = p_j / c_j^{1/2} \\ c_j = \langle p_j, p_j \rangle \\ b_0 = c_0, \quad c_j = b_j c_{j-1} \end{array}\right)$$

$$\varphi_0 = c_0^{-1/2}$$
$$\sqrt{b_1}\,\varphi_1 = (x-a_0)\varphi_0$$
$$\sqrt{b_{j+1}}\,\varphi_{j+1} = (x-a_j)\varphi_j - \sqrt{b_j}\,\varphi_{j-1}$$

$$\underset{\substack{\uparrow \\ \text{columns are} \\ \text{functions of } x}}{(\varphi_0\ \varphi_1\ \cdots\ \varphi_n)} \underbrace{\begin{pmatrix} a_0 & \sqrt{b_1} & & \\ \sqrt{b_1} & a_1 & \ddots & \\ & \ddots & \ddots & \sqrt{b_{n-1}} \\ & & \sqrt{b_{n-1}} & a_{n-1} \\ & & & \sqrt{b_n} \end{pmatrix}}_{(n+1)\times n \text{ matrix}} = x(\varphi_0\ \varphi_1\ \cdots\ \varphi_{n-1})$$

evaluate at the zeros of $\varphi_n(x)$:

Jacobi matrix

$$(\hat\varphi_0\ \hat\varphi_1\ \cdots\ \hat\varphi_{n-1}) \begin{pmatrix} a_0 & \sqrt{b_1} & & \\ \sqrt{b_1} & a_1 & \ddots & \\ & \ddots & \ddots & \sqrt{b_{n-1}} \\ & & \sqrt{b_{n-1}} & a_{n-1} \end{pmatrix} = \begin{pmatrix} x_1 & & \\ & \ddots & \\ & & x_n \end{pmatrix} (\hat\varphi_0\ \cdots\ \hat\varphi_{n-1})$$

$$\hat\Phi A = X\hat\Phi, \quad \hat\Phi_{ij} = (\hat\varphi_j)_i = \varphi_j(x_i) \qquad \begin{array}{l} 0 \le j \le n-1 \\ 1 \le i \le n \end{array}$$

rows of $\hat{\Phi}$ are left eigenvectors of $A$

$\therefore \exists\, \Omega$ s.t. $Q = \Omega^{1/2}\hat{\Phi}$ orthogonal matrix, $\quad QA = XQ$

$$\Omega = \begin{pmatrix} w_1 & & \\ & \ddots & \\ & & w_n \end{pmatrix}, \quad w_i = \left( \sum_{j=0}^{n-1} \varphi_j(x_i)^2 \right)^{-1}$$

commutative diagram of isometries

$$\begin{tikzcd}
 & V = \mathbb{P}_{n-1} \arrow[d, "\text{eval}"] \\
\vec{\alpha} \in \mathbb{C}^n \arrow[ur, "\Phi"] \arrow[r, "\hat{\Phi}"] \arrow[dr, "Q"'] & V_1 \arrow[d, "\Omega^{1/2}"] \\
 & \mathbb{C}^n
\end{tikzcd}$$

$$V: \begin{cases} u(x) = \sum_{j=0}^{n-1} \alpha_j \varphi_j(x) \\ \langle v, u \rangle = \int_a^b \overline{v(x)}\, u(x)\, w(x)\, dx \end{cases}$$

$$V_1: \begin{cases} \hat{u}_i = u(x_i) \\ \langle \hat{v}, \hat{u} \rangle_1 = \sum_{i=1}^{n} \overline{\hat{v}_i}\, \hat{u}_i\, w_i = \bar{\hat{v}}^T \Omega \hat{u} \end{cases}$$

$\text{eval} = \hat{\Phi} \circ \Phi^{-1}$ is an isometry from $V_0$ to $V_1$

$$u(x) \mapsto \hat{u} = \begin{pmatrix} u(x_1) \\ \vdots \\ u(x_n) \end{pmatrix} \begin{matrix} = \hat{u}_1 \\ \\ = \hat{u}_n \end{matrix} \qquad V_1 = \ell^2(\{1, \dots, n\}, \Omega)$$

quadrature weights

Alternatively, define the $w_i$ as Gaussian quadrature weights and derive isometry properties

$$\int \overline{u(x)}\, v(x)\, w(x)\, dx = \sum_{i=1}^{n} \overline{u(x_i)}\, v(x_i)\, w_i = \bar{\hat{u}}^T \Omega \hat{v}$$

$\deg uv \le 2n-2 \le 2n-1$

$$\delta_{kj} = \int_a^b \varphi_k \varphi_j\, w\, dx = \sum_{i=1}^{n} \varphi_k(x_i)\varphi_j(x_i)\, w_i = \hat{\varphi}_k^T \Omega \hat{\varphi}_j$$

or $\hat{\Phi}^T \Omega \hat{\Phi} = I$

O.P.'s are particularly useful for representing numerical solutions on unbounded domains

Often the weight function decays rapidly.

<u>examples:</u>

| | |
|---|---|
| generalized Laguerre | $a=0,\ b=\infty,\ w(x)=x^{\alpha}e^{-x},\ \alpha>-1$ |
| Hermite | $a=-\infty,\ b=\infty,\ w(x)=e^{-x^2}$ |
| Maxwell | $a=0,\ b=\infty,\ w(x)=x^2e^{-x^2}$ |

useful to define $\tilde{\varphi}_j(x) = \varphi_j(x)\,w(x)^{1/2}$, $\tilde{u}(x) = u(x)\,w(x)^{1/2}$

not polynomials

orthogonal in $L^2(dx)$ instead of $L^2(w\,dx)$

uniform oscillations instead of rapid growth

can add a space to the commutative diagram

$$\begin{tikzcd}
 & V=\mathbb{P}_{n-1} \arrow[d, "\text{eval}"] \\
\vec{\alpha}\in\mathbb{C}^n \arrow[ur, "\Phi"] \arrow[r, "\bar{\bar{\Phi}}"] \arrow[dr, "\tilde{\Phi}"'] \arrow[ddr, "Q"'] & V_1 \arrow[d, "W^{1/2}"] \arrow[dd, bend left=60, "\Omega^{1/2}"] \\
 & V_2 \arrow[d, "\tilde{\Omega}^{1/2}"] \\
 & \mathbb{C}^n
\end{tikzcd}$$

$\Phi = (\varphi_0(x),\dots,\varphi_{n-1}(x))$

$W = \mathrm{diag}(w(x_i))$

$\tilde{\omega}_i = \dfrac{\omega_i}{w(x_i)}$

$\tilde{\Phi}_{ij} = \tilde{\varphi}_j(x_i)$

often a ratio of tiny numbers

$$\langle \tilde{u},\tilde{v}\rangle_2 = \sum_{i=1}^{n} \bar{\tilde{u}}_i\,\tilde{v}_i\,\tilde{\omega}_i$$

"case 1"

important spaces here are $V$ and $V_2$
(or $V$ and $V_1$ if $w(x)$ does not decay rapidly)

on the grid of zeros of $\varphi_n(x)$, store

$$\tilde{u}_i = u(x_i)\, w(x_i)^{1/2} = \sum_j \alpha_j \tilde{\varphi}_j(x_i) = \tilde{\Phi}\vec{\alpha}$$

nodal representation (or store $\hat{u}$ on the grid in case 1) modal

(often use both representations in a calculation)

how to compute nodes and weights:

omit in class

given $a_j, b_j$ (often from Rodriguez formula but sometimes have to use Stieltjes procedure to compute them)

compute

$$\{x_i\}_{i=1}^n = \operatorname{eig}(A), \quad A = \begin{pmatrix} a_0 & \sqrt{b_1} & & \\ \sqrt{b_1} & a_1 & \ddots & \\ & \ddots & \ddots & \sqrt{b_n} \\ & & \sqrt{b_n} & a_n \end{pmatrix}$$

then compute $\tilde{\Phi}_{ij} = \tilde{\varphi}_j(x_i)$ via the recurrence for $j = 0, \dots, n-1$

compute $\tilde{\omega}_i$ and $Q$ from $Q = \tilde{\Omega}^{1/2}\tilde{\Phi}$

$$\left( \tilde{\omega}_i = \Big(\sum_{j=0}^{n-1} \tilde{\varphi}_j(x_i)^2\Big)^{-1} \right)$$

store $\tilde{\Phi}^{-1} = Q^T \tilde{\Omega}^{1/2}$

may need to carry an extra integer to extend the floating point exponent range in ⊛

This is a variant of the Golub-Welsch algorithm.

They compute $X, Q$ by solving the eigenvalue problem $AQ^T = Q^T X$

since $Q = \Omega^{1/2}\hat{\Phi}$ up to signs of rows

and $\hat{\Phi}_{i0} = c_0^{-1/2}$, $\quad w_i = c_0 Q_{i0}^2$

This works well in case 1 but if the weight function decays rapidly, the $w_i$ often underflow (the small entries of $Q_{i0}$ can be inaccurate in some eigenvalue solvers, too.)

---

Application: Galerkin and pseudo-spectral methods

goal: solve $u_t = -Lu$ on $[a,b)$ in $H = L^2(w\,dx)$

$$Lu = \frac{-(pu')' + qu}{w} \qquad \text{Sturm-Liouville operator}$$

example: energy diffusion in a plasma

$a=0,\ b=\infty,\ u(x) = f(x)e^{x^2}$

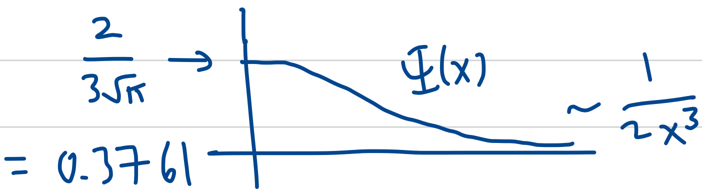
$\left\{ \begin{array}{l} 4\pi x^2 f(x)\,dx = \\ \text{number of particles} \\ \text{moving with speed} \\ \text{between } x \text{ and } x+dx \end{array} \right.$

$$\frac{\partial u}{\partial t} = \frac{1}{w}\left(\Psi w u'\right)', \qquad ' = \frac{d}{dx}$$

$$w = x^2 e^{-x^2} \longleftarrow \text{Maxwell weight function}$$

$$\Psi(x) = \frac{1}{2x^3}\left[\operatorname{erf}(x) - x\operatorname{erf}'(x)\right], \qquad \operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-s^2}\,ds$$

$x^{-1}\cdot$ Chandrasekhar fcn = drag force due to collisions with Maxwellian background distribution




weak form: $u_t = -Lu$

$$\langle v, u_t\rangle = \langle v, -Lu\rangle = \int v \frac{(\Psi w u')'}{w} w\,dx$$

$$= \int -\Psi u' v' w\,dx = -\langle v', \Psi u'\rangle$$

projected dynamics: find $u(t)$ s.t. this holds for $v \in V$

fancy way to write this:

$$u_t = -\left(i\frac{d}{dx}\right)^* \Psi \frac{d}{dx} u = -\left(\frac{d}{dx}\right)^* (P\Psi)\frac{d}{dx} u$$

$\frac{d}{dx} : V \to V$, $i : V \to H$ = inclusion map, $*$ = adjoint

$P = i^* =$ orthogonal projection onto $V$

$$\left( \begin{array}{l} A: X \to Y, \quad A^*: Y \to X, \quad \langle A^* y, x \rangle = \langle y, Ax \rangle \quad \forall\, x \in X,\ y \in Y \\ \langle i^* y, x \rangle_V = \langle y, ix \rangle_H = \langle y, x \rangle_H = \langle P_y, x \rangle_H = \langle P_y, x \rangle_V \end{array} \right)$$

$$\begin{aligned} \Psi &: V \to H \\ & f \mapsto \Psi f \end{aligned} \qquad \text{multiplication by } \Psi$$

pseudo-spectral approach: replace $P\Psi$ by "multiplication by $\Psi$ on the grid"

($P$ becomes an interpolation operator instead of an orthogonal projection)

What does $\frac{d}{dx}$ look like in $V_1$ and $V_2$?

$$\begin{tikzcd} V \arrow[r, "d/dx"] \arrow[d, "E"'] & V \arrow[d, "E = \text{eval}"] \\ V_1 \arrow[r, "D"'] \arrow[d, "W^{1/2}"'] & V_1 \arrow[d, "W^{1/2}"] \\ V_2 \arrow[r, "\tilde{D}"'] & V_2 \end{tikzcd}$$

spectral differentiation matrix:

$$D = E \frac{d}{dx} E^*$$

$$(E^* = E^{-1})$$

$$D_{ij} = \ell_j'(x_i) = \begin{cases} (x_i - x_j)^{-1} \chi_i / \chi_j & i \neq j \\ \sum_{k \neq j} (x_j - x_k)^{-1} & i = j \end{cases}$$

$$\ell_j(x) = \prod_{k \neq j} \frac{x - x_k}{x_j - x_k} \qquad \chi_j = \prod_{k \neq j} (x_j - x_k)$$

(Lagrange polynomials)

$D^* = \Omega^{-1} D^T \Omega$ , adjoint in $V_1$

$\langle \hat{u}, \hat{v} \rangle_1 = \bar{\hat{u}}^T \Omega \hat{v}$

$\tilde{D} = W^{1/2} D W^{-1/2}$ , $\tilde{D}^* = \tilde{\Omega}^{-1} \tilde{D}^T \tilde{\Omega}$

$$\tilde{D}_{ij} = w(x_i)^{1/2} D_{ij} w(x_j)^{-1/2} = \begin{cases} (x_i - x_j)^{-1} \tilde{x}_i / \tilde{x}_j & i \neq j \\ \sum_{k \neq j} (x_j - x_k)^{-1} & i = j \end{cases}$$

$$\tilde{x}_j = w(x_j)^{1/2} \prod_{k \neq j} (x_j - x_k)$$

$$u_t = -\left(\frac{d}{dx}\right)^* (p\Psi) \frac{d}{dx} u \implies \bar{u}_t =$$

$$\tilde{u}_t = -\tilde{D}^* \Psi \tilde{D} \tilde{u} = -\tilde{\Omega}^{-1} \tilde{D}^T \tilde{\Omega} \Psi \tilde{D} \tilde{u}$$

$$= \underbrace{-\tilde{\Omega}^{-1/2} \tilde{J}^T \tilde{J} \tilde{\Omega}^{1/2}}_{\tilde{L}} \tilde{u}$$

(SVD)

$$\tilde{J} = \Psi^{1/2} \tilde{\Omega}^{1/2} \tilde{D} \tilde{\Omega}^{-1/2} = U S V^T$$

$$\tilde{J}^T \tilde{J} = V S^2 V^T$$

$Z = \tilde{\Omega}^{-1/2} V$ evecs

$\tilde{L} Z = Z S^2$ , $S^2 = \begin{pmatrix} s_1^2 & & \\ & \ddots & \\ & & s_n^2 \end{pmatrix}$ evals solution is $\tilde{u} = \underbrace{Z e^{-S^2 t} Z^{-1}}_{e^{-\tilde{L}t}} \underbrace{\tilde{u}_0}_{\text{initial condition}}$

note $Z^{-1}\tilde{u}_0 = V^T \tilde{\Omega}^{-1/2} \tilde{u}_0 = \underbrace{(\tilde{\Omega}^{-1/2} V)^T \Omega \tilde{u}_0}$

inner products of eigenvectors with initial condition give mode amplitudes

Computing the SVD of $J$ avoids "squaring the condition number" (only one $D$ involved)