Numerical Analysis Preliminary Exam

10 AM TO 1 PM, AUGUST 20, 2018

INSTRUCTIONS. You have three hours to complete this exam. Submit solutions to four (and no more) of the following six problems. Please start each problem on a new page. You MUST prove your conclusions or show a counter-example for all problems unless otherwise noted. **Do not write your name on your exam.** Write your student ID number.

Problem 1: Rootfinding

Iterative sequences very reminiscent of those arising when using Newtons method for root finding (either for a single equation or for a system of equations) can arise also in a number of other contexts. For exanple, if one starts with

$$a_0 = \sqrt{2} - 1, \quad b_0 = 6 - 4\sqrt{2}$$

and then iterates for $k = 0, 1, 2, \ldots$

$$a_{k+1} = \frac{1 - (1 - a_k^4)^{1/4}}{1 + (1 - a_k^4)^{1/4}}, \quad b_{k+1} = b_k(1 + a_{k+1})^4 - 2^{2k+3} a_{k+1}(1 + a_{k+1} + a_{k+1}^2)$$

it will turn out that $a_k \to 0$ and $b_k \to 1/\pi$.

(a) In the same sense as we describe typical Newton iteration convergence as quadratic, determine the convergence rate of each of the two sequences above.

(b) Give a rough estimate of how high we need k to be if we want the approximation for π to become correct to over 1,000 decimal places.

Problem 2: Interpolation & Approximation

(a) Let the entries of $\mathbf{x} = [x_0, x_1, \ldots, x_{N-1}]^T$ be N discrete samples of a continuous function f, observed at timeponts $t_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$. What is the connection between the trigonometric interpolation of f at (t_k, x_k), $k = 0, 1, \ldots, N-1$, and the discrete Fourier transform $\mathbf{X} = F_N \mathbf{x}$, where $F_N = [f_{pq}]$ is the Vandermonde matrix with

$$\begin{aligned} f_{pq} &= \omega_N^{pq} \\ \omega_N &= e^{-2\pi i/N}. \end{aligned}$$

(b) Denote the DFT operator as $\mathcal{F}$, the inverse DFT operator as $\mathcal{F}^{-1}$, and a time series of data as $\boldsymbol{x}$. Assuming the existence of a software that efficiently computes a DFT, mathematically explain a way that this code can be used to efficiently compute an inverse DFT. In other words, how can you compute $\mathcal{F}^{-1}(\boldsymbol{x})$ using only the code that computes $\mathcal{F}$ and the data $\boldsymbol{x}$?

Problem 3: Quadrature

(a) Explain how to find weights w_i and nodes x_i such that the quadrature $\sum_{i=0}^{n} w_i f(x_i)$ gives the exact solution to $\int_0^\infty e^{-x} f(x) \mathrm{d}x$ whenever f is a polynomial of degree $\leq n$.

(b) Find nodes x_0 and x_1 and weights w_0 and w_1 such that the quadrature $\sum_{i=0}^{n} w_i f(x_i)$ gives the exact solution to $\int_0^\infty e^{-x} f(x)\mathrm{d}x$ whenever f is a polynomial of degree ≤ 1.

(c) Formulate a convergent quadrature for the integral $\int_0^1 e^x/\sqrt{x}\mathrm{d}x$.

(d) Let $I[f] = \int_a^b f(x)\mathrm{d}x$, and let $I_n[f] = \sum_{i=0}^{n} w_i f(x_i)$. Prove that if I_n integrates polynomials up to degree n exactly and the weights w_i are all positive then the quadrature is convergent for any $f \in C([a,b])$, i.e. $\lim_{n\to\infty} I_n[f] = I[f]$.

Problem 4: Numerical Linear Algebra

(a) Prove that the Gauss-Jacobi iteration is convergent whenever the coefficient matrix is strictly diagonally dominant.

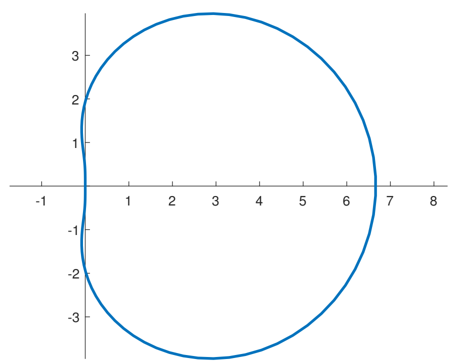
(b) Formulate the Modified Gram-Schmidt algorithm to produce an orthogonal basis for the range of a matrix $\mathbf{A}$. You may assume that $\mathbf{A}$ has full column rank.

(c) Suppose that $\bar{\lambda}$ is a very good approximation (but not exact) to a simple eigenvalue λ of the matrix $\mathbf{A}$. Formulate an iterative method that will obtain a good approximation to the associated eigenvector after a very small number of iterations.

Problem 5: ODEs

The following matlab code produces the following figure

```
r = exp(complex(0,linspace(0,2*pi)));
xi = (11/6*r.^3-3*r.^2+3/2*r-1/3)./r.^3;
plot(xi,'LineWidth',2)
axis equal; ax = gca; box off;
ax.XAxisLocation = 'origin';
ax.YAxisLocation = 'origin';
```



This figure displays the boundary of the stability domain for a certain consistent linear multistep method (LMM).

(a) Write down the formula for this LMM in the conventional form of coefficients for $y(t)$ and $y'(t) = f(t)$ at a sequence of equispaced t levels. Does this scheme go under a well-known name?

(b) Determine if the stability domain is given by the inside or the outside (or neither) of the shown curve.

(c) Determine if the scheme satisfies the root condition.

Problem 6: PDEs

Consider the partial differential equation defined in the (t, x)-domain

$$\begin{aligned} u_{tt} &= 2u_{xx} \\ -1 \leq\ & x \leq 1 \\ 0 <\ & t \end{aligned}$$

with boundary and initial conditions

$$\begin{aligned} u(t,-1) = u(t,1) = 0 &\quad ; \quad t > 0 \\ u(0,x) = e^{-x^2} - e^{-1} &\quad ; \quad -1 \leq x \leq 1 \\ u_t(0,x) = (x+1)(x-1) &\quad ; \quad -1 \leq x \leq 1 \end{aligned}$$

and spatial discretization with stepsize h and time discretization with stepsize k.

(a) Create an explicit $\mathcal{O}(h^2 + k^2)$ finite difference approximation to the solution.

(b) How would one accurately compute the solution at the first timepoint $t_1 = k$?

(c) How would one choose the sizes of h and k sufficient to maintain stability?