# Lecture 17

## Conditioning and Interpolation

L. Olson

Department of Computer Science
University of Illinois at Urbana-Champaign

Slides based on slides from Numerical Methods with Matlab by Gerald Recktenwald

March 14, 2006

# Today:

## Objectives

- finish conditioning
- interpolation with monomial basis functions
- interpolation with Lagrange basis functions

## Material

- wrap-up sections 8.4
- Section 10.2

# Condition number of $A$

The **condition number**

$$\kappa(A) \equiv \|A\|\|A^{-1}\|$$

indicates the sensitivity of the solution to perturbations in $A$ and $b$. The condition number can be measured with any $p$-norm.
The condition number is always in the range

$$1 \leqslant \kappa(A) \leqslant \infty$$

- *$\kappa(A)$ is a mathematical property of $A$*
- *Any algorithm will produce a solution that is sensitive to perturbations in $A$ and $b$ if $\kappa(A)$ is large.*
- *In exact math a matrix is either singular or non-singular. $\kappa(A) = \infty$ for a singular matrix*
- *$\kappa(A)$ indicates how close $A$ is to being numerically singular.*
- *A matrix with large $\kappa$ is said to be **ill-conditioned***

# The Residual

Let $\hat{x}$ be the numerical solution to $Ax = b$. $\hat{x} \neq x$ ($x$ is the exact solution) because of roundoff.

The **residual** measures how close $\hat{x}$ is to satisfying the original equation

$$r = b - A\hat{x}$$

It is not hard to show that

$$\frac{\|\hat{x} - x\|}{\|\hat{x}\|} \leqslant \kappa(A) \frac{\|r\|}{\|b\|}$$

Small $\|r\|$ does not guarantee a small $\|\hat{x} - x\|$.

If $\kappa(A)$ is large the $\hat{x}$ returned by Gaussian elimination and back substitution (or any other solution method) is not guaranteed to be anywhere near the true solution to $Ax = b$.

# Rules of Thumb (1)

- Applying Gaussian elimination with partial pivoting and back substitution to $Ax = b$ yields a numerical solution $\hat{x}$ such that the residual vector $r = b - A\hat{x}$ is small *even if* the $\kappa(A)$ is large.
- If $A$ and $b$ are stored to machine precision $\varepsilon_m$, the numerical solution to $Ax = b$ by any variant of Gaussian elimination is correct to $d$ digits where

$$d = |\log_{10}(\varepsilon_m)| - \log_{10}(\kappa(A))$$

# Rules of Thumb (2)

$$d = |\log_{10}(\varepsilon_m)| - \log_{10}(\kappa(A))$$

**Example:**
MATLAB computations have $\varepsilon_m \approx 2.2 \times 10^{-16}$. For a system with $\kappa(A) \sim 10^{10}$ the elements of the solution vector will have

$$\begin{aligned} d &= |\log_{10}(2.2 \times 10^{-16})| - \log_{10}\left(10^{10}\right) \\ &= 16 - 10 \\ &= 6 \end{aligned}$$

correct digits

# Summary of Limits to Numerical Solution of $Ax = b$

1. $\kappa(A)$ indicates how close $A$ is to being numerically singular
2. If $\kappa(A)$ is “large”, $A$ is **ill-conditioned** and *even the best* numerical algorithms will produce a solution, $\hat{x}$ that cannot be guaranteed to be close to the true solution, $x$
3. In practice, Gaussian elimination with partial pivoting and back substitution produces a solution with a small residual

$$r = b - A\hat{x}$$

*even if* $\kappa(A)$ is large.

# Linear Algebra

**read!**

- Section 8.4
- LU factorization: stores pieces in the Gaussian elimination process to produce lower triangular $L$ and upper triangular $U$ so that $A = LU$
- Cholesky factorization: similar, but for symmetric positive definite matrices

# Interpolation

- Given $x_i = x_1, \ldots, x_n$ and data $y_i = y_1, \ldots, y_n$, find a polynomial $p(x)$ so that $p(x_i) = y_i$ for all $i = 1, \ldots, n$.
- try picking

$$p_{n-1}(x) = c_1 x^{n-1} + c_2 x^{n-2} + \cdots + c_{n-1} x + c_n$$

So for each $x_i$ we have

$$y_i = p_{n-1}(x_i) = c_1 x^{n-1} + c_2 x^{n-2} + \cdots + c_{n-1} x + c_n$$

OR

$$\begin{aligned} c_1 x_1^{n-1} + c_2 x_1^{n-2} + \cdots + c_{n-1} x_1 + c_n &= y_1 \\ c_1 x_2^{n-1} + c_2 x_2^{n-2} + \cdots + c_{n-1} x_2 + c_n &= y_2 \\ &\vdots \\ c_1 x_n^{n-1} + c_2 x_n^{n-2} + \cdots + c_{n-1} x_n + c_n &= y_n \end{aligned}$$