

Numerical Linear Algebra for Computational Science and Information Engineering Floating-Point Arithmetic and Error Analysis

Nicolas Venkovic
nicolas.venkovic@tum.de

Chair of Computational Mathematics
School of Computation, Information and Technology
Technical University of Munich



Outline I

- | | | |
|---|---|----|
| 1 | Number representation and arithmetic on digital computers
Section 3.2 in Darve & Wootters (2021) | 1 |
| 2 | Principles of error analysis
Section 3.3 in Darve & Wootters (2021) | 7 |
| 3 | Analysis of linear systems
Section 3.3 in Darve & Wootters (2021) | 12 |
| 4 | Analysis of eigenvalue problems | 17 |

Number representation and arithmetic on digital computers

Section 3.2 in Darve & Wootters (2021)

Number representation on computers

- ▶ Computers store numbers with **finite memory**, leading to limitations:
 - **Representation errors**: Most real numbers cannot be exactly represented.
 - **Rounding errors**: Arithmetic operations result in quantities which cannot be exactly represented either.
 - **Overflow/Underflow**: Numbers may exceed their representable range.
- ▶ These limitations introduce challenges in numerical computations, such as maintaining
 - **Accuracy**: How close is the computed result to the true value?
Affected by accumulation of representation and rounding errors, and by algorithmic choices.
 - **Stability**: Does the method prevent error growth for small input changes?
Specific to both the problem and the algorithm together.
- ▶ **Error analysis** helps understand these challenges by focusing on
 - **Perturbation**: effect of small input changes on the true solution of a problem.
 - **Propagation**: cumulative effects of rounding errors through calculations.
- ▶ Understanding these concepts is essential to prevent unwanted behaviors when using numerical methods.

Bit representation of integers

- ▶ Digital computers represent integers using a fixed number b of bits, e.g., 8, 16, 32, or 64 bits.
- ▶ For every **unsigned integers** x ranging from 0 to $2^b - 1$,

$$\exists! (d_0, \dots, d_{b-1}) \in \{0, 1\}^b \text{ s.t. } x = \sum_{i=0}^{b-1} d_i 2^i.$$

We say that x is represented as $d_{b-1} \dots d_0$.

Attempting to represent integers out of the range from 0 to $2^b - 1$, leads to **underflow** or **overflow**.

Example: integers from 0 to 7 can be represented as follows using 3 bits:

integer	binary representation	decomposition
0	000	$0 \times 1 + 0 \times 2 + 0 \times 4$
1	001	$1 \times 1 + 0 \times 2 + 0 \times 4$
2	010	$0 \times 1 + 1 \times 2 + 0 \times 4$
3	011	$1 \times 1 + 1 \times 2 + 0 \times 4$
4	100	$0 \times 1 + 0 \times 2 + 1 \times 4$
5	101	$1 \times 1 + 0 \times 2 + 1 \times 4$
6	110	$0 \times 1 + 1 \times 2 + 1 \times 4$
7	111	$1 \times 1 + 1 \times 2 + 1 \times 4$

Bit representation of integers, cont'd

- Different systems exist in order to encode **signed integers** with bits. In particular, we consider the **two's complement representation**:

For every integer x ranging from -2^{b-1} to 2^{b-1} ,

$$\exists! (d_0, \dots, d_{b-2}, d_{b-1}) \in \{0, 1\}^b \text{ s.t. } x = -d_{b-1}2^{b-1} + \sum_{i=0}^{b-2} d_i 2^i.$$

Example: integers from -4 to 3 can be represented as follows using 3 bits:

integer	binary representation	decomposition
0	000	$-0 \times 4 + 0 \times 1 + 0 \times 2$
1	001	$-0 \times 4 + 1 \times 1 + 0 \times 2$
2	010	$-0 \times 4 + 0 \times 1 + 1 \times 2$
3	011	$-0 \times 4 + 1 \times 1 + 1 \times 2$
-4	100	$-1 \times 4 + 0 \times 1 + 0 \times 2$
-3	101	$-1 \times 4 + 1 \times 1 + 0 \times 2$
-2	110	$-1 \times 4 + 0 \times 1 + 1 \times 2$
-1	111	$-1 \times 4 + 1 \times 1 + 1 \times 2$

Clearly, the most significant bit d_{b-1} represents the sign (0 for +, 1 for -). Arithmetic operations on two's complement numbers follow the same rules as unsigned arithmetic.

Bit representation of floating-point numbers

- ▶ **Floating-point numbers** are used to represent a wide range of **real numbers**, including fractions and very large or small numbers.
- ▶ A **floating-point number** x is given by $x = (-1)^s \times m \times 2^{e-2^{b-p-1}}$ where
 - s is the **sign bit** (0 for +, 1 for -).
 - $m = 1 + \sum_{i=1}^{p-1} q_i 2^{-i} \in [1, 2)$ is the **significand** (or **mantissa**), encoded by $p-1$ **fraction bits**, where p is the **precision** of the numerical system.
 - $e - 2^{b-p-1}$ is the **exponent** represented by $b-p$ bits with $e = \sum_{i=0}^{b-p-1} d_i 2^i$.

The associated bits are stored in the form

s	$d_{b-p-1} \dots d_0$	$q_1 \dots q_{p-1}$
-----	-----------------------	---------------------

.

- ▶ Example: Half precision (1 sign bit, 5 exponent bits, 10 fraction bits)
 - Then, the floating-point number $\text{fl}(\pi)$, which best approximates $\pi = 3.1416\dots$, is represented as **0100001001001000** so that

$$\begin{aligned} s &= 0 & e &= 1 \times 2^4 & m &= 1 + 1 \times 2^{-1} + 1 \times 2^{-4} + 1 \times 2^{-7} \\ & & &= 16 & &= 1 + 0.5 + 0.0625 + 0.0078125 \\ & & & & &= 1.5703125 \end{aligned}$$

$$\text{and } \text{fl}(\pi) = (-1)^0 \times 1.5703125 \times 2^{16-2^{4-1}} = 1.5703125 \times 2 = 3.140625.$$

Bit representation of floating-point numbers, cont'd

- ▶ Most real numbers cannot be exactly represented due to the finite number of bits used for the mantissa. The **machine epsilon** and the **unit roundoff** are used to characterize the rounding error of a numerical system.

Definition (Machine epsilon & unit roundoff)

- The (interval) **machine epsilon**, often denoted by ϵ_{mach} , is the distance between 1 and the next floating-point number.
 - The **unit roundoff** u is half the machine precision, i.e., $u = \epsilon_{mach}/2$.
-
- ▶ Common floating-point formats:
 - **Half precision** (16 bits): 1 sign bit, 5 exponent bits, 10 significand bits and unit roundoff $u = 2^{-11} \approx 4.88 \times 10^{-4}$.
 - **Single precision** (32 bits): 1 sign bit, 8 exponent bits, 23 significand bits and unit roundoff $u = 2^{-24} \approx 5.96 \times 10^{-8}$.
 - **Double precision** (64 bits): 1 sign bit, 11 exponent bits, 52 significand bits and unit roundoff $u = 2^{-53} \approx 1.11 \times 10^{-16}$.
 - ▶ The **distribution** of floating-point numbers is **not uniform** within the range of a numerical system.

Floating-point conversion and arithmetic

- ▶ For every number x within the range of a floating-point number system, it can be shown that the associated rounding $\text{fl}(x)$ is such that

$$\text{fl}(x) = (1 + \delta)x \text{ for some } \delta \text{ s.t. } |\delta| \leq u.$$

- ▶ When performing arithmetic operations between floating-point numbers, i.e., $\text{fl}(x) \circ \text{fl}(y)$ with $\circ \in \{+, -, \times, \div\}$, the result is not necessarily a floating-point number. Further rounding applies.

Floating-point number systems follow the **standard model of arithmetic**, which states they must satisfy

$$\text{fl}(\text{fl}(x) \circ \text{fl}(y)) = (1 + \delta)(\text{fl}(x) \circ \text{fl}(y)) \text{ for some } \delta \text{ s.t. } |\delta| \leq u.$$

- ▶ Properties of floating-point arithmetic:

- **Not associative**, e.g., $\text{fl}(\text{fl}(x) + \text{fl}(y)) + \text{fl}(z) \neq \text{fl}(x) + \text{fl}(\text{fl}(y) + \text{fl}(z))$.
- **Not distributive**, e.g.,

$$\text{fl}(x) \times \text{fl}(\text{fl}(y) + \text{fl}(z)) \neq \text{fl}(\text{fl}(x) \times \text{fl}(y)) + \text{fl}(\text{fl}(x) \times \text{fl}(z)).$$

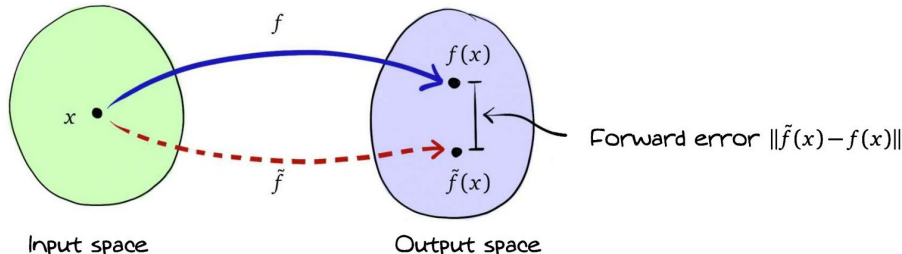
- Subtraction of nearly equal numbers can lead to **catastrophic cancellation**.

Principles of error analysis

Section 3.3 in Darve & Wootters (2021)

Forward error

- ▶ **Error analysis** is crucial for understanding the accuracy and stability of numerical algorithms.
- ▶ Let f be a function and $\tilde{f}$ be its computed approximation for an input x .
- ▶ The **forward error** $\|f(x) - \tilde{f}(x)\|$ measures the **distance between the true value $f(x)$ and the computed approximation $\tilde{f}(x)$** .



Darve, E., & Wootters, M. (2021). Numerical linear algebra with Julia. Society for Industrial and Applied Mathematics.

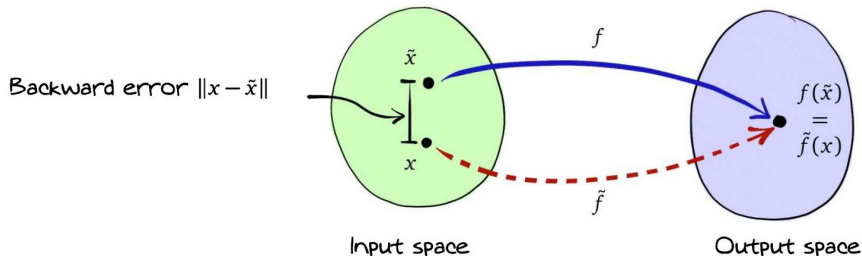
- ▶ The **relative forward error** is given by $\|f(x) - \tilde{f}(x)\| / \|f(x)\|$.
- ▶ In practice, we often do not know $f(x)$, which makes the forward error **difficult to evaluate**.

Backward error

- For an approximation $y := \tilde{f}(x)$ of a true quantity $f(x)$ for some input x , the **backward error** $\eta(x, y)$ is the **smallest perturbation to the input** whose exact map equates the approximation, i.e.,

$$\eta(x, y) = \min_{\tilde{x}} \{ \|x - \tilde{x}\| \text{ s.t. } f(\tilde{x}) = y \}.$$

This can be represented as



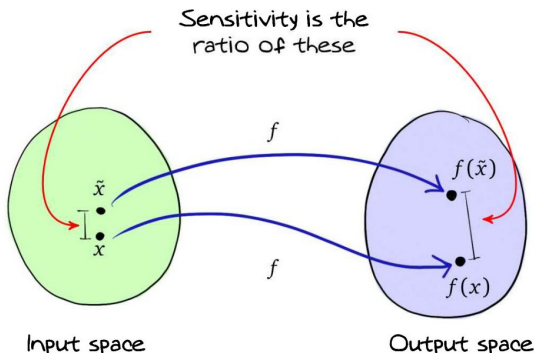
Darve, E., & Wootters, M. (2021). Numerical linear algebra with Julia. Society for Industrial and Applied Mathematics.

- The **relative backward error** is given by $\eta(x, y)/\|x\|$.

Sensitivity of a problem

- **Sensitivity** measures how much the output of a function changes relative to small changes in the input:

$$\text{sensitivity} = \frac{\text{forward error}}{\text{backward error}} = \frac{\|f(x) - f(\tilde{x})\|}{\|x - \tilde{x}\|}.$$



Darve, E., & Wootters, M. (2021). Numerical linear algebra with Julia. Society for Industrial and Applied Mathematics.

- The **relative sensitivity** is given by $\frac{\|f(x) - f(\tilde{x})\| / \|f(x)\|}{\|x - \tilde{x}\| / \|x\|}.$

Conditioning of a problem

- ▶ The (relative) **condition number** $\kappa(x)$ of a problem $x \mapsto f(x)$ bounds the **relative sensitivity for small perturbations in the input data**:

$$\kappa(x) = \lim_{\varepsilon \rightarrow 0} \sup_{\|\delta x\| \leq \varepsilon} \frac{\|f(x + \delta x) - f(x)\| / \|f(x)\|}{\|\delta x\| / \|x\|}.$$

- ▶ A fundamental result of numerical analysis states

$$\text{relative forward error} \lesssim \text{condition number} \times \text{relative backward error}$$

also written as $\frac{\|f(x) - y\|}{\|f(x)\|} \lesssim \kappa(x) \frac{\eta(x, y)}{\|x\|}$ for any approximation y of $f(x)$.

- ▶ A problem $x \mapsto f(x)$ with a large condition number $\kappa(x)$ is **ill-conditioned**
- ▶ The **approximation** $\tilde{f}(x)$ of an **ill-conditioned problem** can have a **large forward error**, even if $\tilde{f}(x)$ has a small backward error.
- ▶ The condition number is **problem-dependent**, i.e., it is specifically defined for linear system solving, least-squares solving, eigenvalue solving, ...
- ▶ The condition number **does not depend on the algorithm**.

Backward stability of an algorithm

- ▶ In practice, we develop algorithms of the form $x \mapsto \tilde{f}(x)$ to approximate the solution of the problem $x \mapsto f(x)$ that minimize the associated backward error $\eta(x, \tilde{f}(x))$.
- ▶ In particular, an algorithm is **backward stable** if the associated backward error remains small, i.e.,

$$\frac{\eta(x, \tilde{f}(x))}{\|x\|} = \mathcal{O}(u)$$

irrespective of x , where u is, typically, the unit roundoff of the floating-point number system.

- ▶ For **well-conditioned problems**, a **backward stable algorithm** ensures **small forward errors**.
- ▶ But, for **ill-conditioned problems**, even **backward stable algorithms** may produce **large forward errors**.

Analysis of linear systems

Section 3.3 in Darve & Wootters (2021)

Perturbation of linear systems

- ▶ Consider the problem of solving for x such that $Ax = b$ for some invertible matrix A and non-zero vector b .
- ▶ Let us assume $\tilde{x} := x + \delta x$ is the true solution of a non-singular perturbed problem $(A + \delta A)\tilde{x} = b + \delta b$. Then, the following remainder is obtained

$$\begin{array}{r} (A + \delta A)(x + \delta x) = b + \delta b \\ - \quad \quad \quad Ax = b \\ \hline A\delta x + \delta Ax + \delta A\delta x = \delta b \end{array}$$

Multiplying the remainder by A^{-1} , we get

$$\delta x + A^{-1}\delta Ax + A^{-1}\delta A\delta x = A^{-1}\delta b.$$

Then, assuming the matrix norm is consistent with the vector norm:

$$\|\delta x\| \leq \|A^{-1}\| \cdot \|\delta A\| \cdot \|x\| + \|A^{-1}\| \cdot \|\delta A\| \cdot \|\delta x\| + \|A^{-1}\| \cdot \|\delta b\|.$$

Dividing by $\|x\|$, and neglecting the 2nd order term $\|\delta A\| \cdot \|\delta x\|$, we get

$$\frac{\|\delta x\|}{\|x\|} \lesssim \|A^{-1}\| \cdot \|\delta A\| + \frac{\|A^{-1}\| \cdot \|\delta b\|}{\|x\|}.$$

Perturbation of linear systems, cont'd

We can then factor by $\|A^{-1}\| \cdot \|A\|$, which leads to

$$\frac{\|\delta x\|}{\|x\|} \lesssim \|A^{-1}\| \cdot \|A\| \cdot \left(\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|A\| \cdot \|x\|} \right).$$

But since $Ax = b$ implies $\|b\| \leq \|A\| \cdot \|x\|$, we obtain

$$\frac{\|\delta x\|}{\|x\|} \lesssim \|A^{-1}\| \cdot \|A\| \cdot \left(\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|b\|} \right).$$

The relative forward error $\|\delta x\|/\|x\|$ is approximately bounded by $\|A^{-1}\| \cdot \|A\|$ times the sum of relative input perturbations $\|\delta A\|/\|A\|$ and $\|\delta b\|/\|b\|$.

- ▶ Therefore, the **condition number** of the linear system solving problem $A \mapsto x := A^{-1}b$ is given by $\kappa(A) = \|A^{-1}\| \cdot \|A\|$.
- ▶ When using the 2-norm, we have $\kappa(A) = \sigma_{\max}(A)/\sigma_{\min}(A)$, in which $\sigma_{\max}(A)$ and $\sigma_{\min}(A)$ are the maximal and minimal singular values of A , respectively.

Backward errors of linear systems

- ▶ Let $\tilde{x}$ be an approximation of the solution x of the linear system $Ax = b$, and define the associate residual $r := b - A\tilde{x}$.
- ▶ Then, we are interested in the backward error $\eta_{A,b}(x)$ defined as
$$\eta_{A,b}(\tilde{x}) = \min\{\varepsilon \text{ s.t. } (A + \delta A)\tilde{x} = b + \delta b, \|\delta A\| \leq \varepsilon\|A\|, \|\delta b\| \leq \varepsilon\|b\|\}.$$
- ▶ To find $\eta_{A,b}(x)$, we first rearrange the perturbed system as follows:

$$\begin{aligned}(A + \delta A)\tilde{x} &= b + \delta b \\ \delta A\tilde{x} &= b - A\tilde{x} + \delta b \\ \delta A\tilde{x} &= r + \delta b.\end{aligned}$$

Then, considering a matrix norm consistent with the vector norm, we have

$$\|\delta A\| \cdot \|\tilde{x}\| \geq \|r + \delta b\| \geq \|r\| - \|\delta b\|.$$

Applying the prescribed bounds $\|\delta A\| \leq \varepsilon\|A\|$ and $\|\delta b\| \leq \varepsilon\|b\|$, we get

$$\varepsilon\|A\| \cdot \|\tilde{x}\| \geq \|r\| - \varepsilon\|b\|$$

Backward errors of linear systems, cont'd₁

which we re-order as

$$\varepsilon \geq \frac{\|r\|}{\|A\| \cdot \|\tilde{x}\| + \|b\|}$$

and whose minimum, i.e., the backward error $\eta_{A,b}(\tilde{x})$, is

$$\eta_{A,b}(\tilde{x}) = \frac{\|r\|}{\|A\| \cdot \|\tilde{x}\| + \|b\|}.$$

When using 2-norms, the bound is attained for

$$\delta A = \frac{\|A\|_2}{\|\tilde{x}\|_2 \cdot (\|A\|_2 \cdot \|\tilde{x}\|_2 + \|b\|_2)} r \tilde{x}^T \quad \text{and} \quad \delta b = -\frac{\|b\|_2}{\|A\|_2 \cdot \|\tilde{x}\|_2 + \|b\|_2} r.$$

- Note that $r \tilde{x}^T$ is a matrix of rank 1, so that the approximate solution $\tilde{x}$ to the linear system $Ax = b$ is the exact solution to a linear system whose matrix is a rank-1 perturbation of A .
- $\eta_{A,b}(\tilde{x})$ is sometimes referred to as the **normwise relative backward error**, so as to be distinguished from other definitions of backward error.

Backward errors of linear systems, cont'd₂

- ▶ In practice, evaluating $\eta_{A,b}(\tilde{x})$ can be challenging due to the need of $\|A\|$.
- ▶ Then, the backward error $\eta_b(\tilde{x})$ is considered, where only b is perturbed:

$$\eta_b(\tilde{x}) = \min\{\varepsilon \text{ s.t. } A\tilde{x} = b + \delta b, \|\delta b\| \leq \varepsilon\|b\|\}.$$

Since we then have $\|\delta b\| = \|A\tilde{x} - b\| = \|r\|$, the backward error is

$$\eta_b(\tilde{x}) = \frac{\|r\|}{\|b\|}.$$

- ▶ Note that $\eta_b(\tilde{x}) \geq \eta_{A,b}(\tilde{x})$ for all A, b and $\tilde{x}$, so that the design of a stopping criteria on the basis of $\eta_b(\tilde{x})$ is **conservative**, and **good practice**.
- ▶ Some implementations of iterative linear solvers monitor the convergence of iterates $x_0, \dots, x_k$ through $\|r_k\|/\|r_0\|$. But, if $x_0 \neq 0$ and $\|r_0\| \gg \|b\|$, we have

$$\eta_b(x_k) = \frac{\|r_k\|}{\|b\|} = \frac{\|r_k\|}{\|r_0\|} \frac{\|r_0\|}{\|b\|}$$

so that, even if $\|r_k\|/\|r_0\| \leq \varepsilon$, we actually have $\eta_b(x_k) \gg \varepsilon$.

Thus, this practice is **not recommended**, especially for ill-conditioned systems with poor non-zero initial guess.

Analysis of eigenvalue problems

Backward error of an eigenpair

- ▶ Let $(\tilde{\lambda}, \tilde{u})$ be an approximation of the eigenpair (λ, u) such that $Au = \lambda u$.
- ▶ Then, the associated normwise backward error $\eta_A(\tilde{\lambda}, \tilde{u})$ is given as

$$\eta_A(\tilde{\lambda}, \tilde{u}) = \min\{\varepsilon \text{ s.t. } (A + \delta A)\tilde{u} = \tilde{\lambda}\tilde{u}, \|\delta A\| \leq \varepsilon\|A\|\}.$$

To find $\eta_A(\tilde{\lambda}, \tilde{u})$, we reorder the perturbed eigenvalue problem as

$$\begin{aligned}(A + \delta A)\tilde{u} &= \tilde{\lambda}\tilde{u} \\ \delta A\tilde{u} &= \tilde{\lambda}\tilde{u} - A\tilde{u}.\end{aligned}$$

Assuming consistent matrix and vector norms, we obtain

$$\varepsilon\|A\|\|\tilde{u}\| \geq \|\delta A\|\|\tilde{u}\| \geq \|\tilde{\lambda}\tilde{u} - A\tilde{u}\|$$

so that $\eta_A(\tilde{\lambda}, \tilde{u}) = \frac{\|r\|}{\|A\| \cdot \|\tilde{u}\|}$, where $r = A\tilde{u} - \tilde{\lambda}\tilde{u}$ is the eigen-residual.

Backward error of an eigenpair, cont'd

- ▶ When using 2-norms, the minimal norm perturbation is achieved with

$$\delta A = -\frac{r\tilde{u}^H}{\|A\|_2 \cdot \|\tilde{u}\|_2^2}$$

which, again, is a rank-1 perturbation.

- ▶ So, computing an approximation $(\tilde{\lambda}, \tilde{u})$ of the eigenpair (λ, u) such that

$$\frac{\|r\|}{\|A\| \cdot \|\tilde{u}\|} \leq \varepsilon$$

for a small value of ε should ensure the good quality approximation, **if the problem is well-conditioned**.

But, **what is the conditioning of solving for an eigenpair** (λ, u) of A ?

- ▶ In practice, convergence is often monitored with the criterion

$$\frac{\|r\|}{|\tilde{\lambda}| \cdot \|\tilde{u}\|} \leq \varepsilon$$

which, for larger eigenvalues of the spectrum, is generally not an issue.

Perturbation of the normal eigenvalue problem

- ▶ Let λ be a simple eigenvalue of a matrix A with normalized right-eigenvector u and left-eigenvector v , i.e.,

$$Au = \lambda u, \quad v^H A = \lambda v^H \quad \text{and} \quad \|u\|_2 = \|v\|_2 = 1.$$

We consider the approximate eigenpair $(\tilde{\lambda}(\varepsilon), \tilde{u}(\varepsilon))$ of A with the linear perturbation $\tilde{A}(\varepsilon) := A + \varepsilon E$ along a matrix E s.t. $(\tilde{\lambda}(0), \tilde{u}(0)) = (\lambda, u)$ and

$$\tilde{A}(\varepsilon)\tilde{u}(\varepsilon) = \tilde{\lambda}(\varepsilon)\tilde{u}(\varepsilon).$$

Multiplying both sides of this expression by the left-eigenvector v^H associated with λ , we obtain:

$$v^H \tilde{A}(\varepsilon)\tilde{u}(\varepsilon) = \tilde{\lambda}(\varepsilon)v^H \tilde{u}(\varepsilon)$$

$$v^H A \tilde{u}(\varepsilon) + \varepsilon v^H E \tilde{u}(\varepsilon) = \tilde{\lambda}(\varepsilon)v^H \tilde{u}(\varepsilon)$$

$$(v^H A = \lambda v^H)$$

$$\lambda v^H \tilde{u}(\varepsilon) + \varepsilon v^H E \tilde{u}(\varepsilon) = \tilde{\lambda}(\varepsilon)v^H \tilde{u}(\varepsilon)$$

so that $(\tilde{\lambda}(\varepsilon) - \lambda)/\varepsilon = v^H E \tilde{u}(\varepsilon)/(v^H \tilde{u}(\varepsilon))$.

Perturbation of the normal eigenvalue problem, cont'd

- ▶ The rate of change in $\tilde{\lambda}$ induced by the linear perturbation of A along E is then given by:

$$\lim_{\varepsilon \rightarrow 0} \frac{\tilde{\lambda}(\varepsilon) - \lambda}{\varepsilon} = \frac{v^H E \tilde{u}(0)}{v^H \tilde{u}(0)} = \frac{v^H E \tilde{u}}{v^H \tilde{u}}.$$

Using the Cauchy-Schwartz inequality, we obtain $|v^H E u| \leq \|E\|_2$ and

$$|\tilde{\lambda} - \lambda| \lesssim \frac{\|E\|_2}{|v^H u|},$$

so that solving for the simple eigenvalue λ of A has conditioning given by $\kappa(A, \lambda) = 1/|v^H u|$.

- ▶ **Normal** (and thus **symmetric**) **matrices** have aligned right- and left-eigenvectors, which implies $\kappa(\lambda, A) = 1$, i.e., **solving for a simple eigenvalue of a normal matrix is a well-conditioned problem**.
- ▶ For general matrices, if u and v are nearly orthogonal, we have $\kappa(\lambda, A) \gg 1$, and **solving for the eigenvalue λ is an ill-conditioned problem**.