

# Lineáris egyenletrendszerek

Kormányos Andor

Komplex Rendszerek Fizikája Tanszék

2023 október 24.

**Általában:**  $M$  darab egyenlet  $N$  változóval, az  $a_{ij}$  és  $b_j$  értékek ismertek:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1N}x_N &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2N}x_N &= b_2 \\ &\vdots \\ a_{M1}x_1 + a_{M2}x_2 + \dots + a_{MN}x_N &= b_M \end{aligned}$$

Felírhatjuk mátrix alakban is:

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$$

Cél: az  $x_i$  ismeretlenek ( $\mathbf{x}$  vektor) meghatározása.

$M = N$  esetén

- Egyértelmű megoldást kapunk, kivéve ha a mátrix szinguláris:  $\det(\mathbf{A}) = 0$
- ha a mátrix szinguláris:
  - vagy nincs megoldás
  - vagy végtelen sok megoldás van

Mikor szinguláris egy mátrix?

- van olyan sor, ami más sorok lineárkombinációjaként előáll
- ugyanakkor ez úgy is látható, hogy van olyan oszlop, ami más oszlopok lineárkombinációjaként előáll

A számítógép a valós számokat véges precizitással ábrázolja, emiatt kerekítési hibák adódnak.

$N \gg 1 \Rightarrow$  az egyenletrendszer megoldása sok műveletet igényel

- a kerekítési hibák összeadódnak
- a program lefut, de a végeredmény hibás lesz
- a kapott  $x$  megoldás visszahelyettesítésével lehet ellenőrizni a hibát
- a kerekítési hibák miatt a számolás során előfordulhat 0-val osztás  $\Rightarrow$  a mátrix *numerikusan szinguláris*

A számítógép a valós számokat véges precizitással ábrázolja, emiatt kerekítési hibák adódnak.

$N \gg 1 \Rightarrow$  az egyenletrendszer megoldása sok műveletet igényel

- a kerekítési hibák összeadódnak
- a program lefut, de a végeredmény hibás lesz
- a kapott  $x$  megoldás visszahelyettesítésével lehet ellenőrizni a hibát
- a kerekítési hibák miatt a számolás során előfordulhat 0-val osztás  $\Rightarrow$  a mátrix *numerikusan szinguláris*

Speciális algoritmus alkalmazása nélkül

- néhány 10 változós egyenlet általában elég könnyen megoldható
- néhány 100 egyenlethez duplapontosságú aritmetika kell: `double` típusú változók használata
- 1000 egyenlet fölött már biztosan jelentkeznek a problémák

- a memóriaigény  $N^2$ -tel skálázik
- a számításigény  $N^3$ -bel skálázik

Speciális alakú mátrixok esetében van gyorsabb megoldás.

- tridiagonális (ld. FiznumII: spline interpoláció)
- Vandermonde-típusú (ld. FiznumII: interpolált polinom együtthatói)

Ha  $M < N$ , vagy  $M = N$ , de az egyenletrendszer degenerált (kevesebb egyenlet, mint ismeretlen): az egyenletrendszer *alulhatározott*

- általában nincsen megoldás, vagy
- a megoldás egy egész altér:  
 $\mathbf{x}_p$  egy lehetséges megoldás, és ehhez jön még  $N - M$  vektor tetszőleges lineárkombinációja

Ha  $M > N$  (több egyenlet mint ismeretlen) akkor az egyenletrendszer *túlhatározott*

- általában nincsen megoldás, de
- van értelme egy lehetséges legjobb megoldásról beszélni
- ez az az  $\mathbf{x}$ , ami az alábbi minimumot megvalósítja

$$\min_{\mathbf{x}} \sum_i \left( b_i - \sum_j a_{ij} x_j \right)^2 = \min_{\mathbf{x}} (\mathbf{b} - \mathbf{Ax})^2$$



Optimalizált programcsomagok léteznek

- LINPACK, LAPACK
- Párhuzamosított változat: ScaLAPACK
- Intel processzorokra optimalizált: MKL

Optimalizált programcsomagok léteznek

- LINPACK, LAPACK
- Párhuzamosított változat: ScaLAPACK
- Intel processzorokra optimalizált: MKL

## **Singular value decomposition**

- (közel) szinguláris, túlhatározott, alulhatározott egyenletrendszer esetén is hasznos lehet az [szinguláris érték szerinti felbontás](#) (angol: SVD) módszer