How to Write Fast Numerical Code

Spring 2011

Lecture 12

Instructor: Markus Püschel

TA: Georg Ofenbeck

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Miscellaneous

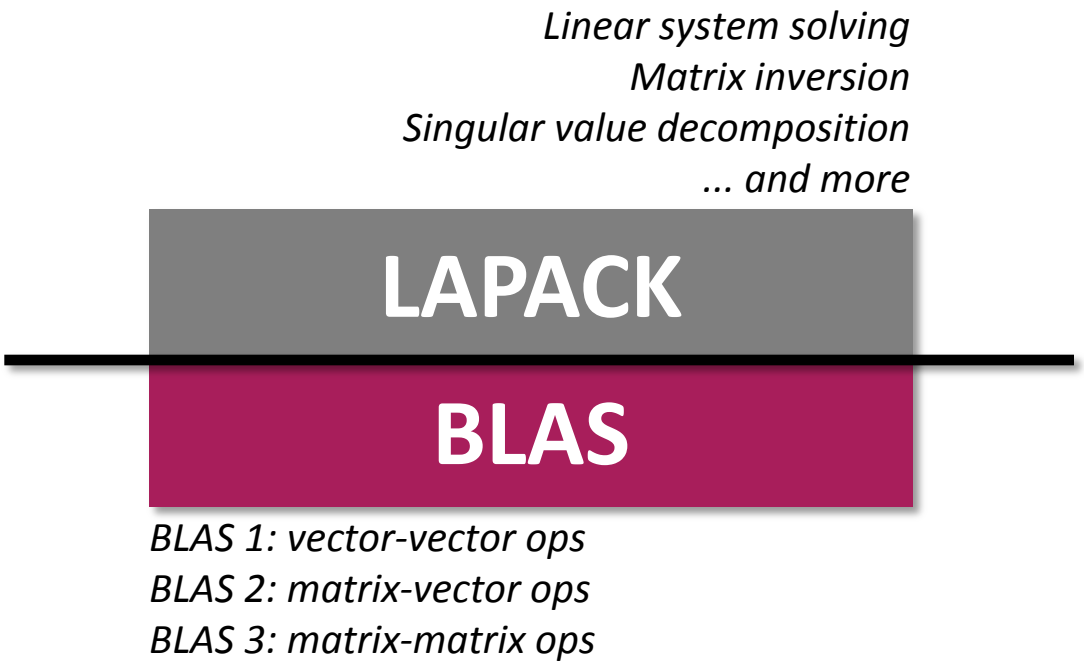
- **Start of research project**
- **No class next Monday, April 11th (Sechseläuten)**
- **Midterm exam: Friday, April 15th**

Today

- **Linear algebra algorithms and optimization**
  - Solving linear systems (Gauss elimination)
  - Matrix inversion
  - Determinant

Reminder: LAPACK

- **Implements linear algebra algorithms**
- **Implemented on top of BLAS using BLAS 3 as much as possible (by "blocking" the algorithms)**



Example: Linear Systems and Related

- Solving linear systems
- PLU factorization
- Matrix inversion
- Determinant

Complexity

- ***Source:*** **Buergisser, Clausen, Shokrollahi "Algebraic Complexity Theory," Springer 1997, pp. 426**

- ***Definition:*** **P(n), n > 0, a sequence of problems (n = problem size), complexity measure = number of adds + mults, then**

$$w(P) = inf(g \mid complexity(P(n)) = O(n^g))$$

- **Problems:**
  - MMM(n): multiplying two n x n matrices
  - MInv(n): inverting an n x n matrix
  - PLU(n): computing PLU factorization of an n x n matrix
  - Det(n): computing the determinant of an n x n matrix

Complexity Results

- **Example (we had that before): $2 \leq w(MMM(n)) < 2.38$**

- ***Theorem:***
  $w(MMM(n)) = w(MInv(n)) = w(PLU(n)) = w(Det(n))$

- **Cost of the usual implementations:**
  - $MMM(n) = 2n^3 + O(n^2)$
  - $MInv(n) = 8/3\ n^3 + O(n^2)$
  - $PLU(n) = 2/3\ n^3 + O(n^2)$
  - $Det(n) = 2/3\ n^3 + O(n^2)$

How it’s Implemented

- **Blackboard**

Chapter 2 in James W. Demmel, Applied Numerical Linear Algebra, SIAM, 1997