THE TOP 10 ALGORITHMS FROM THE 20TH CENTURY

Alex Townsend
Cornell University
THE TOP 10 LIST

1946: The Metropolis Algorithm
1947: Simplex Method
1950: Krylov Subspace Method
1951: The Decompositional Approach to Matrix Computations
1957: The Fortran Optimizing Compiler
1959: QR Algorithm
1962: Quicksort
1965: Fast Fourier Transform
1977: Integer Relation Detection
1987: Fast Multipole Method

Dantzig von Neumann Hestenes Householder Backus Hoare Greengard
WHAT IS AN ALGORITHM?

Definition:
“An algorithm is a sequence of finite computational steps that transforms an input into an output” [Cormen and Leiserson, 2009]

- Making tea: Set of instructions
- Baking a cake: Recipe
- Finite: while(1), end
NUMERICAL ANALYSIS

A definition

“The study and development of algorithms that use numerical approximation”

How many of the top 10 algorithms are in numerical analysis?

Potentially all of them

Floating point arithmetic

\[
1/3 \approx (-1)^s \left( 1 + \sum_{i=1}^{52} b_{52-i} 2^{-i} \right) \times 2^{e-1023}
\]

Algorithms implemented in floating point arithmetic are studied and developed by numerical analysts
OVERVIEW OF TALK

A top 10 algorithm

How it works?

How do I use it?

Open problem
1959: QR ALGORITHM

The Tacoma Narrows bridge in Nov 1940

Collapsed in 80km/h winds
NUMERICAL SIMULATIONS

Resonant frequencies are eigenvalues: \( A \vec{v} = \lambda \vec{v} \quad \vec{v} \neq 0 \)
HOW DOES IT WORK?

$A = \begin{bmatrix} \end{bmatrix}$ $Q = \begin{bmatrix} \end{bmatrix}$ $R = \begin{bmatrix} \end{bmatrix}$

$A = \text{symmetric}$

for $k = 1, 2, \ldots$

$A = Q*R$

$A = R*Q$

end

The final diagonal matrix contains all the eigenvalues.
How do I use it?

Rootfinding and global optimization

Matrix determinant

\[ p(x) = \pm \det (A - xI) \]

characteristic polynomial of \( A \)

Identity matrix

A tiger’s tail
Let \( p(x, y) \) be a degree \((n, n)\) polynomial. Construct \( n \times n \) matrices \( A, B, \) and \( C \) such that

\[
p(x, y) = \det(A + xB + yC).
\]

Compare to: \( p(x) = \pm \det(A - xI) \)

Need it to solve:

\[
p(x, y) = q(x, y) = 0
\]
1965: THE FAST FOURIER TRANSFORM

“Mozart could listen to music just once and then write it down from memory without any mistakes” [Vernon, 1996]

A simple example:

$$\text{sound}(t) = 3 \cos(2\pi 10t + 0.2) + \cos(2\pi 30t - 0.3) + 2 \cos(2\pi 40t + 2.4)$$
HOW DOES IT WORK?

Given equally spaced samples \( f(0/n), f(1/n), \ldots, f((n-1)/n) \), find \( a_k \) so that

\[
f(j/n) = \sum_{k=-n/2}^{n/2-1} a_k e^{2\pi ik (j/n)}, \quad 0 \leq j \leq n-1.
\]

\( F \) has a sparse factorization. For \( n = 16 \) we have

\[
\begin{pmatrix}
    f(0/n) \\
    \vdots \\
    f((n-1)/n)
\end{pmatrix} = F \begin{pmatrix}
    a_{-n/2} \\
    \vdots \\
    a_{n/2-1}
\end{pmatrix}, \quad F_{jk} = e^{2\pi ik (j/n)}
\]
HOW DO I USE IT?

An automatic way to tell us how “complicated” a function is.
OPEN PROBLEM

Let everyone be a Mozart

An example with chords:

Eight playing

My sheet music for cellos

Play back
1987: THE FAST MULTIPOLE METHOD

In 4 billion years time...
HOW DOES IT WORK?
The SVD gives the best low rank approximations:

Original  rank 1  rank 3  rank 10  rank 50

The low rank format saves computational time and storage costs.
OPEN PROBLEM

Why are so many matrices/functions in practice of low rank?

A random matrix is of full rank so “average” matrices are not...

...but, these are of low rank.

Even the American flag is of low rank!
THE TOP 10 LIST (AGAIN)

1946: The Metropolis Algorithm
1947: Simplex Method
1950: Krylov Subspace Method
1951: The Decompositional Approach to Matrix Computations
1957: The Fortran Optimizing Compiler
1959: QR Algorithm
1962: Quicksort
1965: Fast Fourier Transform
1977: Integer Relation Detection
1987: Fast Multipole Method

Dantzig  Neumann  Hestenes  Householder  Backus  Hoare  Greengard
THANK YOU

What will be the top 10 algorithms of this century?

Alex Townsend
Assistant Professor
Math Department
townsend@cornell.edu