

A Taste of Scientific Computing

Victor Eijkhout

2022

What is Scientific Computing about?

You know the science; what more is there?

- Science often gives an implicit description.
How do you turn it into something computational.
- Algorithms are not unique:
There are many ways to solve a linear system

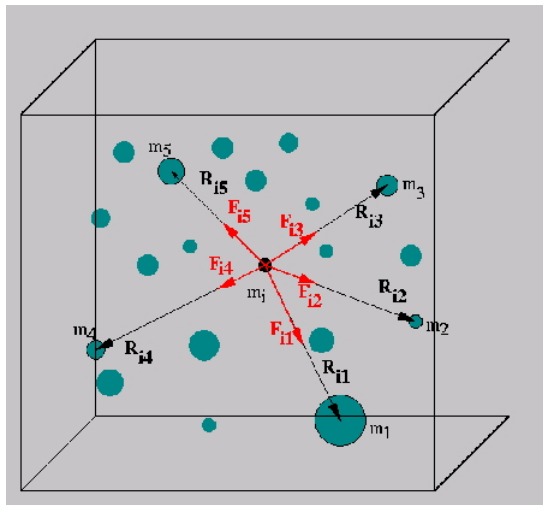
$$x: Ax = b$$

What are pros and cons of the choices?

- Algorithms can be implemented multiple ways, depending on your processor.

Algorithmic choices

Summing forces



Particle interactions

for each particle i

for each particle j

let \vec{r}_{ij} be the vector between i and j ;

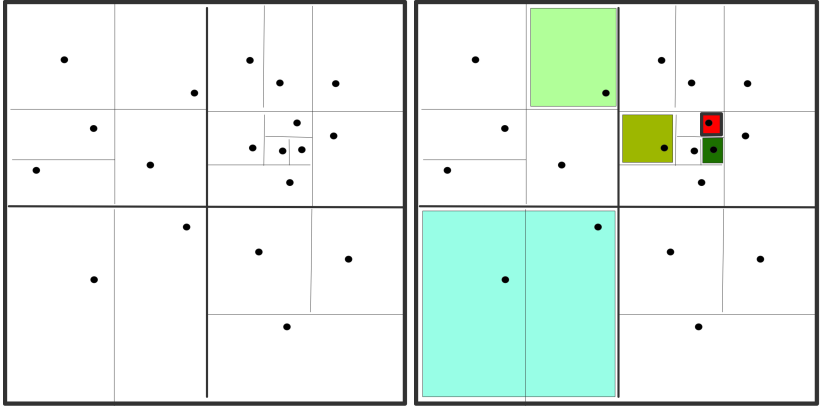
then the force on i because of j is

$$\vec{f}_{ij} = -\vec{r}_{ij} \frac{m_i m_j}{|\vec{r}_{ij}|^3}$$

(where m_i, m_j are the masses or charges) and

$$\vec{f}_{ji} = -\vec{f}_{ij}.$$

Naive all-pairs algorithm: $O(N^2)$



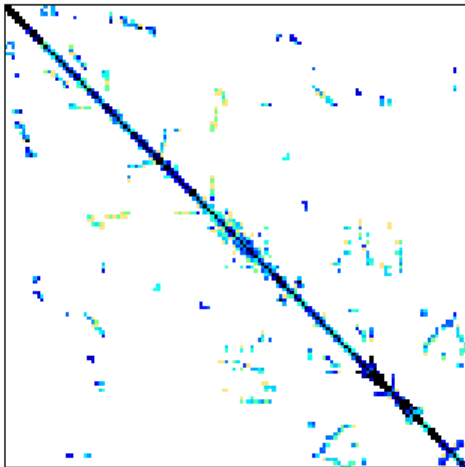
Clever algorithms: $O(N \log N)$, sometimes even $O(N)$

Algorithm aspects

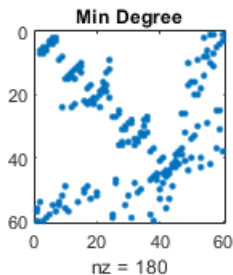
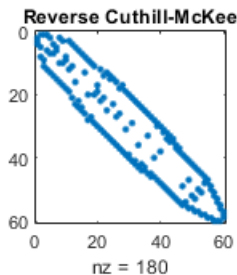
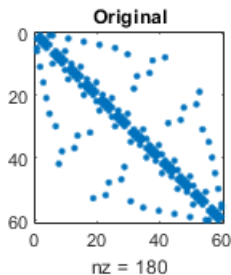
$$\underset{x}{?}: Ax = b$$

- Inversion: N^3 operations, unstable
- Gaussian elimination: N^3 but lower constant, stable
- Sparse Gaussian elimination: $N^{3/2}$, hard to program
- Iterative methods: $N \cdot \kappa^{1/2}$, not always successful
- Multigrid: $O(N)$, very limited applicability.

Sparse matrices



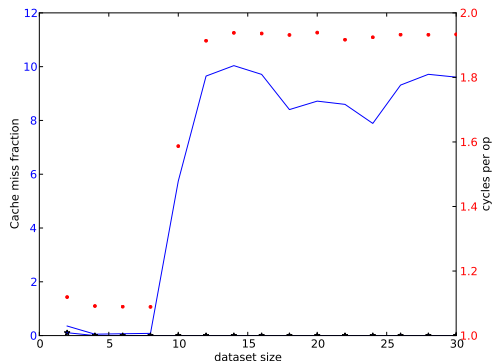
Permuting the matrix



The influence of your architecture

Fitting data to cache

```
for (j=0; j<size; j++)  
    array[j] = 2.3*array[j]+1.2;
```

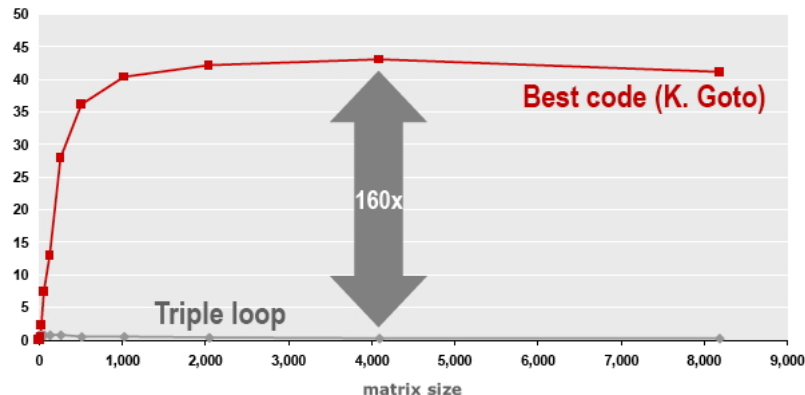


Matrix-matrix product

Lots of small optimizations add up:

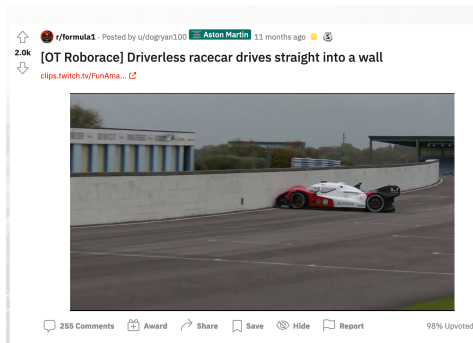
Matrix-Matrix Multiplication (MMM) on 2 x Core 2 Duo 3 GHz (double precision)

Gflop/s



Computer arithmetic

Computer numbers are not real numbers. If you don't pay attention to this you may lose you a race car, or a rocket



So what is scientific computing about?

Between science and computing

- Modeling
- Numerical analysis
- Linear algebra
- Computer architecture
- ... and the interaction between any and all of these.