

# 科学计算

Scientific Computation

潘建瑜

华东师范大学 数学科学学院

# 为什么科学计算

---

计算科学是 21 世纪确保国家核心竞争能力的战略技术之一.

—— **计算科学: 确保美国竞争力, 2005 年总统信息技术咨询委员会报告**

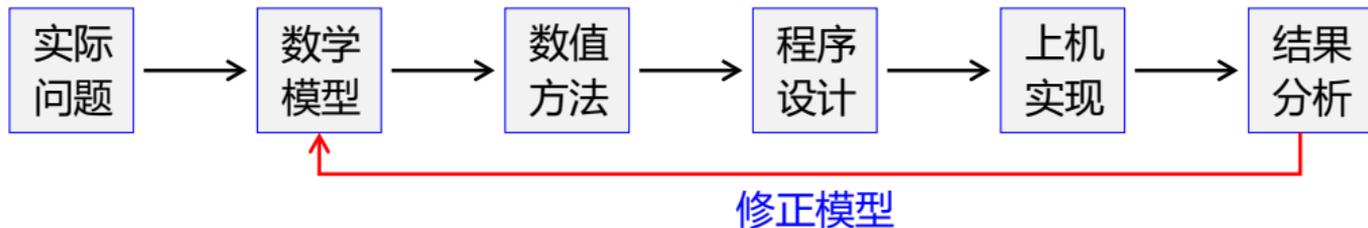
**科学计算**是 20 世纪重要科学技术进步之一, 已与**理论研究**和**实验研究**相并列成为科学研究的第三种方法. 现今科学计算已是体现国家科学技术核心竞争力的重要标志, 是国家科学技术创新发展的关键要素

—— **国家自然科学基金·重大项目指南, 2014**

科学计算的核心/数学基础: **计算数学**.

# 数值算法

借助计算机的高速计算能力, 解决现代科学、工程、经济、人文等领域中的各类复杂 (数学) 问题, 是数学与计算机的有机结合.



# 主要任务

## 设计求解各种实际问题的高效可靠的数值方法

- **算法设计**: 构造求解各种数学问题的数值方法
- **算法分析**: 收敛性、稳定性、复杂性、计算精度等
- **算法实现**: 编程实现、软件开发
- **数学模型分析**: 数学问题的结构特点, 数学模型的评估 (合理性、准确性)

对于同一个问题, 不同的算法在计算性能上可能相差百万倍或者更多!

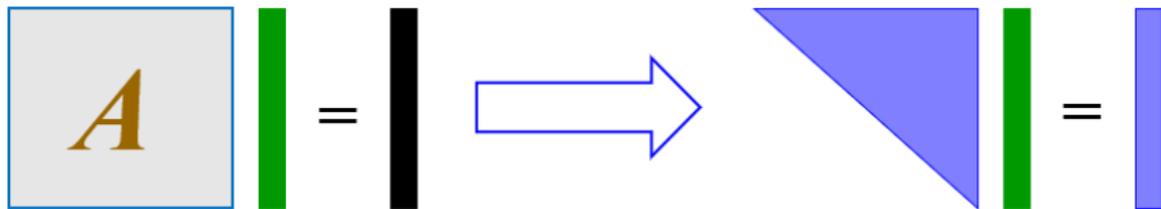
# 应用举例：线性方程组求解

## 线性方程组求解

求解一个  $n$  阶线性方程组, 若使用**克莱姆法则**, 需要计算  $n + 1$  个  $n$  阶行列式, 大约需要  $n!(n^2 - 1)$  次乘除运算. 而使用**高斯消去法**, 只需约  $2n^3/3$  次加减乘除运算.

🔴 当  $n = 20$  时,  $n!(n^2 - 1) \approx 9.7^{20}$ ,  $2n^3/3 \approx 5.3 \times 10^3$

如果用每秒运算 30 亿次 (主频 3.0G) 的计算机求解, 克莱姆法则大约需要 10000 年!  
但高斯消去法一秒不到就能解决!



# 应用举例：矩阵乘积 (Deepmind 团队)



nature

Explore content ▾

About the journal ▾

Publish with us ▾

[nature](#) > [articles](#) > article

Article | [Open access](#) | [Published: 05 October 2022](#)

## Discovering faster matrix multiplication algorithms with reinforcement learning

[Alhussein Fawzi](#) , [Matej Balog](#), [Aja Huang](#), [Thomas Hubert](#), [Bernardino Romera-Paredes](#), [Mohammadamin Barekatin](#), [Alexander Novikov](#), [Francisco J. R. Ruiz](#), [Julian Schrittwieser](#), [Grzegorz Swirszcz](#), [David Silver](#), [Demis Hassabis](#) & [Pushmeet Kohli](#)

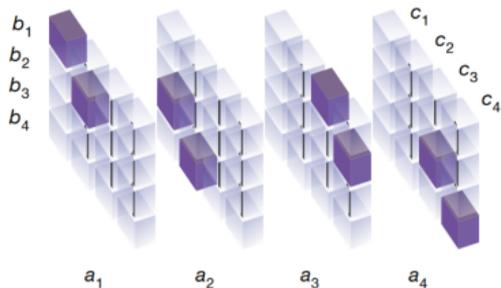
[Nature](#) **610**, 47–53 (2022) | [Cite this article](#)

**563k** Accesses | **131** Citations | **3658** Altmetric | [Metrics](#)

# 应用举例：矩阵乘积

**a**

$$\begin{pmatrix} c_1 & c_2 \\ c_3 & c_4 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \cdot \begin{pmatrix} b_1 & b_2 \\ b_3 & b_4 \end{pmatrix}$$



**b**

$$m_1 = (a_1 + a_4)(b_1 + b_4)$$

$$m_2 = (a_3 + a_4)b_1$$

$$m_3 = a_1(b_2 - b_4)$$

$$m_4 = a_4(b_3 - b_1)$$

$$m_5 = (a_1 + a_2)b_4$$

$$m_6 = (a_3 - a_1)(b_1 + b_2)$$

$$m_7 = (a_2 - a_4)(b_3 + b_4)$$

$$c_1 = m_1 + m_4 - m_5 + m_7$$

$$c_2 = m_3 + m_5$$

$$c_3 = m_2 + m_4$$

$$c_4 = m_1 - m_2 + m_3 + m_6$$

**c**

$$\mathbf{u} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & -1 \end{pmatrix}$$

$$\mathbf{v} = \begin{pmatrix} 1 & 1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

$$\mathbf{w} = \begin{pmatrix} 1 & 0 & 0 & 1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

事实上,除了计算复杂度外,数据在内存/缓存中的移动/访问方式也会影响计算效率.

(GEMM\_Performance.m)

# 数值方法基本概念

---

- 解析解、精确解、真解、真值, 数值解、近似解
- 数值方法的特点
  - 求的是近似解, 即求出的解是有误差的
  - 与计算机紧密结合, 易于上机实现
- 算法的评价
  - 时间复杂度 (计算机运行所需的时间)
  - 空间复杂度 (所占用的计算机存储空间)
  - 逻辑复杂度 (影响程序开发的周期以及后续维护的难易程度)

# 好的数值方法

---

- 有可靠的理论分析, 即收敛性、稳定性等有数学理论保证
- 有良好的计算复杂性 (时间和空间)
- 易于在计算机上实现
- 要有具体的数值试验来证明是行之有效的

# 课程主要内容

---

- 非线性方程的数值解法
- 线性方程组的直接解法
- 线性方程组的迭代解法
- 线性最小二乘问题
- 函数插值
- 数值积分与数值微分

# 预备知识和学习建议

## 预备知识

- 微积分
- 线性代数
- 一定的编程基础（北太天元, MATLAB 或 Python）

## 学习建议

- 掌握数值方法的基本思想和原理 (不能仅仅靠记忆);
- 掌握数值方法的设计和分析的一些常用技术和技巧;
- 适量的数值计算训练 (包括编程实践和手工推导, 有助于加深对算法的理解).

# 成绩评定与参考资料

 成绩评定: 平时成绩 (课后作业 + 算法实践), 期末成绩 (笔试)

## 主要参考资料

-  潘建瑜, **科学计算简明讲义**, 2025
-  李庆扬等, **数值分析**, 第五版, 清华大学出版社, 2008
-  喻文健, **数值分析与算法**, 第三版, 清华大学出版社, 2020
-  Sauer, **Numerical Analysis**, 3rd Edition, 2018
-  Leader, **Numerical Analysis and Scientific Computation**, 2nd Edition, 2022
-  Salgado and Wise, **Classical Numerical Analysis: A Comprehensive Course**, 2023
-  课程主页: <http://math.ecnu.edu.cn/~jypan/Teaching/SC>

谢谢  
THANK YOU

