数学，计算复杂性，计算理论，计算机

数学和计算的奥秘

Michael Shub

译、校者注：Michael Shub 是理论计算机科学工作者，与 L. Blum, S. Smale 合作提出了实数环上的计算模型 (称为 BSS 机) 及其相应的计算复杂性理论. Shub 是 IBM 研究院 T. J. Watson 研究中心数学科学部门的研究人员，特别数学研究计划的负责人. 本文作为理论联系实践的方式之一，讨论了 BSS 机产生的原因，介绍了如何用于动力系统的研究；并陈述了冯诺伊曼，图灵对现代计算机的贡献及其历史，以说明追溯思想在实践中产生的根源是相当困难的.

去年五月间，皇后学院 (Queens college) 计算机科学系系主任 Ted Brown 打电话给我，邀请我作一个题为 "计算理论的未来" 的讲座，当时我感到非常惊讶. 我刚从加州呆了两个月回来，还没有打开 e-mail 信箱，所以我意识到 Ted 并不是请我作一个简明的、非正式的报告. 他给皇后学院计算机科学系的工业委员会 (Industrial Affiliation Board) 的每个委员去了一封信，请他们参加于 6 月在学院举行的会议.

大约两年前，我写了一个书评，评论 David Ruelle 的关于动力系统和分支 (bifurcation) 理论的书；这使我意识到我所想的东西在动力系统中是重要的. 对这结论 [20] 我感到非常高兴. 仅部分知道这方面的工作，所以我不太情愿地答应给一个讲座，只是借此机会反思一下计算的理论. 我所考虑的问题之一是理论与实际的关系 —— 过去、现在和将来；这特别是因为基金的来源越来越要求科学界估计科学研究的效用. 在皇后学院的讲座通知发出后，我被邀请在我所在的系以及 Courant 研究所做报告，每次我都对报告进行修改.

理论联系实际至少有四种方式，对每一种我将先给一个计算机科学的例子，然后，举一个数学的例子.

1.**渐进的**：在一个明确的研究分支里理论和实际是相互不断促进的.

这时我头脑里出现的例子有数据分类、编译设计算法的不断改进；在数学中，物理、工程和工程设计中微分方程数值解方面的不断进展.

2.**结构的**：理论提供语言和结构并由此讨论和分析所存在的实际.

一个显著的例子是复杂性理论，P 与 NP 完全问题，或者语言和逻辑构造在计算机

原题：Mysteries of Mathematics and Computation. 译自：*Mathematical Intelligencer*, Vol. 16, No.1, 1994, pp. 10–15.

语言和编程理论中的应用. 在数学中，我们可以用常微分方程中动力系统所具有的结构效果 (organizing effect) 作为例子.

3.**预期的**: 由于其内部原因，理论产生的结构将来可能对实际有重要意义.

这里显明的例子是图灵 (Turing) 的通用机可看作现代计算机的先驱，至今仍是机器的主要理论模型 (我将回头再来讨论这一点). 在数学中，标准的例子是黎曼几何的发展后来对爱因斯坦来说是非常有用的.

在一个更小范围，但更现代和更接近我们的例子是，在本系里，Brian Marcus 和 Roy Adler 在符号动力学的分类问题方面的工作，后来发现在编码中是有用的，并且用来改进磁盘的存储容量.

4.**非正式的**: 理论产生的背景可以被认为是体现了作为所能完成的东西的范围的知识现状.

我们有 Gödel 和图灵的定理，以及深信 NP 完全问题的不可简化性.

在数学中，我们仍能引用 Gödel 和图灵的工作，或者混沌系统的不可预测性.

对于每个类型都有许多例子，当然，结构的、预期的和非正式的类型均可归为渐进的类型.

就本讲座的主题而言，我将尽可能广泛地讨论理论联系实际问题. 理论的未来在一定程度上依赖于未来的实践，我们得等待观望未来. 目前，计算的理论紧密地联系于机器设计，数据管理和相伴组合优化问题. 翻一下 1969 年计算理论会议目录我惊奇地发现第一次会议看起来多么象逻辑或可计算函数理论的会议，其中特别强调可计算函数的复杂性. 而到 1992 年复杂性理论已相当成熟了，在 24 届计算理论 ACM[1] 年会上出现了一些新问题，如平行计算和容错. 但 1992 年的 STOC[2] 的重点仍在解决象通信问题那样的机器的逻辑结构问题.

过去几十年里，数据处理占据了大部分计算机的时间，但工作站的发展使得我们可以在办公桌前进行精深的计算. 计算数学，甚至符号计算数学发展了. CAD-CAM[3] 和出现的计算机可视交互技术，机器人，和象蛋白质复合这样的科学问题将要求大量的科学计算. 平行和分布式计算机将大大提高计算能力. 计算机是一个工具；部分由于更加精深的硬件，数值算法也随之更加精深. 我们能很容易地预见理论将随着实践的发展而相互发展. 目前关于小波问题的大量研究就是一个明显的例子.

存储程序式数值计算机是用来进行科学计算的. 这里的理论，数值分析和实践看起来几乎完全是渐进式发展的. Wilkinson 发明的回溯错误分析和条件数是两个例外，我称他们是结构的. 一个科学计算的结构理论是由什么组成的？让我们以一个例子看科学来源以及我们所遇到的困难问题.

计算机图形学对于低维动力系统的研究是极其有用的工具. Lorenz 吸引子，Julia 集，Mandelbrot 集和 Henon 吸引子是几个最早和最有名的例子. 如果我们考虑复多项

---

[1] Association for Computing Machinery, (美国) 计算机协会. —— 译注.

[2] Symposium on Theory of Computing, 由美国计算机协会举办的计算的理论国际会议. —— 译注.

[3] Computer-Aided Design/Computer-Aided Manufacturing, 计算机辅助设计与计算机辅助制造. —— 译注.

式

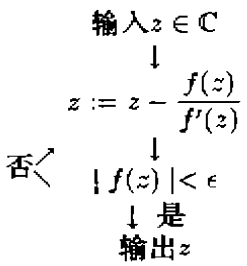
$$f(z) = (z^2 - 1)(z^2 + 0.16)$$

的牛顿法， Scott Sutherland [1] 给出了一个漂亮的图形（见本期封面的彩图），我们知道， $N_f(z) = z - f(z)/f'(z)$ 是一个黎曼曲面上的动力系统， N_f 的不动点是 f 的根. 这些根是在实轴和虚轴上的. 红、绿、黄或蓝色调区域在 N_f 的叠代下将收敛于这些根，越浅的部分收敛越快. 黑色区域是一个开点集，在叠代下收敛于两个周期为 2 的吸引点，因而不收敛于 f 的根. 任何两个色彩之间的边界，包括黑色区域，是 Julia 集 (的逼近). 它的分形特征在图中很明显. 这个计算机图形对理解动力学和这种特殊类型的牛顿法是一个极好的启发.

现在让我们反过来去解释产生这图形的机器和图形本身. 图灵机模型看起来并不适合于做这件事. 首先，开区域的分数维和边界对离散点集是没有意义的. 为真正地理解这个图形，我们必须设制一个复数上运算的机器，使得我们能通过 Fatou 和 Julia 的工作理解复解析动力 $N_f(z) = z - f(z)/f'(z)$, 理解 Sullivan 的非游荡区域定理 (nonwandering domains),Julia 集的双曲线及其逼近等. 牛顿法只是一个例子. 科学计算和数值分析所处理的问题，其自然定义区域是复数. 为了能讨论这方面的可计算性、有效性以及复杂性， Lenore Blum, Steve Smale 和我 [Blum-Shub-Smale [2](=BSS)] 引入了在一个环上运算的机器. 环的例子有整数环 $\mathbb{Z}$, 实数环 $\mathbb{R}$ 和复数环 $\mathbb{C}$.

环	函数	分支
$\mathbb{Z}$	多项式	≤ 0或> 0
$\mathbb{R}$	有理函数	≤ 0 或 > 0
$\mathbb{C}$	有理函数	$= 0$ 或 $\neq 0$

我们的机器，现已称为 BSS 机器，基本上是一个流程图式机器，是由下列五种结点组合的有向图：输入结点、输出结点、计算结点、分支结点和一种存取计算结果的结点. 所有计算只含有限多个变量，且只有一个输入结点. 这种机器的一个例子是完成牛顿法的机器：



对于适当选择的 ϵ, 这个机器再带一个作为子路径的计数器，可以用来产生本期封面上的那张图.

定义在各种环上运算的机器的产生的部分原因是，使数学分析用于研究 $\mathbb{Z}$ 上的可

计算性和复杂性的同时，也可用于研究 $\mathbb{R}$ 或 $\mathbb{C}$ 上的问题. 受递归函数理论启示，我们的第一个结果是关于不可判定性的. 给定一个输入子集 $S \subset I$, 称机器 M 判定 S, 或 S 称为可判定的，如果对于输入 $x \in I$, 当 $x \in S$ 时 M 输出 "是"; 而当 $x \notin S$ 时输出 "否".

定理 1(BSS). 如果 $f(z) = \Sigma_{j=0}^{d} a_j z^j$ 是一个有三个或三个以上不同根的复多项式，则没有机器能判定对任何输入 $z \in \mathbb{C}$, 牛顿法是否收敛于 f 的根，即，当 $k \to \infty$ 是否 $f(N_f^k(z)) \to 0$.

这样，封面图中的 Julia 集的逼近仅仅是一个逼近.

对于可判定问题，同样有关于输入规模的多项式代价内 (译注：确定机器) 可判定的问题，称为 P 类判定问题. 为了定义 P, 对这三个基本例子我们还需要两个数据.

环	输入规模	代价
$\mathbb{Z}$	比特	比特
$\mathbb{R}$	维数	代数的
$\mathbb{C}$	维数	代数的

NP 类可看成是在输入规模的多项式代价内 (译注：非确定机器) 可判定的问题类. 可类似于 Cook[1] 定义 P 类和 NP 类，在 $\mathbb{Z}$ 上是与 Cook [3] 定义的类相同. 基本问题：是否 "P≠ NP?" 在三个环上都有意义. $\mathbb{Z}$ 上的 NP 类的重要性是由于 Cook[3] 和 Karp [4] 提出的大量的 NP 完全问题，见 Garey 和 Johnson [5]. 我们知道一个问题是 NP 完全的，如果任何其它 NP 问题在多项式代价内可归约为该问题.

定理 2(BSS). 下列问题是 NP 完全的:

在 $\mathbb{Z}$ 上:

(a) 给定 $f \in \mathbb{Z}[x_1, ..., x_n]$ 和一个界 $b \in \mathbb{Z}$，是否存在点 $(x_1, ..., x_n) \in \mathbb{Z}^n$ 使得 $f(x_1, ..., x_n) = 0$ 且 $\Sigma x_i^2 \leq b^2$? (围界希尔伯特第十问题)

(b) 给定 $n \times m$ 整数矩阵 A、整数向量 b, c 和一个整数 k, 是否存在一个整数点 x 使得 $Ax \leq b$ 且 $cx \geq k$? (整数线性规划)

在 $\mathbb{R}$ 上:

(c) 给定 4 次多项式 $f : \mathbb{R}^n \to \mathbb{R}$, 是否存在一个 $x \in \mathbb{R}^n$ 使得 $f(x) = 0$? (四次可行性)

在 $\mathbb{C}$ 上:

(d) 给定一个多项式组 $f_1, ..., f_k$, 其中 $f_j : C^n \to C$、是否存在 f_j 的公共根，即，一 $x \in \mathbb{C}^n$ 使得对 $j = 1, ..., k$, $f_j(x) = 0$? (希尔伯特的零化子 Nullstellensatz)

我们都知道 $\mathbb{Z}$ 上的这些问题是 NP 完全的. 用 BSS 机器，我们给出了 (a),(c) 和 (d) 的完整证明；(b) 的证明只是稍为困难些.

$\mathbb{R}$ 上的搜索问题中，我们不仅要判定一个问题是否有解，而且要在有解的情况下逼近这个解. 既使是求解 $x^2 - a = 0$, 即计算 $\sqrt{a}$ 精确到 ϵ, 随着 $\epsilon \to 0$ 我们需要越来越多的代数运算，因为 $\sqrt{a}$ 不是 a 的有理函数.

因此，逼近的精确度必须是问题输入规模的一个组成部分. 对于容许输入或计算误

差的搜索问题更是如此. 问题的条件，即，解对数据波动的敏感性，将起作用. 甚至对于两个联立线性方程，解对于数据是不连续的，所以在容许误差的情况下，就会有无限病态问题.

最近， Steve Smale 和我借助于同伦的条件，对于度量步数的 Bezout 定理 [6-8] 的同伦方法作了分析，给出了条件的几何解释，以及一个问题是病态的概率估计.

下面是我们对 Bezout 问题研究的结论之一. 给定 $n+1$ 个复变量、次数为 $d_1,...,d_n$ 的 n 个齐次方程，求其解线 (solution line). 设 $H_{(d)}$ 是这样系统的向量空间，$(d)=(d_1,...,d_n)$.

我们集中于导引簇 (incidence variety) $V\subset P(H_{(d)})\times P(n)$，其中 $P(H_{(d)})$ 是 $H_{(d)}$ 的投影空间，$P(n)$ 是 $\mathbb{C}^{n+1}$ 的投影空间，且 $V=\{(f,x)\mid f(x)=0\}$. 酉群带以通常的 Hermite 结构作用于 $\mathbb{C}^{n+1}$，导出一个在 $H_{(d)}$ 上的作用，取一个 $H_{(d)}$ 上的酉不变 Hermite 结构，这结构首先是由 Kostlan[9] 讨论的. 这样，我们有一个 $P(H_{(d)})$ 和 $P(n)$ 上的 Fubini-Study 测度，和 V 上的导出测度. 我们用 [10] 中介绍的投影牛顿法代替牛顿法. 当 $f(x)=0$，f 在 x 处的导数逆的范数决定 (f,x) 的经典条件. 我们用 $\mu(f,x)$ 表示这个数的投影，对于一个同伦 $F=f_t$，设 $\mu(F)$ 为 $\sup_{f_t(x_t)=0}\mu(f_t,x_t)$.

定理(Bez I). 足以跟踪一个同伦的投影牛顿法的步数 $\leq cLD^{3/2}\mu^2$，其中 c 是一个常量 <10，$D=\max d_i$，L 是同伦的长度.

一个根的情况下，类似定理也成立. 因此，如果考虑一个问题的条件，步数可能有节制地依赖于条件. 在我们的分析中酉群是至关重要的. 下面我们对条件数作一个几何解释，这将推广 Demmel[11, 12] 的工作.

设 $\Sigma'=\{(f,x)\in V\mid \text{秩 } Df(x)<n\}$. Σ' 是 (f,x) 的单变簇的推广，这里的 x 是 f 的多重根，我们定义了到 Σ' 的垂直距离 ρ 并且证明

定理(Bez I). 对于 $(f,x)\in V$，

$$\mu(f,x)=\frac{1}{\rho((f,x),\Sigma')}.$$

最后我们估计了病态问题的量 (volumn).

定理(Bez II). 对于 $(f,x)\in V,\rho(f,x)<\rho_0<1/\sqrt{n}$ 的概率 $\leq KnN\rho_0^2$，其中 N 为 $H_{(d)}$ 的维数，且 K 是一个小常数.

为了证明这个量的估值，我们考虑了两个投影

$$\begin{array}{ccc} & V\subset P(\mathcal{H}_{(d)})\times P(n) & \\ & \swarrow\quad\searrow & \\ P(\mathcal{H}_{(d)}) & & P(n) \end{array}$$

当我们将这技术用到带导出黎曼结构的实齐次系统时，得到

定理(Bez II). 实齐次系统的实根平均数是复根数的平方根，即 $D^{1/2}$，其中 $D=\Pi d_i$ 为 Bezout 数.

当 d_i 均相等时，这定理是 Kostlan [13] 中的一个结论．这不同于 Kac[14] 的单变结论，这结论渐近地给出 $(2/\pi)\ln d$, 其中 d 为度．差别是由酉群引起的．

这是一个对解方程的复杂性理论的贡献．为科学计算的复杂性理论找适当的、现实的模型和好的结论体系仍是一个大问题．

现在我想从完全不同的角度考虑理论和实际．我已经指出图灵通用机是一个很好的说明理论先自身发展，而后应用于实际的例子．图灵和冯诺伊曼常被人看作是存储程序式计算机的发明者．但在美国宾州大学 Moore 学院的两个工程师 J.P. Eckert 和 J.W. Mauchly 已经建成了 Eniac[4] 并设计一种其后的机器．

冯诺伊曼参与了计划，并以冯诺伊曼一个人的名字写了一份关于 Edvac[5] 的报告草案． Joel Shurkin [15] 写道：

> 这文章有争议地成为这领域的唯一最重要的文件．几乎所有关于计算机或冯诺伊曼的历史都将存储记忆式计算机的荣誉归于冯诺伊曼．在科学界，计算机至今仍称为冯诺伊曼机．
>
> 冯诺伊曼不是存储程序式计算机的创始人，如我们所见，这样的机器的思想在冯诺伊曼到达 Moore 学院的一年前就在该学院讨论过了，并且 Eckert 在冯诺伊曼听说 Moore 学院计划的六个月前就写了一个计划的备忘录．

Shurkin 接着引用了 H. H. Goldstine 关于该报告的话：

> 这个报告对从 1944 年秋到 1945 年春期间所有关于 Edvac 的思想作了强有力的分析和综合，不是其中的每个想法都是他的，但本质部分是……
>
> 很明显，冯诺伊曼在写该报告时结晶了计算机领域的思想，这是其他人没有做到的，在 Moore 学院所有研究成员中他是必不可少的一位，唯有冯诺伊曼对整个计划具有本质的重要性．

比较 B. Randell 引用 S.Frankel 的话：

> 许多人称冯诺伊曼是计算机之父 (在现代意义下)，但我肯定他没有犯这样的错误[6]．他可能最适合于称为助产士．但他对我，我肯定他还对其他人，特别强调基本概念是属于图灵的 — 且不提 Babbage、Lovelace 以及其他人．我认为冯诺伊曼的根本作用是他使世界了解这些由图灵发现的基本概念以及在 Moore 学院和其它地方进行的工作．

我能想像冯诺伊曼比 Eckert 还要理解 Eckert 的思想，因为他一定知道图灵的工作．这与 Goldstine 和 Frankel 的观点一致．我没有试图去查找原始文件．两件事情是清楚

---

[4] 电子数字积分计算机． —— 译注．

[5] 电子离散变量自动计算机． —— 译注．

[6] 此句意为： 但我肯定他没有如此称呼自己． —— 译注．

的：追踪一种思想在实践中的起源的极其困难性，以及当思想可以应用时真正理解这些思想的人的价值. 在现代这个经济时代，作为理论工作者我们决不能失去对好的理论工作价值的信念，即使当这项工作在短时期内看起来远离实际.

我就这样结束了于 6 月在皇后学院的讲座，其他两个报告到 10 月份才得到安排. 我将材料放一边，害怕在没有史学家的技巧和训练的情况下进一步卷入这个历史问题，而很高兴自己致力于我正在写的数学文章. 但我总不能忘记是否冯诺伊曼知道图灵通用机的问题. 这结论是相当模糊的. 图灵在解决了判定问题之后在普林斯顿待了两年，冯诺伊曼曾请他作自己的助手. 但这结论有一个非常烦人的特点. *Edvac* 报告中提到了 McCulloch 和 Pitts，他们在 1943 年的一个报告中发现了神经网络，但报告却没有提到图灵. 这是事实， McCulloch 和 Pitts [16] 提起了图灵机并断言他们的神经网络给出相同的可计算函数："这是非常有趣的，对可计算性的图灵定义及其等价定义， Church 的 λ- 可定义性和 Kleene 的原始递归性， 提出了一个心理学证据：如果一个数能被某有机体计算，那么它在这些定义里也是可计算的，反之亦然"(预印本的 p.35). 但没有提到通用机，并且在文献中也没有引用图灵的文章.

除了 Frankel 外， Goldstine ([17], p.174) 这样评述冯诺伊曼：

> 正是他的形式逻辑的训练使他非常清楚并感兴趣于预示现代计算机的这一结果. 这一点在 Emil L. Post 和 Alan M.Turing 于 1936 年独立发表的文章中均提到过. Post 在纽约市立学院教书，图灵是在普林斯顿大学 (1936-1938) 学习的英国人. 两位都想像到了现称为自动机的东西，并用类似的机械学术语描写自动机. 他们独立地工作，相互之间不了解对方. 毫无疑问冯诺伊曼完全知道图灵的工作，但明显地不知道 Post 的工作.

在 1937 年冯诺伊曼写了一封推荐图灵的信，这信在 Hodge 写的图灵传 [18] 中被引用并评论：

> 一九三七年六月一日
>
> 先生：
>
> A. M. 图灵先生告诉我他在申请做 1937–1938 学年由剑桥到普林斯顿大学的学监访问学者. 我愿意支持他的申请，并告知你我在几年前就非常了解图灵：在 1935 年的最后一学期间，我在剑桥是客座教授，1936–1937 年间图灵在普林斯顿. 我有机会了解他的科学工作. 他在我所感兴趣的数学分支——几乎周期函数理论和连续群理论方面作了好工作.
>
> 我想他是最应得到学监奖学金的人选. 如果你有可能将这资格给他，我将非常高兴.
>
> 致敬
>
> 冯诺伊曼

上星期我打电话给 Goldstine, 给他读了冯诺伊曼的信. 他觉得这不可思议. 他认为"如果 Johnny(冯诺伊曼) 知道图灵通用机的话, 他肯定会提起的." 他也说到在 Edvac 时期或 IAS 计划的早期阶段, 没有提起图灵的工作. 只有后来冯诺伊曼讲自动机理论时才提到图灵的工作. Goldstine 在某时刻从冯诺伊曼那儿知道了图灵, 并读了图灵的工作, 但他不能肯定这是在什么时候. Hodge 在 1983 年的图灵传中也记录了一些其它证据说明冯诺伊曼知道图灵的工作: Ulam 于 1938, 1939 年的回忆及 Frankel 的回忆录中称冯诺伊曼在 1943 或 1944 年间向他俩指出了图灵的工作. 另一方面, Hodge 叙说冯诺伊曼声称在 Gödel 的 1931 年文章之后没有读过任何其它逻辑文章. 所以冯诺伊曼在写 Edvac 报告时不知道图灵通用机似乎是真的, 但当他知道这工作后他通情达理地将荣誉给了图灵.

作为一个业余爱好者, 我尽自己可能进行调查. 我不知道史学家能否解决这个问题. Hank Tropp 说我打开了一罐蛆. 这能否改变我对在这个例子中理论的预示价值的评价? 实际上不. 无论如何, 图灵的工作预示到了现代计算机, 并作为一个计算模型是极其重要的, 冯诺伊曼的逻辑背景当然也是他在研究 Edvac 的工作中的一个重要组成部分. 有一件事情更加清楚了, 那就是追踪各种实践中的思想的智力方面的起源是难以置信地困难.

当我在 Courant 研究所报告时, Martin Davis 将他的极好文章 [10] 提供给我. Martin 更肯定冯诺伊曼知道通用机. 我们一致认为, 关于这件事, 要想得到如数学家所喜欢的那种的证明是很难的.

参 考 文 献

[1] Sutherland, S., Finding roots of complex polynomials with Newton's method, Preprint, Institute for Math. Sciences, SUNY, Stony Brook, 1989.

[2] Blum, L., Shub, M. and Smale S., On a theory of computation and complexity over the real numbers: NP-completeness, Recursive functions and Universal Machines, *Bull. Amer. Math. Soc. (New Series)*. Also abstracted in the *IEEE* 1988 *FOCS* **21** (1989), 1–46.

[3] Cook, S. A., The complexity of theorem-proving procedures, Proceedings 3rd ACM STOC, 1989, 151–158.

[4] Karp, R. M., *Reducibility among combinatorial problems*, in *Complexity of Computer Computations*. (R. E. Miller, and J. W. Thatcher, eds.), Plenum, New York, 1972, 85–104.

[5] Garey, M. and Johnson D., *Computers and Intractability*, Freeman, San Francisco, 1979.

[6] Shub, M. and Smale, S., Complexity of Bezout's Theorem I: Geometrical aspects, *J. Amer Math. Soc.* **6** (1993), 459–501.

[7] Shub, M. and Smale, S., Complexity of Bezout's Theorem II: Volumes and probabilities, *Computational Algebraic Geometry* (F. Eysette & A. Galliger, eds.) Progress in Mathematics, Vol. 109, Birkhäuser Boston, (1993), 267–285.

[8] Shub, M. and Smale, S., Complexity of Bezout's Theorem III: Condition number and packing, *J. Complexity* **9**, (1993), 4–14.

[9] Kostlan, E., Random polynomials and the statistical fundamental theorem of algebra. Preprint, University of Hawaii, 1987.

[10] Shub, M., Some remarks on Bezout's Theorem and complexity theory, in *From Topology to Computation*, Proceedings of the Smalefest (M. Hirsch, J. Marsden and M. Shub, eds.) Springer, 1993, 443–455.

[11] Demmel, J., On condition numbers and the distance to the nearest ill-posed problem, *Numer. Math.* **51** (1987), 251–289.

[12] Demmel, J., The probability that a numerical analysis problem is difficult, *Math. Comp.* **50** (1988), 449–480.

[13] Kostlan, E., On the distribution of the roots of random polynomials, in *From Topology to Computation*, Proceedings of the Smalefest, (M. Hirsch, J. Marsden and M. Shub, eds.) Springer, 1993, 419–432.

[14] Kac, M., On the average number of real roots of a random algebraic equation, *Bull. Amer. Math. Soc.* **49** (1943), 314-320.

[15] Shurkin, J., *Engines of the Mind*, W. W. Norton, New York, 1984.

[16] McCulloch, W. S. and Pitts, W., A logical calculus of the ideas imminent in nervous activity, *Bull. Math. Biophys.* **5** (1943), 115-133. Reprinted in McCulloch, W. S., *Embodiments of Mind*, MIT Press, 1988.

[17] Goldstine, H. H., *The Computer from Pascal to von Neumann*, Princeton University Press, Princeton, NJ, 1972.

[18] Hodges, A., *Alan Turing: The Enigma*. Simon & Schuster, New York , 1983.

[19] Davis, M., Logic and the Origin of Modern Computers, in *The Universal Turing Machine. A Half Century Survey* (Rolf Hecken, eds.), Verlag Kammerer & Unverzagt, Hamburg-Berlin; Oxford University Press, Oxford, 1988.

[20] Shub, M., Book review of *Elements of differentiable dynamics and bifurcation theory* by David Ruelle, *Bull. Amer. Math. Soc.* (New Series) **24** (1991), 199-211.

(眭跃飞 译　杨东屏 校)