arXiv:2302.06244v1 [cs.SC] 13 Feb 2023

浮点系数多项式的快速求值与求根

Rémi Imbach 洛林大学、CNRS、INRIA、LORIA，法国南锡 F-54000

Guillaume Moroz
guillaume.moroz@inria.fr 洛林大学、CNRS、INRIA、LORIA，法国南锡 F-54000

摘要

对系数为浮点数的多项式 $f(z) = f_0 + \cdots + f_d z^d$ 进行求值或求根，是一个普遍存在的问题。通过谨慎利用多项式 f 的牛顿多边形，对 f 进行分段近似，我们改进了求值和求根所需运算次数的当前最优上界。特别是，当多项式 f 的系数以 m 个有效位给出时，我们首次提出一种算法，能够找出 f 的所有根，且其相对条件数低于 2^m；该算法所需的位运算次数与 f 的浮点表示位数准线性相关。值得注意的是，无论在理论上还是实践中，我们的新方法都能高效处理系数范围从 2^{-d} 到 2^d 的多项式。

对系数以浮点数表示的多项式进行求值或求根，在许多领域都有广泛应用。给定两个正整数界 m 和 τ，一个浮点数可表示为 $a2^b$，其中 a 和 b 是满足 $|a| \le 2^m$ 和 $|b| \le \tau$ 的整数。这种表示的显著优点是，可以用固定大小的 $m + \log_2 \tau$ 表示具有不同数量级 [6, 14, 22] 的数。

在现有文献中，分析近似求解多项式根的复杂度时，通常会考虑系数的定点表示（[2, 20, 21, 25] 及其中的参考文献）。在这些分析中，系数要么是整数，要么以定点数表示，且所有系数具有统一的误差。这些方法的缺点是，求出多项式的根 $2^\tau - 2^{-\tau} z = 0$ 需要 $O(\tau)$ 位运算，其中 $\widetilde{O}(\cdot)$ 表示省略对数因子。另一方面，如果使用浮点运算，这个简单的方程只需 $O(\log \tau)$ 次位运算即可求解。

在若干应用中会出现系数具有不同数量级的多项式。例如，将广义超几何函数的级数展开截断到 d 阶，例如 e^z，可能会得到一个多项式，其中第 k 个系数的模长为 $\Theta(1/k!)$。矩阵的特征多项式也具有数量级不同的系数。求解

多元多项式方程组可以通过消去变量，将问题化简为一个系数具有不同数量级的单变量多项式来求解 [1]。

文献中的一些研究讨论了求根问题中如何处理较大的数量级，尤其是在使用基于牛顿图和格雷费迭代的方法时（[8, 18, 23, 24, 27] 等）。在这些方法中，中间值可能具有很大的数量级，因此可以采用归一化过程来减小其规模 [11, 13, 19]。然而，他们所描述的算法关于输入多项式的次数是二次的。

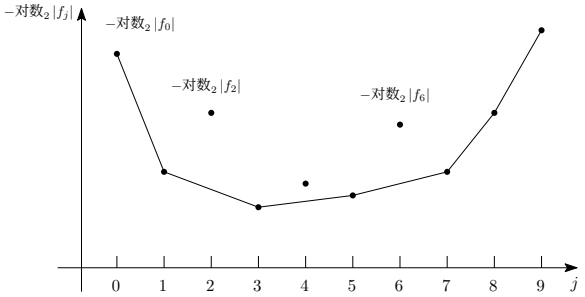

图 1：多项式 f 的牛顿多边形，其中使用赋值 $-\log_2 |f_j|$ 来表示第 j 个系数。

利用 f 的牛顿多边形，我们开发了一种适用于输入系数工作相对精度的新方法。设 f 是一个次数为 d、浮点系数的模长小于 2^τ 并且以 m 个有效位表示的多项式。我们将通过 $O(d/m)$ 个次数为 $O(m)$ 的多项式 g，对 f 进行分段近似，其中每个多项式 g 的系数以 $O(m)$ 个有效数字表示。这种表示将使我们能够改进在 d 个点上计算 f 以及求其根的问题的当前最佳界。

在第 1 节正式陈述主要结果后，我们将在第 2 节和第 3 节说明如何用 $\widetilde{O}(d(m + \log \tau))$ 次位运算计算这样的分段近似。然后，我们将展示如何使用这一数据结构来求值 f（第 4 节），以及如何近似或隔离 f 的根（第 5 节）。最后，我们将介绍原型实现对不同多项式族进行实验的结果（第 6 节）。

1 预备知识

本文中，我们的输入是多项式 $f(z) = f_0 + \cdots + f_d z^d$。我们假设 f 的系数以浮点数表示，我们的目标是设计快速算法，用于求值或求根问题，并达到与使用经典浮点运算算法时相同的误差界。

为此，我们将在第 2 节和第 3 节中，计算定义在复平面某个划分上的 f 分段多项式近似。在陈述主要结果之前，我们先回顾两种将在算法中发挥关键作用的数学工具：条件数和牛顿多边形。

记号。我们将使用以下记号来描述分段多项式近似。对于一个复数 γ 和一个正实数 ρ，我们用 $D(\gamma, \rho)$ 表示以 γ 为圆心、半径为 ρ 的圆盘。对于两个正实数 r_1, r_2，我们用 $R(r_1, r_2)$ 表示

环 $D(0,\ r_2) \setminus D(0,\ r1)$。此外，对于多项式 f，我们定义：

$$\widetilde{f}(|z|) = |f_0| + \cdots + |f_d||z|^d, \qquad \|f\|_1 = |f_0| + \cdots + |f_d|$$
$$\hat{f}(|z|) = \max_{0\le j\le d}(|f_j||z|^j), \qquad f_{\ell,u}(z) = f_l + \cdots + f_u z^{u-\ell}.$$

1.1 相对条件数

在浮点表示中，误差是相对的。令 f_ε 为在 f 的系数上添加了有界相对误差 ε 后得到的多项式。这相当于将所有系数 f_j 乘以 $(1+\varepsilon_j)$，其中 $|\varepsilon_j| \le \varepsilon$。

对于求值问题，任意复点 z 都满足

$$|f(z) - f_\varepsilon(z)| \le \varepsilon\widetilde{f}(|z|)$$

并且这个界是紧的。在定理 1.1 中，我们关注的问题是求出 $f(z)$ 的近似值，并使其具有相同的误差界。

对于求根问题，我们也可以定义相对条件数 [10],[14, §1.6]，它与对 f 系数的相对扰动相关。更确切地说，若 ζ 是 f 的一个根，且 ζ 是 f_ε 中离它最近的根，则 ζ 的相对条件数记为 $\mathrm{cond}(f,\ \zeta)$，定义为 $\lim_{\varepsilon\to 0}|\zeta - .\zeta|/(|\zeta|\varepsilon)$。利用泰勒展开 $f_\varepsilon(\zeta) = (\zeta - .\zeta)f_\varepsilon(\zeta) + O(|\zeta - .\zeta|^2)$ 和 $\neg|f(\zeta) - f_\varepsilon(\zeta)| \le \varepsilon f(|\zeta|)$，可得：

$$\mathrm{cond}(f,\zeta) = \frac{\widetilde{f}(|\zeta|)}{|\zeta||f'(\zeta)|}.$$

这意味着，计算 ζ 的第一个有效位所需的精度位数为 $O(\log(\mathrm{cond}(f,\zeta))$。在第 5 节中，我们重点研究如何用准线性于 $\log(\mathrm{cond}(f,\zeta))$ 且准线性于 d 的位运算次数求出 f 的根。

1.2 牛顿多边形

给定多项式 f 的一个非零复系数 f_j，我们可以将其与数值 $-\log_2|f_j|$ 关联起来。然后，对每个指标 $0 \le j \le d$，令 P_j 为点 $(j, -\log_2|f_j|)$。f 的牛顿多边形是点集 P_j 的下凸包，如图 1 所示。它曾被用于粗略估计多项式 [3, 4, 8, 15, 18, 23, 24, 27] 的根的模。

我们用 H 表示 f 的牛顿多边形，并用它将复平面划分为若干环，使得每个环内 f 的变化有界（第 2 节）。

1.3 主要结果

现在，我们可以陈述主要结果。为便于表述，我们对精度作出一些合理假设，例如 $m > \log d$。

第一个定理表明，我们可以在 d 个点上求值一个次数为 d 的多项式，其复杂度与 d 乘以系数浮点表示的大小呈准线性。特别地，如果系数的模长以 2^τ为界，这意味着我们的算法关于 $d\log\ \tau$呈准线性。这改进了近期的一项结果 [21]，该结果的复杂度关于 $d\tau$呈准线性。

定理 1.1（求值）。设 f是一个次数为 d的多项式，其复数浮点系数的模长小于 2^τ，且 m为正整数。可以在 $\widetilde{O}$（d（$m+\log\tau$））次位运算中计算出一个分段多项式近似 f_{pw}，使得

$$|f(z) - f_{pw}(z)| \le 2^{-m}\widetilde{f}(z)$$

对所有 $z \in \mathbb{C}$均成立。此外，假设 $m > \log d$，并给定 d个复数点 z_1、……、 z_d，其中 $|z_j| \le 2^\tau$，即可求值：

(i) $f_{pw}(z_1)$ *in* $\widetilde{O}(m(m+\log\tau))$ *bit operations,*
(ii) $f_{pw}(z_1),\ldots,f_{pw}(z_d)$ *in* $\widetilde{O}(d(m+\log\tau))$ *bit operations.*

第二个定理使我们能够以关于$\log\tau$呈准线性的位运算次数求出多项式的根，并处理系数很大的多项式，例如两个二元多项式的结式多项式（见第6.2节）。

定理 1.2（求根）。设 f 是一个次数为 d 的多项式，其复数浮点系数的模长小于 2^τ，且 m 是一个正整数。可以通过 $\widetilde{O}(d(m+\log\tau))$ 次位运算，求出多项式 f 的根 ζ 的 m 个最高有效位，使得 $\mathrm{cond}(f,\zeta)$

此外，若 $\kappa = \max_{\zeta|f(\zeta)=0}\mathrm{cond}(f,\zeta)$，则可以通过 $O(d(m+\log\tau+\log\kappa)$ 次位运算，求出 f 的所有根的 $\widetilde{}$ 个最高有效位。

我们的方法分两步计算分段近似。首先，在复平面上的同心圆环划分上计算一个初步近似，详见第 2 节。然后，如第 3 节所述，将每个环均匀划分为角扇区，以此细化分段近似。

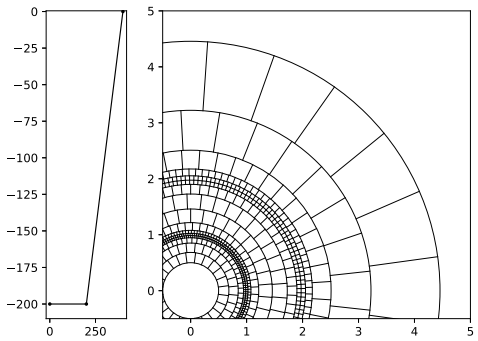

图2：与多项式$(z^{200}-1)(z^{200}-2^{200})$相关的牛顿多边形（左）和扇区划分（右）

2 环上的分段近似

我们的第一个分段近似基于将复平面划分为以原点为圆心的同心圆环 R_n，如图2所示。在算法 1 中，我们计算这些环，并选取 f 的主导单项式。由此得到一个环列表$R_n = R(r_n,\ r_{n+1})$，以及与每个环相关联的一对指标 $(\ell_n,\ u_n)$，使得在 R_n 上计算 $z^{\ell_n}f_{\ell_n,\ u_n}(z)$，足以在给定误差界内计算 f。该算法的目标之一是选择这些环，使多项式 $f_{\ell_n,\ u_n}$ 中单项式的总数与 d 成线性关系。本节的主要结果总结如下引理。

引理 2.1。 给定一个 f次数为 d、具有浮点系数且尾数为 m 位、模长小于 2^τ的多项式，则

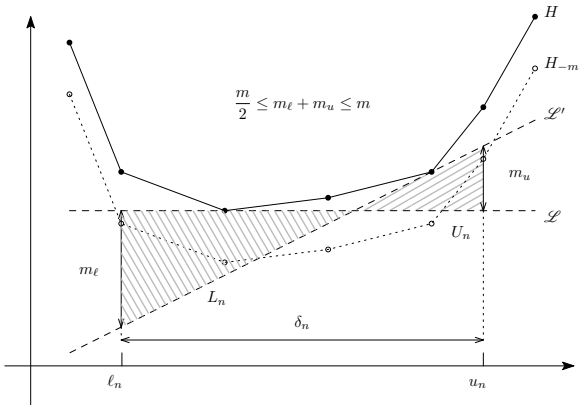

图 3：算法 1 中扫描直线的一步。

~可以在 $O(d(m+\log\tau))$ 次位运算中计算 N环 $R_n = R(r_n, r_{n+1})$，并提取 N次数为 δ_n 的多项式 f_{ℓ_n}，使得：

(i) $$2^{\frac{m}{2}} r_n^{\delta_n} \le r_{n+1}^{\delta_n} \le 2^m r_n^{\delta_n} \quad \text{if } \delta_n \ge 1, \text{ and}$$

(ii) $$\sum_{n=0}^{N-1} (\delta_n + 1) \le 65d + 1.$$

并且对所有 $z \in R_n$：

(iii) $$|f(z) - z^{\ell_n} f_{\ell_n,u_n}(z)| \le (d-\delta_n) 2^{-m} \hat{f}(|z|).$$

我们使用与 f 相关联的牛顿多边形来计算此类近似多项式，从而将满足引理 2.1 的环和指标的计算归结为几何论证。

在第 2.1 节中，我们首先详细介绍算法 1，并证明引理 2.1 的不等式 (i) 和 (iii)。然后在第 2.2 节中，我们证明不等式 (ii)，这是界定所有后续算法复杂度的关键。

2.1 算法

我们的算法以多项式 f 的牛顿多边形 H 为输入。牛顿多边形是点 $(j, -\log_2|f_j|)$ 的下凸包，其中 $0 \le j \le d$；一般情况下，可通过 $O(d\log d)$ 次算术运算计算得出；当这些点已排序时（本例即如此），则可通过 $O(d)$ 次算术运算计算得出。

算法 1 的核心思想是让一条直线与 H 相切，如图 3 所示。给定一条斜率为 s、与 H 相切的直线 L，可按如下非正式方法计算下一条斜率为 s'、与 H 相切的直线 L′，其中 $s' > s$：

1. 令 H_{-m} 为将多边形 H 沿垂直方向平移 $-m$ 后所得的图形。2. 令 ℓ 为 H_{-m} 上位于直线 L 下方的最左点的横坐标。3. 对于新直线 L′，记$-u$ 为 H_{-m} 上位于直线 L′ 下方的最右点的横坐标，$-m_\ell$ 为直线 L 与 L′ 在 ℓ 处的垂直距离，$-m_u$ 为直线 L 与 L′ 在 u 处的垂直距离。4. 从直线 L 的斜率开始，增大直线 L′ 的斜率，直到

$$m/2 \le m_\ell + m_u \le m.$$

在算法 1 中，第 10 行处理上述步骤 1。在第 11 行，我们计算满足 $m_\ell + m_u \le m$ 的最大横坐标 u。然后，我们选择 $s' = s + m/(u-l+1)$ 作为 L′ 的新斜率；这意味着 $m_l + m_u = m(u-l)/(u-l+1) \le m$，并且一旦 $u > l$，我们就有 $m/2 \le m_l + m_u \le m$。因此，若 $\delta_n \ge 1$，则 $(r_{n+1}/r_n)^{\delta_n} = 2^{m(u_n-\ell_n)/(u_n-\ell_n+1)}$ 介于 $2^{m/2}$ 和 2^m 之间，这证明了引理 2.1 的不等式 (i)。

对于不等式 (iii)，令 z 使得 $r_n \le |z| \le r_{n+1}$。这意味着 $\log|z|$ 介于 s_n 和 s_{n+1}之间。令 L″ 为斜率为 $\log|z|$ 且与 H相切的直线。对于小于 ℓ_n的横坐标，它位于 L 下方；对于大于 u_n的横坐标，它位于 L′ 下方。特别地，令$P = (k, y_k)$ 为 H中位于 L″ 上的一个顶点，并令 $Q = (j, y_j)$ 为 H_{-m}中的一点，使得 $j < \ell_n$或 $j > u_n$ 且 $|f_j| \ne 0$。根据构造，我们有 $y_k = -\log_2|f_k|$以及 $y_j \le -\log_2|f_j| - m$。此外，我们知道直线 L″ 位于 P_j下方。由于 L″ 是一条斜率为 $\log|z|$且经过 P_k的直线，这意味着 $y_k + (j-k)\log_2|z| \le y_j$。等价地，这意味着 $-\log_2|f_k| + (j-k)\log_2|z| \le -\log_2|f_j| - m$。两边取平方，这意味着 $|z|^{j-k}/|f_k| \le 2^{-m}/|f_j|$，可改写为$|f_j||z|^j \le 2^{-m}|f_k||z|^k \le 2^{-m}\hat{f}(|z|)$。这意味着对于所有 $z \in R_n$、所有 $j < \ell_n$ 以及所有 $j > u_n$，我们都有 $|f_j||z|^j \le 2^{-m}\hat{f}(|z|)$，进而推出不等式 (iii)。

算法 1： 环和主导单项式

输入： m：一个正整数
H：牛顿多边形上的点列表 $(j,\ h_j)$ 对于 $0 \le j \le d$

输出： 满足引理 2.1 的不等式 (i)、(ii)、(iii)
$(r_n)_{0\le n<N}$：环周围的圆半径列表
$(\ell_n)_{0\le n<N}$：主导单项式的下界指标
$(u_n)_{0\le n<N}$：主导单项式的上界指标

1 **对于** j从 1 到 $d-1$，**执行**
2 　L 中与 H 相切且经过 $(j,\ h_j - m)$ 的直线 ←
3 　S_j^0 ←L 中两条直线的较小斜率
4 　S_j^1 ←L 中两条直线的较大斜率
5 开始：经过 $(0,\ h_0)$ 和 $(1,\ h_1 - m)$ 的直线的 ←斜率
6 结束：经过 $(d-1,\ h_{d-1})$ 和 $(d,\ h_d - m)$ 的直线的 ←斜率
7 $n,\ s_n \leftarrow 1$，开始
8 $r_0,\ r_1,\ \ell_0,\ u_0 \leftarrow 0$，2开始，0，0
9 **当** $s_n <$ 结束 **时执行**
10 　$\ell_n \leftarrow$ 满足 $S_\ell^1 > s_n$的最小指标 ℓ
11 　$u_n \leftarrow$ 满足条件的最大指标 u使得 $(u-\ell_n)(S_u^0 - s) < m$
12 　$s_{n+1} \leftarrow s_n + \frac{m}{u_n - \ell_n + 1}$ **如果** $u_n > \ell_n$ **否则** $S_{u_n+1}^0$
13 　$r_{n+1} \leftarrow 2^{s_{n+1}}$
14 　$n \leftarrow n+1$
15 $r_{n+1}, \ell_n, u_n \leftarrow +\infty, d, d$
16 $N \leftarrow n+1$
17 **返回** $(r_n)_{0\le n\le N}, (\ell_n)_{0\le n<N}, (u_n)_{0\le n<N}$

2.2 复杂度

现在我们可以证明不等式 (ii)，它将使我们能够显著限制第 2.2.2 节中算法 1 的位运算次数。

2.2.1 单项式累计数量的上界。 由于δ_n 的计算方式，其总和呈关于 d 的线性增长并不明显。证明基于几何论证。对于每一步，我们可以

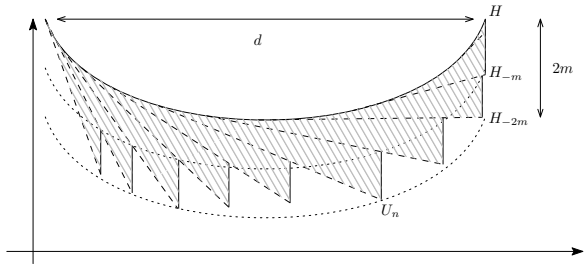

图 4： N 算法中扫描直线的步骤 **1.**

定义一个面积，其大小为 $\Omega(\delta_n m)$，并证明这些面积之和为 $O(dm)$，如图 4 所示。更准确地说，对于满足 $\delta_n \geq 1$的 $0 \leq n < N$，令 V_n 为横坐标 ℓ_n 和 u_n之间的垂直带。令斜率分别为 s_n 和 s_{n+1} 的直线为 L 和 L′，我们定义两个区域：

- U_n 位于 L 上方且位于 L′下方的 V_n 点集
- L_n 位于 L 下方且位于 L′上方的 V_n 点集

首先，由于 V_n 的宽度为 δ_n，且 L 与 L′之间的斜率差为 $m/(\delta_n+1) \geq m/(2\delta_n)$，因此 L_n 和 U_n 的面积之和至少为 $\delta_n m/8$。

接着，令 U为 $0 \leq n < N$对应的 U_n 与 $\delta_n \geq 1$的并集。根据构造，对于不同的指标 n，U_n 两两不同，因此 U的面积大于 U_n的面积之和。同理，令 L 为 L_n 的并集，则L 的面积大于 L_n的面积之和。

最后，令 H_{-2m}为将牛顿多边形 H沿垂直方向平移 $-2m$后得到的多边形。根据构造，指标 u_n的选取满足：L 与 H_{-m} 中横坐标为 u_n 的点之间的垂直距离至多为 m。因此，L 从右侧离开 V_n 的位置高于 H_{-2m}。此外，在横坐标为 ℓ_n 处，直线 L 位于 H_{-m} 上方，所以在垂直带 V_n内，直线 L 始终位于 H_{-2m}上方。类似地，我们可以证明 L′ 位于 H_{-2m} 上方，因为它在横坐标为 u_n 处位于 L 上方，而在横坐标为 ℓ_n处，它与 L 的垂直距离为 $m_{\ell_n} \leq m$。因此，U_n和 L_n都包含在 H与 H_{-2m}之间的带状区域内。这意味着 U（分别是 L）的面积小于 $2dm$。

综合所有几何约束，并令 $|U|$、$|L|$、$|U_n|$、$|L_n|$ 分别表示相应区域的面积，我们得到：

$$\sum_{0\leq n<N \mid \delta_n\geq 1} \frac{\delta_n m}{8} \leq \sum_{0\leq n<N \mid \delta_n\geq 1} |U_n| + |L_n| \leq |U| + |L| \leq 4dm$$

使得 $\sum_{0\leq n<N} \delta_n \leq 32d$并且对于 $\delta_n \geq 1$，我们有 $\delta_n+1 \leq 2\delta_n$，这意味着 $\sum_{0\leq n<N,\ \delta_n\geq 1}(\delta_n+1) \leq 64d$。

为了完成引理 2.1 中不等式 (ii) 的证明，我们还需要考虑 $\delta_n = 0$的情形。当 $\ell_n = u_n$时，就会出现这种情形。满足 $\delta_n = 0$ 的值 ℓ_n互不相同，因此 $\sum_{0\leq n<N,\ \delta_n=0}(\delta_n+1) \leq d+1$。

2.2.2 算法 *1* 的复杂度上界。我们在此证明，算法 1 的位运算次数为 $\widetilde{O}(d(m+\log\ \tau))$。设 N为 **while** 循环的迭代次数。

首先，列表 S^0 和 S^1 可以通过 $O(d)$ 次算术运算计算得出。其次，由于指标 ℓ和 u单调不减，计算序列 ($\ell_n)_{n\leq N}$ $(u_n)_{n\leq N}$ $O(N+)$ 和 ( ) 需要进行 ( ) 次比较。第三，序列 $(r_n)_{n\leq N}$中的各项可以用算术-几何平均 [5]计算得出。这种方法所需正确有效位数的准线性算法。 h_i的模长有界于 τ。因此，变量 s的模长为 $O(\tau)$，只需以 $O(m+\log\ d+\log\ \tau)$ 位正确有效数字进行算术运算和求幂运算，即可使项 r_n达到所需精度。

最后，利用引理 2.1 的不等式 (ii)，N属于 $O(d)$，由此完成证明。

3 角扇区上的分段近似

在上一节中，我们计算了 f 的第一个分段近似，它由多项式 $z^{\ell_n} f_{\ell_n,u_n}(z)$ 给出，定义在复平面的同心圆环分割 R_n 上。尽管这些多项式的单项式总数关于 d 呈线性增长，但单个逼近多项式 f_{ℓ_n,u_n} 的次数可能达到 d。因此，我们需要进一步截断这些多项式，以加快后续求值或求根方法的速度。具体做法是将每个环 R_n 均匀划分为角扇区 $A_{n,k}$，并在每个扇区内计算 f_{ℓ_n,u_n} 的泰勒逼近。更准确地说，设 $\gamma_{n,k}$ 为一个包含 $A_{n,k}$ 的圆盘的圆心，$|\rho_{n,k}|$ 为其半径。然后，我们通过计算多项式 f_{ℓ_n,u_n}与多项式 $\gamma_{n,k}+\rho_{n,k}t$ 复合后所得多项式的前 $4m+1$ 个系数，来逼近 f_{ℓ_n,u_n}。我们将所得多项式记为 $g_{n,k}$。本节的主要结果概括如下引理。

引理 3.1。 给定一个 $f_{\ell_n,\ u_n}$ 满足引理 *2.1* 的条件的多项式，可以通过 $\widetilde{O}(\delta_n(m+$对数 $\tau))$ 位运算计算仿射变量变换 K 的参数 $Z=\gamma_{n,\ k}+\rho_{n,\ k}T$ 以及K 次数为 $\min(\delta_n, 4m)$ 的多项式 $g_{n,\ k}$ 使得：

(i) R_n *is included in the union of the disks* $D(\gamma_{n,k}, |\rho_{n,k}|)$,

(ii) $\|f_{\ell_n,u_n}(\gamma_{n,k}+\rho_{n,k}T) - g_{n,k}(T)\|_1 \leq (\delta_n+1)2^{-m}\hat{f}(r_n)r_n^{-\ell_n}$.

我们使用算法 2 来计算这些逼近。我们将在第 3.1 节描述该算法并给出其复杂度分析，并在第 3.2 节证明不等式 (i) 和 (ii)。

3.1 算法和复杂度分析

在算法 2 中，我们关注由两个半径分别为 r_n 和 r_{n+1}的圆 R_n 定义的一个环。该算法是先前工作中所使用算法的一个变体，其中环由显式公式 [21, §3]给出。主要区别在于，我们需要处理任意环，并且需要为逼近多项式 $g_{n,\ k}$保证不同的误差界。

我们首先计算覆盖环 R_n 的圆盘参数，其中 γ 是 r_n 和 r_{n+1}之间的中点，而 ρ 是区间 $[r_n, r_{n+1}]$ 半径的 3/2 倍。利用这些参数，我们可以验证圆盘 $D(\gamma,\ \rho)$ 包含 R_n 中角度为 $2\rho/\gamma$ 的角扇形（引理 3.2）。因此，覆盖 R_n 所需的圆盘数量至多为 $K \leq 3(\delta/m)+2$，即属于 $O(\max(\delta,\ m)/m)$。接下来，我们要计算对多项式 $f_{\ell_n,\ u_n}$ 中所有 $0 \leq k < K$，经过变量代换 $z = (\gamma+\rho t)e^{i2\pi k/K}$ 后，在 t 中得到的多项式的前 $4m+1$ 个系数；然后我们将说明如何采用婴儿步-巨人步或矩形分裂方法 [6]，以位复杂度 $O(\delta m)$ 完成计算。

为此，请注意，在计算快速傅里叶变换时，$e^{2\pi j/K}$ 对于模 K 同余的指标相等。

算法 2： 分段多项式近似

输入： m：正整数
f：输入多项式
$r_n,\ r_{n+1}$：围绕一个环的两个相邻半径
ℓ_n：主导单项式的下指标
u_n：主导单项式的上指标
输出： 满足引理 3.1 的不等式 (i)、(ii)
$(\gamma_{n,k},\ \rho_{n,k})_{0\le k<K_n}$：变量代换参数：$z=\gamma_{n,k}+\rho_n t$
$(g_{n,k})_{0\le k<K_n}$：逼近多项式列表

```
# 使用 a 计算截断泰勒平移
# 婴儿步-巨人步方法

# A. 变量替换参数
#      z = (γ + ρt)e^{i2π k/K}
γ, ρ, ← (r_{n+1}+r_n)/2, 3/2 (r_{n+1}−r_n)/2
δ ← u_n − ℓ_n
K ← ⌈2π γ/ρ⌉                                   #  K ≤ 3δ/m + 2
m' ← ⌊δ/K⌋

# B. 婴儿步。
#      计算 (γ + ρT)^k      模 T^{4m+1}，归一化
q_0(T) ← (γ+ρT)/(γ+ρ)
执行 k 从 0 到 K − 1 执行
    q_{k+1}(T) ← q_k(T) · (γ+ρT)/(γ+ρ)  模 T^{4m+1}

# C语言。大步法
# C.1 将具有相同指标的单项式按模 K 汇总
M ← m' f̂(γ)/γ^{ℓ_n} (1 + ρ/γ)^δ
对 k 从 0 到 K − 1 执行
    p_k(Z) ← 1/M Σ_{j=0}^{m'} f_{ℓ_n+k+jK} (γ+ρ)^{k+jK} Z^j
# C.2 使用绝对误差的快速复合 2^{−4m}
对 k 从 0 到 K − 1 执行
    r_k(T) ← Σ_{j=0}^{4m} r_{k,j} T^j = p_k(q_K(T)) · q_k(T)
      模 T^{4m+1}

# D. 将近似值扩展到所有角扇区
对于 j 从 0 到 min(δ, 4m) 执行
    r^T_j(W) ← Σ_{k=0}^{K−1} r_{k,j} W^k
    g_{0,j}、……、g_{K−1,j} ← r^T(e^{i2π k/K}) 用于 0 ≤ k < K，使用 FFT
返回(γe^{i2π k/K}, ρe^{i2π k/K}) 对 0 ≤ k < K
        Σ_{j=0}^{4m} M g_{k,j} T^j 对 0 ≤ k < K
```

因此，对于给定的 k（介于 $0\le k<K$之间），我们可以在计算快速傅里叶变换之前，将多项式 $r_k=\sum_{j=0}^{\delta/K} h_{k+jK}$中指标模 K为 k的各项归集起来。这样便可以重新安排运算，在 $\widetilde{O}(\delta m)$ 次位运算内计算出$g_{n,k}$。

更准确地说，在步骤 B（婴儿步）中，我们计算多项式 $\gamma+\rho T$ 模 T^{4m+1} 的幂，直到指数 K，它属于 $O(\delta/m+1)$。在循环的每一步中，利用浮点运算将线性因子归一化，可在 $O(m+\log\tau)$ 次位运算内完成，因为 γ和 ρ 的模长为 $O(2^\tau)$。然后，每次多项式乘法可在$\widetilde{O}(\min(\delta,\ m)m)$ 次位运算内完成。因此，所有多项式 q_k均可在 $O(\min(\delta,\ m)Km+K(m+\log\tau)))$ 内计算完成；或者，由于 Km属于 $O(\max(\delta,\ m))$，也可在 $O(\delta m+K\log\tau)$ 内完成。

然后在步骤 C（巨人步）中，我们结合快速泰勒平移（[27, 定理 8.4]）和快速复合（[26, 定理 2.2]），计算多项式 $r_k$of $q_k\cdot(p_k\circ q_K)\bmod T^{4m+1}$的系数。通过 $\widetilde{O}$ 次位运算，可以使系数的绝对误差小于 2^{-4m}，其中运算量为 $\widetilde{O}(\min(\deg(p_k)\deg(q_k)\ m)m)$。由于 q_k的次数为 K，而 p_k的次数为$m'\le\delta/K$，因此 $p_k\circ q_k$的次数 $\deg(p_k)\deg(q_k)$为 $O(\delta)$，所有 r_k都可以通过 $\widetilde{O}(\min(\delta,m)Km)$ 次位运算计算得出。最后，$\sim Km$ 为 $O(\max(\delta,m))$），因此步骤 C.2 可以通过 $O(\delta m)$ 次运算完成。此外，与步骤 B.类似，在步骤 C.1 中对 p_k 的系数进行归一化需要 $\widetilde{O}(\delta(m+\log\tau))$。

最后，在步骤 D. 中，我们使用 $\min(\delta,\ 4m)$ 次快速傅里叶变换计算多项式$g_{n,k}$，其绝对误差小于 2^{-4m}([27])。这可在 $\widetilde{O}(\min(\delta,\ m)Km)$ 次位运算内完成，即 $\widetilde{O}(\delta m)$。

因此，算法 2 的位运算总次数为$\widetilde{O}(\delta(m+\log\tau))$。

3.2 正确性

这里我们证明引理 3.1 中的不等式 (i) 和 (ii)。

3.2.1 **环覆盖。** 对于引理 3.1 中的不等式 (i)，如果下述引理成立，则环R_n包含于 K_n 个圆盘 $D(\gamma_{n,k},\ |\rho_{n,k}|)$ 的并集之中。

引理 3.2。 圆盘 $D(\gamma,\rho)$ 包含一个角度为 $\frac{\rho}{\gamma}$ 的 R_n角扇形。此外，$-\frac{\gamma}{\rho}\le 1+3\frac{\delta}{m}$。 − −

证明。 $D(\gamma,\ \rho)$所覆盖的角扇形的角度为2β，其中 β是三角形两条边之间的夹角，这两条边的长度分别为 γ和r_{n+1}，该三角形的三边长度为 γ、 r_{n+1}和 ρ。我们有$\beta^2\ge 2(1-\cos(\beta))$。然后利用三角形几何中的论证对$\cos(\beta)$进行界定，可推得 $2\beta>(2/3)\sqrt{5/2}\rho/\gamma>\rho/\gamma$。

然后，对于 $\frac{\gamma}{\rho}$的界，我们注意到它等于$(2/3)(r_{n+1}+r_n)/(r_{n+1}-r_n)=(2/3)(1+1/(r_{n+1}/r_n-1))$。此外， $r_{n+1}/r_n\ge 2^{m/(2\delta)}\ge 1+(\log(2)/2)m/\delta$。结合$2/3<1$和$4/(3\log(2))<3$，即可完成证明。□

3.2.2 **近似误差界。** 为证明引理 3.1 中近似不等式 (ii) 的误差界，我们需要界定两种误差：在 $4m$ 阶截断多项式$f_{\ell_n,u_n}(\gamma+\rho T)$ 时产生的误差，以及在算法 2 中以 2^{-4m} 的绝对误差进行计算时系数上产生的误差。

首先，我们界定由系数误差带来的误差。在步骤 B 和 C.1 中，我们对多项式进行归一化，使其系数绝对值之和小于 1。因此，最后每个系数的绝对误差都小于 $(\delta+1)2^{-4m}M$。我们利用以下关于 $M=m\prime(\hat{}f(\gamma)/\gamma^{\ell})(1+\rho/\gamma)^{\delta}$ 的界，证明其上界为 $(\delta+1)2^{-m-1}\hat{}f(r_n)/r_n^{\ell_n}$。

引理 3.3。对于所有 $m \geq 1$:

$$\|g_{n,k}\|_1 \leq M \leq (\delta+1)2^{3m-1}\hat{f}(r_n)/r_n^{\ell_n}$$

证明。令 j 为满足 $\hat{f}(\gamma) = |f_j|\gamma^j$ 的指标。我们有 $|f_j|\gamma^{j-\ell_n} = |f_j|r_n^{j-\ell_n}(\gamma/r_n)^{j-\ell_n} \leq (\hat{f}(r_n)/r_n^{\ell_n})(r_{n+1}/r_n)^{\delta}$。结合 $(r_{n+1}/r_n)^{\delta} \leq 2^m$，可得 $M \leq m'(\hat{f}(r_n)/r_n^{\ell_n})2^m(1+\rho/\gamma)^{\delta}$。然后可以注意到 $(1+\rho/\gamma)^{\delta} \leq 2(1/\log(2))\,(\delta\rho/\gamma)$。此外，$\rho\ \gamma = 3\ 2\ s_{n+1} - s_n\ 2\ 2\ \rho\ \gamma \leq /(\ /)\tanh((\ )\ 3\ 2\ s_{n+1} - s_n\ 4 \leq 3\ 2\ 4\ m\ \delta \log(\ )/\ )$。因此 $/\log(\ )(\ )/(\log(\ )/\ )(\ /\ )$。由此可得 $(\ 1+\rho\ \gamma\ ^{\delta} \leq 2^{3m/4}\ K \geq 8\pi\ 3\ 2\ \delta\ m\ /\ )$。这也意味着 $(\ /(\log(\ )))(\ /\ )$，因此 $m' \leq m$。最后，这表明 M 有界于 $m2^{7m/4}f\ r_n\ r_n^{\ell_n}\ m \geq 1\ m2^{7m/4} \leq \hat{\ }(\ )/$。此外，对于所有 2^{3m-1}，我们有 。 □

因此，通过对给定 j 下次数大于 $4m$ 的多项式 $|f_j|(\gamma+\rho T)^{j-\ell_n}$ 的系数进行界定，可以得到泰勒展开所产生的误差界。该多项式的 $k\ |f_j|\gamma^{j-\ell_n}\binom{j-\ell_n}{k}\ \rho\ \gamma\ ^k \leq$ 次系 $|f_j|\gamma^{j-\ell_n}\ e\delta\ k\ \rho\ \gamma\ ^k\ \rho\ \gamma$ 数为 $(\ /\ )((\ /\ )(\ /\ ))$。利用上面对 $/$ 的界定，该系数小于 $|f_j|\gamma^{j-\ell_n}((3e\log(2)/4)(m/k))^k$。与之前一样，我们可以将 $|f_j|\gamma^{j-\ell_n}$ 界定为 $2^m\hat{\ }f(r_n)/r_n^{\ell_n}$。对于所有 $k \geq 4m$，第二个因子可以界定为 $(3e\log(2)m/16)^k \leq 1/2^k$。因此，所有次数为 $4m+1$ 或更高的系数之和可界定为 $(\hat{\ }2^m f(r_n)/r_n^{\ell_n})2^{-4m}$。将 j 从 ℓ_n 到 u_n 所产生的误差相加，可得余项误差的界为 $(\delta+1)\hat{\ }f(r_n)/r_n^{\ell_n}2^{-m-1}$。

将两个误差相加，即可得到引理 3.1 中不等式 (ii) 所需的界。

4 求值

根据引理 2.1 和 3.1，算法 2 返回的多项式 $g_{n,\ k}$ 构成角扇区 $A_{n,\ k}$ 上的分段多项式近似，这些角扇区由环 $R(r_n,\ r_{n+1})$ 与圆盘 $D(\gamma_{n,\ k},\ |\rho_{n,\ k}|)$ 的交集确定。下面的推论将此结论形式化。

推论4.1。设 z 为一个复点。则存在 n 和 k，使得 $r_n \leq |z| \leq r_{n+1}$ 且 $z \in D(\gamma_{n,\ k}, |\rho_{n,\ k}|)$。此外，令 $t = (z-\gamma_{n,\ k})/\rho_{n,\ k}$，多项式 $g_{n,\ k}$ 满足:

$$|f(z) - z^{\ell_n}g_{n,k}(t)| \leq (d+1)2^{-m}\hat{f}(z)$$

这给出了一种直接的方法，可在一个点 z 处求值 f，误差为 $O(d2^{-m}\widetilde{f}(|z|))$。首先找到它所属的角扇区 $A_{n,\ k}$，然后计算 $t = (z-\gamma_{n,\ k})/\rho_{n,\ k}$，再求值 $z^{\ell_n}g_{n,\ k}(t)$。要在 d 个点处对 f 求值，我们使用快速多点求值 [17, 28] 对 $g_{n,\ k}$ 求值。

复杂度分析。 假设 z 的模长小于 2^{μ}。通过对最高有效位进行二分查找，可以找到角扇形 $A_{n,\ k}$，总共需要 $\widetilde{O}(\log d+m+\log\ \mu)$ 次位运算。随后，变量代换 $t = (z-\gamma_{n,\ k})/\rho_{n,\ k}$ 也可以在相同复杂度内完成。

最后，我们可以使用经典霍纳法，在 $\widetilde{O}(m^2+\log\ \tau)$ 次位运算内求值 $g_{n,\ k}(t)$；并且利用算术-几何平均法计算对数和指数函数，可以在 $O((m+\log\ \mu+\log\ d)$ 次位运算内求值 $z^{\ell_n} = e^{\ell_n \log z}$ ~。因此，单次求值所需的位运算总数为 $\widetilde{O}(m^2+\log\ d+\log\ \tau+\log\ \mu)$。

此外，要在一组模长小于 $\log\ \mu$ 的 $f\ d$ 个点上进行求值，先按角扇区对这些点进行归集，再使用快速多点求值算法 [17, 28] 在每个扇区上求值 $g_{n,\ k}$，所需的位运算次数为 $\widetilde{O}(d(m+\log\ \mu+\log\ \tau))$。

误差界的计算。 我们还可以利用显式公式 $(d+1)2^{-m}\hat{\ }f(z)$ 高效地计算误差界。计算 $\hat{\ }f(|z|)$ 时，首先需要找到指标 j，使得 $\hat{\ }f(|z|) = |f_j||z|^j$。这相当于在斜率为 $\log\ |z|$、与 H 相切的直线上，找到牛顿多边形 H 上的顶点。如果预先计算出 H 各边的斜率，就可以通过二分查找完成此操作，所需位运算次数为 $\widetilde{O}(\log\ d+m+\log\ \mu)$。随后，计算该公式需要 $\widetilde{O}(m+\log\ \mu+\log\ d+\log\ \tau)$ 次位运算。因此，计算误差界所需的运算次数与近似计算 f 本身相同。

5 求根

我们的根隔离算法包括：首先计算 f 的分段近似，然后求出每个近似多项式的根的近似值 ζ，最后，对于每个近似根 ζ，计算一个以 ζ 为圆心的圆盘；如果 ζ 是良态的，该圆盘包含 f 的一个根 ζ。具体描述见算法 3。

保证我们的算法能够正确近似 f 的所有良态根的主要引理如下。

引理 5.1。 按照算法 *3* 中的记号:

(i) 每个 $D(.\zeta, r) \in .Z$ 都包含 f 的唯一根 ζ，且从圆盘中的任意点出发的牛顿算法都收敛于 ζ(ii) f 的每个根 ζ，只要 $2\log(\mathrm{cond}(f,\zeta)) + 3\log_2(d+1) + 11 < m$，都包含在一个圆盘 $D(.\zeta, r) \in .F$

为确保找到所有满足 $\mathrm{cond}(f,\zeta) \leq 2^m$ 的根 ζ，只需在运行算法前，将 m 改为 $2m + 3\log_2(d+1) + 11$。我们在第 5.1 节证明不等式 (i)，并在第 5.2 节证明不等式 (ii)

对于算法的复杂度，主要耗时部分包括：步骤 A 中分段近似的计算、步骤 B 中的 $O(d/m)$ 根近似，以及步骤 C 中的近似求值 F 和 $.F_{\prime}$。分段近似~的计算和求值可在 $O(d(m+\log\ \tau))$ 次位运算内完成，而每次根近似可在 $\widetilde{O}(m^2)$ 次位运算内完成，由此证明了定理 1.2 中所述的复杂度。

5.1 保证每个返回的圆盘都隔离了 f 的一个根

给定 f 的一个根 ζ，ζ 的牛顿吸引域是使牛顿法收敛到 ζ 的初始点集合。利用 f 的二阶导数上界，Kantorovich 理论可以给出以 ζ 为圆心、且包含于 ζ [7, 的牛顿吸引域中的圆盘半径上界。§3.2],[21, 引理1]。具体而言，设对所有 $y \in D(.\zeta, 4r)$，都有 $r > 2|f(.\zeta)/f_{\prime}(.\zeta)|$，且 $K > |f_{\prime\prime}(y)/f_{\prime}(.z)|$。若 $5rK \leq 1$，则 f 在 $D(.\zeta, r)$ 中有唯一的根 ζ；并且对于所有 $z \in D(\zeta, r)$，从 z 开始的牛顿迭代序列收敛到 ζ。

算法 3： 根隔离

输入： m 一个正整数以及一个次数为 d 的多项式 f

输出： 根 ζ 的隔离圆盘列表，使得 f

$2\log_2(\mathrm{cond}(f,\zeta)) + 3\log_2(d+1) + 11 \le m$

A. 计算分段近似

$A_m(f) \leftarrow f$的分段近似

B. 计算近似根

$Z, \dot{Z} \leftarrow \{\}$

对 $(g,\ \gamma,\ \rho)$在 $A_m(f)$**中执行**

    $.g \leftarrow g$ 的近似因式分解，使得 $\|g - .g\|_1 \le 2^{-4m}\|g\|_1$

    $.Z \leftarrow .Z \cup \{\gamma + \rho.t \mid .t$的根 $.g\}$

C语言。求值 f 和 f' 近似根

$\hat{}_7\ \varepsilon(z) \leftarrow z \mapsto (d+1)2^{-m}\hat{}f(|z|)$

$\varepsilon'(z) \leftarrow z \mapsto d^2 2^{-m}\hat{}f(|z|)/|z|$

$.{}_9\ F \leftarrow$ 对 f 的近似求值，误差为 ε（在 Z 上）

$\dot{F}' \leftarrow$ 对 f' 的近似求值，误差为 ε'（在 $.Z$ 上）

D. 计算隔离圆盘

对 $(.\zeta, .f, .f') \in .Z \times .F \times .F's.t. .f \simeq f(.\zeta)$ 且 $.f' \simeq f'(.\zeta)$**执行**

    $r \leftarrow 2\frac{|\dot{f}| + \varepsilon(\dot{\zeta})}{|\dot{f}'| - \varepsilon'(\dot{\zeta})}$

    $K \leftarrow 2d^3\frac{\hat{f}(|\dot{\zeta}|)}{|\dot{\zeta}|^2(|\dot{f}'| - \varepsilon'(\dot{\zeta}))}$

    如果 $4r < |.\zeta|(2^{1/d} - 1)$ 且 $5rK < 1$

        $Z \leftarrow Z \cup \{D(\dot{\zeta}, r)\}$

返回 L

在我们的情形中，注意，对所有满足 $|z| \le |.\zeta|2^{1/d}$ 的 z，都有 $|f''(z)| \le d^2\widetilde{f}_{2,d}(|z|) \le 2d^2\widetilde{f}_{2,d}(|.\zeta|) \le 2d^2\widetilde{f}(|.\zeta|)/|.\zeta|^2 \le 2d^3\hat{}f(|.\zeta|)/|.\zeta|^2$。这保证了算法 3 第 13 行计算的数值 K 是 $\max_{z\in D(.\zeta,r)}|f''(z)/f'(.\zeta)|$ 的上界。因此，根据 Kantorovich 理论，当满足算法 3 第 14 行的判据时，返回的圆盘可隔离出 f 的一个根。此外，如果两个这样的圆盘相交，那么该根就在它们的交集中，因为从其交集中的任意点开始应用牛顿法，都会收敛到该交集中的唯一根。

5.2 保证每个良态根都包含在返回的圆盘中

给定一个根 ζ，它属于 f，引理 5.1 的不等式 (ii) 的证明分两步完成。首先，我们证明 $.F$ 包含一个足够接近 ζ 的根。然后，我们证明第 14 行的 Kantorovich 判据成立。

在本节中，我们假设 ζ 属于角扇形 $A_{n,k}$，并令 g 为相应的近似多项式，$\gamma + \rho t$ 和为变量替换参数。于是，表示通过算法 3 第 4 行的因式分解得到的近似多项式。最后，令 $.\zeta$ 为最接近的 。点，使得 ζ 满足 $.g((\zeta-\gamma)/\rho) = 0$，并令 $\theta = (\zeta-\gamma)/\rho$ 和 $.\theta = (.\zeta-\gamma)/\rho$，使得 $f(\zeta) = 0$ 且 $.g(.\theta) = 0$。

5.2.1 近似根距离的上界。本节计算 ζ 与 $.\zeta$ 之间距离的上界。首先已知（[12, 定理 6.4e], [3, 定理 9]），对于给定点 θ，θ 与 $.g$ 的最近根 $.\theta$ 之间的距离不超过 $\deg(.g)|.g(\theta)/.g'(\theta)| \le d|.g(\theta)/.g'(\theta)|$。根据构造，$.g$ 满足 $|g(\theta) - .g(\theta)| \le 2^{-4m}\|g\|_1$，且 $|g'(\theta) - .g'(\theta)| \le d2^{-4m}\|g\|_1$。

利用引理 3.3，我们有 $\|g\|_1 \le (\delta+1)2^{-3m-1}\hat{}f(\|\zeta|)/|\zeta|^{\ell_n}$。结合三角不等式，将关于 $|g(\theta) - .g(\theta)|$ 的不等式与引理 2.1 的 (iii) 和引理 3.1 的 (ii) 中的不等式结合，可得 $|.g(\theta)| \le 2(d+1)2^{-m}\widetilde{f}(|\zeta|)/|\zeta|^{\ell_n}$。

对于导数，利用类似的论证可以证明 $|\rho f'_{\ell_n,u_n}(\gamma+\rho\theta) - .g'(\zeta)| \le 2(d+1)^2\widetilde{f}(|\zeta|)/|\zeta|^{\ell_n}$。我们还可以由引理 2.1 的不等式 (iii) 推出 $\rho|f'(\zeta) - \zeta^{\ell_n}f'_{\ell_n,u_n}(\zeta)| \le \rho d^2\widetilde{f}(|\zeta|)/|\zeta|$。最后，注意到 $|\zeta|/\rho \le r_{n+1}/((3/4)(r_{n+1} - r_n)) \le (4/3)(1/(1 - r_n/r_{n+1})) \le (4/3)(1/(1 - 2^{-m}/2^d)) \le (4/3)(2d/m + 1)$。这意味着，对于 $m \ge 4$ 和 $d \ge 1$，我们有 $|\zeta|/\rho \le d + 1$ 和 $|.g'(\theta) - \rho f'(\zeta)/|\zeta|^{\ell}_n| \le 3\rho(d+1)^3 2^{-m}\widetilde{f}(|\zeta|)/|\zeta||\zeta|^{\ell_n}$。

结合前面的所有不等式可知，对于 $m \ge \log_2(\mathrm{cond}(f,\ \zeta)) + 3\log_2(d+1) + 4$，有：

$$\frac{|\zeta - \dot{\zeta}|}{|\zeta|} \le 4(d+1)2^{-m}\,\mathrm{cond}(f,\zeta)$$

5.2.2 保证每个良态根都会被返回。最后，为证明算法 3 的正确性，还需证明：若 $2\,\mathrm{cond}(f,\ \zeta) + 3\log_2(d+1) + 11 \le m$，则在直线 [14] 上应用于 $.\zeta$ 的判据求值结果为真。尽管证明过程较为复杂，其核心思想是变量 r 大致与 $|\zeta - .\zeta|$ 成正比。因此，只要 $|\zeta - \zeta|$ 足够小，rK 就会足够小，从而满足该判据。

首先，若 $m > \mathrm{cond}(f,\ \zeta) + 3\log_2(d+1) + 4$，则 $|\zeta - .\zeta|$ 上的界保证 $|\zeta|2^{-1/d} \le |.\zeta| \le |\zeta|2^{1/d}$。特别地，这意味着 $\widetilde{f}(|.\zeta|) \le 2\widetilde{f}(|\zeta|)$。此外，对于模小于 $|\zeta|2^{1/d}$ 的所有点，都有 $|f'(z)| \le 2d\widetilde{f}(|\zeta|)/|\zeta|$，以及 $|f''(z)| \le 2d^2\widetilde{f}(|\zeta|)/|\zeta|^2$。

。因此，我们可以使用围绕 ζ 的泰勒展开来界定 $|f(\zeta)|$ 和 $|f'(\zeta)|$，其中 $|f(.\zeta)| \le |\zeta - .|$。$\zeta||f'(\zeta)| + d^2|\zeta - .\zeta|^2\hat{}f(|\zeta|)/|\zeta|^2$ 和 $|f'(.\zeta)| \ge |f'(\zeta)| - 2d^2|\zeta - .\zeta|\widetilde{f}(|\zeta|)/|\zeta|^2$。

。令 $C = 6(d+1)\,\mathrm{cond}(f,\zeta)$，由此得到 $|f| + \varepsilon(.\zeta \le |\zeta||f\ \zeta| 2^{-m}C + d2^{-2m}C^3 \quad |f| - \varepsilon\ \zeta \ge)'()()$ 以及 $.''(.)'()|f\ \zeta|(1 - d2^{-m}C^2 - d2^{-m}C)$。

因此，一旦 $d2^{-m}C(C+1) < 1/2$，我们就有 $r \le 6\cdot 2^{-m}C|\zeta|$ 和 $K \le 8d^3\widetilde{f}(|\zeta|)/(|\zeta|^2|f'(\zeta)|)$，使得 $5rK \le 40d2^{-m}C^2$；当 $40d2^{-m}C^2 < 1$ 时，该值小于 1，且 $r/|.\zeta| \le 6\cdot 2^{-m}C2^{1/d}$，当 $18d\cdot 2^{-m}C < \log(2)$ 时，该值小于 $\log(2)/d < 2^{1/d} - 1$。当 $1500(d+1)^3\,\mathrm{cond}^2(f,\zeta) < 2^m$ 时，所有这些约束均得到满足；而当 $m > 2\log_2(\mathrm{cond}(f,\zeta)) + 3\log_2(d+1) + 11$ 时，也满足该条件。

6 基准测试与应用

这里介绍的方法不仅具有理论意义，还可用于解决涉及次数较高、系数数量级各异的良态多项式的问题，例如高斯解析函数的截断级数展开（可得到一阶具有排斥性的多项式），或超几何函数的截断级数展开。我们展示了这些方法在若干类良态多项式上的实际效率，包括与双曲型、椭圆型和平坦分布相关的随机多项式，以及结构更复杂的多项式，例如在通过消元求解多变量方程组时出现的随机二元多项式的结式。

对于 $d >= 0$，我们将双曲型、椭圆型和平坦型基定义为 $(\mathrm{H}_i)_{0\le i\le d} := (1)_{0\le i\le d}$、$(\mathrm{E}_i)_{0\le i\le d} := \left(\sqrt{\binom{d}{i}}\right)_{0\le i\le d}$ 和 $(\mathrm{F}_i)_{0\le i\le d} := \left(1/\sqrt{i!}\right)_{0\le i\le d}$。多项式 f 可以分解为

$$f(z) = \sum_{i=0}^{d} a_i \mathcal{H}_i z^i = \sum_{i=0}^{d} b_i \mathcal{E}_i z^i = \sum_{i=0}^{d} c_i \mathcal{F}_i z^i.$$

当 a_i（分别为 b_i、c_i）是随机数时，称 f 为随机双曲型（分别为随机椭圆型、随机平坦型）。

6.1 实现

本文所述算法的 C语言原型实现[1] 基于 C语言库 Arb[2]（提供多精度球算术）和 MPSolve[3]（参见 [4],，该库提供一个单变量多项式求解器，采用 Ehrlich 迭代（又称 Aberth 迭代）和多精度浮点算术）。

PWEval 实现了第 4 节所述的求值过程，而 PWRoots 实现了算法 3 的根隔离算法；它们以多项式 f 和 m 作为输入，并使用球算术计算 f 的分段近似。PWRoots 输出一组两两不相交的复圆盘 $d_\prime$。如果 $d_\prime = d$，那么 f 的每个根都被隔离在输出中的一个圆盘内。算法 3 的步骤 4 通过调用 MPSolve 中基于世俗方程的求解器来实现。PWEval 还接收复平面 $\mathbb{C}$ 中的点集 Z 作为输入，并为每个 $z \in Z$ 输出一个包含 $f(z)$ 的圆盘，其半径小于 $2^{-m} d\hat{}f(|z|)$。各点逐个求值（不进行多点求值）。

在实现中，我们对算法 1 做了一个小幅变更：在步骤 12 中，我们将 s_{n+1} 计算为 $s_n + c\,\frac{m}{u_n^{-\ell_n+1}}$，其中 c 是 $O(1)$ 中的一个正实参数。在保持整体方法位复杂度不变的同时，这会改变复平面的扇区划分以及分段近似，具体如下：c 越大，环越宽，环中的角扇区数量和近似次数越少，计算分段近似的速度也越快，因为需要计算的近似更少。

图8和图9展示了使用我们的实现并结合 $c = 1$得到的双曲型、椭圆型和平坦型随机多项式在复平面上的牛顿多边形和扇区划分。

我们观察到，PWEval 的瓶颈在于分段近似的计算；我们在 PWEval 中选择 $c = 7/2$ 这使得这一步更快。在 PWRoots 中，主要耗时步骤是使用 MPSolve 逼近逼近多项式的根，其复杂度与多项式的次数呈二次关系；我们选择 $c = 2/5$，以生成次数更低的逼近多项式。

[1]可在 https://gitlab.inria.fr/gamble/pwpoly 获取 [2]https://arblib.org [3] https://numpi.dm.unipi.it/scientific-computing-libraries/mpsolve/

6.2 数值结果

以下给出的时间均为在运行 Linux 的 Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz 机器上测得的串行运行时间，单位为秒。

我们在测试中使用的输入多项式，是随机稠密双曲、椭圆型和平坦多项式的 53 位浮点近似；这些多项式的整数系数因子服从区间 $[2^{-8}, 2^{8}]$ 上的均匀分布。对于根隔离，我们还考虑了两个双变量随机稠密双曲多项式的结式的 53 位浮点近似。

我们生成若干组随机复点，其实部和虚部均为有理数，分子和分母均从 $[2^{-16}, 2^{16}]$ 中均匀选取。

6.2.1 求值。 我们使用 PWEval 对次数不断增加的随机稠密双曲型、椭圆型和平坦多项式进行求值，在 d 个点上进行测试。我们将 PWEval 的运行时间与使用 Arb 中提供的矩形分裂算法（见 [6, 16]）在同一组 d 个点上对 f 求值所需的时间进行比较。我们选择 m（PWEval 的输出精度）以及 Arb 的精度，使得两种求值结果相对于 f 的误差之对数的中位数约为 $-50 \pm 10\%$。在我们的测试中，使用霍纳法则求值始终比使用矩形分裂求值更慢。

图5以直方图展示了这些运行时间，并详细列出了 PWEval 中计算分段近似所用的时间，以及在 d 个点上对其求值所用的时间（在图5中分别称为“pw comp”和“pw eval”）。对于次数高于 40000 的 f，PWEval 的速度快于使用矩形分裂进行求值。还请注意，分段近似一旦计算完成，在单个点上对其求值的速度就比使用矩形分裂进行一次求值（在单个点上）快若干个数量级。

6.2.2 根隔离。我们使用 PWRoots 对测试集中的多项式进行根隔离，精度为 $m = 2(30 + \lceil \log_2(d+1) \rceil)$，这使 PWRoots 始终能够隔离出所有根。

我们首先比较 PWRoots 与 HCRoots 的运行时间。HCRoots 是对 [21][Sec 5, Algo. 3] 中求根算法的 C语言实现。HCRoots 会在给定精度下计算分段近似 m，该实现专用于双曲多项式；根隔离过程嵌套在一个循环中，当所有根尚未隔离出来时，m 会不断加倍。当 m 达到 d 时，则使用复杂度关于 d 为二次的细分根隔离器。表1列出了 PWRoots 和 HCRoots 在 d 取较小值时的运行时间。对于双曲多项式，HCRoots 和 PWRoots 的运行时间都与 d 大致呈线性关系；由于实现上的特定优化，HCRoots 快一个常数倍。HCRoots 不适用于双曲型以外的权重；在这些情况下，其运行时间主要由 f 的二次求解器决定，如表1所示（椭圆型和平坦型情况）。

我们还在图6中比较了 PWRoots 和 MPSolve 对次数高达 20000 的双曲型、椭圆型和平坦型随机稠密多项式的处理情况。可以观察到，PWRoots 的复杂度呈线性 d（对于所考虑的条件良好的多项式），而 MPSolve 的复杂度则呈二次 d。

图?? 比较了 PWRoots 和 MPSolve 在隔离两个次数为 $\sqrt{d}$ 的双变量随机稠密双曲多项式的结式多项式的根时的表现；该结式多项式的次数为 d，其中 $\sqrt{d}$ 最大为 100。当 d 大于 60^2时，PWRoots 的速度开始超过 MPSolve。

参考文献

[1] S. Basu、R. Pollack 和 M.-R. Roy。《实代数几何中的算法》。Springer Berlin Heidelberg，德国柏林海德堡，2006年。[2] Ruben Becker、Michael Sagraloff、Vikram Sharma 和 Chee Yap。基于佩尔莱检验和牛顿迭代的近乎最优复根隔离细分算法。《符号计算杂志》，86：51–96，2018年。[3] Dario A. Bini 和 Giuseppe Fiorentino。多精度多项式求根器的设计、分析与实现。《数值算法》，23(2)：127–173，2000年6月。[4] Dario A Bini 和 Leonardo Robol。求解世俗方程与多项式方程：一种多精度算法。《计算与应用数学杂志》，272：276–292，2014年。[5] Richard P. Brent。多精度求零方法与初等函数求值的复杂度。载于 J.F. Traub 编，《解析计算复杂性》，第151–176页。Academic Press，1976年。[6] Richard P Brent 和 Paul Zimmermann。《现代计算机算术》，第18卷。剑桥大学出版社，2010年。[7] Jean-Pierre Dedieu。《不动点、零点与牛顿法》。数学与应用。Springer Berlin Heidelberg，德国柏林海德堡，2006年。[8] Xavier Gourdon。《代数学基本定理的算法》。研究报告 RR-1852，INRIA，1993年。[9] Ronald L Graham 和 F Frances Yao。求简单多边形的凸包。《算法杂志》，4(4)：324–331，1983年。[10] Stef Graillat。浮点运算中多项式单零点的精确计算。《计算机与应用数学》，56(4)：1114–1120，2008年。[11] A. A. Grau。论格雷夫过程应用中的数值范围缩减。《ACM杂志》，10(4)：538–544，1963年10月。[12] Peter Henrici。《应用与计算复分析》，第1卷。Wiley，纽约，1974年。[13] Peter Henrici。《应用与计算复分析》，第3卷：离散傅里叶分析、柯西积分、共形映射构造、单叶函数，第41卷。John Wiley & Sons，1993年。[14] Nicholas J Higham。《数值算法的精度与稳定性》。SIAM，2002年。[15] Rémi Imbach 和 Victor Y Pan。多项式求根的根半径与细分。载于《科学计算中的计算机代数：第23届国际研讨会，CASC 2021，俄罗斯索契，2021年9月13日至17日，论文集23》，第136–156页。Springer，2021年。[16] Fredrik Johansson。利用矩形分裂计算参数化全纯序列。载于《第39届国际符号与代数计算研讨会论文集》，第256–263页，2014年。[17] Alexander Kobel 和 Michael Sagraloff。快速近似多项式多点求值及其应用。arXiv 预印本 arXiv:1304.8069，2013年。[18] Gregorio Malajovich 和 Jorge P Zubelli。论格雷夫迭代的几何。《复杂性杂志》，17(3)：541–573，2001年。[19] Gregorio Malajovich 和 Jorge P Zubelli。切向格雷夫迭代。《数值数学》，89：749–782，2001年。[20] Kurt Mehlhorn、Michael Sagraloff 和 Pengming Wang。从近似因式分解到根隔离及其在柱面代数分解中的应用。《符号计算杂志》，66：34–69，2015年。[21] Guillaume Moroz。用于单变量多项式逼近的新数据结构及其在根隔离、数值多点求值等问题中的应用。载于《2021年IEEE第62届计算机科学基础年会（FOCS）》，第1090–1099页。IEEE，2022年。[22] Jean-Michel Muller、Nicolas Brisebarre、Florent De Dinechin、Claude-Pierre Jeannerod、Vincent Lefevre、Guillaume Melquiond、Nathalie Revol、Damien Stehlé、Serge Torres 等。《浮点运算手册》。Springer，2018年。[23] Alexandre Ostrowski。关于格雷夫方法以及多项式和劳朗级数的零点的研究。《数学学报》，72(1)：157–257，1940年。[24] Victor Y Pan。复多项式零点的逼近：改进的韦尔四叉树构造与改进的牛顿迭代。《复杂性杂志》，16(1)：213–264，2000年。[25] Victor Y. Pan。单变量多项式：数值因式分解与求根的近乎最优算法。《符号计算杂志》，33 733，2002 5 701( ): – 。[26] Peter Ritzmann。一种快速数值算法，用于复系数幂级数的复合。《理论计算机科学》，44：1–16，1986年。[27] Arnold Schönhage。以计算复杂性表述的代数学基本定理。手稿。图宾根大学，德国

，1982。[28] Joris van der Hoeven。《数值幂级数的快速复合》。技术报告 2008-09，巴黎第十一大学，奥赛，法国，2008年。

A 基准测试

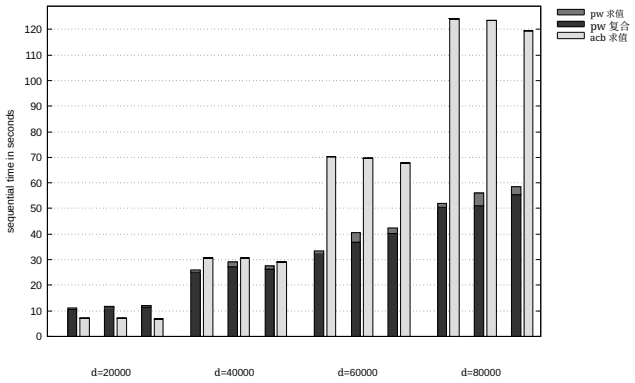

图5：使用 PWEval 和 Arb 中的矩形分裂求值，对次数为 d、位于 d 随机点处的随机稠密多项式进行求值。

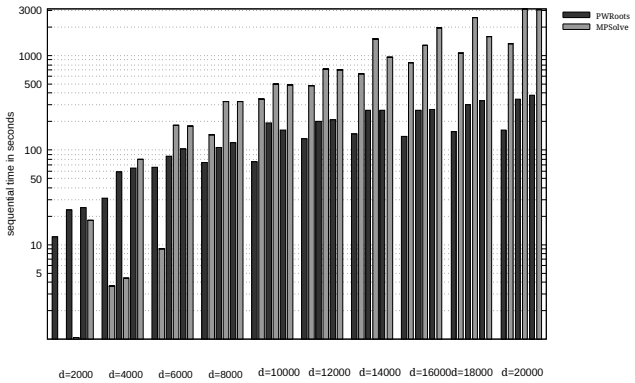

图6：使用 PWRoots 和 MPSolve，对次数递增的随机稠密双曲型、椭圆型和平坦多项式进行复根隔离。

	$d = 200$		$d = 400$		$d = 800$		$d = 1600$	
	PWRoots	HCRoots	PWRoots	HCRoots	PWRoots	HCRoots	PWRoots	HCRoots
双曲的	1.10	0.23	2.50	0.43	6.84	0.92	13.20	1.95
椭圆型	1.63	10.8	3.48	70.2	11.5	611	17.8	> 999
flat	1.69	19.6	3.46	68.7	9.46	511	22.0	> 999

表1：使用 PWRoots 和 HCRoots 隔离随机稠密双曲、椭圆型和平坦型多项式的复根所需的运行时间（秒）；这些多项式的次数逐渐增加 d。

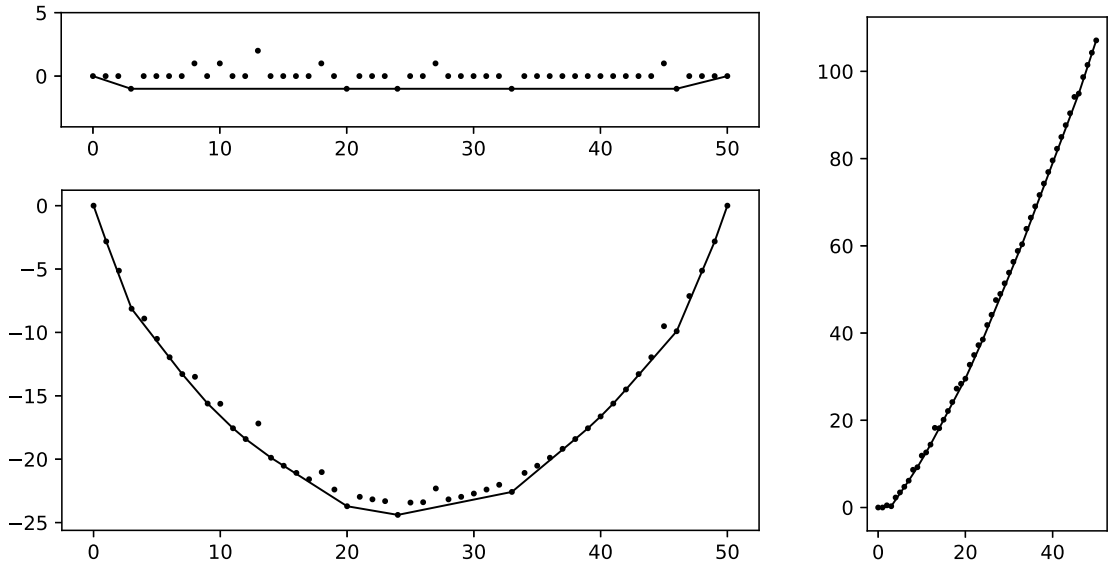

图 8：牛顿多　　项式的双曲的（左上）、椭圆型（左下）和平坦型（右）多项式，次数为50

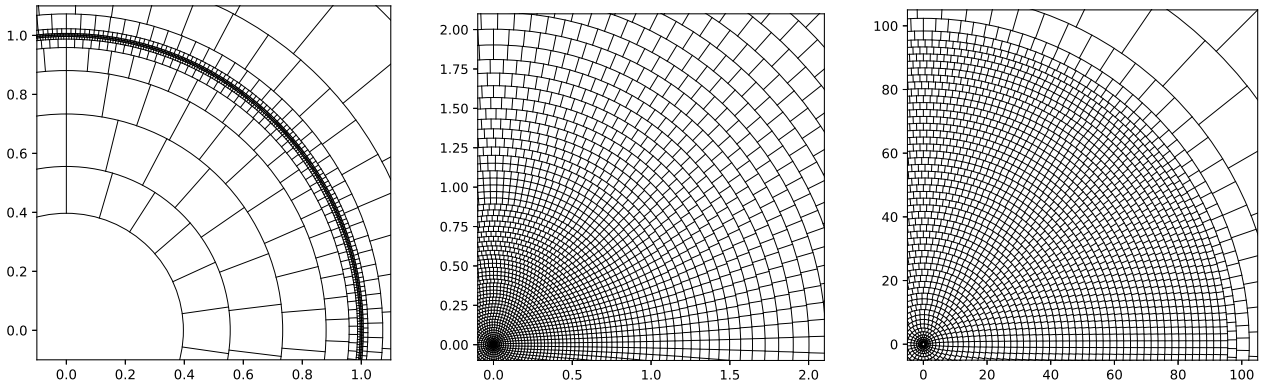

图 9：扇区划分　　与双曲的（左）、椭圆型（中）和平坦型（右）多项式相关的扇区，次数为100　　00