多项式乘法与快速傅里叶变换

于海鑫

2017211240

2019 年 9 月 24 日

多项式

- 以 x 为变量的**多项式**定义在数域 F 上，将函数

$$A(x) = a_{n-1}x^{n-1} + \cdots + a_2x^2 + a_1x^1 + a_0$$

  表示为形式和 $A(x) = \sum_{i=0}^{n-1} a_i x^i$ 的形式
- $a_0, a_1, \cdots, a_{n-1}$ 被称为多项式的**系数**
- 如果 $A(x)$ 的最高次的非零系数是 a_k，则称 $A(x)$ 的**次数**为 k
- 任何严格大于多项式的次数的整数都是该多项式的**次数界**

多项式的表示

系数表达

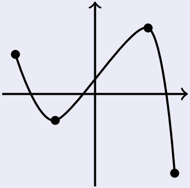
对于多项式 $A(x)$ ，其**系数表达**为由其系数组成的向量

$$\vec{a} = (a_0, a_1, \cdots, a_{n_1})$$

点值表达

次数界为 n 的多项式 $A(x)$ 的**点值表达**为 n 个点值对构成的集合

$$\{(x_0, y_0), (x_1, y_1), \cdots, (x_{n-1}, y_{n-1})\}$$



举例

对于多项式:

$$A(x) = x^3 + x^2 + 1$$

- $A(x)$ 的系数为 3
- $A(x)$ 的次数界为 $4, 5, \cdots$ 等全部大于 3 的数
- $A(x)$ 的系数表示为 $(1, 1, 0, 1)$
- $A(x)$ 的一个点值表达为 $\{(-1, 1), (0, 1), (1, 3), (2, 13)\}$

表达形式的转换

系数表达 $\Rightarrow$ 点值表达 aka 求值

直接对选取的点进行求值即可，时间复杂度为 $\Theta(n^2)$

点值表达 $\Rightarrow$ 系数表达 aka 插值

对于给定的点值对集合 $\{(x_0, y_0), \cdots, (x_{n-1}, y_{n-1})\}$ 使用**拉格朗日公式**

$$A(x) = \sum_{k=0}^{n-1} y_k \frac{\prod_{j \neq k}(x - x_j)}{\prod_{j \neq k}(x_k - x_j)}$$

即可求出 $A(x)$ 的系数表达。时间复杂度为 $\Theta(n^2)$。

多项式的乘法

对于多项式乘法，如果 $A(x).B(x)$ 均为次数界为 n 的多项式，则它们的**乘积** $C(x)$ 是一个次数界为 $2n-1$ 的多项式，对于任一属于多项式定义域的 x，都有 $C(x)=A(x)B(x)$

多项式的乘法

当多项式为**系数**表达时，其乘积的计算方法相当简单，对于表示为

$$\vec{a} = (a_0, a_1, \cdots, a_{n-1})$$

$$\vec{b} = (b_0, b_1, \cdots, b_{n-1})$$

的多项式 $A(x).B(x)$，其乘积 $C(x)$ 的系数表达为：

$$\vec{b} = (c_0, c_1, \cdots, c_{2n-1})$$

其中

$$c_i = \sum_{j=0}^{i} a_j b_{i-j} \qquad \Rightarrow \Theta(n^2)$$

$\vec{c}$ 被称为 $\vec{a}$ 和 $\vec{b}$ 的卷积，记为 $\vec{c} = \vec{a} \otimes \vec{b}$

多项式的乘法

当多项式为**点值**表达时，对于任一点 x_k，由都有 $C(x_k) = A(x_k)B(x_k)$。同时注意到

$$degree(C) = degree(A) + degree(B)$$

因此当 $A(x)$ $B(x)$ 的点值表达分别为 (注意对于 $A(x), B(x)$ 需要 $2n$ 个点值对)

$$A : \{(x_0, y_0), \cdots, (x_{2n-1}, y_{2n-1})\}$$

$$B : \{(x_0, y_0'), \cdots, (x_{2n-1}, y_{2n-1}')\}$$

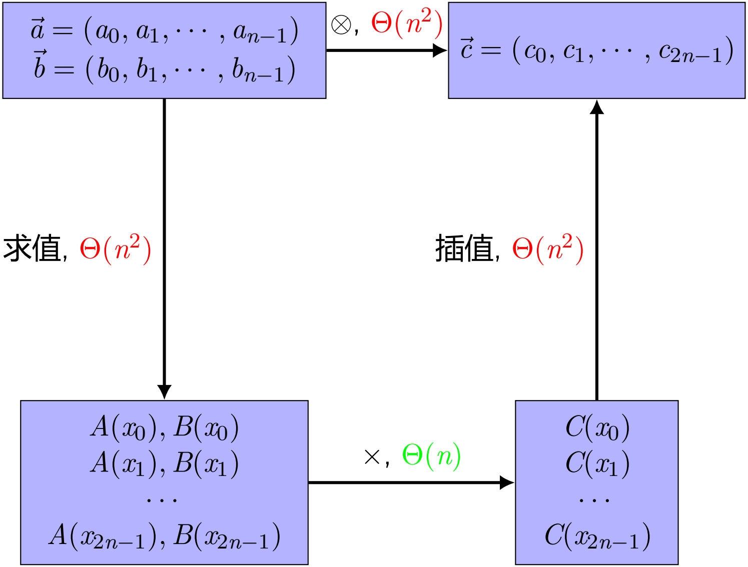
时，$C(x)$ 的点值表达应为：

$$C : \{(x_0, z_0), \cdots, (x_{2n-1}, z_{2n-1})\}$$

其中

$$z_i = y_i y_i' \qquad \Rightarrow \Theta(n)$$

多项式的乘法


$\vec{a} = (a_0, a_1, \cdots, a_{n-1})$
$\vec{b} = (b_0, b_1, \cdots, b_{n-1})$
$\otimes$, $\Theta(n^2)$
$\vec{c} = (c_0, c_1, \cdots, c_{2n-1})$
求值, $\Theta(n^2)$
插值, $\Theta(n^2)$
$A(x_0), B(x_0)$
$A(x_1), B(x_1)$
$\cdots$
$A(x_{2n-1}), B(x_{2n-1})$
$\times$, $\Theta(n)$
$C(x_0)$
$C(x_1)$
$\cdots$
$C(x_{2n-1})$

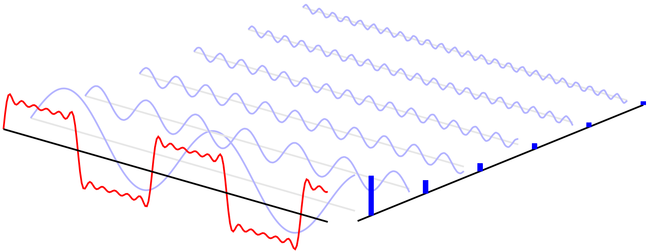
Can we do better?

使用**快速傅里叶变换**，我们可以在 $\Theta(n \log n)$ 的时间复杂度下完成两种表示形式的转换

此时我们可以在 $\Theta(n \log n)$ 的时间复杂度内计算多项式的乘积

傅里叶变换

- 有史以来最伟大、最深刻的发现之一
- 将时域信号转换为频域信号
- ~~不开门课专门讲这个是计算机院学生的损失~~



N 次单位根

称方程 $\omega^n = 1$ 在 $\mathbb{C}$ 上的 n 个解

$$\omega_k = \exp^{\frac{2k\pi i}{n}} = \cos(\frac{2\pi k}{n}) + i\sin(\frac{2\pi k}{n})$$

为 n 次单位根

显然，N 次单位根有如下性质：

- $\omega_{dn}^{dk} = \omega_n^k$ (Cancellation Lemma)
- 当 n 为偶数时，$(\omega_n^k)^2 = (\omega_n^k)^2$ (Halving Lemma)
- 当 $n \geq 1$ 时，对于 $k \neq 0$ 且 k 不被 n 整除，有 $\sum_{j=0} n - 1(\omega_n^k)^j = 0$ (Summation Lemma)

~~证明留做习题~~

离散傅里叶变换（DFT）

可用于求次数界为 n 的多项式 $A(x)$ 在 $(\omega_n^0, \omega_n^1, \cdots, \omega_n^{n-1})$ 处进行求值。为了简化问题，在这里我们假设

- n 是 2 的幂
- A(x) 被表示为系数向量 $\vec{a} = (a_0, a_1, \cdots, a_{n-1})$

我们定义，对于上文所述多项式 $A(x)$，向量 $\vec{y} = (y_0, y_1, \cdots, y_{n-1})$ 被称为系数向量的**离散傅里叶变换**，其中

$$y_k = A(\omega_n^k) = \sum_{i=0}^{n-1} a_i \omega_n^{ki}$$

可记

$$\vec{y} = DFT_n(\vec{a}) \qquad \Rightarrow \Theta(n^2)$$

快速傅里叶变换（FFT）

"The most important numerical algorithm of our lifetime"
by G. Strang, 1994

分而治之

$$A^{[0]}(x) = a_0 + a_2 x + a_4 x^2 + \cdots + a_{n-2} x^{\frac{n}{2}-1}$$

$$A^{[1]}(x) = a_1 + a_3 x + a_5 x^2 + \cdots + a_{n-1} x^{\frac{n}{2}-1}$$

$$A(x) = A^{[0]}(x^2) + xA^{[1]}(x^2)$$

$$T(n) = 2T(\frac{n}{2}) + \Theta(n) \qquad \Rightarrow \Theta(n \log n)$$

正确性证明

对于 $y_0, y_1, \cdots y_{n/2-1}$ 有

$$\begin{aligned} y_k &= y_k^{[0]} + \omega_n^k y_k^{[1]} \\ &= A^{[0]}(\omega_n^{2k}) + \omega_n^k A^{[1]}(\omega_n^{2k}) \\ &= A(\omega_n^k) \end{aligned}$$

对于 $y_{n/2}, y_{n/2+1}, \cdots y_{n-1}$ 有

$$\begin{aligned} y_{k+n/2} &= y_k^{[0]} - \omega_n^k y_k^{[1]} \\ &= y_k^{[0]} + \omega_n^{k+n/2} y_k^{[1]} \\ &= A^{[0]}(\omega_n^{2k}) + \omega_n^{k+n/2} A^{[1]}(\omega_n^{2k}) \\ &= A^{[0]}(\omega_n^{2k+n}) + \omega_n^{k+n/2} A^{[1]}(\omega_n^{2k+n}) \\ &= A(\omega_n^{k+n/2}) \end{aligned}$$