再一次、试着理解 Möbius 反演。

Zetian Lin

January 16th, 2018

注意

- This PDF version is made on December 5th 2023, almost six years after from the original HTML version.
- 有些证明的部分，我没有用 Dirichlet 卷积，所以证明本身会被挑剔的人责备不严谨。
- 下文会用 $[A]$ 代表「如果 A 为真，则为 1；否则为 0」。
- 这点东西可能对于 competitive programming 而言足够了，但是肯定还有很多不够的东西；对于不够的东西只能让阁下自行补全了

1 Möbius 反演

1.1 算术函数 [3]

对于某个函数 $f(x)$，假如 $x \in \mathbb{N}^+ \wedge f(x) \in \mathbb{C}$，就说它是算术函数。而当对于这样的（就是说，是一个算术函数的）$f(x)$，假如：

$$\forall nm, gcd(n,m) = 1 \rightarrow f(nm) = f(n) + f(m)$$

，则说它是**加性**的；当然，类似地，假如：

$$\forall nm, gcd(n,m) = 1 \rightarrow f(nm) = f(n)f(m)$$

，则说它是**积性**的。假如不需要这个前件，即：

$$\forall nm, f(nm) = f(n) + f(m)$$

或：

$$\forall nm, f(nm) = f(n)f(m)$$

，则称为**完全加性/积性**的。

1.2 Möbius 反演

Möbius 反演说的是，假如有两个算术函数 $f(x)$ 和 $g(x)$ 有这样的关系：

$$g(x) = \sum_{d|x} f(x)$$

其中 $d \mid x$ 即「x 可被 d 整除」，如 $\sum_{d|6} d = 1 + 2 + 3 + 6$；那么则有：

$$f(x) = \sum_{d|x} \mu(d) g(\frac{n}{d})$$

其中 $\mu(x)$ 定义如下：

- $\mu(1) = 1$
- $\mu(x) = (-1)^a$，其中 a 是 x 的不同的、质因子的个数。如 $\mu(12) = (-1)^2 = 1$，因为 $12 = 2^2 \times 3$，不同的质因子只有 2 和 3，所以是 2 个，所以是 $(-1)^2$。$\mu(x)$ 被称作 **Möbius 函数**。

1.3 $(1 * \mu)(n) = [n = 1]$

实际上是在说 $\sum_{d|n} \mu(d) = [n = 1]$。这实际上是 Möbius 函数自身的性质。

1.4 第二类 Möbius 反演

定义 $F(x) = \sum_{x|d} f(d)$，有 $f(x) = \sum_{x|d} \mu(\frac{d}{x}) F(d)$。

2 应用

Möbius 反演可用于简化对于某些数论问题的计算：对于 $f(x)$，我们可以构造出（至少是试图构造出）一个更加简单的 $F(x)$，并应用这一定理将对 $f(x)$ 的计算化为对 $\mu(x)$ 和对 $F(x)$ 的计算。

2.1 互质对 [2]

问题：

$$\forall nmij, \sum_{i=1}^{n}\sum_{j=1}^{m}[gcd(i,j)=1] =?$$

让我们先来考察一下直接计算的复杂度。

- 判断 $gcd(i,j)=1$ 需要的时间是计算出 $gcd(i,j)$ 的时间与判断其是否为 1 的时间之和，是 $O(logk)$；
- 这一操作要经历两层循环（$\sum_{i=1}^{n}\sum_{j=1}^{m}$），是 $O(nm)$。

总复杂度为 $O(nmlogk)$，其中 k 为 $min(n,m)$；在 competitive programming 中，n 和 m 的范围一般都是一样的，所以复杂度可以计为 $O(n^2logn)$。那么这样的复杂度能够承受多大的范围呢？按照以往的经验，1000ms 对应 $1\times10^8\sim2\times10^8$，因此保守计算（在 1000ms 的限制下）n 最大可以有 10^3，这显然是不够的。

试图对其应用 Möbius 反演：

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{d|gcd(i,j)}\mu(d)$$

实际上也就是：

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{x=1}^{min(i,j)}[x\mid gcd(i,j)]\mu(d)$$

理由很简单：只有当 $x\mid gcd(i,j)$ 时才会有 $\mu(d)\times1$，于是就相当于只加了 $d\mid gcd(i,j)$ 的 $\mu(d)$。

而这实际上等于：

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{d=1}^{min(i,j)}[d|i][d|j]\mu(d)$$

假如我们先遍历可能的 d，这实际上就是：

$$\sum_{d=1}^{k}\mu(d)\sum_{i=1}^{n}\sum_{j=1}^{m}[d|i][d|j]$$

有一个直观的理由：先计算要不要取（即 $[d|i][d|j]$）然后再计算 $\mu(d)$，与先计算 $\mu(d)$ 然后再决定是否取，结果是一样的。由上可得：

$$\sum_{d=1}^{k}\mu(d)\left(\sum_{i=1}^{n}[d|i]\right)\left(\sum_{j=1}^{m}[d|j]\right)$$

易知有 $\sum_{i=1}^{n}[d|i]=\lfloor\frac{n}{d}\rfloor$，于是实际上就是：

$$f(x)=\sum_{d=1}^{k}\mu(d)\lfloor\frac{n}{d}\rfloor\lfloor\frac{m}{d}\rfloor$$

其中 $k=min(n,m)$。对其分析复杂度：$\mu(d)$ 可以用 $O(n)$ 的时间使用线性筛先行计算，在这一等式中 $\mu(d)$ 应视为 $O(1)$；我们可以直接遍历 $[1..min(n,m)]$，时间复杂度为 $O(n)$；总计时间复杂度为 $O(n)$。于是，通过使用 Möbius 反演，我们将原本 $O(n^2logn)$ 的计算转变成了 $O(n)$ 的计算。

2.2 HDU 1695

问题：

$$\forall bdk,\sum_{i=1}^{n}\sum_{j=1}^{m}[gcd<i,j>=k]=?$$

其中规定 $<i,j>=<j,i>$。

对于其中的规定先不管（因为可以用简单地用「全部 - 重复部分」的方法解决；而「重复部分」的计算与「全部」的复杂度是一样的），其复杂度跟上一个问题是一样的，为 $O(bdlogk)$，而规定本身对复杂度没有影响。那么，要怎么办呢?

1. 定义 $f[n,m](k)=\sum_{i=1}^{n}\sum_{j=1}^{m}[gcd(i,j)=k]$，再定义：

$$\begin{aligned}F[n,m](k)&=\sum_{k|d}f[n,m](d) &(1)\\&=f[n,m](k)+f[n,m](2k)+f[n,m](3k)+\cdots &(2)\\&=\sum_{x=1}^{\lfloor\frac{min(n,m)}{k}\rfloor}f[n,m](xk) &(3)\end{aligned}$$

。为什么是 $\lfloor\frac{min(n,m)}{k}\rfloor$？是因为这里的 k 实际上是受限的：f 的定义中有 $gcd(i,j)=k$，可知 k 最大只能是 $min(i,j)$，即 $min(n,m)$。

2. 通过第二类 Möbius 反演得到 $f[n,m](k)=\sum_{x=1}^{\lfloor \frac{min(n,m)}{k} \rfloor}\mu(x)F[n,m](xk)$。

3. 考察 $F[n,m](k)$。实际上有：

$$F[n,m](k)=f[n,m](k)+f[n,m](2k)+\cdots \tag{4}$$

$$=\sum_{i=1}^{n}\sum_{j=1}^{m}[k|gcd(i,j)] \tag{5}$$

$$=\lfloor\frac{n}{k}\rfloor\lfloor\frac{m}{k}\rfloor \tag{6}$$

4. 整合得 $f[n,m](k)=\sum_{i=1}^{\lfloor \frac{min(n,m)}{k} \rfloor}\mu(i)\lfloor\frac{n}{ik}\rfloor\lfloor\frac{m}{ik}\rfloor$

考察其复杂度：对于 $\lfloor\frac{min(n,m)}{k}\rfloor$，最坏情况有 $\lfloor\frac{min(n,m)}{k}\rfloor\sim min(n,m)\sim n$，即 $O(n)$；$F[n,m](k)$ 的计算很容易知道是 $O(1)$；对于 μ，可以先进行 $O(nlogn)$ 的直接按定义计算或 $O(n)$ 的线性筛进行计算。于是总体复杂度为 $O(nlogn+n\times 1)\sim O(nlogn)$ 或者 $O(n)$（使用线性筛）。

2.3 HYSBZ 2005

要意识到题目中的 k 实际上是 $gcd(i,j)-1$。于是问题为：

$$\forall nm,\sum_{i=1}^{n}\sum_{j=1}^{m}[2(gcd(i,j)-1)+1]=?$$

但是我们不直接对此构造 $f(k)$；即，我们不计算 $2(gcd(i,j)-1)+1$ 的和，而是计算 $gcd(i,j)=k$ 的个数与 $2k+1$ 的乘积；这样就将问题变成了计算 $\sum_k\left[(2k+1)\sum_{i=1}^{n}\sum_{j=1}^{m}[gcd(i,j)=k]\right]$。与上一个问题类似，这里的 k 也是受限的。计算的总复杂度为 $O(n^2)$。

3 优化 [1]

在上一节最后一个问题中我们看到了直接解决的时间复杂度为 $O(n^2)$（我可从来都没有说可以按这个方法 Accepted），按照题目给出的数据范围，这是不够的。我们需要重新考虑一下能够缩减计算时间的地方。用 $f[n,m](k)=\sum_{i=1}^{n}\sum_{j=1}^{m}[gcd(i,j)=k]$ 作为示例对象：

- 对于 μ，我们最好就只能做到 $O(n)$ 了。打表另谈，但是打 10^5 级别的表……且不说会不会有对源代码长度的限制，我们真的需要做得这么硬核打那么长的表吗?

- 我们已经得到 $f[n,m](k)=\sum_{i=1}^{\lfloor \frac{min(n,m)}{k} \rfloor}\mu(i)\lfloor\frac{n}{k}\rfloor\lfloor\frac{m}{k}\rfloor$，但是显然，我们肯定对 $\sum$ 做不了什么。
- 于是我们就只剩下了 $\lfloor\frac{n}{k}\rfloor$。可这是 $O(1)$ 的，我们真的能够对这个做什么吗？我们似乎什么都做不了。那么该怎么做呢？在上一节的最后一个问题里，我们没有直接计算每「一个」的和，而是对每一个可能的 k 的个数然后再乘以 $2k+1$，算的是积；对于这个问题，我们能否不去计算每一个 $\lfloor\frac{n}{k}\rfloor\lfloor\frac{m}{k}\rfloor$，而是计算它与某种东西的积?

试一试吧。假设 $n=32,m=40,k=2$：

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	...
n/ik	16	8	5	4	3	2	2	2	1	1	1	1	1	1	...
m/ik	20	10	6	5	4	3	2	2	2	2	1	1	1	1	...
mu	1	-1	-1	0	-1	1	-1	0	0	1	-1	0	-1	1	...

我们可以看到，对于 μ 我们确实做不了什么（那是当然的），但是 $\lfloor\frac{n}{ik}\rfloor$ 和 $\lfloor\frac{m}{ik}\rfloor$ 都有重复的部分（如 $i=7$ 和 $i=11$）。因此，假如我们能够计算出这样的重复部分的大小，我们就能够将原来的对 $\lfloor\frac{n}{ik}\rfloor\lfloor\frac{m}{ik}\rfloor$ 进行的多次计算缩减为 1 次：假设对于某个重复部分其长度为 α，那么这个重复部分整体的和等于 $\alpha\left(\sum_{i=A}^{B}\mu(i)\right)\lfloor\frac{n}{ik}\rfloor\lfloor\frac{m}{ik}\rfloor$；我们就可以通过计算每一个重复部分得出同样的结果。

于是问题就变成了：

- 重复部分的个数肯定是要比 n 小的；那么究竟小到什么程度?
- 我们该怎么知道重复部分的长度?
- 就算我们有了重复部分，我们还是要计算 μ 的和，这样的话我们还是需要遍历重复部分里的每一个 i。那么我们用于思考改进方法的时间是不是浪费了?

3.1 优化的程度

结论是，$\lfloor\frac{n}{k}\rfloor$ 的取值不会超过 $2\sqrt{n}$ 个，即这样的重复部分不会超过 $2\sqrt{n}$ 个。

- 当 $1\le d\le\sqrt{n}$ 的时候，由于 d（记住，一直都有 $d\in\mathbb{N}^+$）不会多于 $\sqrt{n}$ 个，所以总共最多只有 $\sqrt{n}$ 个。

- 当 $\sqrt{n} \le d \le n$ 的时候，这时变成了 $1 \le \lfloor \frac{n}{d} \rfloor \le \sqrt{n}$ 了，根据上一条可知 $\lfloor \frac{n}{d} \rfloor$ 不会多于 $\sqrt{n}$ 个，因此与其对应的 d 也不会多于 $\sqrt{n}$ 个。因此共计不会多于 $2\sqrt{n}$ 个。对应上一节最后一个问题，假如进行这样的优化，我们可以将时间复杂度从 $O(n^2)$ 缩减至 $O(n\sqrt{n})$。

3.2 重复部分长度的计算

有一个简单到令人惊愕的方法：对于任意 i，计算下一个最小的令 $\lfloor \frac{n}{j} \rfloor \le \lfloor \frac{n}{i} \rfloor$ 的 j；j 就是下一个重复部分的开始。对这一不等式做如下变形：

$$\lfloor \frac{n}{j} \rfloor \le \lfloor \frac{n}{i} \rfloor \tag{7}$$

$$\Rightarrow j\lfloor \frac{n}{j} \rfloor \le j\lfloor \frac{n}{i} \rfloor \tag{8}$$

$$\Rightarrow \left\lfloor \frac{n}{\lfloor \frac{n}{i} \rfloor} \right\rfloor \le j \tag{9}$$

$$\Rightarrow j \ge \left\lfloor \frac{n}{\lfloor \frac{n}{i} \rfloor} \right\rfloor \tag{10}$$

对于 $\lfloor \frac{m}{j} \rfloor$ 同理。回到上一节最后一个问题。我们在判定重复部分的时候必须要满足对于在范围内的所有 i，$\lfloor \frac{n}{ik} \rfloor$ 和 $\lfloor \frac{m}{ik} \rfloor$ 必须相等，于是我们就需要 $min(n/(n/ik), m/(m/ik))$。

3.3 μ 的前缀和

对于第三个问题，答案是「我们根本就不需要在计算时遍历每一个 i」。思考一下，$\sum_{i=n}^{m} f(i)$（假设有 $1 \le n \le m$）实际上等于 $\sum_{i=1}^{m} f(i) - \sum_{i=1}^{n-1} f(i)$，而这个 $\sum_{i=1}^{n} f(i)$，我们可以在使用线性筛得出 μ 时同时计算，因为计算 $\mu(n)$ 时，（按照同样的方法）我们已经得到了 $\mu(n)$ 和 $\sum_{i=1}^{n-1} \mu(i)$。

4 证明，Dirichlet 卷积

4.1 Dirichlet 卷积 [4]

Dirichlet 卷积指的是：

$$(f * g)(n) = \sum_{d|n} f(d)g(\frac{n}{d}) = \sum_{ab=n} f(a)g(b)$$

来看一下它的性质吧。

- $\forall fg, f * g = g * f$
- $\forall fgh, (f * g) * h = f * (g * h)$
- $\forall fgh, f * (g + h) = f * g + f * h = (g + h) * f$，其中定义 $(f + g)(x) = f(x) + g(x)$。
- $\forall fg, f * \epsilon = \epsilon * f = f$，其中定义 $\epsilon(x) = [x = 1]$

这已经跟上述的 Möbius 反演公式看上去很像了。比较一下：

$$f(x) = \sum_{d|x} \mu(d) g(\frac{n}{d})$$

不对啊？看上去应该是

$$(\mu * g)(n) = \sum_{d|x} \mu(d) g(\frac{n}{d})$$

呀？

其实 Möbius 反演定理说的就是：

$$g = 1 * f \leftrightarrow f = \mu * g$$

定义 $1(n) = 1$，$g(x)$ 实际上就是：

$$g(x) = 1 * f(x) = \sum_{ab=x} 1(a) f(b) = \sum_{ab=x} f(b) = \sum_{b|x} f(b)$$

4.2 证明

$\mu(x)$ 有个特别的性质：

$$1 * \mu = \epsilon$$

，即对于任何 $n > 1$：

$$\sum_{ab=n} 1(a)\mu(b) = \sum_{d|n} \mu(d) = 0$$

，对于 $n=1$ 则有 $\sum_{ab=1} 1(a)\mu(b)=\mu(1)=1$。于是证明一下子就变得很简单了：

$$g=1*f \tag{11}$$
$$\Rightarrow \mu*g=\mu*1*f \tag{12}$$
$$\Rightarrow \mu*g=(1*\mu)*f \tag{13}$$
$$\Rightarrow \mu*g=\epsilon*f \tag{14}$$
$$\Rightarrow \mu*g=f \tag{15}$$

5 证明（第二类），基本地

假设对于 $f(k)$ 我们按照如下的公式得出 $F(k)$：

$$F(k)=\sum_{k|d} f(d)$$

应该知道 $F(k)$ 实际上是：

$$F(k)=f(k)+f(2k)+f(3k)+\cdots$$

因为 $k \mid d$，即 d 整除 k，那么 d 就只可能是某某倍的 k。

现在我们证明这个：

$$f(k)=\sum_{k|d} \mu(\frac{d}{k})F(d) \tag{16}$$

因为我们有 $d=k,2k,3k,\cdots$，于是我们有 $\frac{d}{k}=1,2,3,\cdots$。从此我们可以得

出：

$$
\begin{aligned}
&\sum_{k|d}\mu(\frac{d}{k})F(d)\\
&=\mu(1)F(k)+\mu(2)F(2k)+\mu(3)F(3k)+\cdots &(17)\\
&=\mu(1)[f(k)+f(2k)+f(3k)+\cdots]\\
&+\mu(2)[f(2k)+f(4k)+f(6k)+\cdots]\\
&+\mu(3)[f(3k)+f(6k)+\cdots]\\
&+\cdots &(18)\\
&=\mu(1)f(k)+\{[\mu(1)+\mu(2)]f(2k)\}\\
&+\{[\mu(1)+\mu(3)]f(3k)\}+\{[\mu(1)+\mu(2)+\mu(4)]f(4k)\}\\
&+\cdots &(19)\\
&=\sum_{i=1}^{+\infty}\left[f(ik)\sum_{t|i}\mu(t)\right] &(20)
\end{aligned}
$$

对于后面的 $\sum_{t|i}\mu(t)$ 这一部分，注意每一个 $f(k)$ 前面的 μ 的和：如，对于 $f(4k)$，前面的和是 $\mu(1)+\mu(2)+\mu(4)$，而对于 $f(6k)$ 前面的和是 $\mu(1)+\mu(2)+\mu(3)+\mu(6)$，诸如此类。

设 $A=ik$，现在我们有：

$$
\begin{aligned}
i&=\frac{A}{k} &(21)\\
f(k)&=\sum_{i=1}^{+\infty}\left[f(A)\sum_{t|\frac{A}{k}}(\mu(t))\right] &(22)
\end{aligned}
$$

后面的 $\sum_{t|\frac{A}{k}}(\mu(t))$，实际上就是我们在上面提及的 Möbius 函数的性质：当 $\frac{A}{k}=1$，即 $A=k$ 时，我们有 $\sum_{t|\frac{A}{k}}(\mu(t))=1$；其余时候为 0。

于是我们可以得到：

$$
\begin{aligned}
&\sum_{i=1}^{+\infty}\left[f(A)\sum_{t|\frac{A}{k}}(\mu(t))\right]\\
&=f(k)\times 1+f(2k)\times 0+f(3k)\times 0+\cdots &(23)\\
&=f(k) &(24)
\end{aligned}
$$

参考文献

[1] Sengxin's Blog. 莫比乌斯反演简要笔记, 2016. [Online; accessed 5-December-2023].

[2] Nisiyama_Suzune. [tutorial] math note —möbius inversion, 2017. [Online; accessed 5-December-2023].

[3] Wikipedia contributors. Arithmetic function — Wikipedia, the free encyclopedia, 2023. [Online; accessed 5-December-2023].

[4] Wikipedia contributors. Dirichlet convolution — Wikipedia, the free encyclopedia, 2023. [Online; accessed 5-December-2023].