

初等数论三大定理和缩系乘法群

前言

这篇和算法没什么关系，纯粹是基础知识。

初等数论三大定理，指将整个初等数论框架支撑起来的三个定理，分别是Fermat-Euler（费马欧拉）定理、Wilson（威尔逊）定理和Chinese-Residue（中国剩余）定理。

其中FE定理说明取模意义下缩系（简化剩余系/缩剩余系）集合的乘法构成群，Wilson定理揭示了模为素数的乘法群的结构，而CR定理阐述了怎样将群和群结合起来，即多素因子模数乘法群的结构问题。

它们三者的本质，都是解释缩系乘法群的结构问题。而研究缩系乘法群的结构，最终结论的形式是：奇素数幂次群结构、2的幂次群结构、CR定理，三个定理作为最终的最高结论。

Fermat-Euler定理

内容

设欧拉函数 $\varphi(n)$ 是0到n-1里与n互素的数（缩剩余系）的个数，即缩系乘法群的阶。对于缩系中任一元素a有：

$$a^{\varphi(n)} \equiv 1 \pmod{n}$$

特别地，当n是单个素数p的时候 $\varphi(p)$ 是p-1即：（费马小定理）

$$a^{p-1} \equiv 1 \pmod{p}$$

这其实是群论里的定理。任意一个群，群里任意一个元素，自乘群的阶次，一定会回到单位元。即：元素的阶整除群的阶。

证明也简单：对缩系所有元素同时进行乘法操作，构成缩系元素的一个置换。（也可以采用群论中陪集的方法）

这个定理在数学题或者算法中，一般用于简化幂次。例如快速幂函数。

推广

将研究对象转移到缩系以外。在完系（完全剩余系）中，任一元素a有相似结论：

$$a^{t+\varphi(\frac{n}{\gcd(a,n)})} \equiv a^t \pmod{n}$$

对于足够大的整数t成立。意思是a本身自乘很多次后，也会落入循环中，循环节是n去除a与n最大公约数的缩系元素个数的约数。

并且这个足够大的t一般要求a与n重合的那部分素因数被“消除”干净了，即a与n互质的部分素因数的幂次已

经达到或超过了 n 中的相应幂次。

这个证明是显然的，分素因数讨论即可。

由于欧拉函数的积性，循环节显然是 $\varphi(n)$ 的约数。因此弱化一下就是这样：

$$a^{t+\varphi(n)} \equiv a^t \pmod n$$

这个更方便理解和使用。

Wilson定理

内容

对于任一素数 p ， 1 到 $p-1$ 的乘积，模 p 余 -1 。即：

$$(p-1)! \equiv -1 \pmod p$$

或写为比较常见（方便使用）的形式：

$$(n-2)! \equiv \begin{cases} 1 \pmod n & \text{if } n \text{ is prime} \\ 0 \pmod n & \text{otherwise} \end{cases}$$

等价条件，显然可以用于判定素数，像费马小定理都还有无数个特例存在。但是由于阶乘太大了，且判断余数没有速算法，导致时间复杂度比正常因数分解还要高，所以没人选择这么做。

既然要研究缩系乘法群，那么缩系所有元素乘积自然很重要。Wilson定理说明它是 -1 。

证明也特别简单：数论倒数两两配对即可。只有两个无法配对的数， 1 和 -1 ，因此最终结果是 -1 。

这个定理常用于解决剩余问题，在算法中基本不会遇到。

推广

模不是素数的时候，缩系中所有元素的乘积如何？

对于奇素数的幂次：

$$\prod_{(a,p)=1}^a \equiv (-1) \pmod{p^t}$$

对于 2 的幂次： 4 以下仍然是 -1 ，但是 8 以上全是 1 。

当 n 有原根的时候，即 n 属于 $2, 4, p^a, 2p^a$ 的时候，由于缩系是循环群，每个数显然有唯一的数论倒数配对，只有 -1 不能配对。所以结果还是 -1 。

对于其他的整数 n 情形如何？我们继续考虑配对法，那么关键就变为了求这个方程的解：

$$x^2 \equiv 1 \pmod n$$

最后要计算全体解的乘积。

当 n 被 4 整除的时候， 2 的幂的情形已经解决了。对于非 2 的幂的情形， n 含有其他的奇素因子。设其中一个含 a 个，即 n 含有 p^a 。由中国剩余定理，解分为两部分：

$$x \equiv 1 \pmod{p^a}$$

$$x \equiv -1 \pmod{p^a}$$

每一个奇素数 p 均如此。现在说明，同余于-1的那部分解一定是偶数个，从而全体乘积模 p^a 余数为1。

仍旧用配对的思想。对于每一个同余于-1的解，找另外一个与它配对的同余于-1的解。

模2的幂的那个维度与它不相关。每有一个模4余1的解，必然也有一个模4余-1的解与它配对。因此是偶数个。

同理，模2的幂的部分也可以采用配对法。

那么最终结果为：

$$\prod_{(a,n)=1} a \equiv \begin{cases} -1 & n=1,2,4, p^a, 2p^a, p \text{ is odd prime} \\ 1 & \text{otherwise} \end{cases}$$

中国剩余定理

很简单，不同素因子幂乘起来，对应于缩系乘法群的笛卡尔积。因此缩系乘法群的总体构成一个空间，各个素因子的缩系乘法群互不相干，分别构成相应的维度。

当已知这个数在各个维度的坐标，想求这个数的时候，利用线性代数的知识，先求各个维度上的单位向量，然后向量点乘即可。

单位向量的求法，就是一次不定方程。

缩系乘法群的结构

有个经典事实：群的结构与这个素数是不是2有关，当素数是2的时候群的结构会更加复杂。

模为奇素数幂

构成循环群。生成元叫做原根。

不止这类模有原根，事实上1、2、4、奇素数的幂、2倍奇素数的幂都有，也就是说这些缩系乘法群也是循环群，而其余的模都没有。

模为2的幂

当为1、2、4的时候，仍旧是循环群。

当大于等于8的时候，变为一个循环群（元素数为这个数除以4）与 $\{-1,1\}$ 乘法群的笛卡尔积。

著名的Klein四元群与模8的缩系乘法群同构。

离散对数

写在前面

这是一个天坑。关于离散对数的算法数不胜数，甚至是一个P与NP问题。如果未来的您能找到一个多项式时间求解离散对数问题的算法，那么今天的加密算法将半数失效，您不仅可以凭借这个算法轻松拿到图灵奖和菲尔兹奖，甚至可以改写世界历史。当然，如果您证明了不存在多项式时间的求解离散对数问题算法，相当于找到了P与NP问题的有效反例，照样可以拿到图灵奖和菲尔兹奖，只是无法改写历史的进程了而已。

由于本页面不打算涉及算法，那么这部分的算法计划将于暑假再开一个页面（这是因为烤漆实在没时间）。这里仅谈谈离散对数是怎么来的。

定义

离散对数，就来源于循环群。我们知道，原根是缩系乘法群的生成元，那么每个元素是原根的多少次幂呢？

求解幂次，就是标准的对数运算。

我们知道，在复变函数里，指数函数是以 $2\pi i$ 为周期的，也就是说：

$$\ln re^{i\theta} = \ln r + i\theta + 2k\pi i \quad r > 0 \quad k \in \mathbb{Z}$$

这是因为 e 乘上 $2\pi i$ 就回到了乘法单位元1，和Fermat-Euler定理有着异曲同工之妙。

模 n 下，对于原根 g 如果 g 的 t 次方等于 a 那么有：

$$\log_g a = t + k\varphi(n) \quad k \in \mathbb{Z}$$

t 只是对数的主值，即一个代表，一般取0到 $\varphi(n)$ （左闭右开）之间，以 $\varphi(n)$ 为周期。

注意：这里的周期已经不是模数 n 而是 n 的缩系元素个数，所以模 n 记号仅表示模 n 意义下（大范围），并不是这个式子本身的模。

为避免混淆，这里特地记作等号，不是三横线，并将模 n 记号改写在了左边。

例如模13的生成元是2，那么有表格：

$n \bmod 13$	1	2	4	8	3	6	12	11	9	5	10	7
$\log_2 n \bmod 13$	0	1	2	3	4	5	6	7	8	9	10	11

换底公式

更加神奇的是，如果引入取模下对数这个设定，那么换底公式是成立的，只要底一直是原根，并且除法意义变为模 $\varphi(n)$ 意义下。

$$\frac{\log_{g_0} a}{\log_{g_0} g_1} = \log_{g_1} a$$

适用范围

因为离散对数要求是循环群，需要有原根（生成元），所以适用范围是 $1 \leq a \leq p-1$ （ p 为奇素数）。

像是模2的幂（至少为8），一般对数不能直接引入，因为缩系乘法群是一个循环群与 $\{-1,1\}$ 乘法群的笛卡尔积，不是循环群。但是也有办法：

$\{-1,1\}$ 乘法群方向坐标分量：如果 a 为 $4k+1$ 形式的数，该方向分量为1；如果 a 为 $4k+3$ 形式的数，该方向分量是-1。

因此，对于模2的幂（至少为8）缩系乘法群，只取它的一半，即留下 $4k+1$ 形式的一半，则构成循环群，可以引入离散对数。此时始终有固定的生成元为5。那么所有 $4k+1$ 形式的整数都可以求出以5为底的对数。由于底数都给定了，这个对数的求解甚至都可能写出固定的公式，所以不可能用于加密。

另一半 $4k+3$ 形式的数怎么办？由于大背景是模2的幂（至少为8），每一个 $4k+3$ 形式的数都是 $4k+1$ 形式的数乘一个-1。根据对数将乘法变为加法，问题转化为如何定义：

$$c \pmod{2^t} \log_5 (-1) \pmod{c} > 2^t$$

那么这个东西就很玄妙了。如果希望这个新的离散对数具有两个维度的周期，我们可以借助复数来解决这个问题。而与此同时，我们仍旧希望换底公式是成立的。经过尝试，强行将此式定义为：

$$c \pmod{2^t} \log_5 (-1) = 2^{c-3} \pmod{c} > 3^t$$

（当 c 为3，即模数为8的时候，定义为 $1+i$ ——当然这实在没什么用，因为它同构于Klein四元群，习惯采用别的处理方法）

例如模16，有4个“生成元”（只能跑遍半个缩系）3、5、11、13，可以列表验证换底公式（验算不妨将除法改为计算乘法）仍然成立：

$n \pmod{16}$	1	9	5	13	3	11	7	5
$\log_3 n \pmod{16}$	0	2	$3+2i$	$1+2i$	1	3	$2+2i$	$2i$
$\log_{11} n \pmod{16}$	0	2	$1+2i$	$3+2i$	3	1	$2+2i$	$2i$
$\log_5 n \pmod{16}$	0	2	1	3	$3+2i$	$1+2i$	$2+2i$	$2i$
$\log_{13} n \pmod{16}$	0	2	3	1	$1+2i$	$3+2i$	$2+2i$	$2i$

利用复平面上两个维度同时取模（取模构成矩形）意义下的除法，换底公式仍旧成立。虽然完备，只是这么定义没什么实际用途罢了。

计算模2的幂以5为底给定元素的对数

这里给一个算法，计算模2的 t 次幂缩系中一个元素 a 以5为底的对数。要求 a 必须为 $4k+1$ 形式，因为5的幂在这个缩系乘法群中只能跑遍一半。

（如果 a 是 $4k+3$ 形式，就计算 $-a$ ）

首先，类似于快速幂的思想，先计算5、 5^2 、 5^4在模2的 t 次幂意义下的值，总共有 $t-3$ 个，因为：

$$5^{2^{t-2}} \equiv 1 \pmod{2^t}$$

有规律：5是 $4k+1$ 形式， 5^2 是 $8k+1$ 形式， 5^4 是 $16k+1$ 形式.....。

在模2的幂缩系中 $4k+1$ 形式的数以5为底的对数是奇数 $8k+1$ 形式的数以5为底的对数恰好被2整除（不被4整除） $16k+1$ 形式以5为底的对数恰好被4整除。

因此，对数计算算法设计非常简单：

第一步：计算 $a-1$ 在二进制下末尾含多少个0，假设含 h 个。由于 a 是 $4k+1$ 形式 h 至少为2。这意味着 a 以5为

底的对数恰好被 2^{h-2} 整除。

第二步：因为取模下除法（数论倒数）不好计算，因此改算取模乘法。用a乘上 $5^{2^{h-2}}$ 得到b，那么b-1当中2的幂次一定比a-1要高。

第三步：如果新的b为1，则跳出循环，否则用b代替a回到第一步重新执行。

循环中记录下每一个h-2，这些h-2是单调递增的。

循环结束后，我们得到一个二进制数：在每个得到的h-2处为1，其它处为0。因为我们使用了乘法而不是除法，最后用 2^{t-2} 减去得到的二进制数，就得到了所求的对数。

From: <https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:

<https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:namespace:> 群
Last update: 2020/06/17 17:05

