第 19 卷 第 1 期
2002 年 02 月

工 程 数 学 学 报
JOURNAL OF ENGINEERING MATHEMATICS

Vol. 19 No. 1
Feb. 2002

文章编号: 1005-3085(2002) 01-0014-07

数论在密码学中的应用*

王国俊
(西安交通大学基础科学研究中心, 西安 710049; 陕西师范大学数学研究所, 西安 710062)

摘 要: 以背包原理、RSA 系统和秘密分享方案为例介绍了数论在现代密码学中的应用。

关键词: 背包原理; RSA 系统; 秘密分享
分类号: AMS(2000) 11T 71 中图分类号: TN918. 1 文献标识码: A

1 引 言

语言是人类表达思想和进行交流的基本手段。人们在讲话、写文章或者通信时通常总希望能把自己的意思表达得清清楚楚, 使得任何听到讲话或看到文章或信件的人都能正确地领会自己的意思而不致产生任何误解。但在有些情况下, 人们只希望某些特定的对象能理解自己的语言而不希望局外人从语言中获得任何信息。这种事例古已有之。比如, 我国古代不乏以藏头露尾诗的形式把真正的信息隐藏于整个诗篇中, 从而只让某些掌握了规律的人知晓的例子。再如, 古罗马恺撒大帝通过将拼音字母向后移三位的方法向赛查罗发布命令。更早些时候, 古希腊历史学家波里比阿利用删去了 J 的 25 字母方阵创造了通过用表示行和列的两个字母去表示方阵中的一个字母的方法, 等等。这些都可看作是密码学的雏形。直到 20 世纪中叶, 密码学主要应用于军事或外交方面的消息传送上。随着计算机科学的迅速发展和信息时代的到来, 现代密码学的应用范围已经远远超出了军事与外交领域, 它也从狭义的通信保密理论发展为包含了诸如防篡改技术、防假冒技术、安全协议的制订、身份验证、电子签名、电子货币以及秘密分享方案等多方面内容的枝繁叶茂的学科, 因而也在金融、财贸和商业等领域有重要的应用。现代密码学当然是与计算机科学紧密联系的, 而它所使用的数学工具已涉及数论、布尔函数、Walsh 函数、群论、有限域理论、逻辑学乃至代数几何学中的椭圆曲线理论。不过应用最多的还是数论。关于现代密码学已有许多专著, 如文[1~ 10], 这篇短文的目的则是力求用最通俗的语言通过背包原理、RSA 系统和秘密分享这几个专题向读者说明数论方法在现代密码学中应用之一斑。如果读者能通过本文了解到数论这个数学科学中的“皇后”其实也可以在应用领域大显身手, 乃至加盟于密码学研究的

* 收稿日期: 2001-11-15. 作者简介: 王国俊(1935 年 11 月生), 男, 教授, 博士生导师, 研究方向: 非经典逻辑与拓扑.
基金项目: 国家自然科学基金重点项目, 19831040.

队伍, 那将是作者的最大愿望。

2　数字语言

称人们表达思想或传递信息的原始语言为明文, 称将明文以某种方式变换后所得的语言为密文, 由它可以经过相反的变换而恢复出明文来。为了实现这种从明文到密文、再从密文回到明文的变换, 最方便者莫过于使用拼音文字, 因为拼音文字的字母只有有限多个, 建立了这有限个字母间的变换法则也就得到了明文与密文间的变换法则。而这有限多个字母是容易用数字来表示、变换和传递的。所以现代密码学中的语言, 不论是明文还是密文, 实际上都是数字, 我们称其为数字语言。

以英语的 26 个拼音字母为例, 它们既可以用从 1 到 26 的自然数来表达, 也可以用计算机中通用的二进制数来表达, 因为 26 在 2^4 与 2^5 之间, 用 5 位的二进制数表示这 26 个字母已经够用了。为简便计, 不考虑语言中问号、惊叹号、逗号与句号等的区别, 只考虑语言的停顿间隙, 并一律用空格表示, 那么, 组成数字语言的基本单位就有了, 共 27 个, 如下表所示:

表 1　　数字语言的基本单元表

字母	十进制数字	二进制数字	字母	十进制数字	二进制数字
a	1	0 0 0 0 1	o	15	0 1 1 1 1
b	2	0 0 0 1 0	p	16	1 0 0 0 0
c	3	0 0 0 1 1	q	17	1 0 0 0 1
d	4	0 0 1 0 0	r	18	1 0 0 1 0
e	5	0 0 1 0 1	s	19	1 0 0 1 1
f	6	0 0 1 1 0	t	20	1 0 1 0 0
g	7	0 0 1 1 1	u	21	1 0 1 0 1
h	8	0 1 0 0 0	v	22	1 0 1 1 0
i	9	0 1 0 0 1	w	23	1 0 1 1 1
j	10	0 1 0 1 0	x	24	1 1 0 0 0
k	11	0 1 0 1 1	y	25	1 1 0 0 1
l	12	0 1 1 0 0	z	26	1 1 0 1 0
m	13	0 1 1 0 1	空格	0	0 0 0 0 0
n	14	0 1 1 1 0			

有了这张从字母到数字的转换表, 就可以将普通语言转换为数字语言了。比如, I love you 连中间的空格一起就可按上表中的二进制部分转换为如下的数字语言:

0100100000, 0110001111, 1011000101, 0000011001, 0111110101

这里的逗号是为了读者便于和表 1 对照而加上的, 真正的数字语言中没有这些逗号。

3　背包原理

3.1　背包原理　设想有一个长方体形状的背包, 里面恰好装满一组大小不等、形状各异的积木块。又, 旁边还有一堆积木块。如果把背包里的积木块倒在这一堆积木块里搅匀, 那么再从中挑出一组积木块使它们恰好装满背包是十分困难的。类似的数学问题可表述为:

背包问题:　设 $\boldsymbol{A} = (a_1, \cdots, a_n)$, a_i 是正整数, α 也是正整数。问是否存在 $i_1, \cdots, i_k$ (1

$\leq i_j \leq n, j = 1, \cdots, k)$ 使

$$\alpha = \sum_{j=1}^{k} a_{i_j} \tag{1}$$

$\boldsymbol{A}$ 称为背包矢量或背包序列。

这里 A 就像那一大堆积木块, α 是背包, 能不能从 $\boldsymbol{A}$ 中挑出 $a_{i_1}, \cdots, a_{i_k}$ 恰好装满 α 是很难判断的。用专业语言来说, 这是一个 NP 完全问题。

不过对于某些背包矢量 $\boldsymbol{A}$, 上述问题是容易解决的, 这就是矢量 $\boldsymbol{A}$ 构成超递增数列的情形。

3.2 超递增序列

定义 1 正整数序列 $a_1, a_2, \cdots$ 叫超递增序列, 如果

$$a_i > \sum_{j=1}^{i-1} a_j, \quad i = 2, 3, \cdots \tag{2}$$

这一定义对有限序列也适用。

通俗地说, 每一项都大于它前面各项之和的正整数序列叫超递增序列。下面就是两个超递增序列:

$\boldsymbol{A}_1 = (1, 2, 5, 9, 20, 41)$,

$\boldsymbol{A}_2 = (103, 107, 211, 430, 863, 1718, 3449, 6907, 13807, 27610)$。

对于超递增序列 $\boldsymbol{A}$ 而言, 如果给定了一个正整数 α, 是很容易判断 α 是否能表示为 $\boldsymbol{A}$ 中的若干个项之和的。比如 $\boldsymbol{A} = \boldsymbol{A}_1$, $\alpha = 52$。因为 $\alpha > 41$, 41 必须被选中, 否则其它各项全选也不够 41, 从而更达不到 52。然后算出 $52 - 41 = 11$。因为 11 恰大于 $\boldsymbol{A}$ 中的 9, 9 必须被选中, 否则前面各项加起来也不到 9, 更达不到 11。同理, 算出 $11 - 9 = 2$ 正好是 $\boldsymbol{A}$ 中的项, 这样就得到 $52 = 41 + 9 + 2$。再如 $\boldsymbol{A} = \boldsymbol{A}_2$, $\alpha = 10000$。如果 α 能表示为 $\boldsymbol{A}$ 中的若干项之和, 那么从 α 恰大于 6907 知 6907 是一个加项, 然后算出 $10000 - 6907 = 3093$。3093 恰大于 $\boldsymbol{A}$ 中的 1718, $3039 - 1718 = 1375$, $1375 - 863 = 512$, $512 - 430 = 82$。因为 82 不是 $\boldsymbol{A}$ 中的项, 所以 $\alpha = 10000$ 不能表示为 $\boldsymbol{A}$ 中的若干项之和。

背包系统就是将超递增序列用于秘密地传递数字语言的一种密码系统。事实上, 只要会秘密地传送一个字母, 也就会传送整个语言了。而由表 1 知一个字母可以看成一个 5 位的比特串, 比如 i 可看成 01001。选取一个超递增序列 $\boldsymbol{A} = (1, 2, 3, 9, 20)$, 把 i 也想成一个矢量 $\delta = (0, 1, 0, 0, 1)$ 并作 $\boldsymbol{A}$ 与 δ 的内积:

$$\boldsymbol{A} \cdot \delta = (1, 2, 5, 9, 20) \cdot (0, 1, 0, 0, 1) = 22。$$

基于超递增序列 $\boldsymbol{A}$ 就可以把 i 以数字 22 的形式发送给对方, 对方收到数字 22 后, 如果他(她)也知道这个超递增序列 $\boldsymbol{A}$, 就很容易算出 $(0, 1, 0, 0, 1)$ 这个序列从而可以恢复出文字 i 来。如果敌方(或第三者)截获了数字 22, 因为他(她)不知道 $\boldsymbol{A}$, 也就恢复不出 i 来。但以上所述不是真正的背包系统。在实用的背包系统中, 序列 $\boldsymbol{A}$ 的长度 n 远远大于 5, 可以达到 100。同时序列 $\boldsymbol{A}$ 经过用数论方法进行变换后可将所得的非超递增序列 B 公诸与众, 但控制一些秘钥仅为友方所知。

3.3 背包系统的加密和解密方法

(1) 加密对象 数字语言, 即比特串。以 $n = 10$ 为例, 即以传送两个英文字母为例进行说明。

(2) 加密方法　选取一个超递增序列 $\boldsymbol{A}=(a_1,\cdots,a_n)$ 比如 $\boldsymbol{A}$ 为前文中给出的 $\boldsymbol{A}_2$。以 $\Sigma\boldsymbol{A}$ 记 $\boldsymbol{A}$ 中各项之和, 则 $\Sigma\boldsymbol{A}=55205$。取整数 m 使 $m>\Sigma\boldsymbol{A}$, 比如取 $m=55207$。再取整数 t 使 $1\leq t\leq m$ 且 t 与 m 互素, 即 $(m,t)=1$ 比如取 $t=25236$。用 t 乘 $\boldsymbol{A}$ 的各项后再用 m 去除所得各项, 以 $\boldsymbol{B}$ 记所得余数依次构成的序列, 即 $\boldsymbol{B}=(b_1,\cdots,b_n)$,

$$b_i=(ta_i,\ \mathrm{mod}\ m),\ i=1,\cdots,n,\ n=10, \tag{3}$$

这里 $(ta_i,\mathrm{mod}m)$ 表示 ta_i 被 m 除所得之余数 $(i=1,\cdots,10)$。经计算得

$$\boldsymbol{B}=(4579,50316,24924,30908,27110,17953,32732,16553,22075,53620)$$

这个 $\boldsymbol{B}$ 可以公之于众, 友方敌方都知道, 但请注意 $\boldsymbol{B}$ 已经不是超递增序列了。

设 δ 为待传送的 10 比特数字语言, 比如 $\delta=(0100100000)$。将 δ 与 $\boldsymbol{B}$ 作内积得 $\beta=\boldsymbol{B}\cdot\delta=77426$。这个 77426 就是密文, 敌方截获了也很难从 $\boldsymbol{B}$ 和 β 算出 δ, 因为 $\boldsymbol{B}$ 已经不是超递增的了, 正像想从一大堆积木块 $\boldsymbol{B}$ 中找出恰好装满背包 β 的那些积木块一样。当 $n=100$ 时, 如果不借助其它数学理论, 用 20 世纪 90 年代的计算机去试探性地破译背包系统得花上 30 年的时间!

解密方法　友方在收到信息 β 后是容易基于 $\boldsymbol{B}$ 而恢复出 δ 来的, 因为 t 与 m 这两个密钥是通知了友方的。事实上, 因为 t 与 m 互素, 利用辗转相除法可以求出一个最小的正整数 u 使 $ut\equiv1(\mathrm{mod}\,m)$。在本例中可求得 $u=1061$。有了 u 就可以从 $\boldsymbol{B}$ 中恢复出超递增序列 $\boldsymbol{A}$ 来, 同时也可以恢复出 $\boldsymbol{A}$ 与 δ 的内积 α 来: 用 u 去乘 $\boldsymbol{B}$ 的各项再模去若干个 m 即得 $\boldsymbol{A}$。以 $\boldsymbol{B}$ 的第一项 4579 为例, 因为

$$u\cdot4579=4858319,\ 4858319=88\times55207+103$$

所以 $\boldsymbol{A}$ 的第 1 项为 103。同理可得 $\boldsymbol{A}$ 的其余各项, 求得 $\boldsymbol{A}=\boldsymbol{A}_2$。又, 以 u 乘 β 即得 α:

$$\alpha=\boldsymbol{A}\delta\equiv u\boldsymbol{B}\delta=u\beta=1061\times77426=82148986\equiv970(\mathrm{mod}\,m),$$

因为 $\boldsymbol{A}$ 是超递增序列, 有了 $\alpha=970$ 就很容易求出 $\delta=(0100100000)$。

有趣的是密码学的发展往往是在加密和攻击 (即敌方进行破译的手段) 的相互竞争中实现的, 一种保密方法一旦被提出, 立刻就有人去研究如何去攻击它, 而为了防止这种攻击, 更高级的保密方法就不断产生, 然后它又会遭遇新的攻击。前面曾提到用计算机破译背包系统要花 20 年的事, 那是指最笨的试验性方法。实际上为了破译背包系统, 已经有了相当成功的一套方法, 可以大大缩短破译时间。有兴趣的读者可参看文[10]。

4　RSA 系统

一种应用非常广泛的密码系统由 Rivert, Shamir 和 Adleman 提出, 就是如今被称为 RSA 的密码系统。首先回忆一下数论中的有关概念。设 n 是正整数, 用 $\varphi(n)$ 表示小于 n 且与 n 互素的非负整数的个数, 即

$$\varphi(n)=|\{0\leq x<n\mid x\ \text{与}\ n\ \text{的最大公约数为}\ 1\}| \tag{4}$$

容易看出 $\varphi(1)=1$, $\varphi(2)=1$, $\varphi(3)=2$, $\cdots$, $\varphi(p)=p-1$ (p 为素数)。φ 称为欧拉 φ 函数。数论中的欧拉定理说如果 ω 与 n 互素, 则 $\omega^{\varphi(n)}\equiv1(\mathrm{mod}\ n)$, 即 $\omega^{\varphi(n)}$ 一定是 n 的若干倍再加上 1 (参看文[8])。和背包系统一样, 加密对象也是数字语言, 不过这次我们用十进制数作为数字语言。比如, 考虑两个字母 sa, 由于 s 与 a 在表 1 中分别排在第 19 位和第 1 位, 所以与 sa 对应的数字语言为 1901。

(1) 加密方法　选择两个大素数 p 与 q, 令 $n=pq$, 则易验证 $\varphi(n)=(p-1)(q-1)$。再选

择正整数 d 使 d 与 $\varphi(n)$ 互素且 $1 < d < \varphi(n)$。这时可利用辗转相除法求出 e 使

$$ed \equiv 1 \pmod{\varphi(n)},\ 1 < e < \varphi(n) \tag{5}$$

比如, 取 $p = 47$, $q = 59$ 则 $n = 2773$, $\varphi(n) = 2668$。选择 $d = 157$, 则可算出 $e = 17$。再把明文 $\omega = 1901$ 自乘 $e(=17)$ 次并模 $n(=2773)$ 约化就得到密文 c:

$$c = (\omega^e, \bmod n) = (1901^{17}, \bmod 273) = 1281$$

在实际应用时 n 和 e 都是公之于众的, 称为公钥, 而且 n 比我们为举例说明原理用的 $n = 2773$ 大得多, 比如, n 可以由 100 位数组成, 这时想由 n 分解出两个素数 p 与 q 来几乎是不可能的。正是这一对 p 与 q 以及 d 是秘密密钥。为讲清楚解密方法的原理, 需要一个定理。

(2) 还原定理　设 n, e, d 如上所述, 则　$\omega = ((\omega^e, \bmod n)^d, \bmod n)$。　(6)

(3) 解密方法　所谓解密就是要从 $c = \omega^e$ 密文恢复出 ω 来。由还原定理知, 只需将密文 c 再 d 次方然后再模 n 进行约简即得 ω。如, 密文 $c = 1281$ 由 $d = 157$ 知由 c^{157} 模去若干个 n 即得明文　$(1281^{157}, \bmod 2773) = 1901$。

注　在上例中曾涉及 $(1901^{17}, \bmod 2773)$ 与 $(1281^{157}, \bmod 2773)$ 的计算。看起来似乎计算量大得惊人, 其实不然。以第一个计算为例。$1901^2 \equiv 582 (\bmod 2773)$ 是容易算出的, 由此又容易得出 $1901^4 \equiv 582^2 \equiv 418 (\bmod 2773)$, $1901^8 \equiv 418^2 \equiv 25 (\bmod 2773)$, $1901^{16} \equiv 25^2 \equiv 625 (\bmod 2773)$。最后就可得出 $1901^{17}625 \times 1901 \equiv 1281 (\bmod 2773)$。

5　一种秘密分享方案

秘密是一种不公开的信息, 自然也是语言, 从数字语言的角度看, 一个秘密就是一个数字 s。古代曾有过在一个重要的铁门上锁三把锁, 钥匙分别由三个人保存, 只有三个人都到齐了才能开门的例子。现代密码学中也有类似的情况。早在 1979 年 Shamir 就提出了秘密分享的理论。在所谓 (t, n) 门限秘密分享体制中, 秘密被分成了 n 个份额, 至少要掌握 t 个份额才能获得这个秘密。为了搞清什么是秘密分享, 我们先看一个我国古代韩信点兵中的数学问题。据说韩信在统计士兵的人数时用到了一首诗:

三人同行七十稀, 五树梅花廿一枝, 七子团园正月半, 除百去五便得知。

这里"正月半"表示 15。计算人数的方法是: 将 3 人一组分列所剩的人数(只能是 0, 1 或 2) 乘以 70, 将 5 人一组分列所剩的人数乘以 21, 将 7 人一组分列所剩的人数乘以 15, 然后将以上三个结果相加再减去若干个 105 就得到士兵的人数了。比如, 若士兵 3 人一组站立剩 1 人, 5 人一组站立剩 4 人, 7 人一组站立剩 3 人, 那么士兵人数为

$$1 \times 70 + 4 \times 21 + 3 \times 15 - 105 = 94$$

再如, 若士兵人数被 3, 5 和 7 除所得余数分别为 2, 4 和 5, 则士兵人数可从

$$2 \times 70 + 4 \times 21 + 5 \times 15 = 299。$$

中减去 1 到 2 个 105 而求得。究竟是减去 105 还是减去 210 对统兵的人来说是容易确定的, 因为他当然对有多少士兵是心中有数的。比如, 统兵的人知道自己的士兵是 197 人, 那么就应当从 299 中减去 1 个 105 得 194, 他就发现有 3 人缺席, 如果士兵共有 90 人, 则从 299 中减去两个 105 得 89, 表明有 1 人缺席。容易看出, 105 是 3, 5 和 7 的乘积。至于 70, 21 和 15 是如何得到的本文不打算细说, 因为我们的目的是介绍这一方法在秘享分享策略中的应用。首先注意, 由已知某数被 3, 5 和 7 去除所得的余数去求某数只能得出某数的可能值。比如, 在前面的第二例中士兵人数可能是 194, 也可能是 89。再者, 是减去几个 105 还是加上几个

105 也不确定。比如在上例中如果给299加上105, 则所得的数404被3, 5和7去除的余数仍分别为2, 4和5。再加几个105得509, 614, 719 …等也都满足同样的条件。当然, 如果已知人数不超过105, 那么就只有唯一解89了。

现在假如只告诉你某数被3除余2, 被5除余4而要求出某数, 那么可能的答案就有

14, 29, 44, 59, 74, 89, 104, 119, …

等等。进一步, 如果只告诉你某数被3除余2, 那么可能的答案就更多了:

5, 8, 11, 14, …, 89, 92, …

如果设想89是一个秘密, 由3个人分享, 第一人只知道那是一个被3除余2的数, 第二个人只知道那是一个被5除余4的数, 而第三人则只知道那是一个被7除余5的数, 那么这三人中的每一个人都难以断定这个秘密到底是什么。但如果三个人合在一起, 那就可在一定条件下容易地得出89来。一种秘密分享方案正是基于这种思想而得出的。我们先把以上的数学方法加以推广。下面是国际上通称为中国剩余定理的定理, 证明并不复杂, 参看文[8]。

中国剩余定理[8]　设 $m_1, \cdots, m_r$ 是两两互素的正整数, 若正整数 x 满足条件

$$x \equiv a_1 (\mathrm{mod}\ m_1), \cdots, x \equiv a_r (\mathrm{mod}\ m_r)$$

则存在正整数 $k_1, \cdots, k_r$ 使

$$x \equiv \sum_{i=1}^{r} a_i k_i \quad (\mathrm{mod}\ m_1 \cdots m_r)。$$

注意, 如果已知 $x < m_1 \cdots m_r$ 那么由此定理就可唯一地确定出 x。

现在看基于中国剩余定理的一种秘密分享方案。设 s 是一个秘密, 通常它是一个很大的正整数。我们的目的是要把 s 分成 n 个份额, 使得只有掌握够其中的 t 个份额的人才可获得秘密 s。注意, "分成 n 个份额" 并不是指把 s 拆成 n 个数之和, 而是指给定了与 s 有关的 n 个信息(如, 被7除余5等), 由其中不少于 t 个信息即可决定 s。方法如下: 选择两两互素的 n 个正整数 $d_1, d_2, \cdots, d_n$ 和另一个大于 s 的正整数 p 使得

(i) $d_1 < d_2 < \cdots < d_n$;

(ii) p 与 d_i 互素 $(i = 1, \cdots, n)$;

(iii) 前 t 个 d_i 的乘积 N 大于 p 乘以后 $t-1$ 个 d_i 的乘积 M, 即

$$N = d_1 \cdots d_t > p \cdot d_n \cdots d_{n-t+2} = pM。 \quad (8)$$

举一个小数字的例子来说明上述条件。设 $s = 22, p = 23, n = 5, d_1$ 到 d_5 分别为31, 34, 35, 37, 41, 则各 d_i 两两互素并且也都与 p 互素。这时条件(i), (ii)都成立。设 $t = 2$, 则由 $d_1 d_2 = 1054$ 和 $pd_5 = 943$ 知条件(iii)也成立。

以 L 记 $\frac{N}{P}$ 的整数部分, 即 $L = [\frac{N}{P}]$。在区间 $[0, L-1]$ 中任取一个整数 r, 令 $x = s + rp$, 则由 $s < p$ 知 $x < (r+1)p \leq Lp \leq N$, 从而 $x \in [0, N-1]$。令

$$s_j = (x, \mathrm{mod}\ d_j), j = 1, \cdots, n, \quad (9)$$

即 s_j 是 x 用 d_j 除所得之余数 $(j = 1, \cdots, n)$。$s_1, \cdots, s_n$ 就是基于 s 的 n 个信息份额。如果掌握了其中的 t 个份额 $s_{i_1}, \cdots, s_{i_t}$, 那么由(9)知

$$x \equiv s_{i_1} (\mathrm{mod} d_{i_1}), \cdots, x \equiv s_{i_t} (\mathrm{mod} d_{i_t})。 \quad (10)$$

再由 $x \leq N - 1 < d_1 \cdots d_t \leq d_{i_1} \cdots d_{i_t}$ 即可根据中国剩余定理唯一地确定出 x, 然后就可以由 $s = x - rp$ 得出信息 s 来。但是如果只掌握了 $t-1$ 个份额的信息 $s_{i_1}, \cdots, s_{i_{t-1}}$ 当然仍可

按 $$x \equiv s_{i_1} (\mathrm{mod}\ d_{i_1}),\ \cdots,\ x \equiv s_{i_{t-1}} (\mathrm{mod}\ d_{i_{t-1}}) \tag{11}$$

以及中国剩余定理来估算 x, 但是由(8) 式知这时的模太小了, 即

$$D = d_{i_1} \cdots d_{i_{t-1}} \leq d_n \cdots d_{n-t+2} < \frac{N}{P}。$$

如果用 x_0 表示(11) 式的满足条件 $0 \leq x_0 < D$ 的解, 那么

$$x_0,\ x_0 + D,\ x_0 + 2D,\ \cdots,\ x_0 + pD$$

就都是(11) 式的解且全部位于[0, $N-1$] 之中。究竟应当从哪一个解出发减去 rp 而得s 就有 $p+1$ 种可能性, 从而由 $p > s$ 知有比秘密s 自身还要多的可能性, 所以这些解也就失去了意义。关于秘密分享方案的详细研究可参看文[11]。

参考文献:

[1] 王育民, 刘建伟. 通信网的安全——理论与技术[M]. 西安: 西安电子科技大学出版社, 1999

[2] 肖国镇. 密码计算机和通信系统数据安全[M]. 北京: 人民邮电出版社, 1993

[3] 王育民, 何大可. 保密学——基础与应用[M]. 西安: 西安电子科技大学出版社, 1990

[4] 杨义先, 林须端. 编码密码学[M]. 北京: 人民邮电出版社, 1993

[5] 丁存生, 肖国镇. 流密码学及其应用[M]. 北京: 国防工业出版社, 1994

[6] 王新梅, 肖国镇. 纠错码(原理与应用)[M]. 西安: 西安电子科技大学出版社, 1991

[7] 温巧燕, 钮心忻, 杨义先. 现代密码学中的布尔函数[M]. 北京: 科学出版社, 2000

[8] Koblitz N. A Course in Number Theory and Cryptography[M]. New York: Springer-Verlag, 1987

[9] Kranakis E. Primality and Cryptography[M]. New York: John Wiley and Sons, 1986

[10] Saloma A. Public-Key Cryptography[M]. Berlin: Springer-Verlag, 1990. 中译本: 丁存生, 单炜娟译. 北京: 国防工业出版社, 1999

[11] 张福泰. 可验证秘密分享及其应用研究[D]. 西安电子科技大学博士论文, 2001

[12] Shamir A. How to share a secret[J]. Communications of the ACM, 1979; 24(11): 612-613

The Application of Number Theory in Cryptography

WANG Guo-jun

(Xi'an Jiaotong University, Xi'an 710049; Shanxi Normal University, Xi'an 710062)

Abstract Through three typical examples, how to apply the number theory in cryptography is introduced briefly.

Keywords: The principle of knapsack; RSA; Share in a secret