计算机研究与发展
Journal of Computer Research and Development
ISSN 1000-1239/CN 11-1777/TP
45(9): 1561-1566, 2008

基于Frobenius映射的快速标量乘算法

殷新春[1,2] 侯红祥[1] 谢 立[2]
[1](扬州大学信息工程学院 江苏扬州 225009)
[2](南京大学计算机软件新技术国家重点实验室 南京 210093)
(xcyin@yzu.edu.cn)

Fast Scalar Multiplication Algorithm Based on Frobenius Mapping

Yin Xinchun[1,2], Hou Hongxiang[1], and Xie Li[2]
[1](*School of Information and Engineering, Yangzhou University, Yangzhou, Jiangsu* 225009)
[1](*State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing* 210093)

Abstract Elliptic curve cryptosystem (ECC) is a novel public key cryptosystem, which will be the primary standard for application in the future. The capability of ECC depends on the efficiency of scalar multiplication. Furthermore, fast scalar multiplication algorithm on Koblitz curve is the top demanding task in the research of scalar multiplication. After the reduction of TNAF(k), a super operation algorithm based on Frobenius mapping is proposed, which is Comb algorithm. At pre-compute stage, in order to establish a pre-compute table, the algorithm calculates the coordinate of some points on elliptic curve corresponding to any sequence at a fixed length of r with the help of Frobenius mapping. On the other hand, at evaluation stage, the algorithm employs the reduction of TNAF(k) as well as the pre-compute table to improve the efficiency of the whole Comb algorithm. Because of high performance of Frobenius mapping, Comb algorithm doesn't relate to point doubling. And after arranging of Comb matrix, the quantity of point addition needed by the algorithm in this paper is $1/5 \sim 1/4$ times of that needed by traditional algorithms. In addition, the efficiency of the algorithm is faster at least about 67% than the traditional Comb algorithm with arbitrary length of row in any coordinate.

Key words elliptic curve cryptosystem (ECC); scalar multiplication; Koblitz curve; Frobenius mapping; Comb algorithm

摘 要 标量乘法的效率决定着椭圆曲线密码体制的性能,而Koblitz曲线上的快速标量乘算法是标量乘法研究的重要课题,在标量k的TNAF约简基础上,给出了一种基于Frobenius映射的上层运算:Comb算法.在预计算阶段,该算法利用Frobenius映射对宽度为r的序列计算其对应椭圆曲线上的点,从而建立预计算表,在累加赋值阶段结合约简后的TNAF(k)和预计算表来提高效率.Comb算法基于高效的Frobenius映射无须进行倍点运算,经过Comb矩阵的组合,其所需点加量是传统算法的$1/5 \sim 1/4$,当行数r任意时,其效率在任意坐标下比传统Comb算法提高至少67%.

关键词 椭圆曲线密码体制;标量乘法;Koblitz曲线;Frobenius映射;Comb算法

中图法分类号 TP301.6

收稿日期:2007-05-23;**修回日期**:2008-05-19

基金项目:国家自然科学基金项目(60473012);国家"八六三"高技术研究发展计划基金项目(2007AA012448);江苏省"六大人才高峰"项目(06-E-025)

标量乘法是椭圆曲线密码体制[1]中最基本最耗时的运算,其效率决定着整个椭圆曲线密码体制的性能,近几年 Koblitz 曲线上的标量乘法研究很深入,这类快速算法的高效源于该曲线上的 Frobenius 映射的高效.

自从 Koblitz 将 Frobenius 映射引入标量乘法研究领域[2]后,Solinas 给出了 k 的 TNAF 展式[3-4];Avanzi[5]和 Hankerson 等人[6]分别用不同的方法对 TNAF 展式进行约简;国内白国强等人给出了一类安全椭圆曲线及其快速计算的方法[7];胡磊等人给出特征为 3 的 Koblitz 曲线的快速点乘[8];刘铎等人给出优化扩域上快速标量乘算法[9];也有学者将 Frobenius 映射与双基数系统结合[10-11]等,但上层运算领域并没有很好的快速算法.

本文从 Koblitz 曲线和 Frobenius 映射理论基础出发,在 $TNAF(k)$约简后给出基于 Frobenius 映射的 Comb 算法,使用预计算表结合 $TNAF(k)$的一定组合来提高效率.

1 γ-adic 展式

1.1 Koblitz 曲线

定义 1. 在 F_2 域上的非超线性曲线称为 Koblitz 曲线,也被称为不规则二元曲线,一般都同构于以下方程:

$$E_a: y^2+xy=x^3+ax^2+1, a\in\{0,1\}$$

存在两个非同态的 Koblitz 曲线:E_0 和 E_1,且 $\#E_0(F_2)=4$ 及 $\#E_1(F_2)=2$,那么 $\#E_0(F_{2^m})=4n$ 及 $\#E_1(F_{2^m})=2n'(n,n'\in Z)$. n 和 n'是素数当且仅当 m 是素数,否则存在 $E_a(F_{2^m})$的子域 $E_a(F_{2^d})$,其中 $d|m$.

1.2 Frobenius 映射

令 $a\in\{0,1\}$,$E_a(F_{2^m})$上的 Frobenius 映射是:

$$\gamma: E_a(F_{2^m})\to E_a(F_{2^m}),$$

$$\gamma(\infty)=\infty, \gamma(x,y)=(x^2,y^2).$$

定理 1. Frobenius 映射 γ的性质:

1) 令 $\lambda=(-1)^{1-a}$, E_a上 γ 的特征多项式是 $x^2-\lambda x+2$;

2) $(\gamma^2-\lambda\gamma+2)P=\infty$, $P\in E_a(F_{2^m})$,其中 $\gamma=(\lambda+\sqrt{-7})/2$ 是 γ特征多项式的一个根.

二元域上的平方在多项式基下只需在元素之间插入零,然后约简,同样计算其他运算也很简单,正则基下的所有运算只须循环移位.因此计算任何二元域元素的 Frobenius 映射是很快的[4].那么计算标量乘法 kP 时,将 k 在复环 $Z[\gamma]$展开成 γ-adic 形式,就有:

1.3 范函数

定义 2. $\alpha=a_0+a_1\gamma\in Z[\gamma]$的范数是复数 α的绝对值:

$$N(a_0+a_1\gamma)=(a_0+a_1\gamma)(a_0+a_1\bar{\gamma})=a_0^2+\lambda a_0a_1+2a_1^2.$$

定理 2. 范函数的性质:

1) $N(\gamma)=2$ 以及 $N(\gamma-1)=2^{2-a}$;

2) $N(\gamma^m-1)=\#E_a(F_{2^m})$以及 $N((\gamma^m-1)/(\gamma-1))=\#E_a(F_{2^m})/2^{2-a}$;

3) 复环 $Z[\gamma]$是范函数的剩余类,对于任意的 $\alpha,\beta\in Z[\gamma]$且 $\beta\neq 0$,总存在 $\kappa,\rho\in Z[\gamma]$,使得 $\alpha=\kappa\beta+\rho$成立且 $N(\rho)<N(\beta)$.

1.4 TNAF

定理 2 保证任何正整数 k 都可以表示成以 γ 为基的展式, k 的 γ-adic 展式可将 k 不断除 γ 得余数 u_i,由于 $N(\gamma)=2$,所以余数只能是 -1, 0 或 1,进一步为减少展式中非零量的个数,可对该展式应用 NAF,叫做 γ-adic NAF 或 TNAF.

定义 3. 复环 $Z[\gamma]$中任意元素 κ的 TNAF 是:

$$\kappa=\sum_{i=0}^{l-1}u_i\gamma^i,\ u_i\in\{0,\pm1\}\ \text{且}\ u_iu_{i+1}\neq 0.$$

定理 3. TNAF 的性质:

1) κ有唯一的 TNAF,表示成 $TNAF(\kappa)$;

2) TNAF 非零量的平均密度是 1/3;

3) 如果 $l>30$,那么$\log_2(N(k))-0.55<l<\log_2(N(k))+3.52$.

定理 4. 令 $\alpha=a_0+a_1\gamma\in Z[\gamma]$:

1) α被 γ整除当且仅当 a_0是偶数,同时有 $\alpha/\gamma=(a_1+\lambda a_0/2)-(a_0/2)\gamma$;

2) α被 γ^2整除当且仅当 $a_0\equiv 2a_1\pmod 4$.

对于任意 $\kappa\in Z[\gamma]$,根据定理 4 都可给出其 TNAF(κ),算法见附录 A,但由定理 3 TNAF 的性质 3 可知,任何整数 k 的 TNAF 长度大约是$\log_2 N(k)=\log_2(k^2)=2\log_2 k$,大约是 NAF 长度的两倍.由于计算标量乘法 kP 的效率取决于 k 展式的长度和非零量,所以直接使用 $TNAF(k)$效率不会很高.

2 快速算法

2.1 约 简

如果扩域的度是 m,由定理 2 可知:

$$(\gamma^m-1)(P)=\gamma^m(P)-P=P-P=\infty,\quad \forall P\in E_a(F_{2^m}),$$

如果 $k\equiv h(\mathrm{mod}\ \gamma^m-1)$,就有 $kP=hP$,并且 h 的长度接近 NAF(k)的长度.

进一步可发现:

$$(\gamma^m-1)(P)=\left(\frac{\gamma^m-1}{\gamma-1}(\gamma-1)\right)(P),\ m>1,$$

所以 $k\equiv h'(\mathrm{mod}((\gamma^m-1)/(\gamma-1)))$,同样 $kP=h'P$,并且 h'的长度也接近 NAF(k)的长度.计算 $h'\equiv k(\mathrm{mod}((\gamma^m-1)/(\gamma-1)))$,复环 $Z[\gamma]$内的除法算法必须给定,该算法参见附录 B.

2.2 Frobenius Comb 算法

正整数 k 模$(\gamma^m-1)/(\gamma-1)$后展开成 TNAF(k)后,长度 l 大约为 $\log_2 k$,在该展式左边补零使其长度达到 cr(r 事先给定,$c=\lceil l/r\rceil$),那么对于任意一个 $P\in E_a$,计算 kP 除了可用类似二元法的 γ-add 方法[4]外,还可表示成:

$$kP=\sum_{j=0}^{cr-1}u_j\gamma^j(P)=\sum_{i=0}^{r-1}\left(\sum_{j=0}^{c-1}u_{ci+j}\gamma^j\right)\gamma^{ci}(P)=\sum_{j=0}^{c-1}\gamma^j\left(\sum_{i=0}^{r-1}(u_{ci+j}\gamma^{ri}(P))\right)=\sum_{j=0}^{c-1}\gamma^j\left(\sum_{i=0}^{r-1}(u_{ci+j}P_i)\right),$$

$$P_i=\gamma^{ri}(P),0\leqslant i\leqslant r-1,$$

其中 u_j 为 0 的概率为 2/3.对于该展式,可预先计算 0 到 $r-1$ 行对应的 P_i,然后根据 P_i 计算 0 到 $r-1$ 行中所有可能的情况,建立预计算表 T,最后从 $c-1$ 到 0 列搜索 k 转化的 Comb 矩阵查表 T 累加,过程示例图 1 所示:

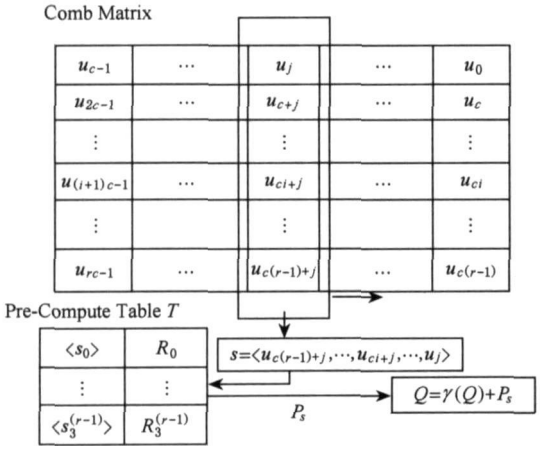

Fig. 1 Demonstration of Frobenius Comb algorithm.

图 1 Frobenius Comb 算法示例图

图 1 中预计算表 T 在计算标量乘法之前存储好,Frobenius Comb 算法使用 $TNAF(k)$排列成的 Comb 矩阵来计算 kP,从左到右扫描 $c-1$ 到 0 列,用一列的元素查预计算表 T,最后使用 γ-add 方法累加.具体算法步骤如下:

算法 1. Frobenius Comb 算法.

输入:$P\in E_a(F_{2^m})$;r 及 $k=\sum_{j=0}^{cr-1}u_j\gamma^j$,$u_j=0,\pm1$;

输出:kP.

Step1. 预计算

① $P_0=O$; $s=$??; $T=$?(s, O)?; $T_1=$??;

② for $i=1$ to $r-1$

$P_i=\gamma^c(P_{i-1})$;

③ for $i=r-1$ to 0

while (! T)

(s, R?$\leftarrow T$;

$s\leftarrow 1$; $R_1=R+P_i$; $T_1\leftarrow$? s, R_1?;

$s\leftarrow -1$; $R_{-1}=R-P_i$; $T_1\leftarrow$? s, R_{-1}?;

$s\leftarrow 0$; $R_0=R$; $T_1\leftarrow$? s, R_0?;

)

$T=T_1$;

Step2. $c=\lceil l/r\rceil$; $Q=O$; $s=$??;

Step3. for $j=c-1$ to 0

① for $i=0$ to $r-1$

$s\leftarrow u_{ci+j}$;

② $Q=\gamma(Q)+P_s$;

Step4. Return Q.

上述算法 Step1(ii)中的左箭头表示向一个集合尾部添加元素或取出元素.

3 算法分析

定理 5. 整个 Frobenius Comb 算法所需的点加量 C_f 为

$$C_f=2r+(1-(2/3)^r)l/r,\ l=cr.$$

证明. Frobenius Comb 算法分为 4 步,Step1 需要 $2r$ 次点加运算,$-P$ 可看做$+(-P)$;Step2 是一些简单的赋值查表操作,几乎不耗时;因为 ?$u_{c(r-1)+j},\cdots,u_{ci+j},\cdots,u_j$? 序列为零的概率是$(2/3)^r$,所以 Step3 需要$(1-(2/3)^r)c$ 次点加运算.得证.

比较 Frobenius Comb 算法与传统的二元法下的窗口算法、Comb 算法和 NAF 算法的计算复杂性,具体从预计算和累加赋值两阶段进行比较.假设

点加和倍点运算相同以及密钥长度为160 b,表1列出以上4种算法在假设前提下需要计算的点加量:

Table 1 Comparison of Algorithm Performance

表1 算法性能比较

Algorithm	Pre-compute Stage	Evaluation Stage
Window Algorithm (binary)	8	191
Comb Algorithm (binary)	174	39
NAF Algorithm (binary)	0	213.33
Frobenius Comb Algorithm	8	32.10

表1中窗口算法的窗口宽度和Comb算法的行数都选为4.从表1中可看出,基于Frobenius映射的算法比二元法下的几种常见算法所需的点加量少大约4～5倍.

图2展示了二元法中Comb算法和Frobenius Comb算法随行数 r 的变化情况,这里只给出行数在1～10之间的比较:

从图2可看出,基于Frobenius映射的Comb算法行数在1～10之间都比二元法中Comb算法高效,且Frobenius Comb算法随行数的变化点加量几乎不变,而二元法中Comb算法的点加量行数在10以后会成指数级上升.

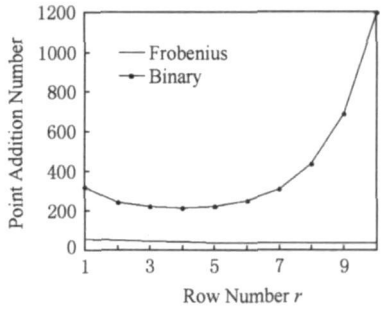

Fig. 2 Comparison of two Comb algorithm.

图2 两种Comb算法性能比较

从文献[12]中可以看出,点加运算和点倍运算的时间复杂度 T_a 和 T_d 有如下近似关系:在仿射坐标中 $T_a=T_d$,在其他坐标中 $T_a=2T_d$.由于传统Comb算法的复杂性为 $C_t=(cr-1)T_d+(2^r+c-2)T_a$,那么可得Comb算法和Frobenius Comb算法在不同密钥长度 l 和不同行数 r 下的时间复杂性比较,如表2、表3所示:

Table 2 Comparison of Time Complexity ($T_a=T_d$)

表2 时间复杂性比较 ($T_a=T_d$)

r	l=160		l=192		l=224	
	Comb	Frobenius Comb	Comb	Frobenius Comb	Comb	Frobenius Comb
4	213	40.10	253	46.52	293	52.94
8	433	35.22	469	39.06	505	42.91
16	65703	41.98	65737	43.98	65771	45.98

Table 3 Comparison of Time Complexity ($T_a=2T_d$)

表3 时间复杂性比较 ($T_a=2T_d$)

r	l=160		l=192		l=224	
	Comb	Frobenius Comb	Comb	Frobenius Comb	Comb	Frobenius Comb
4	133.5	40.10	157.5	46.52	181.5	52.94
8	353.5	35.22	373.5	39.06	393.5	42.91
16	65623.5	41.98	65641.5	43.98	65659.5	45.98

从表2、表3中可看出,在密钥长度 l 为160,192和224以及行数 r 为4,8和16的前提下,Frobenius Comb算法时间复杂性至多是Comb算法的1/3倍,且行数 r 越大倍数越小.

在仿射坐标下,Frobenius Comb算法比Comb算法提高的效率为 imp_1:

$$imp_1=1-\frac{(2r+(1-(2/3)^r)c)T_a}{(cr-1)T_d+(2^r+c-2)T_a}=\frac{l+2^r+(2/3)^r)l/r-2r-3}{l+2^r+l/r-3}.$$

在其他坐标下,Frobenius Comb算法比Comb算法提高的效率为 imp_2:

$$imp_2=1-\frac{(2r+(1-(2/3)^r)c)T_a}{(cr-1)T_d+(2^r+c-2)T_a}=\frac{l+2^{r+1}+2(2/3)^r)l/r-4r-5}{l+2^{r+1}+2l/r-5}.$$

根据 imp_1 和 imp_2 的公式,分析在不同坐标下 Frobenius Comb 算法比 Comb 算法提高的效率随行数 r 的变化情况,如图 3、图 4 所示:

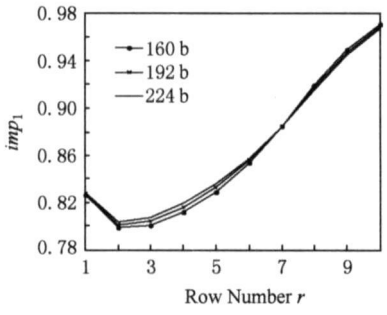

Fig. 3 Chang graph of imp_1 following r.
图 3 imp_1 随 r 变化的情况

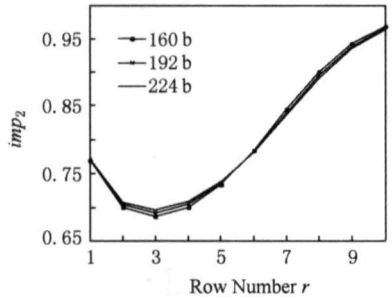

Fig. 4 Chang graph of imp_2 following r.
图 4 imp_2 随 r 变化的情况

从图 3 中可看出,仿射坐标下 Frobenius Comb 算法比 Comb 算法提高效率 imp_1 至少约 80%,从图 4 中可看出,其他坐标下 Frobenius Comb 算法比 Comb 算法提高效率 imp_1 至少约 67%.

4 结 论

本文基于 Frobenius 映射给出了 Koblitz 曲线上一种上层运算:Frobenius Comb 算法,该算法使用预计算表来提高累加赋值阶段的效率,由于新算法使用高效的 Frobenius 映射,无需进行倍点运算,同时经过 Comb 矩阵的一定组合,极大地减少了所需的点加量,所以进一步提高了计算标量乘法的效率.

参 考 文 献

[1] Xu Qiuliang, Li Daxing. Elliptic curve cryptosystem [J]. Journal of Computer Research and Development, 1999, 36(11): 1281-1288 (in Chinese)
(徐秋亮, 李大兴. 椭圆曲线密码体制[J]. 计算机研究与发展, 1999, 36(11): 1281-1288)

[2] Koblitz N. An elliptic curve implementation of the finite field digital signature algorithm [C] //Advance in Cryptology—CRYPTO 1998. Berlin: Springer, 1998: 327-337

[3] Solinas J A. An improved algorithm for arithmetic on a family of elliptic curves [C] //Proc of CRYPTO 1997. Berlin: Springer, 1997: 357-371

[4] Solinas J A. Efficient arithmetic on Koblitz curves [J]. Designs, Codes and Cryptography, 2000, 19(2-3): 195-249

[5] Avanzi R, Heuberger C, Prodinger H. Minimality of the Hamming weight of the r-NAF for Koblitz curves and improved combination with point halving [OL]. [2005-07-12]. http://eprint.iacr.org/2005/225.pdf

[6] Hankerson D, Menezes A, Vanstone S. Guide to Elliptic Curve Cryptography [M]. Berlin: Springer, 2004

[7] Bai Guoqiang, Zhou Tao, Chen Hongyi. A selection of the secure elliptic curve and fast calculation of scalar multiplication [J]. Acta Electronica, Sinica, 2002, 30(11): 1654-1657 (in Chinese)
(白国强, 周涛, 陈弘毅. 一类安全椭圆曲线的选取及其标量乘法的快速计算[J]. 电子学报, 2002, 30(11): 1654-1657)

[8] Hu Lei, Feng Dengguo, Wen Tiehua. Fast multiplication on a family of koblitz elliptic curves [J]. Journal of Software, 2003, 14(11): 1907-1910 (in Chinese)
(胡磊, 冯登国, 文铁华. 一类 Koblitz 椭圆曲线的快速点乘[J]. 软件学报, 2003, 14(11): 1907-1910)

[9] Liu Duo, Dai Yiqi. A new method of scalar multiplication of elliptic curve over OEF [J]. Acta Electronica Sinica, 2002, 30(11): 1654-1657 (in Chinese)
(刘铎, 戴一奇. 优化扩域上椭圆曲线标量乘的一个新算法[J]. 电子学报, 2003, 33(8): 1451-1456)

[10] Avanzi R, Sica F. Scalar multiplication on Koblitz curves using double bases [OL]. [2007-03-15]. http://iacr.org/2006/067.ps.gz

[11] Wong K W, Lee C W, Cheng L M, *et al*. Fast elliptic scalar multiplication using new double-base chain and point halving [OL]. [2006-03-08]. http://eprint.iacr.org/2006/124.pdf

[12] IEEE P1363/ D9 Standard Specifications for Public-Key Cryptography [S]. New York: Institute of Electrical and Electronics Engineers Inc., 2001

Yin Xinchun, born in 1962. Ph. D., professor and Ph. D. supervisor. Senior member of China Computer Federation. His main research interests include parallel and distributed computing, information security, *etc*.

殷新春,1962 年生,博士,教授,博士生导师,中国计算机学会高级会员,主要研究方向为并行与分布式计算、信息安全等.

Hou Hongxiang, born in 1982. Master candidate. His main research interests include information security, *etc*.

侯红祥, 1982 年生, 硕士研究生, 主要研究方向为信息安全等 (houhhx361542@hotmail.com).

Xie Li, born in 1942. Ph. D., professor and Ph. D. supervisor. Senior member of China Computer Federation. His main research interests include parallel and distributed computing, information security, *etc*.

谢 立, 1942 年生, 博士, 教授, 博士生导师, 中国计算机学会高级会员, 主要研究方向为并行与分布式处理、信息安全等.

Research Background

Elliptic curve cryptosystem (ECC) is a novel public key cryptosystem. It is going to be the primary standard for application in the near future. Scalar multiplication is the basic computation of ECC, so improving the efficiency of scalar multiplication has important meaning. This paper improves scalar multiplication based on Frobenius mapping. Our work is supported by the National Natural Science Foundation of China under grant No. 60473012; the National High-Tech R&D Plan of China (863) under grant No. 2007AA012448; and The Six Scientific Fields Summit of Jiangsu Province under grant No. 06-E-025.

附录 A 算法[A1]: 计算 $TNAF(\kappa)$.

输入: $\kappa=a_0+a_1\gamma\in Z[\gamma]$;

输出: $TNAF(\kappa)$.

Step1. $i=0$.

Step2. while $a_0\neq 0$ or $a_1\neq 0$ do

1) if (a_0 is odd)

① $u_i=2-(a_0-2\ a_1 \bmod 4)$;

② $a_0=a_0-u_i$;

2) else $u_i=0$;

3) $t=a_0$;

4) $a_0=a_1+\lambda a_0/2$;

5) $a_1=-t/2$;

6) $i=i+1$.

Step3. output(u_{l-1}, u_{l-2}, …, u_1, u_0).

附录 B 算法[B1]: $Z[\gamma]$内的除法.

输入: $\alpha=a_0+a_1\gamma$, $\beta=b_0+b_1\gamma\in Z[\gamma]$且 $\beta\neq 0$;

输出: $\kappa=q_0+q_1\gamma$, $\delta=r_0+r_1\gamma\in Z[\gamma]$且 $\alpha=\kappa\beta+\delta$及 $N(\delta)\leqslant 4/7\,N(\beta)$.

Step1. $g_0=a_0b_0+\lambda a_0b_1+2a_1b_1$;

Step2. $g_1=a_1b_0-a_0b_1$;

Step3. $N=b_0^2+\lambda b_0b_1+2b_1^2$;

Step4. $\mu_0=g_0/N$, $\mu_1=g_1/N$;

Step5. compute(q_0, q_1)=Round(μ_0, μ_1);

Step6. $r_0=a_0-b_1q_0+2b_1q_1$;

Step7. $r_1=a_1-b_1q_0-b_0q_1-\lambda b_1q_1$;

Step8. $\kappa=q_0+q_1\gamma$;

Step9. $\delta=r_0+r_1\gamma$;

Step10. output(κ, δ).

参 考 文 献

[A1] Solinas J A. Efficient arithmetic on Koblitz curves [J]. Designs, Codes and Cryptography, 2000, 19(2-3): 195-249

[B1] Hankerson D, Menezes A, Vanstone S. Guide to Elliptic Curve Cryptography [M]. Berlin: Springer, 2004