基于多基数系统的有效标量乘算法

逯玲娜,李冬霞
(郑州城市职业学院 基础部,河南 郑州 452370)

摘要:首先给出了椭圆曲线上点P的7倍点公式$7P$,当$[i]/[m]=6$时,它比直接计算节省运算量7.4%~30.56%。作为双基数系统的一个推广,多基数系统具有标量表示长度更短、汉明重量更小的特点,较适用于椭圆曲线标量乘的快速计算。结合以上给出的公式,提出了一个以2,3和7作为基底的多基数系统计算椭圆曲线标量乘的有效算法,所提方法计算量更少。

关键词:椭圆曲线密码体制;标量乘法;双基数系统(DBNS);多基数系统(MBNS)
中图分类号:TP309 **文献标识码**:A **文章编号**:1671-5322(2013)02-0025-06

自1985年Koblitz和Miller各自分别提出椭圆曲线密码体制(ECC,Elliptic Curve Cryptography),ECC就一直得到众多密码学家及密码学研究者的关注。ECC体制的安全性是基于椭圆曲线上离散对数问题求解的困难性,目前还没有找到解决此问题的亚指数时间算法,因而与另一著名的公钥密码RSA相比,ECC密钥短,安全性高,速度快,存储空间占用少和带宽要求低。它的这些特点使得业内人士普遍认为ECC将成为下一代最通用的公钥加密算法标准。

在ECC的快速实现中,最关键的就是标量乘kP的计算。即已知整数k和椭圆曲线上一点P,求kP的运算。从国内外学者的研究现状可知,目前主要从以下几个方面加快椭圆曲线标量乘算法:a)对基点P进行坐标变换(仿射坐标、投影坐标、Jacobian坐标等);b)对标量乘k进行重新编码(二进制、三进制、NAF、ω-NAF等);c)在曲线上进行有效的群运算($2P$,$2P+Q$,$3P$,$3P+Q$等);d)利用预计算先把需要用到的点计算出来进行存储;e)用滑动窗口方法,comb方法,加减法链等有效算法。

经典的标量乘算法是把标量表示成二进制形式通过点倍、点加公式进行标量乘运算。对于一般的曲线,以为基元的双基数系统DBNS标量乘表示算法的有效性已被文献[1]证明。在过去的几年中,DBNS分别被作者[3-5]运用到相应的文献中加速运算。Knudsen和Schroeppel分别独立提出了半点运算,他们认为半点运算可以替代点倍和点加公式中的倍点运算作为新的方法加速标量乘算法[4]。作为DBNS的推广,文献[2]以{2,3,5}作为多基数系统MBNS的基元表示标量k加速算法,同时在该文献中包括点P的五倍点公式$5P$。文献[6,7]也分别使用MBNS加速算法,其中[7]是和半点公式结合起来的。本文首先给出二元域$GF(2^n)$椭圆曲线上点P的7倍点公式$7P$,然后以{2,3,7}作为多基数系统的基元表示标量k,提出一种新的有效的多基链标量乘算法,经验证,与现存算法相比,该标量表示更稀疏且该标量乘算法更有效。

1 相关知识

1.1 椭圆曲线ECC的介绍

定义1 域K上的椭圆曲线E定义为Weierstrass方程:

$$E: y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6 \tag{1}$$

其中$a_1,a_2,a_3,a_4,a_6\in K$,$V\neq 0$,V是E的判别式。在素域$K=F_p(Char(k)>2,3)$上,(1)式可以简化为:

$$y^2=x^3+ax+b \tag{2}$$

其中$a,b\in K$,$\Delta=4a^3+27b^2$。

在二元域$K=F_2$上,非奇异曲线形式用于加

收稿日期:2013-04-16
作者简介:逯玲娜(1985-),女,河南滑县人,助教,硕士,主要研究方向为代数学与密码学。

密,其 Weierstrass 方程可以化简为:

$$y^2 + xy = x^3 + ax^2 + b \tag{3}$$

其中 $a,b \in K, \Delta = b \neq 0$。

满足(1)式的(x,y)成为域 K 上的点;此外,椭圆曲线还定义一个特殊的无穷远点 o;即域 K 上的点集和一个无穷远点 O 组成域 K 上的椭圆曲线 $E(K)$。

在二元域 $K=F_2$ 上,若 $P=(x_1,y_1)$,$Q=(x_2,y_2)$是椭圆曲线上两个点,则 $P+Q=(x_3,y_3)\in E(K)$,倍点:$2P=(x_4,y_4)\in E(K)$且有:

$$\begin{cases} x_3 = \lambda^2 + \lambda + x_1 + x_2 + a \\ y_3 = \lambda(x_1 + x_3) + x_3 + y_1 \end{cases} \quad \text{其中 } \lambda = \frac{y_2 + y_1}{x_2 + x_1} \tag{4}$$

倍点:$2P=(x_4,y_4)\in E(K)$且有:

$$\begin{cases} x_4 = \lambda^2 + \lambda + a \\ y_4 = x_1^2 + \lambda x_4 + x_4 \end{cases} \quad \text{其中 } \lambda = x_1 + \frac{y_1}{x_1} \tag{5}$$

标量乘算法的效率涉及到底层域快速算法,由于仿射坐标下的群运算需要在域上求逆,而这正是域上最耗时的运算,所以为了避免作逆运算,许多文献提出用投射坐标,Jacobian 坐标等表示椭圆曲线上的点进行运算。而点坐标形式的选取很大程度上依赖于$[i]/[m]$的比值,即域上逆运算的时间消耗和域上乘法运算的时间消耗的比值。通常认为二元域上 $3\leqslant[i]/[m]\leqslant 10$,素域上$[i]/[m]\geqslant 30$[8]。因此,我们通常考虑二元域上的仿射坐标形式和素域上的 Jacobian 坐标形式,本文暂且讨论二元域上的仿射坐标。

为了表示域的运算开销,我们分别用$[i]$,$[s]$,$[m]$表示一次逆运算,一次平方运算和一次乘法运算。我们总可以忽略域上的加法开销,并且,在二元域上,平方运算几乎不费时间(如果使用正规基)或者其运算开销可以忽略(线性运算)(详见[9])。二元域上不同运算的开销。

1.2 整数的多基数表示

设 $B=\{b_1,\cdots,b_l\}$ 是个小整数的集合,则任何一个整数 k 都可以表示成 $\sum_{j=1}^{m} s_j b^{e_j}\ (s_j = \pm 1)$ 的形式。本文主要研究 $B=\{2,3,7\}$ 的情况,具体定义如下。

定义 2 设集合 $B=\{2,3,7\}$,则 k 可以表示成如下形式:

$$k = \sum_{i=1}^{m} s_i 2^{bi} 3^{ti} 7^{qi},\quad s_i \in \{-1,1\} \tag{6}$$

该形式即为整数的多基数表示。

当 $B=\{2,3\}$ 时,相应的表示称为双基数表示。DBNS 是高冗余的,而且表示长度非常短;与 DBNS 相比,MBNS 冗余度更高,表示长度也更短。如仅考虑 $s_i=1$ 情况下,100 的双基数表示共有 402 个,而它的多基数表示就有 8425 个。b_1,t_1,q_1 的大小影响标量乘中 2 倍点、3 倍点和 7 倍点运算的运算次数,而 $m-1$ 为标量乘中点加的次数。一个 160 bit 的大整数如果使用双基数系统来表示需要大约 23 项,而如果使用多基数系统则需要约 15 项就可以了[2],而和 $B=\{2,3,5\}$ 相比,使用基元 $B=\{2,3,7\}$ 表示 k 则使 k 的表示更短更稀疏。因此与使用双基数系统计算标量乘相比,使用多基数系统能够大大提高椭圆曲线标量乘法的计算效率。

虽然一般的多基数表示比较短,但不一定适合标量乘运算,我们感兴趣的是有指数限制的特殊表示形式。

定义 3 k 的一个多基数表示 $k=\sum_{i=1}^{m} s_i 2^{bi} 3^{ti} 7^{qi}$ 叫作阶梯形多基数表示(SMBR, step multibase representation),如果它的指数$\{b_i\}$,$\{t_i\}$和$\{q_i\}$分别组成一个单调递减的序列。

通常采用贪婪算法把整数 k 转化为 SMBR 形式,下面给出算法:

算法 1 多基数转换贪婪算法

Input: integer , max2, max3, max7 >0,
array $T=\{0,\cdots,$max 2;$0,\cdots,$max 3;$0,\cdots,$max 7$\}$
Ouput: sequence (s_i,b_i,t_i,q_i)
1: $s\leftarrow 1$
2: while $k>0$ do
for (b =0 to max2, t =0 to max3, p =0 to max7)
3: $z=T[b,t,p]$, the best approximation of k
4: print (s,b,t,p)
5: max $2\leftarrow b$, max $3\leftarrow t$, max $7\leftarrow p$;,
6: if $k<z$ then $s\leftarrow -s$
7: $k\leftarrow |k-z|$
8: return (s_i,b_i,t_i,q_i)

2 二元域上的7P有效计算公式

由于二元域上的比值$[i]/[m]$比较小,故我们采用仿射坐标。下面给出给出仿射坐标下点 P 的 7 倍点公式。设 $P(x,y)$是(3)式给出的椭圆曲线上的一点,点 P 的 7 倍 $7P=(x,y)$,则 x_7 和

y_7 可以按如下公式计算:

对于二元域上的非奇异曲线,其可除多项式为:

$$\psi_1 = 1$$
$$\psi_2 = x$$
$$\psi_3 = x^4 + x^3 + a = A$$
$$\psi_4 = x^6 + ax^2 = x^2(x^4 + a) = x^2(A + x^3) = B \tag{7}$$

高阶的可除多项式可由以下递推公式得到:

$$\psi_{2n+1} = \psi_{n+2}\psi_n^3 - \psi_{n-1}\psi_{n+1}^3$$
$$\psi_2\psi_{2n} = \psi_{n+2}\psi_n\psi_{n-1}^2 - \psi_{n-2}\psi_n\psi_{n+1}^2 \tag{8}$$

利用递推公式得:

$$\psi_5 = \psi_4\psi_2^3 - \psi_1\psi_3^3 = Bx^3 + A^3 = A^3 + Bx^3 = C$$

$$\psi_6 = \frac{\psi_5\psi_3\psi_2^2 - \psi_1\psi_3\psi_4^3}{\psi_2} = \frac{CAx^2 + AB^2}{x} = \frac{(A^3 + Bx^3)Ax^2 + ABx^2(A + x^3)}{x} =$$

$$A(A^3 + Bx^3)x + ABx(A + x^3) = xA^2(A^2 + B)xD$$

其中 $D = A^2(A^2 + B)$

$$\psi_7 = \psi_5\psi_3^3 - \psi_2\psi_4^3 = CA^3 + xB^3 = E$$

$$\psi_8 = \frac{\psi_6\psi_4\psi_3^2 - \psi_2\psi_4\psi_5^2}{\psi_2} = \frac{xDBA^2 + xBC^2}{x} =$$

$$DBA^2 + BC^2 = A^2Dx^2(A + x^3) + x^2(A + x^3)C^2 = x^2F$$

其中 $F = (A + x^3)(A^2D + C^2)$

利用以上整除多项式,曲线上点 $P(x,y)$ 的 n 倍点公式由以下公式给出:

$$[n]P = \left(x + \frac{\psi_{n+1}\psi_{n-1}}{\psi_n^2}, y + \psi_2 on + \frac{\psi_{n+1}^2\psi_{n-2}}{\psi_2\psi_n^3} + h_4\frac{\psi_{n+1}\psi_{n-1}}{\psi_2\psi_n^2}\right) \tag{9}$$

其中 $\psi_2 on = x + \frac{\psi_{n+1}\psi_{n-1}}{\psi_n^2}$, $h_4 = (x_2 + y)$

故若 $7P(x,y) = (x_7, y_7)$,则有:

$$x_7 = x + \frac{\psi_8\psi_6}{\psi_7^2} \tag{10}$$

$$y_7 = y + x_7 + \frac{\psi_8^2\psi_5}{\psi_2^{73}} + (x^2 + y)\frac{\psi_8\psi_6}{\psi_7^2} \tag{11}$$

把 $\psi_1 = 1, \psi_2 = x, \psi_3 = A, \psi_4 = B, \psi_5 = C, \psi_6 = xD, \psi_7 = E, \psi_8 = x^2F$ 代入上式,有:

$$x_7 = x + x^2 \cdot \frac{FD}{E^2} \tag{12}$$

$$y_7 = x + y + x^2 \cdot \frac{FD}{E^2} + x^3 \cdot \frac{F^2C}{E^3}(x^2 + y)x^2 \cdot \frac{FD}{E^2} = x + y + (x^4 + x^2y + x^2) \cdot \frac{FD}{E^2} + x^3 \cdot \frac{F^2C}{E^3} \tag{13}$$

其中:

$$A = x^4 + x^3 + a; \quad D = A^2(A^2 + B);$$
$$B = x^2(A + x^3); \quad E = CA^3 + xB^3;$$
$$C = A^3 + Bx^3; \quad F = (A + x^3)(A^2D + C)$$

从而可以得到二元域上 7 倍点公式开销如下表所示:

表 1　二元域上 7 倍点公式运算开销

Table 1　Two yuan domain 7 times formula computation overhead

表达式	运算开销	表达式	运算开销
$A = x^4 + x^3 + a$	$2[s] + 1[m]$	$B = x^2(A + x^3)$	$1[m]$
$C = A^3 + Bx^3$	$1[s] + 2[m]$	$D = A^2(A^2 + B)$	$1[m]$
$E = CA^3 + xB^3$	$1[s] + 3[m]$	$F = (A + x^3) \cdot (A^2D + C^2)$	$1[s] + 2[m]$
$1/E^3$	$1[i] + 1[s] + 1[m]$	$x_7 = x + x^2FD/E^2$	$3[m]$
$y_4 = x + y + (x^4 + x^2y + x^2) \cdot \frac{FD}{E^2} + x^3\frac{F^2C}{E^3}$	$1[s] + 5[m]$	Total	$1[i] + [s] + 19[m]$

下面讨论所提公式的实现效率:

由上表可以看出,我们提出的计算 $7P$ 公式共计花费:$1[i] + 7[s] + 19[m]$,若忽略平方的花费(二元域),则总花费为:$1[i] + 19[m]$。一般来说,计算 $7P$,除了上述公式外,目前有以下几种计算方式:a) $7P = 2(2P) + 3P$; b) $7P = 2(3P) + P$; c) $7P = 3(2P) + P$; d) $7P = 5P + 2P$,其中,a) 中先用 $DBL(P)$ 和 $TPL(P)$ 公式,分别花费:$1[i] + [m], 1[i] + 7[m]$ 再用 $DA(P,Q)$ 公式,花费 $1[i] + 9[m]$,共计花费:$3[i] + 18[m]$; 同理可算 b), c), d) 的花费分别为:$2[i] + 16[m], 3[i] + 11[m], 2[i] + 15[m]$,而使用我们提出的公式,花

费为:1[i] +19[m]。当时[i]/[m] =6,它比直接用上述方法计算节省运算量7.4% ~30.56%,当由此可见,我们提出的二元域上7P公式加速了标量乘的执行效率。下表是二元域上不同运算的计算开销。

表2 二元域上不同运算的开销

Table 2 Different operational costs two yuan domain

运算	$E(F_{2^m})$	参考文献
$ADD(P+Q)$	$1[i]+1[s]+2[m]$	-
$DBL(2P)$	$1[i]+1[s]+2[m]$	-
$DA(2P+Q)$	$1[i]+2[s]+9[m]$	[1]
$TPL(3P)$	$1[i]+4[s]+7[m]$	[1]
$TA(3P+Q)$	$2[i]+3[s]+9[m]$	[1]
$4P$	$1[i]+5[s]+8[m]$	[3]
$5P$	$1[i]+5[s]+13[m]$	[2]
$7P$	$1[i]+7[s]+19[m]$	

3 基于多系统数目的有效标量乘算法

3.1 算法过程

本文把文献[3]中的双基数系统的两个基元,推广到多基数系统中的3个基元,下面给出了二元域上基于多基数系统(基元2,3,7)的标量乘算法,其中用到了群上的运算如:$ADD, DBL, TPL, w-DBL, w-TPL, DA, TA$

算法2:二元域上基于多基数系统的标量乘算法

Input: 整数 $k=\sum_{i=1}^{m} s_i 2^{bi}3^{ti}7^{qi}$,其中 $s_i \in \{-1,1\}$,

$b_1 \geqslant b_2 \geqslant \cdots \geqslant b_m \geqslant 0; t_1 \geqslant t_2 \geqslant \cdots \geqslant t_m \geqslant 0;$

$q_1 \geqslant q_2 \geqslant \cdots \geqslant q_m \geqslant 0$;点 $P \in E(F_{2^m})$

Ouput: 点 $kP \in E(F_{2^m})$

1: $Z \leftarrow s_1 P$

2: for $i=1,2,\cdots,m-1$ do

3: $u \leftarrow b_i - b_{i+1}$

4: $v \leftarrow t_i - t_{i+1}$

5: $w \leftarrow q_i - q_{i+1}$

6: if $u=0$ then

7: $Z \leftarrow (7^w Z)$

8: if $v \neq 0$ then

9: $Z \leftarrow 3(3^{v-1}Z) + s_{i+1}P$ //(TA used here)

10: else

11: $Z \leftarrow Z + s_{i+1}P$

12: else

13: $Z \leftarrow 7^w Z$

14: $Z \leftarrow 3^v Z$

15: $Z \leftarrow 2^{n-1} Z$

16: $Z \leftarrow 2Z + s_{i+1}P$ //(DA used here)

17: Return Z

算法2是用标量的多基数表示来描述二元域上的标量乘运算。我们注意到,上述算法共需 b_1 次倍点运算,t_1 次3倍点运算,p_1 次7倍点运算,加法运算只有当 k 的展开式中2和3的指数均为零的时候才执行,否则执行 DA 和 TA 运算,因此整体看来,我们不需要做很多次加法。

例1:使用以 $B=\{2,3,7\}$ 为基元的多基链表示计算895712,895712P 的多基链表示如下:

$$895712 = 2^9 3^5 7^1 - 2^9 3^0 7^1 - 2^6 3^0 7^1$$

计算过程如表3所示:

表3 用多基链2,3,7计算895712P的过程

Table 3 With a chain of 2, 3, 7 895712P calculation

公式运算	所有运算
$80=3^4-1$	$3T, TA$
$1919=2^3*3*80-1$	Q, T, DA
$895712=2^6*7*1919$	$7P, 3Q$

计算895712P总的运算量为:

$$7P+4Q+DA+TA+4T =$$
$$(I+7S+19M)+4(I+5S+8M)+$$
$$(I+2S+9M)+(2I+3S+9M)+$$
$$4(I+4S+7M) = 12I+48S+97M =$$
$$72M+38.4M+97M = 207.4M$$

例2:使用以 $B=\{2,3,7\}$ 为基元的多基链表示计算123456789P,123456789的多基链表示如下:

$$123456789 = 2^7 3^9 7^2 + 2^0 3^6 7^1 - 2^0 3^4 7^0 - 2^0 3^2 7^0$$

计算过程如表4所示:

表4 用多基链2,3,7计算123456789P的过程

Table 4 With a chain of 2, 3, 7 123456789P calculation

公式运算	所有运算
$24193=2^7*3^3*7+1$	$7P, 3T, DA, 3Q$
$1524158=3^2*7*24193-1$	T, TA
$13717421=3^2*1524158-1$	T, TA
$123456789=3^2*13717421$	$2T$

计算123456789P总的运算量为:

$$7P+3Q+DA+7T+2TA =$$

$$(I+7S+19M)+3(I+5S+8M)+(I+2S+9M)+7(I+4S+7M)+2(2I+3S+9M)=16I+58S+119M=96M+46.4M+119M=261.4M$$

3.2 典型标量乘算法的比较

通过以上几种算法计算过程的比较,得到表5,5,6,7,经对比可以发现,标量 k 一定的条件下,NAF、双基链、多基链算法所需要的求逆次数依次减少,所需运算量也依次减少。以{2,3,5}为基元的多基链与以{2,3,7}为基元的多基链相比,当标量较小时,二者所需求逆次数一样且运算量相当,但当标量 k 较大时,显然以{2,3,7}为基元的多基链所需求逆次数更少且运算量更少。

表5 典型标量乘算法的比较

Table 5 Typical scalar multiplication algorithm

	NAF	双基链 {2,3}	多基链 {2,3,5}	多基链 {2,3,7}
314159	17I+29S	14I+42S	13I+39S	13I+41S+93M
895712	15I+41S	15I+45S	12I+44S	12I+48S+97M
1069493	16I+39S	15I+45S	13I+45S	13I+46S+112M
2578629	18I+38S	18I+51S	15I+49S	14I+50S+116M
9634021	17I+48S	16I+53S	15I+46S	12I+50S+114M
69072367	18I+56S	19I+58S	18I+66S	18I+62S+146M
123456789	23I+49S	18I+64S	17I+61S	16I+58S+119M

当 $I/M=6$ 时,比较结果如表6所示。

表6 典型标量乘算法的比较($I/M=6$)

Table 6 Typical scalar multiplication algorithm ($I/M=6$)

	NAF	双基链	多基链	多基链
314159	221.2M	211.6M	198.2M	203.8M
895712	224.8M	226M	208.2M	207.4M
1069493	232.2M	234M	221M	226.8M
2578629	248.4M	255.8M	238.2M	240M
9634021	289.4M	263.4M	250.8M	226M
69072367	278.8M	294.4M	320.8M	303.2M
123456789	317.2M	291.2M	287.8M	261.4M

当 $I/M=7$ 时,比较结果如表7所示。

表7 典型标量乘算法运算量的比较($I/M=7$)

Table 7 Typical scalar multiplication algorithm ($I/M=7$)

	NAF	双基链	多基链	多基链
314159	238.2M	225.6M	211.2M	216.8M
895712	239.8M	241M	220.2M	219.4M
1069493	248.2M	249M	234M	239.8M
2578629	266.4M	273.8M	253.2M	254M
9634021	306.4M	279.4M	265.8M	238M
69072367	296.8M	313.4M	338.8M	321.2M
123456789	340.2M	309.2M	304.8M	277.4M

当 $I/M=8$ 时,比较结果如表8所示。

表8 典型标量乘算法运算量的比较($I/M=8$)

Table 8 Typical scalar multiplication algorithm ($I/M=8$)

	NAF	双基链	多基链	多基链
314159	255.2M	239.6M	224.2M	229.8M
895712	254.8M	256M	232.2M	231.4M
1069493	264.2M	264M	247M	252.8M
2578629	284.4M	291.8M	268.2M	268M
9634021	323.4M	295.4M	280.8M	250M
69072367	314.8M	332.4M	356.8M	339.2M
123456789	363.2M	327.2M	321.8M	293.4M

4 结论

本文给出了二元域上7倍点的运算公式并提出一种新的标量的多基数表示和标量乘法,利用多基数系统计算椭圆曲线标量乘不仅能够提高标量乘的运算效率,使得基于椭圆曲线密码体制实现更加便捷和高效,而且由于多基数系统表示的高度冗余性,多次计算同一个标量乘,计算过程可以完全不同,因此使用多基数系统计算椭圆曲线标量乘可以抵抗某些边信道攻击。下一步将从底层运算出发,寻找新的方法以减少底层运算中各种算法的运算量,达到提高标量乘算法效率的目的。

参考文献:

[1] Ciet M, Lauter K, Joye M, et al. Trading inversions for multiplications in elliptic curve cryptography[J]. Designs, Codes and Cryptography,2006,39(2):189-206.

[2] Mishra P K, Dimitrov V S. Ecient Quintuple Formulas for Elliptic Curves and Efficient Scalar Multiplication Using Multibase Number Representation[J]. IJCSI, 2007,4 779:390-406.

[3] Dimitrov V, Imbert L, Mishra P K. Efficient and Secure Elliptic Curve Point Multiplication using Double-Base Chains, Advances in Cryptology[J]. LNCS ,2005,3 788:59-78.

[4] Wong K W, Edward, Lee C W, Cheng L M, et al. Fast Scalar Multiplication using new Double Base Chain and Point Halving[J]. Applied Mathematics and Computation. 2006,183(2):1 000-1 007.

[5] Avanzi R M,Sica F. Scalar multiplication on Koblitz curves using Double bases. Technical Report Available[EB/OL]. http://eprint. iacr. org/2006/067.

[6] Avanzi R M,Ciet M,Sica F. Faster Scalar multiplication on Koblitz Curves Combining Point halving with the Frobenius Endomorphism[J]. LNCS ,2004,2 947:28-40.

[7] Ismail A M,Said M R MD,Mohd Atan K A, et al. An Algorithm to enhance Elliptic Curves scalar Multiplication Combinig MBNR with point halving[J]. Applied Mathematical sciences,2010,4:1 259-1 272.

[8] Fong K, Hankerson D, Lopez J, et al. Field inversion and point halving revisited[J]. IEEE Transactions on Computers, 2004,53(8):1 047-1 059.

[9] Hankerson, D, Lopez Hernandez J, Menezes A. Software implementation of elliptic curve cryptography over binary fields [J]. LNCS,2000,1 965:1-24.

Fast Scalar Multiplication Based on MBNS

LU Ling-na, LI Dong-xia

(Department of Basic Science, Zhengzhou City Department of Career Academy, Zhengzhou Henan 452370, China)

Abstract: Firstly, this paper gives the 7P formula of point P which is on the elliptic curve, and it saves 7.4% ~30.56% than directly computation. Then as a generalization of double base chains, multibase number system is very suitable for efficient computation of scalar multiplication of a point of elliptic curve because of shorter representation length and hamming weight. Combined with the given formulas for computing the 7-fold of an elliptic curve point P, an efficient scalar multiplication algorithm of elliptic curve is proposed using 2, 3 and 7 as basis of the multi based number system and the proposed algorithms cost less.

Keywords: Elliptic Curve Cryptosystems; Scalar Multipication; Double-base number system; Muti-base number system

(责任编辑:张英健)