Smart Grid 智能电网, 2012, 2, 131-136
http://dx.doi.org/10.12677/sg.2012.24023 Published Online December 2012 (http://www.hanspub.org/journal/sg.html)

A Survey of Software Optimization of Elliptic Curve Cryptosystem Algorithm

Jialiang Wang[1], Qifeng Qian[1], Lei Wei[2], Hong Zhu[2]

[1]Nanjing University of Posts & Telecommunications, Nanjing
[2]Nanjing Power Supply Company, Nanjing
Email: wang_jia_l@126.com

Received: Nov. 10th, 2012; revised: Nov. 22nd, 2012; accepted: Nov. 30th, 2012

Abstract: In distribution automation system, workable encryption techniques of elliptical encryption algorithm can be used to solve a security issue of stored in clear text and transmitted in the clear. But the efficiency is poor. Now software optimization of elliptic curve cryptosystem algorithms are classified, analyzed and compared. The result is that several algorithms optimization can be used to improve the speed of algorithm encryption. Therefore, three possible research directions are displayed in the future: firstly, for *k* coding; secondly, formula can be rewritten; thirdly, the way of optimization algorithm of upper field and the way of optimization algorithm of underlying field can be used together.

Keywords: Elliptic Curve Cryptosystem; Software Optimization Algorithm; Underlying Field; Upper Computing

ECC 算法软件优化的研究综述

王家良[1]，钱琦锋[1]，韦　磊[2]，朱　红[2]

[1]南京邮电大学，南京
[2]南京供电公司，南京
Email: wang_jia_l@126.com

收稿日期：2012 年 11 月 10 日；修回日期：2012 年 11 月 22 日；录用日期：2012 年 11 月 30 日

摘　要：针对配电自动化系统中明文存储与明文传输的不安全性，ECC 是一种可行的加密算法，但其加密效率较低。现对 ECC 算法软件优化进行分类、分析和对比，综合使用多种算法优化可以提高加密速度。鉴于此，本文提出了关于 *k* 的编码、公式改写、综合利用上下层算法软件优化这三个具有广大前景的研究方向。

关键词：ECC；软件优化算法；底层域；上层运算

1. 引言

随着国家智能电网的发展，配电自动化系统中信息的安全性也受大了广泛的关注。其中最大的缺点在于：1) 信息的存储没有进行加密处理；2) 信息的传输没有进行加密处理，所以为了保证配电自动化系统的安全性，必须使用加密技术对其安全防护。

对称加密和非对称加密是两类不同的加密方式。对称加密就是同一个密钥被信息的发送者和接收者使用，例如：数据加密标准(DES, Data Encryption Standard)。对称加密的优点：信息的加密在很短时间内就可以完成，加密效率高；加密过程中所需的运算量较小。对称加密的缺点[1]在于：1) 发信者将密钥传送给接收者，其安全性得不到保障。2) 密钥的数量随着通信人数的增多而增多，这就使得密钥的管理和分发变得相当困难。非对称加密算法就是不同密钥被发送方和接收方使用，譬如：公开密钥密码体制(RSA)。私钥由发送方自己保管，降低密钥泄露的可能性，也就提高了信息加密技术的安全性，而且密钥的分发较

为简单。但将其与对称加密相比，其缺点在于：加密过程中运算量较大；整个加密处理的速度慢；当非对称加密密钥位数多于对称加密时，才有可能保证有相同的安全性。ECC 是一种广为人知的非对称加密算法之一。

ECC 加密算法较 RSA 加密算法的优点在于：1) 存储空间小，2) 安全性能高，3) 所需带宽低。缺点在于：对于底层域的求逆、乘法、平方等来说，其运算量较大，其中一次模逆运算所耗时间约为一次模乘的 80 倍，所以要对 ECC 算法进行优化。

为了解决以上问题，对 ECC 算法优化有两种方式：硬件优化和软件优化。对于用硬件实现来说，存在很多的弊端[2]，例如：需要复杂的硬件电路实现以及当算法更新后，其硬件电路要重新设计，增加了资源的消耗，更新过程复杂周期长。而软件实现 ECC 算法优化要求的基本程序存储空间小，所需的数据空间可根据实际环境中的内存资源情况进行灵活调整，做到时间效率和空间资源相对均衡。但其缺点在于底层域运算中的乘法、平方和取模算法的实现效率对软件实现 ECC 算法的效率影响较大，ECC 算法执行时约有 70%的时间消耗在这三个运算上，因而影响了整个 ECC 算法的效率。本文针对配电自动化系统中存在的安全隐患，目前使用 ECC 算法对其进行加密处理。由于 ECC 加密运算过程中计算的复杂性，使得 ECC 加密效率低。因此，本文的主要目的就是对国内外现有的 ECC 算法软件优化方法进行分类、对比、分析、总结，然后提出 ECC 算法软件优化可能的研究方向，例如：k 的编码，公式改写等。

2. ECC 算法的基本思想

椭圆曲线上离散对数问题 *ECDLP* 定义[3]如下：给定素数 p 和椭圆曲线 E，对 $Q = kP$，在已知 P，Q 的情况下求出小于 p 的正整数 k。由 k 和 P 计算 Q 比较容易，而由 Q 和 P 计算 k 则比较困难。将椭圆曲线中的加法运算与离散对数中的模乘运算相对应，将椭圆曲线中的乘法运算与离散对数中的模幂运算相对应，就可以建立基于椭圆曲线对应的密码体制。

ECC 的加密通信步骤一般可分为：1) 用户 A 在椭圆曲线选取一点 P，并将其作为基点，2) 用户 A 通过公式 $Q = kP$ 计算生成公开密钥 Q，其中 k 为用户所选的私有密钥，3) 用户 A 将椭圆曲线和点 Q，P 传给另外用户 B，4) 当信息被传送到用户 B 时，用户 B 将待传的明文信息编码到椭圆曲线上一点 N，同时也会有一个 t 生成，5) 用户 B 计算点 $C1 = N + rQ$；$C2 = rP$，6) 用户 B 将 $C1$，$C2$ 传给用户 A。

在这加密过程中标量乘 $Q = kP$ 的运算量远远大于其他几步中的运算量。因为在标量乘中点加和倍点操作需要不断被执行，而点加和倍点则需要多次的做模加、模乘、模逆等模运算，其中一次模逆运算就相当于若干次的模乘运算，这些都会使得计算量增多，加密速度减缓。因此 ECC 的加密效率很大程度上是由这些复杂的运算的计算效率所决定的。例如：计算 $Q = 314159P$，其中 $k = 314159$。第一步，要将 k 编码，变为 $k = \sum_{i=1}^{m} s_i 2^{h_i} 3^{t_i}, s_i \in \{-1, 0, 1\}$ 或其他形式，此处表示为：$314159 = 2^{15}3^2 + 2^{11}3^2 + 2^8 3^1 + 2^4 3^1 - 2^0 3^0$；第二步，预计算可能用到的底层域运算，此处为：$2^l P$ 、$3^l P(l = 0,1,2,3)$ 等值；第三步，计算 $Q = kP$。

3. ECC 算法软件优化的分类研究

对 ECC 算法执行速率的有影响因素一般有如下几个：坐标的选取、标量乘、椭圆曲线的选取。其中起到决定性的因素为标量乘的计算速率。

标量乘的运算可分为两个层次[4]：第一：上层运算，在椭圆曲线上，点与点加运算构成的有限交换群上的运算；第二：底层运算，在有限域上，通过一些求逆，乘法、平方等算术操作来实现椭圆曲线上点加和倍点运算，例如：$3P$ 、$2^k P$ 、$kP + lQ$ 的计算。

3.1. 上层运算

k 的有效编码是上层运算的主要研究任务,即如何使得 k 的编码最短、非零元素最少，如何划分 k 的编码使得计算量最小，这也就是标量乘运算的第一步。通过 k 的有效编码可以减少点加或倍点的运算次数，例如：基于进制的算法优化、基于双基数链的算法、基于窗口的算法优化、基于重编码的算法优化等。

3.1.1. 基于进制的算法优化

近几年国内外提出了许多基于进制的算法优化[5-7]。该类方法的共同思想就是减少非零元素的个数，降低运算量。文献[5]提出了一种基于从左到右的编码

方法的标量乘法算法，这种方法能够减少非零元。此算法的思想是将 k 从高位开始编码，与传统的从右到左的编码方式不同在于：在编码过程中就对编码进行了处理，使得海明重量达到最小。文献[6]也是将 k 从高位开始编码，但是它是利用混合坐标系来减少计算复杂度，因此在用于标量乘法算法时，需要另外存储空间来存储编码的结果和相关的信息。文献[7]方法相对于文献[6]中的方法不依赖于辅助变量，降低了存储要求。文献[7]提出了对称三进制标量乘法算法，此算法的思想在于将 k 用三进制表示减少了倍点次数，然后再表示成对称三进制的形式即只有–1, 1, 0，从而简化计算。相对于二进制标量乘算法，对称三进制标量的运算平均运算效率提升 5.4%。当进行预计算时，相对于二进制算法和二进制预计算算法，平均效率分别提升 73.18%、15.58%，并且能减少需要存储的点数。

3.1.2. 基于双基数链的算法

双基数系统[8](Double-Based Number System, DBNS)是一种数字表示方法，在该系统中，给定一个数 $l\left(l>0\text{且为整数}\right)$，这个数可以用 $l=\sum_{i=1}^{m}s_i 2^{b_i}3^{t_i}, s_i\in\{-1,0,1\}$ 来表示，在标量乘法中使用这种表示方法可以使得点加的操作次数在很大程度上得到减少，因为一个数 M 能在 $O=\left(\frac{\log l}{\log\log l}\right)$ 的时间复杂度内被表示出来[9,10]。对于给定的 $l>0$，存在 n 个正整数的序列 $\left(L_n\right)_{n>0}$，满足 $L_1=1, L_{m+1}=2^v3^uL_m+t,\cdots,L_n=1$ 其中 $t\in\{-1,1\}, v,u\geq 0$ 则 $\left(L_1,L_2,\cdots,L_n\right)$ 叫做 l 的双基数链[8]。基于双基数链的算法思想是：将整数 l 按上述方式进行改写，然后求出其双基数链 $\left(L_1,L_2,\cdots,L_i,L_{i+1},\cdots,L_n\right)$，那么再由 $L_1P\to L_2P\to\cdots\to L_iP\to L_{i+1}P\to\cdots\to L_nP$ 便可以得到 lP：，而每个 $L_iP\to L_{i+1}P$ 都是运算 $2^v3^uQ\pm P$ (初始 $Q\leftarrow P$)的过程。

3.1.3. 基于窗口的算法优化

基于窗口优化算法的共同思想是将直接影响点加和倍点的运算次数的标量 k 进行编码分段处理，全为零的那一段就不需要计算了。基于窗口的优化算法是一类使用较为广泛的方法，由于它将 k 编码按窗口进行划分从而减少了倍点运算，提高了标量乘的计算速度，因而受到广泛的关注。长度 k 可变的无符号滑动窗口法，划分窗口的原理是：用窗口法对 k 进行从左到右的划分，确定窗口大小 w，确保窗口的开始和结尾为非零元，从非零元开始启动窗口，计算的第一步是预计算窗口的模幂值。

文献[11]提出的无符号滑动窗口算法的基本思想是：通过对 k 进行二进制编码，将编码中首尾为 1 的数据段作为一个窗口划分，以便降低窗口个数，从而来降低乘法运算的次数和预计算量。文献[12]对传统滑动窗口算法[9,13]进行改进，其思想是：首先将划分好的窗口值和窗口权的指数存储在预处理栈中，然后利用预计算表和计算 $2kP + Q$ 算法加快标量乘算法的执行效率。该算法在赋值阶段的效率得到了提升，但也消耗了更多的存储空间。文献[10]提出了一种自适应的滑动窗口标量乘算法，其算法以牺牲存储空间为代价，降低了运行时所需要的时间，计算点加的时间复杂度为：$8A + I + 2M + S$，而计算倍点的时间复杂度为：$3A + I + 2M + S$；并给出实验结果：k、P 的选择很大程度上决定了标量乘的运算速率，因此 k 和 P 的选择不同，其算法的效率提高的程度也不同。文献[10]所提出的算法的思想是：将 k 进行编码，根据 k 的编码中“0”、“1”的分布情况，程序自动设定滑动窗口 w 的值，不需要预先设定好。

3.1.4. 基于重编码的算法优化

基于重编码的算法优化的思想[14]是：首先产生一个新的序列，该序列是由具有倍数关系的两个数进行逐位相与而得到，然后对新产生的序列进行调整，在调整过程中有可能会有借位。其调整步骤为：第一，如果有借位，先将当前位减去两倍的借位。没有借位，则不需要这一步的操作；第二，如果当前位和下一位为 11 时，就把当前位和前一位改写为 10 并使借位标志为 1。若按以上方式进行编码，将会减低海明重量，使运算量得到降低。计算 $kP+lQ$ 时应用该算法，与 JSF(Joint Spare Form)[15]算法效果相比，没有很大的提升，效果相当。

3.2. 底层域的快速运算

减少模乘，模逆，模方的计算量是底层域快速算法优化的共同目标，也是标量乘运算第二步研究的主

要内容。例如：计算 $kP + lQ$ 的算法、基于基转换的正规基快速求逆以及其他相关算法。

3.2.1. 计算 $kP + lQ$ 的算法

文献[16]利用了非相邻形式算法提出了计算 $kP + lQ$ 的算法即用非相邻形式表示 k 和 l，并预计算 $P+Q, P-Q$。基于此算法，对组成的矩阵序列 k、l 的加减法链进行观察可以得出：$\begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix},\begin{pmatrix}\pm1 & 0\\ \pm1 & 0\end{pmatrix},\begin{pmatrix}0 & \pm1\\ 0 & \pm1\end{pmatrix},\begin{pmatrix}\pm1 & 0\\ 0 & \pm1\end{pmatrix},\begin{pmatrix}0 & \pm1\\ \pm1 & 0\end{pmatrix}$ 是相邻两列全部的取值情况，因此可以把每两列作为一个单位进行处理，通过预计算 $P + Q$、$P – Q$、$2P + Q$、$2P – Q$、$2Q + P$、$2Q – P$ 并存储其结果，计算部分可以通过查询预存储表来完成，这样可以进一步的减少运算量，提高运算效率。

算法运行所需要的时间基本与选取的 e 的长度呈线性关系，其中 e 为密钥长度。当选取的 e 的长度为 180 比特时，对于此算法的运行时间为：27.47 ms[16]。此算法极大地提高了算法效率，并且所需的预运算较小，但该算法主要是针对二进制域的椭圆曲线的 $Kp + lQ$ 运算进行研究。

3.2.2. 基于基转换的正规基快速求逆

一般的求逆过程为，对于 $\alpha \in GF\left(2^m\right)$，求满足 $a \cdot b \equiv 1 \left(\mathrm{mod}\, p(t)\right)$ 的 b(最高次数小于 m)，Almost Inverse 算法[17]得到 $a^{-1} \equiv b/t^k \left(\mathrm{mod}\, p(t)\right)$ 是将问题转化为求解满足 $a \cdot b \equiv t^k \left(\mathrm{mod}\, p(t)\right)$ 的 (b,k) 对。用正规基表示被 ECC 选用以后，接着用多项式基取代正规基，这可以通过利用两者的转换模块实现，最后基于正规基的 AI 算法进行求逆变为基于多项式基的 AI 算法进行求逆，这样做是由于在多项式基上的 AI 算法进行求逆比在正规基上的 AI 算法进行求逆效率要高。在多项式基上的 AI 求逆算法可以减少运算量，提高运算效率，这是基于基转换的求逆算法的优点。该算法需要预计算和存储转换矩阵及逆矩阵，其中预计算对求逆算法的效率影响甚微以及其所需存储空间为 $O\left(m^2\right)$。

3.2.3. 其他算法

1997 年 Guajardo 和 Paar[18]提出了计算 4P、8P 的算法，其思想是利用过渡变量将一次求逆操作转化为多次的乘法操作。1999 年，这一“有偿代换”的思想被 Han 和 Tan[19]利用给出了 F_2^m 域上直接计算 $3P$、$5P$、$6P$ 和 $7P$ 的算法和 Lpez、Dahab[20]对文献[18]的改进上，2003 年 Ciet、Joye 和 Lauter[21]将这一“有偿代换”思想变得更加明确，并提出了其他的快速计算算法，例如：$2P + Q$、$3P$、$3P + Q$、$4P$ 等。另一方面，合并、简化一些过渡变量在计算过程中以减少乘法的计算量，从某种程度上来说也就加快了点乘操作的运算速率[22,23]。2001 年计算 $2^k P$ 的算法被 Sakai 和 Sakurai[24,25]所提出，通过对公式进行改写，合并、简化。国内也有此类文章[26,27]。

4. ECC 算法优化的比较分析

4.1. 上层域与底层域算法优化的比较

标量乘主要分为上层运算和底层运算，上层运算集中研究了 k 的有效表示，尽可能的减少海明重量和倍点次数，而底层运算集中研究了底层域的快速算法。上层域通过多进制，窗口法，重编码等方法将 k 进行有效编码，其时间复杂度和空间复杂度对于底层域快速运算来说要小得多。基于重编码的算法优化与基于进制算法优化的目标有相似之处，不过其优化思想不同，基于重编码的优化思想是通过将两个数进行相与，然后进行调整从而减少海明重量。

4.2. 各类上层算法优化之间的比较

降低点加和倍点的操作频率是各类上层算法优化的共同目标。基于进制的算法优化主要改进点在于，第一：通过其他进制进行表示或者多种进制交叉利用以减少倍点次数；第二：用非相邻形式减少非零元素的个数以降低点加的操作频率。而对于基于窗口的算法优化的改进点则不同于基于进制的算法优化，基于窗口的算法优化是将 k 的有效编码进行了划分，分成若干段，对于全是零的小段就免去了计算，减少了运算量。基于折半的算法优化主要在于将 k 表示成二分之一的倍数，用折半运算代替倍点运算，因为折半运算的效率高于倍点运算。以上相关的时间复杂度如表 1。

表格中 A 代表点加运算，D 代表倍点运算，DDA 代表直接计算 $2^k P + Q$，DD 代表直接计算 $2^k P$，w 为滑动窗口的宽度，M 代表一次乘法的运算时间，I 代表一次模逆运算的时间，其他字符为各运算的次数。

Table 1. Time complexity of upper field algorithms
表 1. 上层域算法及其时间复杂度

上层域算法	时间复杂度
NAF 算法	$l'D + A\left(\left\|k'_0\right\| + \cdots + \left\|k'_{l-1}\right\|\right)$
传统滑动窗口算法	$1D + \left(2^{w-1} - 1\right)A + \left\lceil b/(w+1)\right\rceil A + bD$
改进的滑动窗口标量乘算法	$1D + \left(2^{w-1} - 1\right)A + (l-1)DDA + \delta DD$

4.3. 各类底层算法优化之间的比较

各类底层算法优化的共同目标就是要减少运算量，尽可能少的占用存储空间，提高运算效率。底层运算主要研究了求逆、平方、乘法三种运算，以上各类算法优化分别针对不同的运算进行了优化，加快了计算速度。

表 2 给出了一些底层域快速算法所需的运算量。

表中的 I、S、M 分别代表求逆、平方、乘法。表中的数据分别是各类底层域算法在运算过程中所使用的快速算法及其运算量。从表中可知，每个底层域快速算法中求逆运算的次数远小于其他运算次数，这样便可以尽可能的提高标量乘的运算速率。

4.4. 分析与结论

底层域的快速运算和 k 的有效编码共同决定了标量乘的实现速率。若 k 不能被有效的编码表示，底层运算则会变得相当的复杂；反之，若 k 被有效表示，但是没有一个很好的底层算法优化，同样会使得运算量大大增加，降低标量乘的计算效率。鉴于此，可以将底层的算法优化和上层域的算法有机的结合在一起，形成多元化的思想，提高计算效率。

5. 总结与展望

通过研究分析 ECC 算法软件优化的方法，给出了现有研究方向的关系图(图 1)。

Table 2. Computational complexity of fast algorithms of underlying field
表 2. 底层域快速算法及其运算量

底层域快速算法	所需运算量
$2P \pm Q$ [21]	1I + 2S +9M
$P \pm Q$ [21]	I + S + 2M
$4P$ [21]	1I + 5S + 8M
$2^k P$ [25]	1I + (4k + 2)S + (4k + 1)M
$3^k P$ [28]	1I + (5k + 1)S + 12 kM

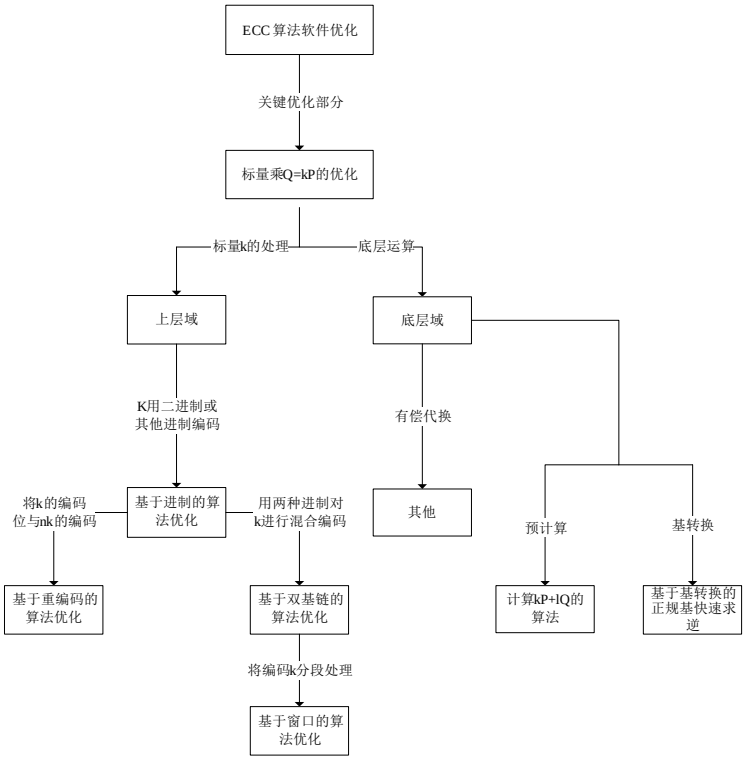

Figure 1. The relationship of the methods of ECC algorithm software optimization
图 1. ECC 算法软件优化方法关系图

使用 ECC 加密的各种优化算法，可以有效的简化运算过程，减少运算量，从而加快运算速率。但是以上 ECC 算法优化只能在上层域或底层域中一个发挥作用，如：基于进制的优化算法，只能对 k 进行有效地表示，从而减少海明重量，但是对于底层域的运算考虑较少，而基于基转换的正规基快速求逆的优化算法只能简化求逆的运算量，而对于 k 的有效表示则未有考虑。因此根据以上分析对于 ECC 优化算法可以从以下三方面进行更深入的优化探索：

1) k 的进制序列中，编码长度决定 ECC 算法中倍点运算的次数，进制序列中非零元素个数决定 ECC 算法中点加运算的次数。因此可以将 k 表示成高进制的序列，从而减少了倍点的次数，综合利用 k 和 nk 逐位相与变换得到正则数，然后利用自适应窗口的思想减少点加次数。

2) 将公式 $Q = kP$ 改写成 $Q = (-k)(-P)$。在椭圆曲线上求 $-P$，运算量较小且易于实现。这样的处理是为了当 k 的进制序列中非零元素的个数大于零元素的个数时，用取反的方法使得零元素个数增多，从而减少海明重量。

3) 综合使用上层域的算法优化和底层域的算法优化，使标量乘的计算速度得到提高。

参考文献 (References)

[1] 郭庆瑞. 基于椭圆曲线密码体制的配电自动化系统信息安全研究[D]. 华北电力大学, 2010.

[2] 胡瑞元, 陈文宇, 甘骏人等. 椭圆曲线加密的硬件实现[J]. 电子设计应用, 2012, 33(2): 117-121.

[3] 张志华, 周捷, 丁可. 非对称数字签名技术在配电自动化系统的应用[J]. 计算机技术及其应用, 2012, 34(3): 39-41.

[4] 王圆圆. 椭圆曲线标量乘法快速实现研究[D]. 扬州大学, 2007.

[5] 黄世中, 羊红光. NAF 编码方法的分析与应用[J]. 计算机研究, 2012, 5: 4-6.

[6] K. Okeya, K. Schmidt-Samoa and S. Cetal. Signed binary representations revisited. Proceedings of Crypto'04, New York: Springer-Verlag, 2004: 123-139.

[7] 邓维勇, 缪祥华. 对称三进制在椭圆曲线标量乘法中的应用[J]. 计算机工程, 2012, 38(5): 152-154.

[8] 王圆圆. 椭圆曲线标量乘法快速实现研究[D]. 扬州大学, 2007.

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

[10] 赵佳, 韩臻. 自适应的椭圆曲线滑动窗口标量乘法[J]. 北京交通大学学报, 2007, 31(2): 6-9.

[11] 王玉华, 王邦菊, 张焕国. 新的无符号滑动窗口算法及其在模幂中的应用研究[J]. 海军工程大学学报, 2009, 21(1): 13-17.

[12] 殷新春, 侯红祥. 改进的滑动窗口标量乘算法[J]. 小型微型计算机系统, 2008, 29(5): 863-866.

[13] A. D. Essame, M. Ramlan, R. Mohammad, et al. A new addition formula for elliptic curves over $GF(2^n)$. IEEE Transactions on Computers, 2002, 51(8): 972-975.

[14] 殷新春, 侯红祥. 基于重编码的快速标量乘算法[J]. 计算机应用研究, 2008, 25(7): 2143-2145.

[15] J. A. Solinas. Low-weight binary representations for pairs of integers, 2001. http://www.cacr.math.uwaterloo.ea/teehm.ports/2001/eorr2001-41.ps.

[16] 胡越梅, 温静静. ECC $kP + lQ$ 点乘算法的优化研究[J]. 计算机与现代化, 2012, 4: 163-166.

[17] R. Schroepel, H. Orman, S. O'Malley, et a1. Fask key exchange with elliptic curve systems. In: D. Coppersmith, Ed., Advances in Cryptology. Santa Barbara: 15th Annual International Cryptology Conference, 1995: 43-56.

[18] J. Guajardo, C. Paar. Efficient algorithms for elliptic curve cryptosystems. Santa Barbara: Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology, 17-21 August 1997: 343-356.

[19] Y. Han, P. C. Tan. Direct computation for elliptic curve cryptosystem. Workshop on Cryptographic Hardware Embedded Systems, Springer-Verlag, 1999: 328-340.

[20] J. L. Pez, R. Dahab. An improvement of the Guajardo-Paar Method for multiplicationon non-supersingular elliptic curves. Proceedings of the XVIII International Conferences of the Chilean Society of Computer Science, 9-14 November 1998: 91-95.

[21] M. Ciet, M. Joye, K. Lauter, et al. Trading inversions for multiplications in elliptic curvecryptography, 2003: 257-277. http://research.microsoft.com/en-us/um/people/klauter/cietjoyelm.pdf

[22] K. Eisentrger, K. Lauter and P. L. Montgomery. Faster elliptic curve arithmetic and improved well pairing evaluation. In: M. Joye, Ed., Proceedings of the 2003 RSA Conference on the Cryptographers' Track, 2003, 2612: 343-354.

[23] 侯保花, 叶震, 尹家生. GF(2m)域上椭圆曲线标量乘算法的改进[A]. 杨义先, 2005 通信理论与技术新进展——第十届全国青年通信学术会议论文集[C], 北京: 北京邮电大学出版社, 2005: 989-994.

[24] Y. Sakai, K. Sakurai. Efficient scalar multiplications on elliptic curves without repeated doublings and their practical performance. Lecture Notes in Computer Science, 2000: 59-73.

[25] Y. Sakai, K. Sakurai. Efficient scalar multiplications on elliptic curves with direct computations of several doublings. IEICE Transactions on Fundamentals, 2001, E84-A(1): 120-129.

[26] 牛力, 祝跃飞. 直接计算的一般算法[J]. 信息工程大学学报, 2003, 4(1): 3-4.

[27] 李湛. 一种改进的椭圆曲线密码实现算法[J]. 电子科技, 2004, 178(7): 31-33.

[28] 殷新春, 王圆圆, 侯红祥. 一种新的基于双基数链的标量乘法快速算法[A]. 王小云, 杨义先, 主编, 密码学进展——CHINA CRYPT, 2006 第九届中国密码学学术会议论文集[C]. 北京: 中国科学技术出版社, 2006: 59-66.