$GF(3^m)$椭圆曲线群快速算术运算研究

端木庆峰[1] 王衍波[1] 张凯泽[1] 雷凤宇[2]
(解放军理工大学通信工程学院 南京 210007)[1] (华中科技大学计算机学院信息安全系 武汉 430074)[2]

摘 要 详细研究了$GF(3^m)$上椭圆曲线基本算术运算,给出并证明$GF(3^m)$上超奇异和非超奇异椭圆曲线仿射坐标系下点加、倍点、3倍点和3^k倍点计算公式,提出高效3^k倍点递归算法,在逆乘率较高时,其效率要优于逐次3倍点算法。在此基础上,提出一种新的变长滑动窗口wrNAF标量乘算法,其在保证较少点加法运算优点的同时可有效降低3倍点的计算量。

关键词 椭圆曲线,标量乘法,小素数扩域,三元域
中图法分类号 TN918.1 **文献标识码** A

Research on Fast Arithmetic of Elliptic Curve over $GF(3^m)$

DUANMU Qing-feng[1] WANG Yan-bo[1] ZHANG Kai-ze[1] LEI Feng-yu[2]
(Institute of Communications Engineering, PLA University of Science & Technology, Nanjing 210007, China)[1]
(College of Computer Science & Technology, Huazhong University of Science & Technology, Wuhan 430074, China)[2]

Abstract The arithmetic on supersingular and non-supersingular elliptic curve over $GF(3^m)$ were researched, and computational formulas of point addition, $2P$, $3P$ and 3^kP in affine coordinate were given and confirmed. Based on these, a new 3^kP recursive algorithm was proposed which was prior to multiple tripling point algorithms when the speed ratio of field inversion to field multiplication was high. Furthermore, we proposed a new variable length sliding window base-3 wrNAF scalar multiplication algorithm which can reduce the cost of tripling points needed in kP computation.

Keywords Elliptic curve, Scalar multiplication, Small prime extension field, Characteristic three field

1 引言

椭圆曲线公钥密码体制(ECC)[1]是Neal Koblitz和Victor Miller在1985年分别独立提出的基于有限域上椭圆曲线群离散对数困难问题的一种公钥密码体制。与RSA等公钥密码体制相比,ECC不存在亚指数攻击算法,使得以相对较小的密钥长度即可达到很高的安全性。二元域$GF(2^m)$和大素数域$GF(p)$(p为大素数)是最为常用的2个有限域,因而其上的椭圆曲线密码体制目前研究得也最为充分。相比而言,$GF(3^m)$-ECC的研究工作还很少开展。$GF(3^m)$-ECC有类似于$GF(2^m)$-ECC计算速度快的某些特性,但也有自己的特殊性质,这使得其上算术运算效率非常高,极适合作为安全密码算法的载体。特别是,随着Weil对和Tate对理论研究的不断进展以及基于此而设计的各种安全协议的广泛应用,使得人们对$GF(3^m)$-ECC更是产生浓厚兴趣。

本文首先介绍椭圆曲线基础理论,对$GF(3^m)$上椭圆曲线进行分类,进而研究$GF(3^m)$椭圆曲线群在仿射坐标系下各基本点的算术运算公式。最后,对$GF(3^m)$-ECC核心标量乘法运算进行分析。

2 椭圆曲线

设K为域,K上仿射坐标系下的Weierstrass方程[2]为

$$y^2+a_1xy+a_3=x^3+a_2x^2+a_4x+a_6 \tag{1}$$

其中$a_1,a_2,a_3,a_4,a_6\in K$。

椭圆曲线为仿射平面$A^2(K)=K\times K$中所有满足方程点及无穷远点O的集合,$\frac{E}{K}$在平面$K^2=K\times K$中的点称为K-有理点,$\frac{E}{K}$全体K-有理点(包括O)记为$E(K)$。令$b_2=a_1^2+4a_2$,$b_4=2a_4+a_1a_3$,$b_6=a_3^2+4a_6$,$b_8=a_1^2a_6+4a_2a_6-a_1a_3a_4+a_2a_3^2-a_4^2$,$c_4=b_2^2-24b_4$,$c_6=b_2^3+36b_2b_4-216b_6$,则方程(1)判别式$\Delta=-b_2^2b_8-8b_4^3-27b_6^2+9b_2b_4b_6$,$j$-不变量$j(E)=c_4^3/\Delta$($\Delta\neq0$),$E(K)$非奇异当且仅当$\Delta\neq0$。

当$K=GF(3^m)$时,$E(K)$称为定义在$GF(3^m)$上的椭圆曲线。根据j-不变量,$GF(3^m)$上的椭圆曲线可分为超奇异和非超奇异椭圆曲线。若$b_2=a_1^2+a_2=0$,则式(1)可变换为$E/K:y^2=x^3+b_4x+b_6$,$b_4,b_6\in K$。$\Delta(E)=-b_4^3$,$j(E)=0$,该曲线为超奇异的。若$b_2=a_1^2+a_2\neq0$,则变量允许代换$(x,y)\to(x+b_4/b_2,y)$,可将曲线(1)变换为$E/K:y^2=x^3+d_2x^2+d_6$,其中$d_2=b_2$,$d_6=(2b_2^2b_4^2+b_4^3+b_2^3b)/b_2^3$。$\Delta(E)=-d_2^3d_6$,$j(E)=-d_2^3/d_6$,该曲线为非超奇异的。

椭圆曲线$E(K)$及其关于椭圆曲线点"弦和切线"加法运算可构成一个加法交换群。给定$E(K)$,n阶点P所生成的子群记为$[P]$,Q为$[P]$中任意一点。椭圆曲线离散对数问题

到稿日期:2008-06-25 本文受国家自然科学基金课题(60703048)资助。
端木庆峰(1980-),男,博士研究生,研究方向为应用密码学和信息隐藏,E-mail:duanmuziyun@126.com;王衍波(1961-),男,教授,研究方向为网络安全和现代密码学;张凯泽(1968-),男,副教授,研究方向为网络安全;雷凤宇(1980-),女,博士研究生,研究方向为公钥密码、可证明安全性、密码学理论和实践。

(ECDLP)就是在已知 P 和 Q，求满足 $Q=kP$ 的整数 k，$0\leqslant k\leqslant n-1$。点 P 称为基点，而 $k=\log_P Q$ 称为 Q 关于基点 P 的离散对数，计算 $k=\log_P Q$ 是困难的。椭圆曲线公钥密码体制(ECC)是基于椭圆曲线离散对数问题构建的，因而其安全性与 ECDLP 困难性等价。

3 $GF(3^m)$上椭圆曲线群算术运算

$GF(3^m)$有限域运算[3]是 $GF(3^m)$上椭圆曲线群算术运算的基础。采用多项式基，$GF(3^m)=GF(3)[x]/f(x)$，$f(x)$定义为 $GF(3)$上 m 次不可约多项式，$\forall a\in GF(3^m)$可唯一表示为 $a=a_{m-1}x^{m-1}+\cdots+a_1x+a_0$，$a_i\in GF(3)$。元素加法和减法即为多项式加减运算，其计算量可以忽略。乘法和平方运算可采用经典行列扫描算法计算乘积，然后进行约减，也可采用 Karatsuba 和 Ofman 优化算法。$GF(3^m)$特征为 3，则 $\forall a\in GF(3^m)$，$3a=0$，进而 $\forall A(x)\in GF(3^m)$，$A(x)^3=A(x^3) \bmod f(x)$，因而元素 3 次方运算比较快。若选择不可约 3 项式 $f(x)=x^m+p_tx^t+p_0$，$t<m/3$，则仅需 1 次约减[4]。通常 3 次方要比乘法运算快至少 10 倍左右。域逆是有限域最费时运算，可采用小素数扩域上各种标准求逆算法。令 M 表示域乘法，S 表示域平方，C 表示 3 次方，I 表示域逆，A 表示加法。一般而言，$t(I)>>t(M)>>t(S)>>t(C)$，$t(X)$表示计算一次域上 X 运算所需要的时间，定义逆乘率 α 为逆和乘法性能比 $\alpha=t(I)/t(M)$。

3.1 非超奇异椭圆曲线群算术

E 为非超奇异椭圆曲线 $(j(E)\neq 0)$，$E/GF(3^m):y^2=x^3+ax^2+b$，$a,b\in GF(3^m)$，$P=(x_1,y_1)$，$Q=(x_2,y_2)$，$P,Q\neq O$，$P\neq\pm Q$。

(1)点加法。令 $R=P+Q=(x_3,y_3)$，则 $x_3=k^2-a-x_1-x_2$，$y_3=k(x_1-x_3)-y_1$，其中 $k=(y_2-y_1)/(x_2-x_1)$，点加需 $1I+2M+1S$。若将 x_3 代入 y_3 可得 $x_3=k^2-a-x_1-x_2$，$y_3=-k^3+ka+y_1+y_2$，点加需 $1I+2M+1S+1C$。若 $a=1$，则需 $1I+1M+1S+1C$，优于文献[5]的结果。

(2)倍点。令 $R=2P=(x_3,y_3)$，则 $x_3=k^2-a+x_1$，$y_3=k(x_1-x_3)-y_1$，其中 $k=ax_1/y_1$，倍点需 $1I+3M+1S$，若 $a=1$，则需 $1I+2M+1S$。将 x_3 代入 y_3 可得 $x_3=k^2-a+x_1$，$y_3=-k(k^2-a)-y_1$。此时，倍点需 $1I+3M+1S$，若 $a=1$，则需 $1I+2M+1S$。

(3)$3P$ 和 3^kP。令 $R=3P=(x_3,y_3)$，则先倍点再点加可得 3 倍点计算公式，令 $A=a^{-2}$，$B=a^{-3}$ 和 $C=ab$(可预先计算)，$x_3=\dfrac{(x_1^3+b)^3-a^3bx_1^3}{a^2(x_1^3+b)^2}=A\cdot(x_1^3+b)-C\cdot x_1^3\cdot\left(\dfrac{1}{x_1^3+b}\right)^2$，$y_3=\dfrac{y_1^9-a^3y_1^3(x_1^3+b)^2}{a^3(x_1^3+b)^3}=B\cdot\left(\dfrac{y_1^3}{x_1^3+b}\right)^3-\dfrac{y_1^3}{x_1^3+b}$，则 3 倍点运算需 $1I+5M+1S+3C$，若 $a=1$，$A=B=1$，则需 $1I+3M+1S+3C$。该结果比文献[5]给出的结果要优。

令 $a=1$，$x_3=\dfrac{(x_1^3+b)^3-bx_1^3}{(x_1^3+b)^2}$，$y_3=\dfrac{y_1^9-y_1^3(x_1^3+b)^2}{(x_1^3+b)^3}$，令 $C_0=1$，$A_1=x_1$，$B_1=C_1=A_1^3+b$ 和 $D_1=y_1^3$，则 $3P=(x_3,y_3)$，$x_3=(B_1^3-bA_1^3C_0^{12})/C_1^2$，$y_3=(D_1^3-D_1C_1^3)/C_1^3$，算法计算 x_3 和 y_3 总共需要 $1I+3M+1S+3C$。

令 $A_2=B_1^3-bA_1^3$，$B_2=A_2^3+b(C_1^2)^3$，$C_2=B_2C_1^3=B_2B_1^3$ 和 $D_2=(D_1^3-D_1C_1^3)^3$，则 $3^2P=(x_{3-2},y_{3-2})$，$x_{3-2}=(B_2^3-bA_2^3C_1^{12})/C_2^2$，$y_{3-2}=(D_2^3-D_2C_2^3)/C_2^3$，则 A_1,B_1,C_1 和 D_1 需 $2C$，计算 A_2 需要 $1M+1C$，B_2 需要 $1M+1S+2C$，C_2 需要 $1M$，D_2 需要 $1M+2C$(C_1^2 已知)，计算 x_{3-2} 和 y_{3-2}需要 $1I+5M+2S+3C$，总共需要 $1I+9M+3S+10C$。相比而言，计算 3^2P 采用 2 次 3 倍点公式需要 $2I+6M+2S+6C$。当 $t(1I)>t(3M+1S+4C)$，递归算法就比较优。同理，3^3P 递归计算需要 $1I+14M+5S+16C$，而 3 次 3 倍点需要 $3I+9M+3S+9C$。

令 $A_k=B_{k-1}^3-bA_{k-1}^3C_{k-2}^{12}$，$B_k=A_k^3+b(C_{k-1}^2)^3$，$C_k=B_kC_{k-1}^3$ 和 $D_k=(D_{k-1}^3-D_{k-1}C_{k-1}^3)^3$，则 $3^kP=(x_{3-k},y_{3-k})$，$x_{3-k}=(B_k^3-bA_k^3C_{k-1}^{12})/C_k^2$，$y_{3-k}=(D_k^3-D_kC_k^3)/C_k^3$。算法计算 $A_1,B_1,C_1,D_1,A_2,B_2,C_2,D_2$ 需要 $4M+1S+7C$，迭代计算 A_n,B_n,C_n,D_n，$n=3,\cdots,k$，A_n 需 $2M+1S+1C$(C_{k-2}^6已计算)，B_n 需 $1M+2C$，C_n 需 $1M+1C$，D_n 需 $1M+1S+2C$，总共需要$(n-2)(6C+2S+5M)$，计算 x_{3-k}和 y_{3-k}需 $1I+5M+2S+3C$，则计算$3^kP(k>2)$需要 $1I+(5k-1)M+(2k-1)S+(6k-2)C$。而 k 次 3 倍点计算 3^kP 需要 $kI+3kM+kS+3kC$。当 $t(1I)>t(2M+1S+3(1-(k-1)^{-1})C)$或者 $\alpha>3$ 时，3^kP 递归算法明显要好于 k 次 3 倍点算法。由公式可以看出，3^kP 递归算法主要以域乘和平方运算来取代费时的域逆运算，算法仅需 1 次域逆。

3.2 超奇异椭圆曲线群算术

E 为超奇异椭圆曲线 $(j(E)=0)$，$E/GF(3^m):y^2=x^3+ax+b$，$a,b\in GF(3^m)$，令 $P=(x_1,y_1)$，$Q=(x_2,y_2)$，$P,Q\neq O$，$P\neq\pm Q$。

(1)点加法。令 $R=P+Q=(x_3,y_3)$，则 $x_3=k^2-x_1-x_2$，$y_3=k(x_1-x_3)-y_1$，其中 $k=(y_2-y_1)/(x_2-x_1)$，点加需 $1I+2M+1S$。若将 x_3 代入 y_3 可得 $x_3=k^2-x_1-x_2$ 和 $y_3=-k^3+y_1+y_2$，点加需 $1I+1M+1S+1C$。

(2)倍点。令 $R=2P=(x_3,y_3)$，则 $x_3=k^2+x_1$，$y_3=k(x_1-x_3)-y_1$，其中 $k=-a/y_1$，倍点需 $1I+2M+1S$，若 $a=1$，则需 $1I+1M+1S$。将 x_3 代入 y_3 可得 $x_3=k^2+x_1$ 和 $y_3=-k^3-y_1$，倍点需 $1I+1M+1S+1C$，若 $a=1$，则需 $1I+1S+1C$。

(3)$3P$ 和 3^kP。令 $R=3P=(x_3,y_3)$，则通过先倍点再点加可得 3 倍点计算公式为 $x_3=x^9/a^4+b^3/a^4-b/a=(x^3+b)^3/a^4-b/a$，$y_3=-y^9/a^6$。令 $A=a^{-4}$，$B=b^3a^{-4}-ba^{-1}$ 和 $C=-a^{-6}$(可预先计算)，则 $x_3=A\cdot x^9+B$，$y_3=C\cdot y^9$，3 倍点运算需 $2M+4C$。若 $a=1$，则 $A=C=1$，3 倍点需要 $4C$。

若 $a\neq 1$，则 $A\neq 1$，$C\neq 1$。由 3 倍点公式可得，$3^2P=(A\cdot A^9x^{9\times 9}+AB^9+B,C\cdot C^9y^{9\times 9})$，若 $A\cdot A^9$，AB^9+B 和 $C\cdot C^9$ 预先计算好，则算法需 $2M+8C$。同理，$3^3P=(A\cdot A^9A^{9\times 9}x^{9\times 9\times 9}+A\cdot A^9B^{9\times 9}+AB^9+B,C\cdot C^9C^{9\times 9}y^{9\cdot\times 9\times 9})$，若 $A^{9\times 9+9+1}$，$C^{9\times 9+9+1}$和 $A\cdot A^9B^{9\times 9}+AB^9+B$ 预先计算好，则算法需 $2M+12C$。令 $A_k=\prod_{i=0}^{k}A^{9^i}$ $(A_{-1}=1)$，$B_k=\sum_{i=0}^{k}(A_{i-1}B^{9^i})$和 $C_k=\prod_{i=0}^{k}C^{9^i}$(可预先计算)，则 $3^kP=(A_{k-1}x^{9^k}+B_{k-1},C_{k-1}y^{9^k})$，算法需要 $2M+4kC$。相比，计算 3^kP 采用 k 次 3 倍点公式需要 $2kM+4kC$，比 3^kP 递归算法多 $2(k-1)M$。

若 $a=1$，则 $A=C=1$，$x_3=x^9+B$，$y_3=y^9$。由 3 倍点公式得，$3^2P=(x^{9\times 9}+B^9+B,y^{9\times 9})$，若 B^9+B 预先计算好，则算法需要 $8C$。递推可得 $3^kP=(x^{9^k}+B_{k-1},y^{9^k})$，$B_k=\sum_{i=0}^{k}$

B^{9^i}（可预先计算），算法需要 $4kC$。相比而言，计算 3^kP 采用 k 次 3 倍点需要 $4kC$，但多 $k-1$ 次加法。

$3P$ 和 3^kP 需要计算元素的 3^i 次方运算（记为 Φ），$3P$ 需要计算 x^9 和 y^9，3^kP 需要计算 $x^{3^{2k}}$ 和 $y^{3^{2k}}$。$\forall A(x)\in GF(3^m)$，$A(x)^{3^i}=\sum_{j=0}^{m-1}(a_jx^j)^{3^i}=\sum_{j=0}^{m-1}a_jx^{j3^i}$，该多项式次数为 $3^i(m-1)$，需要进行约减运算。根据 Frobenius 映射运算，令 $x^{j3^i} \bmod f(x)=s_{0,j}^{(i)}+s_{1,j}^{(i)}x+\cdots+s_{m-1,j}^{(i)}x^{m-1} \bmod f(x)$，其中 $j=1,2,\cdots,m-1$，则 $A(x)^{3^i} \bmod f(x)$

$$=\begin{bmatrix}1 & s_{0,1}^{(i)} & s_{0,2}^{(i)} & \cdots & s_{0,m-1}^{(i)}\\ 0 & s_{1,1}^{(i)} & s_{1,2}^{(i)} & \cdots & s_{1,m-1}^{(i)}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ 0 & s_{m-1,1}^{(i)} & s_{m-1,2}^{(i)} & \cdots & s_{m-1,m-1}^{(i)}\end{bmatrix}\begin{bmatrix}a_0\\ a_1\\ \vdots\\ a_{m-1}\end{bmatrix} \bmod f(x)$$

$$=T^{(i)}\begin{bmatrix}a_0\\ a_1\\ \vdots\\ a_{m-1}\end{bmatrix} \bmod f(x)。$$

对 $GF(3^m)$，矩阵 $T^{(i)}$ 中的元素 $s_{k,j}^{(i)}\in\{0,1,-1\}$，$k,j=0,\cdots,m-1$，且 $a_i\in\{0,1,-1\}$。域多项式 $f(x)$ 选择为不可约三项式，因此 $T^{(i)}$ 中的元素大部分都为 0。通过以上分析，$GF(3^m)$ 中元素 3^i 次方运算最多需要 $GF(3)$ 中 $m(m-2)+1=(m-1)^2$ 次加法，比直接计算 i 次 3 次方运算要优（需 i 次约减）。$f(x)$ 选定后，矩阵 $T^{(i)}$ 固定，$T^{(i)}$ 可以预先计算好。此时，计算 3^kP，$a\neq1$ 时需要 $2M+2\Phi$，而 $a=1$ 时仅需要 2Φ。

以上讨论仿射坐标系各基本点计算量具体如表 1 所列。

表 1 $GF(3^m)$ EC 仿射坐标系基本点计算量比较

EC 操作		非超奇异椭圆曲线	超奇异椭圆曲线
P + Q	a≠1	1I + 2M + 1S	1I + 1M + 1S + 1C
	a=1	1I + 1M + 1S + 1C	1I + 1M + 1S + 1C
2P	a≠1	1I + 2M + 1S	1I + 1M + 1S + 1C
	a=1	1I + 2M + 1S	1I + 1S + 1C
3P	a≠1	1I + 5M + 1S + 3C	2M + 4C
	a=1	1I + 3M + 1S + 3C	4C
3^2P	a≠1	—	2M + 8C
	a=1	1I + 9M + 3S + 10C	8C
3^3P	a≠1	—	2M + 12C
	a=1	1I + 14M + 5S + 16C	12C
3^kP	a≠1	—	2M + 4kC 或 2M + 4Φ
	a=1	1I + (5k−1)M + (2k−1)S + (6k−2)C	4kC 或 2Φ

4 标量乘法运算

$Q=kP$ 是 ECC 的核心运算，称为标量乘法运算（Scalar Point Multiplication），其可分解为椭圆曲线群上一系列基本点运算组合，如点加和倍点等[6,7]。目前，其主要计算方法是将整数 k 以不同类型基展开 $kP=\sum_{i=0}^{l}k_ir^i(P)(P)=r(\cdots r(rk_lP+k_{l-1}P)+\cdots+k_1P)+k_0P$，$k_i\in D_r$，$D_r$ 为系数数字集合，进而采用加减链等技术计算。由于 ECC 的特殊性质，k 以不同形式展开时，kP 计算量存在较大差别。令 A 表示点加，D 表示倍点，TD 表示 3 倍点，$H_r(k)$ 表示为 k 的 r 进制汉明距离。

类似 $GF(2^m)$，$GF(3^m)$-ECC 标量乘法 $Q=kP$ 可采用倍点加算法，其需要 $\lfloor\log_2k\rfloor$ 次倍点和 $H_2(k)-1$ 次点加。考虑到 $GF(3^m)$ 上特殊 3 倍点公式，N P Smart[5] 推广倍点加提出 3 倍点加标量乘算法。若 k 为 n 比特，令 $m=\lfloor\log_32\rfloor+1$，则算法需要 $m-1$ 次 3 倍点和 $H_3(k)-1$ 次点加运算。相比倍点加算法，点加平均减少 $0.5nA-2(\lfloor n\log_32\rfloor+1)A/3\approx0.08nA$，倍点平均减少 $nD-(\lceil n\log_32\rceil+1)TD$。

为进一步减少点加，需要保持 k 展开长度同时减少非零数字密度。定义整数 k 的基 r 非邻接表示型（rNAF）[8,9] 和窗宽 w 基 r 非邻接表示型（wrNAF）。当基数 $r=3$ 时，整数 krNAF 展开数字集 $D_r=\{0,\pm1,\pm2,\pm4\}$，非零数字密度为 $(r-1)/(2r-1)=2/5$，算法平均需要 $\lfloor\log_3k\rfloor+1$ 次 3 倍点和 $2(\lfloor\log_3k\rfloor+1)/5$ 次点加。整数 k wrNAF 展开数字集 $D_{w,r}=\{0,\pm1,\pm2,\cdots,\pm\lfloor(3^w-1)/2\rfloor\}$，非零数字密度为 $(r-1)/(w(r-1)+1)=2/(2w+1)$，算法平均需要 $\lfloor\log_3k\rfloor+1$ 次 3 倍点和 $2(\lfloor\log_3k\rfloor+1)/(2w+1)$ 次点加。可见，整数 k wrNAF 具有更低的非零数字密度。

整数 k 基 3 窗宽 wrNAF 展开式中任意 w 个邻接数字中最多有 1 个非零数字[8]，因此任意两个非零数字之间至少有 $w-1$ 个 0。由上节分析可知，在逆乘率较高时，3^tP 递归算法要比逐次 3 倍点性能要优。综合以上两方面内容，在计算标量乘法过程中，可以直接递归计算 3^tP 来取代逐次计算多个 3 倍点，借此减少算法中 3 倍点总的计算量。据此思想，可以构造类似于变长滑动窗口技术的整数 k wrNAF 标量乘算法。具体算法如下。

算法 1 变长滑动窗口 wrNAF 标量乘算法

输入：窗宽 w，k，P∈E(GF(3^m)) 输出：kP

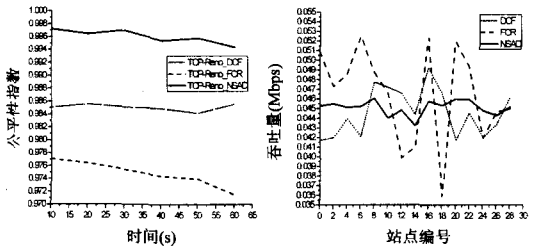
1. 构造整数 k 基 3 宽度 wrNAF 展开型，3rNAF(k)=(k_{m-1},…,k_0)
2. 对于 i∈{1,2,…,⌊(3^w−1)/2⌋}，预计算 P_i=iP
3. Q←k_{m-1}P，i←m−2
4. 当 i≥0 时，重复执行
   - 4.1 若 $k_i=\cdots=k_0=0$，则 u←(k_i,…,k_0)，t=i+1，否则找最大 t 使 u←(k_i,…,k_{i-t+1})满足 $k_i=\cdots=k_{i-t+2}=0$，$k_{i-t+1}\neq0$。
   - 4.2 计算 Q←3^tQ（使用 3^tP 递归算法）
   - 4.3 若 u>0，则 Q←Q+P_u；若 u<0，则 Q←Q−P_{-u}
   - 4.4 i←i−t
5. 返回 Q

令 k 窗宽 wrNAF 展开长度为 m，由于整数 k 基 3 窗宽 wrNAF 展开式中任意两个非零数字之间至少有 $w-1$ 个 0，因而步 4 最多执行 $\lfloor m/w\rfloor+1$ 次，每次执行一次 3^tQ（可证 $t\geqslant w$）和一次点加法，算法最多需要 $\lfloor m/w\rfloor+1$ 次 $3^tQ(t\geqslant w)$ 运算和 $\lfloor m/w\rfloor+1$ 次点加法运算。考虑到 k wrNAF 展开非零数字密度为 $(r-1)/(w(r-1)+1)$，步 4 循环平均执行 $m(r-1)/(w(r-1)+1)=2m/(2w+1)$ 次，算法平均需 $2m/(2w+1)$ 次 $3^tQ(t\geqslant w)$ 运算和 $2m/(2w+1)$ 次点加法运算。由于算法采用 $3^tQ(t\geqslant w)$ 递归算法，在逆程率较高时可以减少标量乘法总的 3 倍点计算量。此外，由于滑动的窗口可变，因而可以更有效增加零窗口长度，减少点加运算数量。

结束语 3 特征小素数扩域作为 $GF(p^m)$ 更加特殊的一种类型，定义于其上的椭圆曲线密码算法更加优越。本文对 $GF(3^m)$-ECC 算术运算进行研究，给出 $GF(3^m)$ 上超奇异和非超奇异椭圆曲线仿射坐标系下点加、倍点、3 倍点以及 3^k 倍点计算公式，提出高效 3^kP 递归算法。$GF(3^m)$ 非超奇 EC3^kP 递归算法，当 $a=1$ 时，需 $1I+(5k-1)M+(2k-1)S+(6k-2)C$。在逆乘率 $a>3$ 时，其要优于逐次 3 倍点。而对 $GF(3^m)$ 超奇异 EC3^kP 递归算法，当 $a\neq1$ 时，算法需要 $2M+4kC$ 或

（下转第 114 页）

绝对不公平。



(a) TCP 公平性指数　(b) 站点吞吐量

图 4　TCP 与 MAC 层的公平性对比图

由图 4(a)可知，无线局域网 MAC 层采用 FCR 时，TCP 的公平性最差，各个站点的 TCP 流吞吐量不平衡，这是因为 FCR 使无线信道带宽在各个站点的分配不公平。如图 4(b)所示，无线局域网 MAC 层采用 FCR 时，6 号、16 号和 20 号站点的吞吐量远大于 12 号、18 号和 24 号站点的吞吐量，必然导致 6 号、16 号和 20 号站点的 TCP 吞吐量大于 12 号、18 号和 24 号站点的 TCP 吞吐量。根据 FCR 的工作原理可知，FCR 使成功发送数据的站点保持很小的竞争窗口值，再次抢占信道的概率大，而其它站点则维持较大的竞争窗口值，抢占信道的概率小，造成某些站点占用较多的信道带宽资源，而引起不公平。无线局域网 MAC 层采用 NSAD 时，各站点吞吐量基本维持在 0.0445Mbps 左右，各站点较公平地占用信道带宽资源，保证了 TCP 较好的公平性。因此，MAC 机制的公平性将直接影响无线局域网中 TCP 的公平性。

结束语　IEEE802.11 无线局域网 MAC 层分别为 DCF、FCR 和 NSAD 3 种机制时的 TCP 性能仿真实验结果表明，由 MAC 机制所决定的信道冲突率、单站点连续冲突率、MAC 帧丢失率、MAC 帧传输时延抖动幅度和频率、网络吞吐量和公平性将直接影响无线局域网中 TCP 的吞吐量、公平性和稳定性。因此，在设计或改进无线局域网 MAC 机制时，不仅以提高网络吞吐量与信道利用率为目标，还应重视网络帧丢失率、时延抖动、公平性等性能的改善，减小 MAC 机制对 TCP 性能的影响。

参 考 文 献

[1] IEEE Std. 802.11, 1999 Edition, PartII: IEEE 802.11 Wireless LAN medium access control (MAC) and physical (PHY) layer specifications[S]. 1999

[2] IEEE802.11e/D13.0. Draft supplement to part11: wireless medium access control (MAC) and physical layer (PHY) specifications: Medium access control (MAC) enhancements for quality of service (QoS) [S]. January 2005

[3] Bianchi G. Performance analysis of the IEEE 802.11 distributed coordination function[J]. IEEE JSAC, 2000, 18(3)

[4] Cali F, Conti M, Gregori E. Dynamic Tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit[J]. IEEE/ACM Transactions on Networking, 2000, 8(6): 785-799

[5] Ramaiyan V, Kumar A. Fixed point analysis of single cell IEEE 802.11e WLANs: Uniqueness, multistability and throughput differentiation[C]// Proc. of ACM SIGMETRICS. June 2005

[6] 彭泳，程时端. 一种自适应无线局域网协议[J]. 软件学报，2004，4：604-615

[7] Younggoo K, Yugang F, Haniph L. A Novel MAC Protocol with Fast Collision Resolution for Wireless LANs[C]// Proc. of INFOCOM2003. July 2003

[8] Paxson V, Allman M, Stevens W. TCP Congestion Control[S]. RFC 2581, 1999

[9] Xiang C, Hongqiang Z, Jianfeng W, et al. A Survey on Improving TCP Performance over Wireless Networks[J]. Network Theory and Applications, 2005, 16(5): 657-695

[10] Jiwei C, Mario G, Yengzhong L, et al. TCP with delayed ack for wireless networks[J]. Adhoc Networks, 2008, 6(7): 1098-1116

[11] Qian W, Mingwei G, Williamson C. TCP fairness issues in IEEE 802.11 wireless LANs[J]. Computer Communications, 2008, 31(10): 2150-2161

(上接第 98 页)

$2M+2\Phi$，当 $a=1$ 时，算法需要 $4kC$ 或 2Φ，其要优于逐次 3 倍点。在此基础上，给出一种变长滑动窗口 wrNAF 标量乘算法，其可有效降低 3 倍点的计算量。优化 $GF(3^m)$-ECC 核心标量乘法运算的性能是需要作进一步研究的课题。

参 考 文 献

[1] Menezes A J. Elliptic Curve Public Key Cryptosystems [M]. Boston: Kluwer Academic Publishers, 1993

[2] Barbosa M, Moss A, Page D. Compiler assisted elliptic curve cryptography[EB/OL]. Cryptology ePrint Archive, 2007/053. http://eprint.iacr.org/2007/081, 2007

[3] Ahmadi O, Hankerson D, Menezes A. Software implementation of arithmetic in F_{3^m}[C]// International Workshop on the Arithmetic of Finite Fields (WAIFI 2007). Berlin: Springer-Verlag, 2007: 85-102

[4] Bertoni G, Guajardo J, Kumar S, et al. Efficient GF(p^m) arithmetic architectures for cryptographic applications[C]// CTRSA2003. Berlin: Springer-Verlag, 2003: 158-175

[5] Smart N P, Westwood E J. Point multiplication on ordinary elliptic curves over fields of characteristic three[J]. Applicable Algebra in Engineering, Communication and Computing, 2003, 13(6): 485-497

[6] Negre C. Scalar multiplication on elliptic curves defined over fields of small odd characteristic[C]// INDOCRYPT 2005. Berlin: Springer-Verlag, 2005: 389-402

[7] Kim K H, Kim S I, Choe J S. New fast algorithms for arithmetic on elliptic curves over finite fields of characteristic three[EB/OL]. Cryptology ePrint Archive. Technical Report 2007/179. http://eprint.iacr.org/2007/179, 2007

[8] Takagi T, Yen S M, Wu B C. Radix-r non-adjacent form[C]// 7th Information Security Conference, ISC 2004. Berlin: Springer-Verlag, 2004: 99-110

[9] Han D G, Takagi T. Some Analysis of Radix-r Representations [EB/OL]. Cryptology ePrint Archive. Technical Report 2005/402. http://eprint.iacr.org/2005/402, 2005