1Y-4

ActionScript による GF(3) 上の η_T ペアリング演算ライブラリ

本郷 考一 † 毛利 公美 † 白石 善明 ‡
† 岐阜大学 ‡ 名古屋工業大学

1 はじめに

楕円曲線上で定義されるペアリングは，これまで実現できなかった ID ベース暗号などの新しい暗号プロトコルの構築が可能であることから注目されており，演算ライブラリの実装や高速なアルゴリズムの研究などが盛んに行われている．一方で，Web アプリケーションによるサービスにおいて，ユーザ側の Web ブラウザ上で暗号化／復号を行うシステムのニーズが高まっている．

先に，著者らはセキュアな Web アプリケーションの開発手段として，ActionScript による GF(3) 上の η_T ペアリング演算ライブラリを開発した [1]．しかし，その有用性については十分な検証を行ったとは言えない．

本稿では，[1] で開発したライブラリの Web アプリケーション開発における有用性を示す．また，ペアリングを用いたセキュアな Web アプリケーションの開発に必須となる，楕円曲線上の点を計算する MapToPoint を関数として提供する．ライブラリの有用性は，我々の開発したライブラリと公開されている他のライブラリとの速度比較を行い，その評価を行う．さらに，ペアリングを用いたセキュアな Web アプリケーションへの適用例として BF 方式の ID ベース暗号 [5] を実装し，その速度評価から Web アプリケーション開発に役立つことを示す．

2 η_T ペアリング

本研究では，標数 3 有限体での η_T ペアリングを扱う．このペアリングは，楕円曲線 E 上の 2 点を入力，有限体 F_{3^m} の 6 次拡大体の元を出力とする 2 入力 1 出力の写像関数である．このとき，楕円曲線 E は以下のように表現される．

$$E : y^2 = x^3 - x + b, (b = \pm 1)$$

また，F_{3^m} における E 上の点の集合 $E(F_{3^m})$ を以下のように定義する．

$$E(F_{3^m}) = \{O\} \cup \{(x, y) \in F_{3^m} \times F_{3^m} | y^2 = x^3 - x + b\}$$

このとき，O は楕円曲線における無限遠点を表す．さらに，標数 3 の η_T ペアリングは以下のように定義される．

$$\eta_T : E(F_{3^m})[r] \times E(F_{3^m})[r] \to \mu_r$$
$$E(F_{3^m})[r] = \{P \in E(F_{3^m}) | rP = O\}, \mu_r = F_{3^{6m}}/(F_{3^{6m}})^r$$

なお，η_T ペアリングは双線形性 $\eta_T(aP, Q) = \eta_T(P, aQ) = \eta_T(P, Q)^a$, $P, Q \in E(F_{3^m}), a \in \mathbf{Z}$ を満たす．

3 ペアリング演算ライブラリの要件

標数 3 有限体における η_T ペアリングを計算するためには，標数 3 の有限体およびその拡大体，楕円曲線上の演算が必要である．

また，ペアリングを用いた Web アプリケーションの開発を行う際にはビット列から楕円曲線上の点を求める MapToPoint の関数が必須となるが，[3][4] のライブラリではこの関数が用意されていない．この場合，開発者自身が用意する必要があるが，そのためにはライブラリの中での有限体の表現方法や楕円曲線の構成・演算について理解しなければならず，ペアリングに対する知識が乏しい開発者にとって大きな負担となる．このことが，ペアリングを用いた Web アプリケーションの開発・普及を妨げる一因となっており，Web アプリケーション開発者の視点からは，MapToPoint もペアリング演算ライブラリ中で関数として用意すべきである．

表 1 [1] で実装を行った演算関数

	F_{3^m}	$F_{3^{3m}}$	$F_{3^{6m}}$
加算	○	○	○
減算	○	○	○
乗算	○	○	○
除算	○	○	
逆元	○	○	
2 乗算		○	
3 乗算	○	○	○
平方根	○	○	
立方根	○		○

	楕円加算	楕円減算	スカラー倍算
$E(F_{3^m})$	○	○	○

○：実装した関数
／：η_T ペアリングの演算には必要ない関数

さらに，Web アプリケーションの開発に用いることを考えた場合，既存のライブラリと同等の演算速度が必要である．

以上より，ペアリング演算ライブラリの要件を次のように定義する．

【要件 1】 有限体，拡大体，楕円曲線上の演算およびペアリング演算が可能であること

【要件 2】 ビット列から楕円曲線上の点へ写像する MapToPoint がライブラリの関数として用意されていること

【要件 3】 既に公開されているペアリング演算ライブラリと比較して，同程度の実行時間で演算ができること

4 提案するペアリング演算ライブラリ

4.1 有限体，拡大体，楕円曲線上の演算

[1] で開発したライブラリでは，表 1 に示す標数 3 の有限体，拡大体，楕円曲線上の演算，およびそれらを用いた η_T ペアリング演算が可能である．これより，[1] のライブラリは要件 1 を満たしていると言える．

4.2 MapToPoint

MapToPoint はビット列から楕円曲線上の点を計算する演算である．3 で述べたように，ペアリングを用いた Web アプリケーションでは必須であるにもかかわらず，[3][4] のライブラリでは提供されていない．本研究では，標数 3 の超特異楕円曲線上における MapToPoint 写像 [2] をもとに実装を行った．以下に，ビット列から楕円曲線上の点を求める手順を示す．

1. [変数 $i \parallel$ ビット列] からハッシュ値を計算する．ただし，i の初期値は 0 とする
2. ハッシュ値を F_{3^m} の元へ変換する．ハッシュ値を 3 進数に変換し，元の各項の係数とする
3. Algorithm 1 に従い F_{3^m} の元から楕円曲線上の点を計算する
4. 得られた値が楕円曲線上の点で無い場合，i の値を 1 増やして 1. に戻る
5. 楕円曲線上の点が得られるか i の値が閾値を超えた場合，処理を終了する

η_T pairing arithmetic library in characteristic three by ActionScript
† Koichi Hongo and Masami Mohri · Gifu University
‡ Yoshiaki Shiraishi · Nagoya Institute of Technology

表 2　実行環境

	提案ライブラリ	jPBC Library[4]
CPU	Intel(R) Core(TM)2 Duo CPU E4600 2.40GHz 1.20GHz	
メモリ	2.00 GB	
OS	Windows Vista Business SP2	
Web ブラウザ	FireFox3.6	
仮想マシン	Adobe Flash Player 10.1	Java(TM) 6.22

表 3　ペアリング演算の実行時間 (CPU 時間)

	提案ライブラリ	jPBC Library[4]
ペアリングの種類	η_T ペアリング	Tate ペアリング
有限体 F_q の要素数 q	3^{97}	512 ビットの素数
拡大次数 k	6	2
MOV security	922	1024
演算の実行時間 [msec]	524	157

Algorithm 1　MapToPoint in $E(F_{3^m})$[2]
$(E : y^2 = x^3 - x + b, \{b = \pm 1\})$

Input:$y \in F_{3^m}$
Output:$P = (x, y) \in E(F_{3^m})$
1. $s \leftarrow t \leftarrow c \leftarrow y^2 - b$
2. $r \leftarrow m \bmod 3$
3. for i from 1 to $(m - r)/3$
4. $\quad t \leftarrow t^{3^3}$
5. $\quad s \leftarrow s + t$
6. end for
7. $s \leftarrow s^3 - s$
8. if $r = 1$ then
9. $\quad s \leftarrow c - s$
10. end if
11. return (s, y)

MapToPoint の関数を実装したことにより，要件 2 を満たしたと言える．

5　開発したペアリング演算ライブラリ [1] の評価

本節では，開発したライブラリの有用性を既存のライブラリとの比較によって示す．

5.1　比較における条件

既に公開されているペアリング演算ライブラリとして PBC Library[3] や jPBC Library[4] が挙げられる．しかし，PBC Library[3] は C 言語で開発が行われたため Web アプリケーションの開発に適しておらず，jPBC Library[4] は java 言語で開発されているものの標数 3 における η_T ペアリング演算が提供されていない．そのため，以下のような条件を設けて開発したライブラリの安全性・実行速度を評価する．

【条件 1】　開発したライブラリは Flash アプリケーション，jPBC Library[4] は java applet として，それぞれ Web ブラウザ上で動作させる

【条件 2】　本研究にて開発した標数 3 における η_T ペアリングと同程度の安全性を持つペアリングを jPBC Library で提供されている演算から選ぶ

なお，条件 2 に挙げた“標数 3 における η_T ペアリングと同程度の安全性”の基準は，MOV security に基づくものとする．

上記の条件および表 2 に示す実行環境の下で，開発したライブラリと既存のライブラリとの速度比較を行う．

5.2　比較結果に基づくライブラリの評価

表 3 にそれぞれのライブラリを用いて Web ブラウザ上で 1 回のペアリング演算を行うために要した実行時間 [msec] を示す．

表 3 中の各欄に示したとおり，厳密には有限体やペアリングの種類は異なるものの，同程度の MOV security のペアリング演算（条件 2）に対する Web ブラウザ上での実行時間は，ともに数百ミリ秒であることがわかる．これより，本研究で開発したライブラリは，実行速度に関して既存ライブラリと同程度であると評価できる．

表 4　BF 方式の ID ベース暗号 [5] における各処理の実行時間 (CPU 時間)[msec]

F_{3^m}	ユーザ鍵の生成	暗号化	復号
$m = 79$	27	7486	1282
$m = 97$	54	4219	542
$m = 193$	63	14424	2934

6　適用例：ID ベース暗号（BF 方式）[5]

6.1　暗号化／復号の手順

ID ベース暗号は，以下の手順で暗号化／復号を行う．

【鍵設定】

鍵生成機関 (PKG) は，以下のパラメータを設定する．

- ペアリング演算が可能な楕円曲線 E
- PKG のマスター秘密鍵 $s \in \mathbf{Z}_n$
- MapToPoint の関数
- ハッシュ関数 $h(x)$ $(x \in e(P, Q)$, $h(x) \in \mathbf{Z}_n)$
- E 上の点 P と sP

【ユーザ鍵生成】

PKG は MapToPoint を用いてユーザ A の ID から E 上の点 $P_A \in E(F_q)$ を計算する．そして sP_A を計算し，これをユーザ A の秘密鍵 S_A として，ユーザ A へ誰にも知られないように渡す．ユーザ B に対しても同様に，秘密鍵 S_B を渡す．

【暗号化】

送信者 A は，以下の手順で暗号化を行う．

手順 1.　送信者 A は乱数 $x \in \mathbf{Z}_n$ を生成する
手順 2.　送信者 A は受信者 B の ID から P_B を計算する
手順 3.　送信者 A は平文 $m \in \mathbf{Z}_n$ から暗号文 $C = (C_1, C_2) = (xP, m \cdot h(e(P_B, xsP)))$ を計算する
手順 4.　送信者 A は暗号文 C を受信者 B へ送る

【復号】

受信者 B は秘密鍵 S_B を用いて $m = C_2 / h(e(S_B, C_1))$ を計算し，平文 m を得る．

6.2　実行速度

BF 方式の ID ベース暗号 [5] を本研究にて開発したライブラリを用いて実装し，表 2 に示した環境で実行した．その実行速度を表 4 に示す．

標数 3 有限体での η_T ペアリングは，$m = 97$ のとき 1024bit の RSA 暗号と同程度の安全性を持つとされているため，これを判断基準とする．得られた実行速度から，Web ブラウザ上において 5 秒以内に暗号化／復号を実行可能であることがわかる．

7　結論

本稿では，文献 [1] で与えた標数 3 における η_T ペアリング演算ライブラリの有用性を評価した．提案ライブラリと既存のライブラリ [4] の実行時間を比較したところ，Web ブラウザ上での演算において同程度に有用な演算速度であるという結果が得られた．また，ペアリングを用いた具体的な暗号システムへの適用例として BF 方式の ID ベース暗号 [5] の実装を行い，ライブラリが実用に耐えうることを示した．以上より，提案ライブラリはペアリングを利用したセキュアな Web アプリケーション開発に有用であると評価できる．

参考文献

[1] 本郷考一, 毛利公美, 白石善明,“Web クライアント上で暗号化を行うための ActionScript によるペアリング演算ライブラリの開発,” 情報処理学会第 72 回全国大会, 3ZE-2, 2010.

[2] 川原祐人, 小林鉄太郎, 高橋元, 高木剛,“標数 3 の超特異楕円曲線における高速な MapToPoint 写像,” 2009 年暗号と情報セキュリティシンポジウム予稿集（CD-ROM）, 3C3-3, 2009.

[3] Ben Lynn, “PBC Library - Pairing-Based Cryptography (online),” http://crypto.stanford.edu/pbc/ (accessed 2011-01-14).

[4] Angelo De Caro, “Java Pairing-Based Cryptography Library (online),” http://gas.dia.unisa.it/projects/jpbc/(accessed2011-01-14).

[5] Dan Boneh,Matthew Franklin,“Identity-Based Encryption from the Weil Pairing,” CRYPTO 2001, LNCS, vol.2139, pp.213-229, Springer-Verlag, 2001.