A Simple Method For Hashing To Elliptic Curves

Amir Mehdi Yazdani Kashani* University of Kashan
Hassan Daghigh University of Kashan

Abstract

In cryptography, it has been an important problem to transform a random value in $\mathbb{F}_q$ into a random point on an elliptic curve in a deterministic and efficient method. In this paper we propose a simpler form of Shallue-Woestijne-Ulas algorithm in order to hash an element of finite field to a point of an elliptic curves. This subject can be used in cryptosystems based on elliptic curves.

Keywords: Elliptic curves, Quadratic residue, Hash
Mathematics Subject Classification [2010]: 14H52, 11T71

1 Introduction

For a number of elliptic curve cryptosystems it is necessary to hash into an elliptic curve. For instance Boneh-franklin identity based scheme [1]. Before 2006 the usual method was to take $x \in \mathbb{F}_q$ and check whether this value corresponds to a valid abscissa of a point on the elliptic curve. If not, try another abscissa until one of them works. One defect of this algorithm is that the number of operation is not constant. namely the number of steps depends on the input x.
The first algorithm for generating elliptic curve points in deterministic polynomial time was published in ANTS 2006 by Shallue and Woestijne [5] .
The algorithm is based on the skalba equality which says that there exist four maps $X_1(t), X_2(t), X_3(t), X_4(t)$ such that

$$g(X_1(t))g(X_2(t))g(X_3(t)) = (X_4(t))^2$$

where $g(x) = x^3+ax+b$. Then in a finite field for a fixed parameter t, there exists $1 \leq j \leq 3$ such that $g(X_j(t))$ is a quadratic residue, which implies that this $(X_j(t), \sqrt{X_j(t)})$ is a point on the elliptic curve $y^2 = g(x)$.
The maps were simplified and generalized to hyperelliptic curves by Ulas in 2007 [4]. We recall these maps in the following result.

Lemma 1.1. *Let*

$$X_1(t,u) = u \qquad X_2(t,u) = \frac{-b}{a}(1 + \frac{1}{t^4g(u)^2 + t^2g(u)})$$
$$X_3(t,u) = t^2g(u)X_2(t,u) \qquad U(t,u) = t^3g(u)^2g(X_2(t,u))$$

*Speaker