On the Construction of the Asymmetric Chudnovsky Multiplication Algorithm in Finite Fields Without Derivated Evaluation

S. Ballet a, N. Baudru a, A. Bonnecaze a, M. Tukumuli a

aAix Marseille Univ, CNRS, Centrale Marseille, I2M, Marseille, France
case 907, 163 avenue de Luminy, F-13288 Marseille Cedex 9 France
bAix-Marseille Université, Laboratoire d’Informatique Fondamentale
case 901, 163 avenue de Luminy, F-13288 Marseille Cedex 9 France

Received *****; accepted after revision +++++

Presented by £££££

Abstract

The Chudnovsky algorithm for the multiplication in extensions of finite fields provides a bilinear complexity uniformly linear with respect to the degree of the extension. Recently, Randriambololona has generalized the method, allowing asymmetry in the interpolation procedure and leading to new upper bounds on the bilinear complexity. In this note, we describe the construction of this asymmetric method without derivated evaluation. To do this, we translate this generalization into the language of algebraic function fields and we give a strategy of construction and implementation.

To cite this article: A. Name1, A. Name2, C. R. Acad. Sci. Paris, Ser. I 340 (2005).

Résumé

Construction effective de l’algorithme asymétrique de multiplication de Chudnovsky dans les corps finis. L’algorithme de multiplication dans les corps finis de Chudnovsky a une complexité bilinéaire uniformément linéaire en le degré de l’extension. Randriambololona a récemment généralisé cette méthode en introduisant l’asymétrie dans la procédure d’interpolation et en obtenant ainsi de nouvelles bornes sur la complexité bilinéaire. Dans cette note, nous décrivons la construction de cette méthode asymétrique sans évaluation dérivée. Pour ce faire, nous traduisons cette généralisation dans le langage des corps de fonctions algébriques, et nous donnons une stratégie de construction et d’implantation.

1. Introduction

Let $q$ be a prime power, $F_q$ the finite field with $q$ elements and $F_q^n$ the degree $n$ extension of $F_q$. Among all algorithms of multiplications in $F_q^n$, those based on Chudnovsky-Chudnovsky [6] method are known to provide the lowest bilinear complexity. This method is based on interpolation on algebraic curves defined over a finite field and provides a bilinear complexity which is linear in $n$. The original algorithm uses only points of degree 1, with multiplicity 1. Ballet and Rolland [4,5] and Arnaud [1] improved the algorithm introducing interpolation at points of higher degree or higher multiplicity. The symmetry of the original construction involves 2-torsion points that represent an obstacle to the improvement of upper bilinear complexity bounds. To eliminate this difficulty, Randriambololona [8] allowed asymmetry in the interpolation procedure, and then Pieltant and Randriambololona [7] derived new bounds, uniform in $q$, of the bilinear complexity. Unlike symmetric constructions, no effective implementation of this asymmetric construction has been done yet. When $q = 1$, it is known [2] that an asymmetric algorithm can always be symmetrized. However, for greater values of $q$, it may not be the case. Thus, it is of interest to know an effective construction of this asymmetric algorithm. So far, no effective implementation has been proposed for such an algorithm.

1.1. Multiplication algorithm and tensor rank

The multiplication of two elements of $F_q^n$ is an $F_q$-bilinear application from $F_q^n \times F_q^n$ onto $F_q^n$. Then it can be considered as an $F_q$-linear application from the tensor product $F_q^n \otimes F_q^n \otimes F_q^n$ onto $F_q^n$. Consequently, it can also be considered as an element $T_m$ of $F_q^n \otimes F_q^n \otimes F_q^n$ where $\star$ denotes the dual. When $T_m$ is written

$$T_m = \sum_{i=1}^{r} x_i^* \otimes y_i^* \otimes c_i,$$

(1)

where the $r$ elements $x_i^*$ as well as the $r$ elements $y_i^*$ are in the dual $F_q^n$ of $F_q^n$ while the $r$ elements $c_i$ are in $F_q$, the following holds for any $x, y \in F_q^n$: $x \cdot y = \sum_{i=1}^{r} x_i^*(x)y_i^*(y)c_i$. The decomposition (1) is not unique.

**Définition 1.1** Every expression $x \cdot y = \sum_{i=1}^{r} x_i^*(x)y_i^*(y)c_i$ defines a bilinear multiplication algorithm $U$ of bilinear complexity $\mu(U) = r$. Such an algorithm is said symmetric if $x_i = y_i$ for all $i$.

**Définition 1.2** The minimal number of summands in a decomposition of the tensor $T_m$ of the multiplication is called the bilinear complexity (resp. symmetric bilinear complexity) of the multiplication and is denoted by $\mu_q(n)$ (resp. $\mu_q^{sym}(n)$):

$$\mu_q(n) = \min_U \mu(U)$$

where $U$ is running over all bilinear multiplication algorithms (resp. all bilinear symmetric multiplication algorithms) in $F_q^n$ over $F_q$.

1.2. Organisation of the note

In Section 2, we give an explicit translation of the generalization of the Chudnovsky algorithm given by Randriambololona [8, Theorem 3.5]. Then in Section 3, by defining a new design of this algorithm, we give a strategy of construction and implementation. In particular, thanks to a suitable representation of the Riemann-Roch spaces, we present the first construction of asymmetric effective algorithms of multiplication in finite fields. These algorithms are tailored to hardware implementation and they allow computations to be parallelized while maintaining a low number of bilinear multiplications. In Section 4, we give an analysis of the not asymptotical complexity of this algorithm.
2. Multiplication algorithms of type Chudnovsky: Generalization of Randriambololona

In this section we present a generalization of Chudnovsky type algorithms, introduced in [8, Theorem 3.5] by Randriambololona, which is possibly asymmetric. Since our aim is to describe explicitly the effective construction of this asymmetric algorithm, we transform the representation of this algorithm, initially made in the abstract geometrical language, in the more explicit language of algebraic function fields.

Let $F/F_q$ be an algebraic function field over the finite field $F_q$ of genus $g(F)$. We denote by $N_q(F/F_q)$ the number of places of degree one of $F$ over $F_q$. If $D$ is a divisor, $\mathcal{L}(D)$ denotes the Riemann-Roch space associated to $D$. We denote by $\mathcal{O}_Q$ the valuation ring of the place $Q$ and by $F_Q$ its residue class field $\mathcal{O}_Q/Q$ which is isomorphic to $\mathbb{F}_{q^{\deg Q}}$ where $\deg Q$ is the degree of the place $Q$.

In the framework of algebraic function fields, the result [8, Theorem 3.5] of Randriambololona can be stated as in Theorem 2.1. Note that we do not take into account derivated evaluations, since we are not interested in asymptotic results. It means that we describe this asymmetric algorithm with the divisor $G = P_1 + \cdots + P_N$ where the $P_i$ are pairwise distinct closed points of degree $\deg P_i = d_i$.

Let us define the following Hadamard product in $\mathbb{F}_{q^{\deg Q}}$, by $G \times F/Q = F$, we denote by $\mathcal{L}(D)$ the Riemann-Roch space $\mathcal{O}_Q/D$, where $\deg Q$ is the degree of the place $Q$.

Let $\mathcal{O}_Q$ be an algebraic function field over the finite field $\mathbb{F}_q$ of genus $g$. Let $\mathcal{O}_Q$ be an algebraic function field over the finite field $\mathbb{F}_q$ of genus $g$. Suppose there exists a place $Q$ of degree $n$. Let $\mathcal{O}_Q$ be an algebraic function field over the finite field $\mathbb{F}_q$ of genus $g$. Suppose there exist two effective divisors $D_1, D_2$ of $F/F_q$ such that:

(i) The place $Q$ and the places of $\mathcal{O}_Q$ are not in the support of the divisors $D_1$ and $D_2$.

(ii) The natural evaluation maps $E_i$ for $i = 1, 2$ defined as follows are surjective

$$E_i : \begin{cases} \mathcal{L}(D_i) \longrightarrow \mathbb{F}_{q^n} \\ f \longmapsto f(Q) \end{cases}$$

(iii) The natural evaluation map defined as follows is injective

$$T : \begin{cases} \mathcal{L}(D_1 + D_2) \longrightarrow \mathbb{F}_{q^{\deg P_1}} \times \mathbb{F}_{q^{\deg P_2}} \times \cdots \times \mathbb{F}_{q^{\deg P_N}} \\ f \longmapsto (f(P_1), f(P_2), \ldots, f(P_N)) \end{cases}$$

Then for any two elements $x, y$ in $\mathbb{F}_{q^n}$, we have:

$$xy = E_Q \circ T^{-1} \circ T \circ E_2^{-1}(y),$$

where $\mathcal{O}_Q$ denotes the canonical projection from the valuation ring $\mathcal{O}_Q$ of the place $Q$ in its residue class field $\mathbb{F}_Q$, $\circ$ the standard composition map, $T^{-1}$ the restriction of the inverse map of $T$ on the image of $T$, $E_i^{-1}$ the inverse map of the restriction of the map $E_i$ on the quotient group $\mathcal{L}(D_i)/\ker E_i$ and $\odot$ the Hadamard product in $\mathbb{F}_{q^{\deg P_1}} \times \mathbb{F}_{q^{\deg P_2}} \times \cdots \times \mathbb{F}_{q^{\deg P_N}}$; and $\mu_q(n) \leq \sum_{i=1}^N \mu_q(\deg P_i)$.

3. Effective algorithm

3.1. Method and strategy of implementation

The construction of the algorithm is based on the choice of the place $Q$, the effective divisors $D_1$ and $D_2$, the bases of spaces $\mathcal{L}(D_1)$, $\mathcal{L}(D_2)$ and $\mathcal{L}(D_1 + D_2)$ and the basis of the residue class field $\mathbb{F}_Q$. 

In practice, following the ideas of [3], divisors $D_1$ and $D_2$ are chosen as places of degree $n + g - 1$. Furthermore, we require additional properties described below.

3.2. Finding good places $D_1$, $D_2$ and $Q$

In order to obtain the good places, we proceed as follows:
- We draw at random an irreducible polynomial $Q(x)$ of degree $n$ in $\mathbb{F}_q[X]$ and check that this polynomial is primitive and totally decomposed in the algebraic function field $F/\mathbb{F}_q$.
- We choose a place $Q$ of degree $n$ among the places of $F/\mathbb{F}_q$ lying above the polynomial $Q(x)$.
- We draw at random a place $D_1$ of degree $n + g - 1$ and check that $D_1 - Q$ is a non-special divisor of degree $g - 1$ i.e. $\dim L(D_1 - Q) = 0$.
- We draw at random a place $D_2$ of degree $n + g - 1$ and check that $D_2 - Q$ is a non-special divisor of degree $g - 1$ i.e. $\dim(D_2 - Q) = 0$.

3.3. Choosing good bases of the spaces

The residue field $F_Q$.
We choose the canonical basis $B_Q$ generated by a root $\alpha$ of the polynomial $Q(x)$, namely $B_Q = (1, \alpha, \alpha^2, \ldots, \alpha^{n-1})$. From now on we identify $\mathbb{F}_q^n$ to $F_Q$, as the residue class field $F_Q$ of the place $Q$ is isomorphic to the finite field $\mathbb{F}_q^n$.

The Riemann-Roch spaces $L(D_1)$ and $L(D_2)$.
We choose as basis of $L(D)$ the reciprocal image $B_D$ of the basis $B_Q = (\phi_1, \ldots, \phi_n)$ of $F_Q$ by the evaluation map $E$, namely $B_D = (E^{-1}_i(\phi_1), \ldots, E^{-1}_i(\phi_n))$.
Let us denote $B_D = (f_{i,1}, \ldots, f_{i,n})$ with $f_{i,1} = 1$ for $i = 1, 2$.

The Riemann-Roch space $L(D_1 + D_2)$. Note that since $D_1$ and $D_2$ are effective divisors, we have $L(D_1) \subset L(D_1 + D_2)$ and $L(D_2) \subset L(D_1 + D_2)$.

Lemma 3.1 Let $D_1$ and $D_2$ be two effective divisors with disjoint supports. Then $L(D_1) \cap L(D_2) = F_q$.

Proposition 3.1 Let $D_1$ and $D_2$ be two effective divisors with disjoint supports. Then $L(D_1) \cap L(D_2) = F_q$.

where $L_r(D_2)$ is such that $L(D_2) = F_q \oplus L_r(D_2)$ and $\oplus$ denotes the direct sum. In particular, if $g = 0$, then $\mathcal{M} = \ker \Lambda$ is equal to $\{0\}$.
We use $L_r(D_2)$ as basis of $L(D_1 + D_2)$ the basis $B_{D_1+D_2}$ defined by $B_{D_1+D_2} = (f_1, \ldots, f_n, f_{n+1}, \ldots, f_{2n+g-1})$ where $B_{D_1} = (f_1, \ldots, f_n)$ is the basis of $L(D_1)$, $(f_{n+1}, \ldots, f_{2n-1})$ is a basis of $L_r(D_2)$ such that $f_{n+j} = f_{j+1}$ $\in B_{D_2}$ with $B_{D_1}$ and $B_{D_2}$ defined in Section 3.3 and $B_{\mathcal{M}} = (f_{2n}, \ldots, f_{2n+g-1})$ is a basis of $\mathcal{M}$.

3.4. Product of two elements in $F_q^n$

Let $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$ be two elements of $F_q^n$ given by their components over $F_q$ relative to the chosen basis $B_Q$. According to the previous notation, we can consider that $x$ and $y$ are identified to respectively $f_x = \sum_{i=1}^n x_i f_{i,1} \in L(D_1)$ and $f_y = \sum_{i=1}^n y_i f_{i,2} \in L(D_2)$.

The product $f_x f_y$ of the two elements $f_x$ and $f_y$ is their product in the valuation ring $O_Q$. This product lies in $L(D_1 + D_2)$ since $D_1$ and $D_2$ are effective divisors. We consider that $x$ and $y$ are respectively the elements $f_x$ and $f_y$ embedded in the Riemann-Roch space $L(D_1 + D_2)$, via respectively the embeddings $I_i$:
$\mathcal{L}(D_1) \rightarrow \mathcal{L}(D_1 + D_2)$ defined by $I_1(f_x)$ and $I_2(f_y)$ as follows. If, $f_x$ and $f_y$ have respectively coordinates $f_x$ and $f_y$ in $\mathcal{B}_{D_1 + D_2}$ where $i \in \{1, \ldots, 2n + g - 1\}$, we have: $I_1(f_x) = (f_{x_1} := x_1, \ldots, f_{x_n} := x_n, 0, \ldots, 0)$ and $I_2(f_y) = (f_{y_1} := y_1, 0, \ldots, 0, f_{y_{n+1}} := y_2, \ldots, f_{y_{2n+1}} := y_n, 0, \ldots, 0)$. Now it is clear that knowing $x$ (resp. $y$) or $f_x$ (resp. $f_y$) by their coordinates is the same thing.

**Theorem 3.2** Let $P_{\mathcal{M}^s}$ be the projection of $\mathcal{L}(D_1 + D_2)$ onto $\mathcal{M}^s = \mathcal{L}(D_1) \oplus \mathcal{L}(D_2)$ and let $\Lambda$ be the map defined as in Proposition (3.1). Then, for any elements $x, y \in \mathbb{F}_{q^n}$, the product of $x$ by $y$ is such that

$$xy = \Lambda \circ P_{\mathcal{M}^s} \left( T^{-1}_{\text{Im } T} (T \circ I_1 \circ E_1^{-1}(x)) \odot T \circ I_2 \circ E_2^{-1}(y) \right),$$

where $\circ$ denotes the standard composition map, $T^{-1} \mid_{\text{Im } T}$ the restriction of the inverse map of $T$ on the image of $T$, and $\odot$ the Hadamard product as in Theorem 2.1.

We can now present the setup algorithm (Algorithm 1), which is done only once.

**Algorithm 1** Setup algorithm

**INPUT:** $F/\mathbb{F}_q$, $Q, D_1, D_2, P_1, \ldots, P_N$.

**OUTPUT:** $T$ and $T^{-1}$.

(i) The representation of the finite field $\mathbb{F}_q = \langle \alpha \rangle$, where $\alpha$ is a fixed primitive element.

(ii) The function field $F/\mathbb{F}_q$, the place $Q$, the divisors $D_1$ and $D_2$ and the points $P_1, \ldots, P_N$ are such that Conditions (ii) and (iii) in Theorem 2.1 are satisfied. In addition, we require that $\sum_{i \leq N} \deg P_i = 2n + g - 1$.

(iii) Represent $\mathbb{F}_{q^n}$ in the canonical basis $\mathcal{B}_Q = \{1, \alpha, \alpha^2, \ldots, \alpha^{n-1}\}$, where $\mathbb{F}_{q^n} = \langle \alpha \rangle$ with $\alpha$ a primitive element as in Section 3.3.

(iv) Construct a basis $(f_1, \ldots, f_n, f_{n+1}, \ldots, f_{2n+g-1})$ of $\mathcal{L}(D_1 + D_2)$ where $(f_1, \ldots, f_n)$ is the basis of $\mathcal{L}(D_1)$, $(f_1, f_{n+1}, \ldots, f_{2n-1})$ the basis of $\mathcal{L}(D_2)$ and $(f_{2n}, \ldots, f_{2n+g-1})$ the basis of $\mathcal{M}$, defined in Section 3.3.

(v) Compute the matrices $T$ and $T^{-1}$.

(vi) Compute the matrix $\Lambda$.

The multiplication algorithm (Algorithm 2) is presented hereafter.

4. Complexity analysis

In terms of number of multiplications in $\mathbb{F}_q$, the complexity of this multiplication algorithm is as follows: calculation of $z$ and $t$ needs $2(2n^2 + nq - n)$ multiplications, calculation of $u$ needs $(2n + 2g - 2 + r) \sum_{1 \leq i \leq r} \mu_n(i)$ bilinear multiplications and calculation of $2n - 1$ first components of $w$ needs $(2n + g - 1)(2n - 1)$ multiplications (remark that in Algorithm 2, we just have to compute the $2n - 1$ first components of $w$). The calculation of $xy$ needs $n + g$ multiplications. The total complexity in terms of multiplications is bounded by $8n^2 + n(4g - 5) + (2n + 2g - 2 + r) \sum_{1 \leq i \leq r} \mu_n(i)$.

The general construction of the set-up algorithm involves some random choice of divisors having prescribed properties over an exponentially large set of divisors. To get a polynomially constructible algorithm with linear complexity, one needs to construct explicitly (i.e. polynomially) points of corresponding degrees $n$ on curves of arbitrary genus with many rational points. Unfortunately, so far it is unknown how to produce such points (cf. [9, Section 4, Remark 5] and [8, Remark 6.6]). Hence, the asymptotic complexity of such a construction is an open problem.
Algorithm 2 Multiplication algorithm

INPUT: $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$. OUTPUT: $xy$.

(i) Compute

$$
\begin{pmatrix}
  z_{1,d_1} \\
  \vdots \\
  z_{n,d_n} \\
  z_{n+1,d_{n+1}} \\
  \vdots \\
  z_{N,d_N}
\end{pmatrix}
= T^{n+1}
\begin{pmatrix}
  z_1 \\
  \vdots \\
  z_n \\
  z_{n+1} \\
  \vdots \\
  z_{2n+g-1}
\end{pmatrix}
$$

and

$$
\begin{pmatrix}
  t_{1,d_1} \\
  \vdots \\
  t_{n,d_n} \\
  t_{n+1,d_{n+1}} \\
  \vdots \\
  t_1
\end{pmatrix}
= T^{n+1}
\begin{pmatrix}
  t_1 \\
  \vdots \\
  t_n \\
  t_{n+1} \\
  \vdots \\
  y_1
\end{pmatrix},
$$

where $\sum_{i=1}^{N} d_i = 2n + g - 1$, $(z_{i,j}, t_{i,j}) \in (\mathbb{F}_{q^{d_j}})^2$, $(z_i, t_i) \in (\mathbb{F}_q)^2$ and $\mathbf{0}$ stands for the null vector.

(ii) Compute the Hadamard product $u = (u_{1,d_1}, \ldots, u_{N,d_N}) = (u_1, \ldots, u_{2n+g-1})$, where $u_i, d_i = z_i, t_i, d_i$, in $\mathbb{F}_{q^1} \times \mathbb{F}_{q^2} \times \cdots \times \mathbb{F}_{q^N}$ as in Theorem 2.1.

(iii) Compute $w = (w_1, \ldots, w_{2n+g-1}) = T^{-1}(u)$.

(iv) Extract $w' = (w_1, \ldots, w_{2n-1})$ (in step (iii), just the $2n - 1$ first components have to be computed).

(v) Return $xy = \Lambda(w')$.

References


