NOTE

On an Involution Concerning Pairs of Polynomials over $\mathbb{F}_2$

Astrid Reifegerste

Institut für Algebra und Geometrie, Otto-von-Guericke-Universität Magdeburg,
Postfach 4120, 39016 Magdeburg, Germany
E-mail: astrid.reifegerste@mathematik.uni-magdeburg.de

Communicated by the Managing Editors
Received July 9, 1999

From the work of S. Corteel et al. (1998, J. Combin. Theory Ser. A 82, 186-192),
the number of coprime $m$-tuples of monic polynomials of degree $n$ over $\mathbb{F}_2$ is equal
to $q^{mn} - q^{(n-1)m + 1}$. In particular, among the ordered pairs of polynomials of degree
$n$ over $\mathbb{F}_2$ there are as many relatively prime as non-relatively prime ones. We give
an involution that proves this result. © 2000 Academic Press

Key Words: pairs of polynomials over $\mathbb{F}_2$.

1. INTRODUCTION

The subject of [1] is the enumeration of all coprime $m$-tuples of elements
of order $n$ in a prefab, i.e., of such tuples whose elements have no prime
factor in common. To this end, a general form of the pentagonal number
sieve was proved which implies, among other things, the following result:
the number of coprime $m$-tuples of monic polynomials of degree $n$ over $\mathbb{F}_q$
is $q^{mn} - q^{(n-1)m + 1}$. Considering the special case $q = m = 2$ yields the nice
consequence that the sets of ordered pairs of coprime polynomials of
degree $n$ over $\mathbb{F}_2$ and of non-coprime ones, respectively, are equinumerous.
In [1], the authors asked for a combinatorial proof of this fact. Here we
present such a proof.

Our bijection is based on elementary properties of the resultant of two
polynomials over $\mathbb{F}_2$.
Let \( f(x) = \sum_{i=0}^{m} a_i x^i \) and \( g(x) = \sum_{i=0}^{n} b_i x^i \) be polynomials of formal degree \( m \) resp. \( n \) over a field \( K \). The determinant

\[
\text{res}(f, g) = \begin{vmatrix}
  a_m & a_{m-1} & \cdots & a_0 & 0 & \cdots & 0 \\
  0 & a_m & a_{m-1} & \cdots & a_0 & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
  0 & \cdots & 0 & a_m & a_{m-1} & \cdots & a_0 \\
  b_n & b_{n-1} & \cdots & b_0 & 0 & \cdots & 0 \\
  0 & b_n & b_{n-1} & \cdots & b_0 & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
  0 & \cdots & 0 & b_n & b_{n-1} & \cdots & b_0 \\
\end{vmatrix}_{m+n}
\]

is called the resultant of \( f \) and \( g \). If \( a_m \neq 0 \) and \( f(x) = a_m(x - \alpha_1) \cdots (x - \alpha_m) \) in the splitting field of \( f \) over \( K \), then \( \text{res}(f, g) \) is also given by the formula

\[
\text{res}(f, g) = a_m^n \prod_{i=1}^{m} g(\alpha_i).
\]

We have \( \text{res}(f, g) = 0 \) if and only if \( f \) and \( g \) have a common divisor of positive degree in \( K[x] \). Define \( \text{res}(1, 0) := 1 \) for \( \text{char } K \neq 0 \).

**Lemma.** Let \( f, g \) be polynomials of degree \( n \geq 1 \) over \( \mathbb{F}_2 \).

(a) Then \( \text{res}(f, g) = \text{res}(f, f + g) \).

(b) If \( f = sq + r \) for some \( s, q, r \in \mathbb{F}_2[x] \) then \( \text{res}(f, s) = \text{res}(s, r) \).

**Proof.** (a) This is obvious. (b) Let \( s(x) = (x - \alpha_1) \cdots (x - \alpha_k) \) in the splitting field of \( s \) over \( \mathbb{F}_2 \). Then \( \text{res}(s, r) = \prod_{i=1}^{k} r(\alpha_i) = \prod_{i=1}^{k} (f(\alpha_i) + s(\alpha_i) q(\alpha_i)) = \text{res}(s, f) = \text{res}(f, s) \). \( \square \)

2. THE BIJECTION

In this section we describe the recursive construction of a one-to-one mapping \( \varphi_n \) from the set \( M_n \) of ordered pairs of polynomials of degree \( n \) over \( \mathbb{F}_2 \) onto itself which satisfies \( \text{res}(\varphi_n(f, g)) = \text{res}(f, g) + 1 \) for all \( (f, g) \in M_n \).

For \( i = 1, 2 \) we denote by \( (f, g)^{(i)} \) the \( i \)th component of the pair \( (f, g) \).

Set \( \text{deg } 0 = 0 \).

**Input:** \( (f, g) \in M_n \)

**Step 1.** Determine \( s = f + g; \ k := \text{deg } s. \)

**Step 2.** If \( k > 0 \) then go to step 3, otherwise put \( f' := f, \ g' := g + 1 \) and terminate.
Step 3. Determine $r = f + sx^n - k$; $m := \deg r$. 

* $m \leq n - 1$ *

Step 4. Set

$$(s', r') := \begin{cases} 
\phi_4(s, r) & \text{if } k = m \\
(\phi_m(s + r, r)^{(1)} + \phi_m(s + r, r)^{(2)}, \phi_m(s + r, r)^{(2)}) & \text{if } k < m \\
(\phi_4(s, s + r)^{(1)}, \phi_4(s, s + r)^{(1)} + \phi_4(s, s + r)^{(2)}) & \text{if } k > m.
\end{cases}$$

Step 5. Put $f' := s'x^n - k$, $g' := f + s'$. 

Output: $\phi_4(f, g) := (f', g')$

We use the notation above throughout this paper.

**Theorem 1.** For any pair $(f, g) \in M_n$ of polynomials the above algorithm determines a pair $(f', g') \in M_n$ satisfying $\text{res}(f', g') = \text{res}(f, g) + 1$. The number of iterations is at most $n$.

**Proof.** The finiteness of the algorithm immediately follows from its description ($k, m \leq n - 1$).

Clearly, by construction, the theorem is true for $k = 0$. Assume $k \geq 1$ and so $n \geq 2$.

First we will show by induction on $n$ that $(f', g') \in M_n$.

For $n = 2$ one has $k = 1$ and $m \leq 1$. If $m = 1$ then $(s', r') = \phi_4(s, r)$ and hence $\deg s' = \deg r' = 1$. Otherwise, setting $(\tilde{s}, \tilde{r}) = \phi_4(s, s + r)$, we obtain $\deg \tilde{s} = \deg \tilde{r} = 1$. Thus $s' = \tilde{s}$ is linear and $r' = \tilde{s} + \tilde{r} \in F_2$. In any case, $\deg f' = \deg g' = 2$.

Assume now $n > 2$. Suppose that the degrees of $s$ and $s'$ resp. $r$ and $r'$ are equal, so $s' = s'x^n - k$ and $g' = f + s'$ are obviously of degree $n$. In fact, we have $\deg s = \deg s'$ and $\deg r = \deg r'$ at every stage of the algorithm.

Let $k' = \deg s'$, $m' = \deg r'$; the polynomials and the corresponding degrees will be indexed to indicate the stage of their appearance in the course of the algorithm. We first consider the last stage denoted by $e$. As seen above, $e < \infty$. By construction, one has $k_{e+1} = 0$ and $k_i > 0$ for $i \leq e$.

Therefore we may assume that $k_e \geq m_e$. Otherwise $f_{e+1} = s_e + r_e$, $g_{e+1} = r_e$, and hence $s_e = s_{e+1}$ is of degree $k_e = k_{e+1} = 0$. In case $k_e = m_e$, we obtain $f_{e+1} = s_e$, $g_{e+1} = r_e$ which implies $s'_e = f_{e+1} = s_e$, $r'_e = g_{e+1} = r_e + 1$. For $k_e > m_e$, setting $f_{e+1} = s_e$, $g_{e+1} = s_e + r_e$ gives $s'_e = f_{e+1} = s_e$, $r'_e = f_{e+1} + g_{e+1} = s_e + (s_e + r_e + 1) = r_e + 1$ again. In particular, in both cases $k_e = k'_e$, $m_e = m'_e$ is satisfied. So we may assume the existence of $i < e$ such that $k_{i+1} = k'_i$. Consider the $i$th stage according to the different cases of Step 4 in the algorithm:
(i) \( k_i = m_i \). Since \( k_i, m_i < n \), we are done by inductive assumption.

(ii) \( k_l < m_l \). By induction, we have \( m_l = m'_l \). Compute \( \varphi_{m_l}(f_{i+1}, g_{i+1}) \) where \( f_{i+1} = s_i + r_i \) and \( g_{i+1} = r_i \). Then \( s_{i+1} = f_{i+1} + g_{i+1} = s_i \) and so \( k_i = k_{i+1} = k'_i + 1 \) by assumption. On the other hand, \( s'_i = f'_{i+1} + g'_{i+1} = s'_{i+1} \), so \( k'_i = k'_{i+1} + 1 \). Therefore \( k_i = k'_i \).

(iii) \( k_l > m_l \). Here the inductive assumption yields \( k_i = k'_i \). Compute \( \varphi_{m_l}(f_{i+1}, g_{i+1}) \) where \( f_{i+1} = s_i + r_i \) and \( g_{i+1} = s_i + r_i \). Then \( s_{i+1} = f_{i+1} + g_{i+1} = r_i \) and, by assumption \( k_{i+1} = k'_{i+1} + 1 \). Setting \( r'_i = f'_{i+1} + g'_{i+1} = s'_{i+1} \), we get \( m_l = m'_l \). Consequently \( m_l = m'_l \).

By iteration, we derive the assertion. To prove the remaining statement \( \text{res}(f', g') = \text{res}(f, g) + 1 \), we use induction by \( n \) once more. For \( n = 1 \) it holds trivially. Assume \( n \geq 2 \). By the lemma, we obtain \( \text{res}(f, g) = \text{res}(f, s) = \text{res}(s, r) \). Applying the inductive assumption and the lemma, we deduce depending on the cases in Step 4 that:

(i) \( \text{res}(s', r') = \text{res}(s, r) + 1 \) because \( k = m < n \);

(ii) \( \text{res}(s', r') = \text{res}(s, r) + 1 \) because \( k = m < n \); and \( \text{deg } r = \text{deg } r' = m' \) throughout the algorithm, we can inductively derive that \( \varphi_{m_l}(s, r) = \text{res}(s, r) + 1 \) analogously as (ii).

Therefore, \( \text{res}(f', g') = \text{res}(f', s') = \text{res}(s', r') = \text{res}(f, g) + 1 \).

**Theorem 2.** The mapping \( \varphi_n \) is an involution.

**Proof.** Certainly the assertion holds for \( k = 0 \), i.e., for \( \varphi_n: M_n \to M_n \), \( f, f + c \mapsto f, f + c + 1, c \in \mathbb{Z} \). So assume \( k \geq 1 \). Since \( \text{deg } s = \text{deg } s' = k' \) and \( \text{deg } r = \text{deg } r' = m' \) throughout the algorithm, we can inductively derive that \( \varphi_n \) is an involution for all \( n \).

The mapping \( \varphi_n \) can be described explicitly: \( (x, x^2 + x) \mapsto (x^2 + 1, x^2 + x + 1) \). Let \( (f', g') = \varphi_n(f, g) \) where \( f' = s' x^{n-k} + r', g' = f' + s' \) and \( n > 2 \). For computing \( \varphi_n(f', g') \), we have to treat the three cases of Step 4:

(i) \( k' = m' \). By induction, \( (s', r') = \varphi_n(s', r') = \varphi_n(s, r) = (s, r) \) since \( k = k' < n \) and \( m = m' \).

(ii) \( k' < m' \). One has \( s' = s' + r', r' = s_r \) where \( (s', r') = \varphi_n(s' + r', r') = \varphi_n(m(s + r, r) = (s + r, r) \) by induction for \( m = m' < n \). Note that \( k = k' \).

Thus \( s'' = s, r'' = r \).

(iii) \( k' > m' \). \( s'' = s, r'' = r \), analogously as (ii).

Therefore, \( f'' = s'' x^{n-k} + r'' = s x^{n-k} + r = f, g'' = f'' + s'' = f + s = g \).

**Example.** To illustrate the algorithm, we compute the image of \((f, g)\) for \( f = x^5 + x^4 + x \) and \( g = x^5 + x^3 + x^2 + x + 1 \) with respect to \( \varphi_3 \).
Thus \( \varphi(f, g) = (x^5 + x^4 + x^3 + x^2 + x + 1) \).

Remark. (1) As mentioned before, our bijection is \( k \)- resp. \( m \)-preserving. In particular, it has the nice property that it can be restricted to the set of pairs \((f, g) \# M_n \) with fixed degree of \( f + g \).

(2) Since \( \text{res}(f, g) = \text{res}(s, r) \), the bijection also proves the similar result that

\[
|\{(f, g) \in \mathbb{F}_2[x] : 1 \leq n_1, n_2 \leq n, \gcd(f, g) = 1\}| = |\{(f, g) \in \mathbb{F}_2[x] : 1 \leq n_1, n_2 \leq n, \gcd(f, g) \neq 1\}| \text{ for all } n \in \mathbb{N}, \text{ where } n_1 = \deg f, n_2 = \deg g.
\]

(3) Any polynomial \( h \in \mathbb{F}_2[x] \) of degree \( d \in \{1, \ldots, n-1\} \) is the gcd for exactly \( 2^{n-d} - 1 \) pairs \((f, g) \in M_n \). Moreover, for \( 2^{n-d} \) of these pairs \( f + g = h \) is satisfied (otherwise \( \deg (f + g) > d \)). In this case the mapping \( \varphi_n \) can be described explicitly. We have \( \varphi_n(f, g) = (f + 1, g + 1) \); to prove this, we compute \( \varphi_n(f, g) \) following the algorithm. So we get \( s = h, k = d > 0 \) and \( r = h(f + x^{-k}), m = \deg r \), where \( f = h f \). If \( f + x^{-k} \in \mathbb{F}_2 \) then \( r = 0 \) and \( r = s \), respectively, and in particular \( m \leq k \). In this case we have \( s = s, r = r + 1 \) and hence \( t = s x^{-k} + r = f + 1, g = f' + s = g + 1 \). If \( \deg f = x^{-k} \geq 1 \), and so \( k = m \), we obtain \( s_{new} = s \) and \( r_{new} = h(1 + f + x^{-k} + x^{-k} + h x^{-k}) \) at the next stage. Continue by considering \( k_{new} = k \) and \( m_{new} < m \) until one comes back to the case already discussed.

REFERENCE