SIAM J. COMPUT.
Vol. 20, No. 3, pp. 591–602, June 1991

012

TESTS FOR PERMUTATION POLYNOMIALS*

JOACHIM VON ZUR GATHEN†

Abstract. If $\mathbb{F}_q$ is a finite field and $f \in \mathbb{F}_q[x]$, then f is called a *permutation polynomial* if the mapping $\mathbb{F}_q \to \mathbb{F}_q$ induced by f is bijective. This property can be tested by a probabilistic algorithm whose number of operations is polynomial (in fact, essentially linear) in the input size, i.e., in $\deg f \cdot \log q$. This is extended to "almost permutation polynomials," whose value set consists of almost all elements of $\mathbb{F}_q$.

Key words. permutation polynomials, values of polynomials, finite fields, Euclidean remainder sequence, subresultant, probabilistic algorithm

AMS(MOS) subject classifications. 11T06, 12Y05, 68Q40

1. Introduction.

A univariate polynomial $f \in \mathbb{F}_q[x]$ over a finite field $\mathbb{F}_q$ with q elements (q a power of a prime number) induces a function $\mathbb{F}_q \to \mathbb{F}_q$ via $a \mapsto f(a)$. If this function is bijective, then f is called a *permutation polynomial*. Permutation polynomials have been studied since Hermite [14] and Dickson [9], and recent interest stems from potential applications in public-key cryptography (see Lidl and Mullen [18]); reference to other uses is given in the latter article. A list of all permutation polynomials of degree at most 5 is given in Dickson [10] and Lidl and Niederreiter [19]. We may always assume, without loss of generality, that $\deg f < q$.

Given an arbitrary polynomial $f \in \mathbb{F}_q[x]$ of degree n, one can test whether it is a permutation polynomial simply by producing its list of values (see §2). Another general test goes back to Hermite and Dickson (see § 3). In their survey paper, Lidl and Mullen [18] pose as an open problem:

(P1) Find an algorithm of lower complexity than $O(qn)$ to test whether a given polynomial is a permutation polynomial of $\mathbb{F}_q$.

For such a test, the input size—the number of bits required to represent f—is about $n \log q$. The above-mentioned tests use exponential time, for large q, and no polynomial-time tests are in the literature. Lidl and Mullen [18] quote some criteria in terms of the coefficients of f. We present a probabilistic test whose number of operations in $\mathbb{F}_q$ is essentially $O(n \log q)$, i.e., essentially linear in the input size $n \log q$.

In §2, we briefly consider the "simple" test and find that off-the-shelf techniques from computer algebra already improve the running time slightly, without any new insights into the problem. Hermite's classical test has been one of the most important tools in the study of permutation polynomials, both for theoretical and practical purposes. Section 3 gives a probabilistic variant of this test, reducing the running time from $\Omega(q^2)$ to essentially $O(q)$. In §4, we derive a criterion saying that f is a permutation polynomial if and only if $g_f = 0$, where $g_f \in \mathbb{F}_q[y]$ is a new polynomial

* Received by the editors May 18, 1989; accepted for publication (in revised form) August 31, 1990. Part of this work was done while the author was a Visiting Fellow at the Computer Science Laboratory, Australian National University, Canberra, Australia, and supported by Natural Sciences and Engineering Research Council of Canada grant A-2514. A first version appeared as Tech. Report TR-CS-89-08, Computer Sciences Laboratory, Australian National University, and extended abstracts of partial results appeared in Proc. 30th Annual IEEE Symposium on Foundations of Computer Science, Research Triangle Park, NC, 1989, pp. 88–92, and Proc. International Symposium on Symbolic and Algebraic Computation, Tokyo, Japan, Association for Computing Machinery Press, 1990, pp. 140–144.

† Department of Computer Science, University of Toronto, Toronto, Ontario M5S 1A4, Canada (gathen@theory.toronto.edu).

whose coefficients are polynomials in the coefficients of f. This criterion is equivalent to one given by Raussnitz [23]. The main result of this paper is in §5, where we show how to calculate $g_f(u)$ fast for randomly chosen u in some finite extension field of $\mathbb{F}_q$. The resulting polynomial-time probabilistic test always gives the correct answer if the input is a permutation polynomial. If it is not, it may give the incorrect answer, but with controllably small probability ϵ. The running time is essentially proportional to $\log \epsilon^{-1}$. If we use $\epsilon = q^{-1}$, the running time is $O(n \log q)$, up to factors $\log n$: *softly linear* running time.

Precious few classes of permutation polynomials are known (see Lidl and Mullen [18]), and a random polynomial in $\mathbb{F}_q[x]$ of degree less than q is a permutation polynomial with very small probability $q!/q^q \approx e^{-q}$. (Recall that the polynomials of degree less than q correspond bijectively to the functions $\mathbb{F}_q \to \mathbb{F}_q$.)

To enlarge the pool of candidate polynomials, we generalize the notion of permutation polynomial as follows. Suppose some $\rho \in \mathbb{N}$ is given, and let $V(f) = \#f(\mathbb{F}_q)$ be the size of the image of f. We say that f is *ρ-large* if the image of the mapping f has at least $q - \rho$ elements: $V(f) \geq q - \rho$. Thus f is 0-large if and only if f is a permutation polynomial.

Section 6 gives a criterion for ρ-large polynomials analogous to the criterion in §4 for permutation polynomials, and §7 gives the resulting test. It is a probabilistic algorithm with expected time polynomial in $n\rho \log q$; in fact, the time is softly linear in $n\rho \log q$.

The method presented here suggests the following general question: which (special) problems can one solve in (random) polynomial time for polynomials of exponential degree given by small arithmetic circuits? Section 8 briefly discusses this.

A "naive" test for permutation polynomials is to choose some elements $u \in \mathbb{F}_q$ at random and check whether each has exactly one preimage under f. At first sight, it looks as if this test has little chance of success, e.g., for a polynomial whose values leave out only very few elements of $\mathbb{F}_q$. However, a geometric study of permutation polynomials, initiated by Hayes [13], leads to the essentially equivalent notion of *exceptional polynomials*. This property can also be tested in random polynomial time, and the approach shows that the above naive test has a good chance of success. Its running time is about the square of the time for the algorithm presented here (von zur Gathen [12]). Shparlinskiy [26] presents a deterministic test using essentially $O(n^3 q^{1/2})$ operations.

2. The simple test revisited. Given $f \in \mathbb{F}_q[x]$ of degree n, one can produce its list of values and sort them, to determine whether f is a permutation polynomial. This takes $O(nq)$ arithmetic operations, plus $O(q \log^2 q)$ binary operations for sorting. Alternatively, one can test whether the q values are distinct with $O(q \log^2 q \log\log q)$ arithmetic operations (Baur and Strassen [3]).

Since we know what the q values have to be, we can do better by checking the condition

$$\prod_{v \in \mathbb{F}_q} (x - f(v)) = x^q - x,$$

which is equivalent to f being a permutation polynomial. All $f(v)$ can be computed in $O(q \log^2 n \log\log n)$ arithmetic operations, and the product can be calculated at the same cost (see Borodin and Munro [4]).

3. Hermite's test revisited. Hermite's criterion says that $f \in \mathbb{F}_q[x]$ is a permutation polynomial if and only if

(i) f has exactly one root in $\mathbb{F}_q$,
(ii) $\forall i, 1 \le i \le q-2$, $\deg(f^i \text{ rem } (x^q - x)) \le q-2$
(Lidl and Niederreiter [19, Thm. 7.4]). Here, $(g \text{ rem } h) \in \mathbb{F}_q[x]$ is the remainder of g on division by h: $(g \text{ rem } h) \equiv g \bmod h$ and $\deg(g \text{ rem } h) < \deg h$ (assuming $h \neq 0$). The obvious implementation of Hermite's test requires about q multiplications of f with a polynomial of degree less than q, each followed by a reduction modulo $x^q - x$. Even when $n = \deg f$ is small, say constant, this may require $\Omega(q^2)$ operations in $\mathbb{F}_q$. We now implement this test more efficiently.

With a new indeterminate y, we have

$$\begin{aligned}(f+y)^{q-1} &= \sum_{0 \le i \le q-1} \binom{q-1}{i} y^{q-1-i} f^i \\ &\equiv y^{q-1} + r_{q-1} + \sum_{1 \le i \le q-2} \binom{q-1}{i} y^{q-1-i} r_i \bmod x^q - x,\end{aligned}$$

where $r_i = f^i \text{ rem } x^q - x$ for $1 \le i < q$. Each of these binomial coefficients is nonzero in $\mathbb{F}_q$ (Lucas [20]; Lidl and Niederreiter [19, Exercise 7.1]). Let

$$r = ((f+y)^{q-1} - f^{q-1}) \text{ rem } (x^q - x) \in \mathbb{F}_q[x, y],$$

and $s \in \mathbb{F}_q[y]$ be the coefficient of x^{q-1} in r. Then we have

$$\text{(ii)} \iff s = 0.$$

Since $\deg_y (f+y)^{q-1} = q-1$, we have $\deg_y s \le q-1$. Computing $(f+y)^{q-1} \text{ rem } (x^q - x)$ as a bivariate polynomial would again result in cost $\Omega(q^2)$. However, we can substitute a randomly chosen $u \in \mathbb{F}_{q^m}$ for y, from a suitable extension $\mathbb{F}_{q^m}$ of $\mathbb{F}_q$, and compute

$$r(u) = ((f+u)^{q-1} - f^{q-1}) \text{ rem } (x^q - x) \in \mathbb{F}_{q^m}[x],$$

and $s(u)$ as the coefficient of x^{q-1} in $r(u)$. We return "YES" if $s(u) = 0$, and "NO" otherwise. (We also check condition (i): $\deg \gcd(x^q - x, f) = 1$.)

To estimate the cost, let $M : \mathbb{N} \to \mathbb{R}$ denote a "universal" cost of multiplication, i.e., let it be such that two polynomials of degree at most n over a ring R can be multiplied in $O(M(n))$ arithmetic operations in R, and two n-bit integers can be multiplied with $O(M(n))$ bit operations. We can choose $M(n) = n \log n \log\log n$ (Schönhage and Strassen [24], Cantor and Kaltofen [7]). If $g, h \in \mathbb{F}_q[x]$ are polynomials of degree at most n, then the division with remainder of g by h (if $h \neq 0$) can be performed in $O(M(n))$ operations in $\mathbb{F}_q$.

PROPOSITION 1. *The probabilistic algorithm given above can be implemented with $O(M(q) \log q \cdot M(m))$ arithmetic operations in $\mathbb{F}_q$. Its output is correct with probability at least $1 - q^{1-m}$.*

Proof. The algorithm can be performed in $O(M(q) \log q)$ operations in $\mathbb{F}_{q^m}$, using "repeated squaring." Elements of $\mathbb{F}_{q^m}$ are represented by their coordinates in $(\mathbb{F}_q)^m$, and a single arithmetic operation on such elements can be performed with $M(m)$ operations in $\mathbb{F}_q$. Finally, the gcd condition can be checked with $O(\log q M(n))$ operations in $\mathbb{F}_q$ (Aho, Hopcroft, and Ullman [1, §8.9]). Thus the total cost is $O(M(q) \log q \cdot M(m))$ operations in $\mathbb{F}_q$. (We have neglected the cost of constructing $\mathbb{F}_{q^m}$; see §5.)

Assume that (i) holds. If f is a permutation polynomial, then $s = 0$ and $s(u) = 0$. If f is not a permutation polynomial, then $s \neq 0$ and $s(u) = 0$ occurs with probability at most $\deg s / q^m < q^{1-m}$. □

We can make the error probability arbitrarily small, by choosing m appropriately. The running time of this test is still at least linear in q. However, the idea of random substitutions will lead to a substantial improvement in §5.

In practical applications of Hermite's test, one stops of course as soon as $\deg r_i = q-1$ is found for some $i \leq q-2$. In the same vein, our algorithm would first calculate $f^{q-1} \operatorname{rem} x^q - x$ by repeated squaring, and stop whenever one of the (few) powers of $f(\operatorname{rem} x^q - x)$ calculated is found to have degree $q-1$. One proceeds similarly in the subsequent computation of certain $\left((f+u)^i - f^i\right) \operatorname{rem} x^q - x$ by repeated squaring.

4. A criterion for permutation polynomials. By definition, a polynomial $f \in \mathbb{F}_q[x]$ is a permutation polynomial if and only if for all $u \in \mathbb{F}_q$ there exists a unique $v \in \mathbb{F}_q$ with $f(v) = u$, or, equivalently, $x - v$ divides $f - u$ in $\mathbb{F}_q[x]$. Since $\mathbb{F}_q$ is a finite set, surjectivity is sufficient:

$$\begin{array}{ll} & f \text{ is a permutation polynomial} \\ \iff & \forall u \in \mathbb{F}_q \; \exists v \in \mathbb{F}_q \quad f(v) = u \\ \iff & \forall u \in \mathbb{F}_q \; \exists v \in \mathbb{F}_q \quad x - v \mid f - u. \end{array}$$

For two nonzero polynomials $a = a_m x^m + \cdots + a_0$, $b = b_n x^n + \cdots + b_0 \in \mathbb{F}_q[x]$, we denote by $\gcd(a,b)$ their *monic* gcd. If $a_m b_n \neq 0$, then

$$R(a,b) = \begin{pmatrix} a_m & a_{m-1} & \cdots & a_0 & & & & \\ & a_m & a_{m-1} & \cdots & a_0 & & & \\ & & \ddots & \ddots & & \ddots & & \\ & & & a_m & a_{m-1} & \cdots & \cdots & a_0 \\ b_n & b_{n-1} & \cdots & \cdots & b_0 & & & \\ & b_n & b_{n-1} & \cdots & \cdots & b_0 & & \\ & & \ddots & \ddots & & & \ddots & \\ & & & b_n & b_{n-1} & \cdots & \cdots & b_0 \end{pmatrix} \in \mathbb{F}_q^{(m+n)\times(m+n)}$$

is their Sylvester matrix, consisting of n rows of coefficients of a, and m rows of coefficients of b. Furthermore, $\operatorname{res}(a,b) = \det R(a,b)$ is their resultant. A fundamental fact is that

$$\gcd(a,b) \neq 1 \iff \operatorname{res}(a,b) = 0$$

(see van der Waerden [28]). Since $x^q - x = \prod_{b \in \mathbb{F}_q} (x-b)$, we have

$$\begin{array}{ll} & f \text{ is a permutation polynomial} \\ \iff & \forall u \in \mathbb{F}_q \;\; \gcd(x^q - x, f - u) \neq 1 \\ \iff & \forall u \in \mathbb{F}_q \;\; \operatorname{res}(x^q - x, f - u) = 0 \\ \iff & y^q - y \mid \operatorname{res}(x^q - x, f - y). \end{array}$$

Here, y is a new indeterminate, $h_f = \operatorname{res}(x^q - x, f - y) \in \mathbb{F}_q[y]$, and the divisibility condition is in $\mathbb{F}_q[y]$.

What are the degree and leading coefficient of h_f? The constant term $f(0) - y$ of $f - y \in \mathbb{F}_q[y][x]$ occurs q times on the lower right part of the diagonal of $R(x^q - x, f - y)$, and nowhere else. The cofactor in the Laplace expansion for the determinant of that part of the diagonal is the upper left $n \times n$-submatrix, which is upper triangular with 1's on the diagonal, and has determinant 1. Thus h_f has degree exactly q,

and leading coefficient $(-1)^q$. It follows that the condition "$y^q - y \mid h_f$" is equivalent to "$h_f = (-1)^q(y^q - y)$." We have proved the following criterion for permutation polynomials.

THEOREM 2. *Let $f \in \mathbb{F}_q[x]$, and*

$$g_f = \operatorname{res}(x^q - x, f - y) - (-1)^q(y^q - y) \in \mathbb{F}_q[y].$$

Then f is a permutation polynomial if and only if $g_f = 0$.

Easy matrix manipulations show that Raussnitz's criterion [23], when expressed in terms of the Sylvester matrix, is equivalent to the above.

5. Testing permutation polynomials. Let $f \in \mathbb{F}_q[x]$ have degree n. We want to use Theorem 2 to test efficiently whether f is a permutation polynomial or not. Computing the resultant as the determinant of the $(q+n) \times (q+n)$-Sylvester matrix would be very costly. We can, however, use the Euclidean algorithm to compute the resultant as follows.

Let F be any field, $a_0, a_1 \in F[x]$ of degrees $n_0 \geq n_1 \geq 0$, respectively, and consider the Euclidean scheme for (a_0, a_1), consisting of the remainders $a_0, a_1, a_2, \cdots, a_l \in F[x]$ and the quotients $q_1, \cdots, q_l \in F[x]$ in the Euclidean algorithm for a_0 and a_1, defined by

$$a_{i-1} = q_i a_i + a_{i+1} \text{ and } \deg a_{i+1} < \deg a_i \tag{1}$$

for $1 \leq i \leq l$, using $a_{l+1} = 0$. This scheme always exists and is unique. $(q_1, \cdots, q_l, a_l)$ is called the *Euclidean representation* of (a_0, a_1) (Knuth [17] and Strassen [27]). Furthermore, let $n_i = \deg a_i$, $d_i = \deg q_i$, $\alpha_i \in F$ be the leading coefficient of a_i, and $\gamma_i \in F$ the leading coefficient of q_i. The "fundamental theorem on polynomial remainder sequences" says that if $n_l \geq 1$, then $\operatorname{res}(a_0, a_1) = 0$, and if $n_l = 0$, then

$$\operatorname{res}(a_0, a_1) = (-1)^s \alpha_l^{n_{l-1}} \prod_{1 \leq i < l} \alpha_i^{n_{i-1} - n_{i+1}}, \tag{2}$$

where $s = \sum_{0 \leq i < l} n_i n_{i+1}$ (Collins [8] and Brown and Traub [6]).

Equation (1) implies that $\alpha_{i-1} = \gamma_i \alpha_i$ and $n_{i-1} = d_i + n_i$ for all i. It follows that

$$\alpha_i = \alpha_1 \prod_{2 \leq j \leq i} \gamma_j^{-1}$$

for $2 \leq i \leq l$. Substituting this into (2) and collecting powers of γ_i, we find

$$\operatorname{res}(a_0, a_1) = (-1)^s \alpha_1^{n_0 + n_1} \prod_{2 \leq i \leq l} \gamma_i^{-(n_{i-1} + n_i)}, \tag{3}$$

if $n_l = 0$.

Thus we could calculate our g_f by executing the Euclidean algorithm for $x^q - x$ and $f - y$ in $\mathbb{F}_q(y)[x]$. Again, this would be very inefficient since the first division of $x^q - x$ by $f - y$ may already leave us with a remainder whose degree in y is very large—about q/n. We circumvent this problem by substituting a random element $u \in \mathbb{F}_{q^m}$ for y, using an appropriate extension $\mathbb{F}_{q^m}$ of $\mathbb{F}_q$.

ALGORITHM TEST FOR PERMUTATION POLYNOMIAL.

Input: Coefficients of a monic polynomial $f \in \mathbb{F}_q[x]$ of degree n, $2 \le n < q$, and confidence parameter $\epsilon > 0$. [Intuitively, $\epsilon \ll 1$. Note that any linear polynomial $ax + b$ with $a \neq 0$ is a permutation polynomial.]

Output: YES or NO.

1. Set $m = 1 + \lceil \log_q(2\epsilon^{-1}) \rceil$, and find an irreducible polynomial $\varphi \in \mathbb{F}_q[z]$ of degree m.
2. Choose (uniformly) a random element $u \in \mathbb{F}_{q^m} = \mathbb{F}_q[z]/(\varphi)$.
3. Set $a_0 = x^q - x$, $a_1 = f - u$, and compute the coefficients of $a_2 = (x^q - x) \text{ rem } (f - u) \in \mathbb{F}_{q^m}[x]$. This division with remainder is performed by "repeated squaring" of x, reducing modulo $f - u$ after each multiplication step.
4. Compute the Euclidean representation $(q_2, \cdots, q_l, a_l)$ for (a_1, a_2) in $\mathbb{F}_{q^m}[x]$, let $d_i = \deg q_i$ and $\gamma_i \in \mathbb{F}_{q^m}$ be the leading coefficient of q_i, for $2 \le i \le l$. If $\deg a_l \ge 1$, return YES and stop.
5. Compute $n_0 = q$, $n_1 = n$, and $n_2, \cdots, n_l$ from $n_i = n_{i-1} - d_i$, and calculate $s = \sum_{0 \le i < l} n_i n_{i+1} \text{ rem } 2$.
6. Compute
$$v = (-1)^s \prod_{2 \le i \le l} \gamma_i^{-(n_{i-1}+n_i)} - (-1)^q (u^q - u) \in \mathbb{F}_{q^m}.$$
7. Return YES if $v = 0$, and NO otherwise. □

It is convenient to ignore logarithmic factors using the "soft O" notation, introduced by von zur Gathen [11] and Babai, Luks, and Seress [2]:

$$g = \tilde{O}(h) \iff \exists k \; g = O(h(\log_2 h)^k).$$

THEOREM 3. *The algorithm can be performed with m random choices in $\mathbb{F}_q$, plus the cost of finding an irreducible polynomial of degree m, and $O(\log q \cdot M(n)M(m))$ arithmetic operations in $\mathbb{F}_q$, where $m = 1 + \lceil \log_q(2\epsilon^{-1}) \rceil$. These are $\tilde{O}(n \log_2 \epsilon^{-1})$ operations in $\mathbb{F}_q$ if $\epsilon \le q^{-1}$. If f is a permutation polynomial, the output is* YES. *If f is not a permutation polynomial, the output is* NO *with probability at least $1 - \epsilon$.*

Proof. Step 3 can be done in $O(\log q \cdot M(n))$ operations in $\mathbb{F}_{q^m}$. The usual algorithm for the Euclidean scheme calculates all quotients and remainders in $O(n^2)$ arithmetic operations. However, the Euclidean representation $(q_2, \cdots, q_l, a_l)$ can be computed in only $O(M(n) \cdot \log n)$ arithmetic operations by the Knuth–Schönhage algorithm (see Aho, Hopcroft, and Ullman [1, §8.9] and Strassen [27]). Each $\gamma_i^{n_{i-1}+n_i}$ can be calculated in $O(\log n)$ operations, and thus step 6 requires $O(l \log n + \log q)$ or $O(n \log n + \log q)$ operations in $\mathbb{F}_{q^m}$. Since one operation in $\mathbb{F}_{q^m}$ can be simulated with $O(M(m))$ operations in $\mathbb{F}_q$, the total cost is $O(\log q \cdot M(n)M(m))$ operations in $\mathbb{F}_q$. This is $\tilde{O}(n \log \epsilon^{-1})$ if $\epsilon \le q^{-1}$, since then $\log_2 q \cdot \lceil \log_q(2\epsilon^{-1}) \rceil = O(\log_2 \epsilon^{-1})$.

Set $A_1 = f - y \in \mathbb{F}_q(y)[x]$. Then $h_f = \text{res}_x(a_0, A_1)$. Since $a_1 = A_1(u) \in \mathbb{F}_{q^m}[x]$ has the same degree as A_1, we have $\text{res}_x(a_0, a_1) = h_f(u)$ and $v = g_f(u)$ in step 6. If f is a permutation polynomial, then $g_f = 0$ and $v = 0$. If f is not a permutation polynomial, then $g_f \in \mathbb{F}_q[y]$ is a nonzero polynomial of degree less then q, and $v \neq 0$ with probability more than $1 - q/q^m \ge 1 - \epsilon/2$.

If $\deg a_l \ge 1$ in step 4, then $f(v) = u$ for some $v \in \mathbb{F}_q$, and thus $u \in \mathbb{F}_q$; the probability of this happening is at most $q/q^m \le \epsilon/2$. □

If we choose ϵ polynomial in q^{-1}, then the algorithm uses $O^{\sim}(n \log q)$ operations. Some Boolean operations occur in the algorithm, e.g., in the calculation of m, the n_i, and s; we have neglected the small cost of these.

We have not specified a method for finding an irreducible polynomial in step 1. Rabin [22] gives a probabilistic algorithm for this problem, using $O(kmM(m) \log m \log q)$ operations in $\mathbb{F}_q$, and returning successfully with probability at least $1 - m^{-k}$, for any k. Choosing $k = \log \epsilon^{-1}$ gives failure probability at most ϵ for this step, and cost $O(\log \epsilon^{-1} mM(m) \log m \log q)$, which is $O^{\sim}(\log n \log^3 \epsilon^{-1})$ if $\epsilon \leq q^{-1}$.

We actually do not need φ of degree exactly m, but degree between m and $2m$, say, is sufficient. This may be useful if a table of irreducible polynomials is available, or if a particular degree is preferable, say powers of 2. One could even take a random $\varphi \in \mathbb{F}_q[z]$ of degree m without linear factors (i.e., $\gcd(z^q - z, \varphi) = 1$) and compute in the "pretend-field" $\mathbb{F}_q[z]/(\varphi)$; if a division by a zero-divisor turns up in the algorithm, this gives a factorization $\varphi = \varphi_1 \cdot \varphi_2$, and one can continue in the two rings $\mathbb{F}_q[z]/(\varphi_1)$ and $\mathbb{F}_q[z]/(\varphi_2)$. The algorithm would still work (by the Chinese Remainder Theorem), but the analysis is slightly more complicated.

Yet another possibility would be to take φ of degree 2 if $n \leq q/2$ (or n is not too close to q), and of degree 3 otherwise; run the algorithm with several random choices $u_1, \cdots, u_{2k+1}$ in $\mathbb{F}_{q^2}$ (respectively, $\mathbb{F}_{q^3}$); and take a majority vote on the individual outcomes. Each individual run has an error probability at most qn/q^2 (respectively, qn/q^3), and for $k = \log_2 \epsilon^{-1}$ (respectively, $k = \log_q \epsilon^{-1}$), the total error probability is at most ϵ. The cost of this implementation is $O^{\sim}((n + \log q) \log \epsilon^{-1})$ (respectively, $O^{\sim}(n \log \epsilon^{-1})$), plus the cost of finding φ.

Instead of studying polynomials inducing permutations on $\mathbb{F}_q$, Brawley, Carlitz, and Levine [5] consider permutations of the matrix algebra $\mathbb{F}_q^{d \times d}$, and prove that $f \in \mathbb{F}_q[x]$ is a permutation polynomial of $\mathbb{F}_q^{d \times d}$ if and only if f is a permutation polynomial of $\mathbb{F}_q, \mathbb{F}_{q^2}, \cdots, \mathbb{F}_{q^d}$, and the derivative f' does not have a root in $\mathbb{F}_q, \mathbb{F}_{q^2}, \cdots, \mathbb{F}_{q^k}$, where $k = \lfloor d/2 \rfloor$.

COROLLARY 4. *Let $f \in \mathbb{F}_q[x]$ have degree n, and $0 < \epsilon \leq q^{-d}$. There is a probabilistic algorithm which determines whether f is a permutation polynomial on $\mathbb{F}_q^{d \times d}$ correctly with probability at least $1 - \epsilon$. Apart from random choices and the finding of certain irreducible polynomials, it uses $O^{\sim}(d \log \epsilon^{-1}(n/\log q + d))$, or $O^{\sim}(d^2 n \log \epsilon^{-1})$, operations in $\mathbb{F}_q$.*

Proof. We simply implement the first of the Brawley, Carlitz, and Levine conditions using the Algorithm Test for Permutation Polynomial, with

$$O^{\sim}((n + j \log q) \log_{q^j}(n/\epsilon))$$

arithmetic operations in $\mathbb{F}_{q^j}$, for $1 \leq j \leq d$, each costing $O^{\sim}(j)$ operations in $\mathbb{F}_q$. This leads to the stated bound. The second condition

$$\gcd(x^{q^j} \dot{-} x, f') = 1 \quad \text{for } 1 \leq j \leq k$$

can also be tested at this cost. □

6. A criterion for large polynomials. Let $\rho \in \mathbb{N}$, and recall the notion of ρ-large from the introduction, and $h_f \in \mathbb{F}_q[y]$ from §4. Then we have:

$$\begin{aligned} f \text{ is } \rho\text{-large} &\iff \exists P \subseteq \mathbb{F}_q \ (\#P \geq q - \rho \text{ and } \forall u \in P \ \exists v \in \mathbb{F}_q \ f(v) = u) \\ &\iff \exists P \subseteq \mathbb{F}_q \ (\#P \geq q - \rho \text{ and } \forall u \in P \ \exists v \in \mathbb{F}_q \ x - v \mid f - u) \\ &\iff \exists P \subseteq \mathbb{F}_q \ (\#P \geq q - \rho \text{ and } \forall u \in P \ \gcd(x^q - x, f - u) \neq 1) \end{aligned}$$

$$\begin{aligned}
&\iff \exists P \subseteq \mathbb{F}_q \ (\#P \geq q - \rho \text{ and } \forall u \in P \ \ h_f(u) = 0) \\
&\iff \exists P \subseteq \mathbb{F}_q \ \ (\#P \geq q - \rho \ \text{ and } \ \forall u \in P \ \ y - u \,|\, h_f) \\
&\iff \deg(\gcd(y^q - y, h_f)) \geq q - \rho \\
&\iff \deg((y^q - y)/k_f) \leq \rho \\
&\iff \deg(h_f/k_f) \leq \rho,
\end{aligned}$$

where $k_f = \gcd(y^q - y, h_f) \in \mathbb{F}_q[y]$. Thus we have the following criterion for ρ-large polynomials.

THEOREM 5. *Let $f \in \mathbb{F}_q[x]$, and $h_f = \operatorname{res}_x(x^q - x, f - y)$ and $k_f = \gcd(y^q - y, h_f)$ in $\mathbb{F}_q[y]$. Then f is ρ-large if and only if $\deg((y^q - y)/k_f) \leq \rho$.*

If f is ρ-large, then all but at most 2ρ elements u of $\mathbb{F}_q$ have exactly one preimage v under f; this unique v can be easily found from $x - v = \gcd(x^q - x, f - u)$.

7. A test for large polynomials. Let $a_0 = x^q - x$, $A_1 = f - y \in \mathbb{F}_q(y)[x]$, and $Q_1, \cdots, Q_l, A_l \in \mathbb{F}_q(y)[x]$ be the Euclidean representation of (a_0, A_1) (i.e., the quotients and the gcd as calculated by the Euclidean algorithm). If $u \in \mathbb{F}_{q^m}$, $a_1 = A_1(u) = f - u \in \mathbb{F}_{q^m}[x]$, and $q_1, \cdots, q_{l'}, a_{l'}$ is the Euclidean representation of (a_0, a_1) in $\mathbb{F}_{q^m}[x]$, then $l = l'$, $q_i = Q_i(u)$ for all i, and $a_l = A_l(u)$. In particular, $\deg q_i = \deg Q_i$.

The Euclidean algorithm for (a_0, A_1) requires, of course, tests for zero or branching, in order to determine the degree sequence $(\deg Q_1, \cdots, \deg Q_l, \deg A_l)$. However, the computation using u gives us the correct degree sequence, and then all entries of the Euclidean scheme are rational functions in the coefficients of a_0 and A_1; the subresultant theory provides explicit formulas. In fact, we obtain an *arithmetic circuit* (or straight-line program) for h_f, i.e., a computation using only the coefficients of f and the operations $+, -, *, /$. The size of an arithmetic circuit is the number of arithmetic operations in it.

It is not clear how to calculate efficiently $k_f = \gcd(y^q - y, h_f)$, if we regard the degree q of $B_1 = y^q - y$ and $B_2 = h_f$ as exponentially large. However, in

$$B_0 = \frac{B_1}{B_2} = \frac{y^q - y}{h_f} = \frac{(y^q - y)/k_f}{h_f/k_f} = \frac{C_1}{C_2}$$

the two polynomials $C_1 = (y^q - y)/k_f$ and $C_2 = h_f/k_f$ are relatively prime. We can now call Kaltofen's [15] Algorithm Rational Numerator and Denominator, to calculate those two polynomials from the arithmetic circuits for h_f (discussed above) and $y^q - y$ (repeated squaring).

FACT 6 (Kaltofen [15]). *Suppose an arithmetic circuit α of size s with one input y over a field F computes $B_0 = C_1/C_2 \in F(y)$, with $C_1, C_2 \in F[y]$ relatively prime, and that $u \in F$ is such that no division by zero occurs in α on input $y \leftarrow u$. Then, given α and u, and an integer ρ, one can compute (deterministically) with $O(M(\rho)(s + \log \rho))$ arithmetic operations in F an arithmetic circuit β of size $O(M(\rho)(s + \log \rho))$ over F which computes two polynomials c_1 and c_2 in $F[y]$ of degree at most ρ such that if $\deg C_1, \deg C_2 \leq \rho$, then $c_1 = C_1$ and $c_2 = C_2$. β has no divisions by zero on input $y \leftarrow u$.*

This is a special case of Kaltofen's theorem 8.1 [15]. To derive it, we note that we can replace Kaltofen's "Step FT" with $a_1 = u$, and all the nasty possibilities that complicate Kaltofen's proof (for multivariate polynomials) vanish in our simple case. In particular, this variant is deterministic, while in the general case, Kaltofen needs probabilistic choice. Note that, if $\deg C_1 > \rho$ or $\deg C_2 > \rho$, then $B_0 \neq c_1/c_2$. (We

ignore the Boolean cost of the procedure, and assume, also in the sequel, a reasonable convention for $\rho = 0$ in the O-notation.)

The following algorithm results from the above discussion.

ALGORITHM TEST FOR POLYNOMIAL WITH LARGE IMAGE.

Input: Coefficients of a monic polynomial $f \in \mathbb{F}_q[x]$ of degree n, $2 \le n < q$, some $\rho \in \mathbb{N}$ with $0 \le \rho \le q$, and a confidence parameter $\epsilon > 0$.

Output: YES or NO.

i. Set $m = 1 + \lceil \log_q(3\epsilon^{-1}) \rceil$, and find an irreducible polynomial $\varphi \in \mathbb{F}_q[z]$ of degree m.

ii. Perform steps 2, 3, 4, and 5 of Algorithm Test for Permutation Polynomial.

iii. Compute $b_2 = (-1)^s \prod_{2 \le i \le l} \gamma_i^{-(n_{i-1}+n_i)} \in \mathbb{F}_{q^m}$, and $b_1 = u^q - u$.

iv. Let $A_1 = f - y \in \mathbb{F}_q(y)[x]$. Consider the following arithmetic circuit α over $\mathbb{F}_q$ with one input y, working in five stages.

a. Compute $A_2 \equiv x^q - x \bmod A_1$.

b. Compute the "Euclidean representation" $(Q_2, \cdots, Q_l, A_l)$ for (A_1, A_2) in $\mathbb{F}_q(y)[x]$, using the degree sequence $(d_2, \cdots, d_l)$ computed in step ii.

c. Let $\Gamma_i \in \mathbb{F}_q(y)$ be the leading coefficient of Q_i, and compute

$$B_2 = h_f = (-1)^s \prod_{2 \le i \le l} \Gamma_i^{-(n_{i-1}+n_i)} \in \mathbb{F}_q[y].$$

d. Compute $B_1 = y^q - y$, by repeated squaring.

e. Compute the output $B_0 = B_1/B_2$.

[Note that in this step we do not actually calculate B_0, but rather describe an arithmetic circuit for B_0.]

v. Call Kaltofen's Algorithm Rational Numerator and Denominator (Fact 6) with input α and u, and degree bound ρ both for numerator and denominator. The output is an arithmetic circuit β computing two polynomials c_1 and c_2 in $\mathbb{F}_q[y]$ of degree at most ρ. [If $\deg C_1 \le \rho$, with $C_1 = B_1/k_f$ as above, then $B_0 = c_1/c_2$.]

vi. Execute β with input u to calculate $c_1(u)$ and $c_2(u)$, and compute $c_3 = b_1 \cdot c_2(u) - b_2 \cdot c_1(u)$. If $c_3 = 0$, then output YES; otherwise output NO.

THEOREM 7. *The algorithm can be performed with $m = 1 + \lceil \log_q(3\epsilon^{-1}) \rceil$ random choices in $\mathbb{F}_q$, plus the cost of finding an irreducible polynomial of degree m, and*

$$O(M(m)M(n)M(\rho)\log q)$$

arithmetic operations in $\mathbb{F}_q$. These are $\tilde{O}(n\rho \log_2 \epsilon^{-1})$ operations if $\epsilon \le q^{-1}$. If f is ρ-large, the output is YES. *If f is not ρ-large, the output is* NO *with probability at least $1 - \epsilon$.*

Proof. If $n_l \ge 1$ in step ii, then $\gcd(x^q - x, f - u) \ne 1$ and hence $f(v) = u$ for some $v \in \mathbb{F}_q$, and thus $u \in \mathbb{F}_q$. This occurs with probability at most $q/q^m \le \epsilon/3$.

The subresultant theory (Brown and Traub [6]) guarantees that h_f is correctly computed in step iv.c, by (3). Recall $B_0 = B_1/B_2 = C_1/C_2$, with $C_1, C_2 \in \mathbb{F}_q[y]$ relatively prime, and let $C_3 = C_1 \cdot c_2 - C_2 \cdot c_1 \in \mathbb{F}_q[y]$. If the correct output is YES, so that $\deg C_1, \deg C_2 \le \rho$, then Fact 6 says that $c_1 = C_1$ and $c_2 = C_2$. Then $C_3 = 0$ and $c_3 = C_3(u) = 0$, and the correct answer YES will be output.

If the correct output is NO, then we know that $\deg C_1$ and $\deg C_2$ are larger than ρ. Step v will output two polynomials c_1 and c_2 of degree at most ρ, essentially

unrelated to our problem. Now $C_3 \in \mathbb{F}_q[y]$ is nonzero of degree at most $q+\rho$, and an incorrect output in step vi implies that $C_3(u) = 0$, which happens with probability at most $(q+\rho)q^{-m} \leq 2\epsilon/3$ (Schwartz [25]). The total error probability is at most $\epsilon/3 + 2\epsilon/3 = \epsilon$.

We assume the standard representation of elements of $\mathbb{F}_{q^m}$ by vectors in $\mathbb{F}_q^m$. The number of random choices in $\mathbb{F}_q$ is m. Using $n < q$, one finds that the size s of α is $s = O(M(n)\log q)$, and $\log \rho = O(s)$. The number of arithmetic operations in $\mathbb{F}_q$ is $O(sM(m))$ in steps i through iii, negligible in step iv (which just sets up a circuit), $O(sM(\rho))$ in step v, and $O(sM(m)M(\rho))$ in step vi. Thus the total number of arithmetic operations in $\mathbb{F}_q$ is $O(M(m)M(n)M(\rho)\log q)$, which is $O^{\sim}(n\rho \log_2 \epsilon^{-1})$ if $\epsilon \leq q^{-1}$. □

For $f \in \mathbb{F}_q[x]$, let $\rho_f = q - V(f)$, so that f is ρ_f-large and not $(\rho_f - 1)$-large. By a "binary search on ρ" one can compute ρ_f with $O^{\sim}(n\rho_f \log_2 \epsilon^{-1})$ operations correctly with probability at least $1-\epsilon$.

8. Manipulating polynomials of large degree. A central ingredient of the algorithms presented here are efficient computations (for special problems) with polynomials of large (exponential) degree and small (polynomial-size) arithmetic circuits. It remains open how to put this development into a more general framework. Suppose we have two polynomials f and g (over a field, in many variables), given by two arithmetic circuits of size s and t, respectively, and an integer ρ, not larger than 2^s and 2^t. Kaltofen's methods can decide, e.g., whether $\deg f \leq \rho$ in random polynomial time $(\rho s)^{O(1)}$. If $\deg f = \deg g$ is known, then the method given above (namely, computing the reduced numerator and denominator of f/g) can decide whether $\deg \gcd(f,g) \geq \deg f - \rho$ in random polynomial time $(\rho st)^{O(1)}$; $\deg f$ may be exponentially large.

In fact, this method only requires an estimate $e \geq |\deg f - \deg g|$, and uses time $(\rho ste)^{O(1)}$. As an application, suppose that $\mathrm{char}(F) = 0$ and, for simplicity, that $f \in F[x]$. One can easily find a small arithmetic circuit for $f' = \partial f/\partial x$; Baur and Strassen [3] produce one of size at most $5s$ even in the multivariate case. Then $\deg f' = \deg f - 1$, and we can test in time $(\rho s)^{O(1)}$ whether $\deg \gcd(f,f') \geq \deg f - \rho$, i.e., whether the squarefree part of f has degree at most ρ.

Here is a list of a few problems in manipulation of polynomials of exponentially large degree that one would like to answer in time $(\rho st)^{O(1)}$. For some, $\deg f$ might be an additional input.

(1) Test whether $\deg f \leq \rho$ in time polynomial in $\log \rho$.
(2) Does g divide f?
(3) Is f squarefree? Does the squarefree part of f have degree at least $(\deg f) - \rho$?
(4) Is $\deg \gcd(f,g) \leq \rho$? (This is probably a difficult problem; Plaisted [21] shows that the question "is $\gcd(f,g) \neq 1$?" is NP-hard for $f, g \in \mathbb{Q}[x]$.)
(5) Can one compute the Euclidean representation of (f,g) in time polynomial in the input *plus output* size, say in the sparse representation? For this, it seems sufficient to have a (probabilistic) polynomial-time test for

$$\deg f \overset{?}{\leq} d$$

(given f as above and the binary representation of $d \in \mathbb{N}$), due to the rational nature of the Euclidean scheme for fixed degree sequence.
(6) Do some (or all) irreducible factors of f have degree at most ρ? Degree at least $(\deg f) - \rho$?

The positive results mentioned above easily carry over to the "black box" model; Kaltofen and Trager [16] present the necessary (probabilistic) algorithms. For questions (2)–(6), one might start by considering the sparse representation of f.

Acknowledgments. I thank Rudolf Lidl for discussions during my visit to the University of Tasmania and for his continued support with pointers and references, Brendan McKay for help with some computations, and Victor Shoup for an improvement in the proof of Theorem 3.

REFERENCES

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison–Wesley, Reading MA, 1974.

[2] L. Babai, E. M. Luks, and Á. Seress, *Fast management of permutation groups*, in Proc. 29th IEEE Symposium on Foundations of Computer Science, White Plains, NY, 1988, pp. 272–282.

[3] W. Baur and V. Strassen, *The complexity of partial derivatives*, Theoret. Comput. Sci., 22 (1982), pp. 317–330.

[4] A. Borodin and I. Munro, *The Computational Complexity of Algebraic and Numeric Problems*, American Elsevier, New York, 1975.

[5] J.V. Brawley, L. Carlitz, and J. Levine, *Scalar polynomial functions on the $n \times n$ matrices over a finite field*, Linear Algebra Appl., 10 (1975), pp. 199–217.

[6] W. S. Brown and J. F. Traub, *On Euclid's algorithm and the theory of subresultants*, J. Assoc. Comput. Mach., 18 (1971), pp. 505–514.

[7] D. G. Cantor and E. Kaltofen, *Fast multiplication of polynomials over arbitrary rings*, Tech. Report 87-35, Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, 1987; Acta Inform., to appear.

[8] G. E. Collins, *The calculation of multivariate polynomial resultants*, J. Assoc. Comput. Mach., 18 (1971), pp. 515–532.

[9] L.E. Dickson, *The analytic representation of substitutions on a power of a prime number of letters with a discussion of the linear group*, Ann. of Math., 11 (1897), pp. 65–120, 161–183.

[10] ———, *Linear Groups with an Exposition of the Galois Field Theory*, Teubner, Leipzig, Stuttgart, 1901; Dover, New York, 1958.

[11] J. von zur Gathen, *Irreducibility of multivariate polynomials*, J. Comput. System Sci., 31 (1985), pp. 225–264.

[12] ———, *Values of polynomials over finite fields*, Bull. Austral. Math. Soc., 43 (1991), pp. 141–146.

[13] D. R. Hayes, *A geometric approach to permutation polynomials over a finite field*, Duke Math. J., 34 (1967), pp. 293–305.

[14] C. Hermite, *Sur les fonctions de sept lettres*, C.R. Acad. Sci. Paris, 57 (1863), pp. 750–757; also in Œuvres, Vol. 2, Gauthier-Villars, Paris, 1908, pp. 280–288.

[15] E. Kaltofen, *Greatest common divisors of polynomials given by straight-line programs*, J. Assoc. Comput. Mach., 35 (1988), pp. 231–264.

[16] E. Kaltofen and B. M. Trager, *Computing with polynomials given by black boxes for their evaluations: Greatest common divisors, factorization, separation of numerators and denominators*, J. Symb. Comp., 9 (1990), pp. 301–320.

[17] D. E. Knuth, *The analysis of algorithms*, in Proc. Internat. Congress of Mathematicians, Vol. 3, Nice, France, 1970, pp. 269–274.

[18] R. Lidl and G.L. Mullen, *When does a polynomial over a finite field permute the elements of the field?*, Amer. Math. Monthly, 95 (1988), pp. 243–246.

[19] R. Lidl and H. Niederreiter, *Finite Fields*, Vol. 20, Encyclopedia of Mathematics and Its Applications, Addison–Wesley, Reading MA, 1983.

[20] E. Lucas, *Sur les congruences des nombres eulériens et des coefficients différentiels des fonctions trigonométriques, suivant un module premier*, Bull. Soc. Math. France, 6 (1877/78), pp. 49–54.

[21] D. A. Plaisted, *New NP-hard and NP-complete polynomial and integer divisibility problems*, Theoret. Comput. Sci., 31 (1984), pp. 125–138.

[22] M. O. Rabin, *Probabilistic algorithms in finite fields*, SIAM J. Comput., 9 (1980), pp. 273–280.

[23] G. Raussnitz, *Zur Theorie der Congruenzen höheren Grades*, Math. Naturwiss. Ber. Ungarn, 1 (1883), pp. 266–278.

[24] A. SCHÖNHAGE AND V. STRASSEN, *Schnelle Multiplikation großer Zahlen*, Computing, 7 (1971), pp. 281–292.
[25] J. T. SCHWARTZ, *Fast probabilistic algorithms for verification of polynomial identities*, J. Assoc. Comput. Mach., 27 (1980), pp. 701–717.
[26] I.E. SHPARLINSKIY, *Private communication*, August 1990.
[27] V. STRASSEN, *The computational complexity of continued fractions*, SIAM J. Comput., 12 (1983), pp. 1–27.
[28] B. L. VAN DER WAERDEN, *Algebra*, Vol. 1, Seventh Edition, Frederick Ungar, New York, 1970.