Facultad de Ciencias

Permutation Polynomials and Applications

Polinomios Permutacionales y Aplicaciones

Trabajo Fin de Máster para acceder al

Máster en Matemáticas y Computación

Autora: Andrea Altemir Castán

Tutor: Jaime Gutierrez Gutierrez

Septiembre - 2023

Resumen

El objetivo de este Trabajo Fin de Máster es el estudio de los polinomios permutacionales y localmente permutacionales en varias variables definidos sobre cuerpos finitos. En la primera parte de esta memoria se introduce el concepto de polinomios permutacionales en una variable, así como una serie de resultados básicos. El segundo capítulo extiende este concepto a varias variables además de presentar los polinomios localmente permutacionales, proporcionando propiedades, caracterizaciones y construcciones. El tercer capítulo se centra en polinomios localmente permutacionales en dos variables, exhibiendo una familia de estos denominada *e-Klenian polynomials*, y estudiando la relación entre polinomios localmente permutacionales bivariados y cuadrados latinos. Además, se trata el tema de la ortogonalidad, hablando de Sistemas Ortogonales de Polinomios y de Cuadrados Latinos Mutuamente Ortogonales (MOLS, por sus siglas en inglés). Por último, se incluye en esta memoria un capítulo dedicado a dos de las aplicaciones más notables de los cuadrados latinos en los ámbitos de la Teoría de Códigos y Criptografía.

Ciertos resultados y herramientas para la manipulación de estos polinomios han sido implementados en el sistema de computación simbólica SageMath.

Abstract

The goal of this dissertation is the study of permutation and local permutation polynomials in several variables defined over finite fields. In the first part of this work the concept of permutation polynomials in one variable is introduced alongside a series of basic results. The second chapter extends this concept to several variables, providing properties, characterizations and constructions. The focus of the third chapter is local permutation polynomials in two variables, presenting a family of such polynomials denoted e-Klenian polynomials, and studying the relation between bivariate local permutation polynomials and Latin squares. Also, the topic of orthogonality is discussed, talking about Orthogonal Polynomial Systems and Mutually Orthogonal Latin Squares (MOLS). Finally, this report includes a chapter dedicated to two of the most notable applications of Latin squares in the areas of Coding Theory and Cryptography.

Some results and tools to manipulate these polynomials have been implemented in the symbolic computation system SageMath.

Contents

Introduction

Let $\mathbb{F}_q$ be the finite field with $q = p^r$ elements, where $p, r \in \mathbb{N}$, p prime. A polynomial $f \in \mathbb{F}_q[x]$ is called a *Permutation Polynomial* (PP) of $\mathbb{F}_q$ if the induced mapping $x \to f(x)$ is a permutation of $\mathbb{F}_q$. The study of permutation polynomials over finite fields has a long history. There are numerous books and survey papers on the subject, see for instance [5], [16], [11] to mention a few of them.

As for their applications, permutations of finite fields have become of considerable interest in the construction of cryptographic protocols, where bijective functions can be used to encrypt and decrypt messages. Of course, in order to be useful in a cryptography system, these functions must have several additional properties, see [22]. PPs are also useful in multiple combinatorial applications, see [14].

The generalisation of permutation polynomials to $n \geq 2$ variables was first defined in [23]. A polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ is a *Permutation Polynomial* if the equation $f(x_1, \ldots, x_n) = a$ has q^{n-1} solutions in $\mathbb{F}_q^n$ for each $a \in \mathbb{F}_q$. Note how, if $n = 1$, f is a univariate permutation polynomial as in the previous definition.

A closely related concept is the following: a polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ is called a *Local Permutation Polynomial* (LPP) if for each $i \in \{1, \ldots, n\}$, the polynomial $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a permutation polynomial in $\mathbb{F}_q[x_i]$, for all choices of $(a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n) \in \mathbb{F}_q^{n-1}$. Any local permutation polynomial is a permutation polynomial, but the opposite is not true in general.

Contrary to the many papers and results on permutation polynomials in one variable, there are few for permutation and local permutation polynomials in several variables.
A classification of permutation polynomials in $\mathbb{F}_q[x_1, \ldots, x_n]$ of degree at most two is given in [23], see also [16] for several properties and results and the particular case $n = 1$. The author of [20] and [21] gives necessary and sufficient conditions for polynomials in two and three variables to be local permutations polynomials over a prime field $\mathbb{F}_p$. These conditions are expressed in terms of the coefficients of the polynomial.
A result about degree bounds for n local permutation polynomials defining a permutation of $\mathbb{F}_q^n$ is presented in [1].
A significant part of the results displayed this dissertation are part of the recent papers [9] and [10].

An important concept strongly related to Local Permutation Polynomials is **Latin Squares**, namely $t \times t$ matrices with entries from a set T of size t such that each element of T occurs exactly once in every row and every column of the matrix.

All Latin squares such that $T = \mathbb{F}_q$ can be represented by a bivariate local permutation polynomial $f(x, y) \in \mathbb{F}_q[x, y]$ (see Lemma 3.20), and the relevance of this representation for the study of Latin squares, as well as Latin cubes, are described in [20] and [21].

Latin squares occur in many structures such as group multiplication tables and Cayley tables. To be precise, Latin squares are referred to as the multiplication tables of an algebraic structure called a quasigroup.

Two Latin squares L_1 and L_2 of order t are orthogonal if, when superimposed, each position has a different pair of ordered entries, and a set of **Mutually Orthogonal Latin Squares (MOLS)** is a set of Latin squares that are pairwise orthogonal. The construction of sets of MOLS is a notoriously difficult combinatorial problem and it is one of the most studied research topics in design theory [19]. This interest is also due to the numerous applications that MOLS have in other fields such as Cryptography [26], Coding Theory and many others, see [13, 17, 27].

We focus on Latin squares of order p^r, where p is prime and $r \geq 0$.

The remainder of the dissertation is structured as follows.

We start by presenting in Chapter 1 some general properties and preliminary results on permutation polynomials in one variable for later use.

Then, in Chapter 2 we introduce the concepts of permutation and local permutation polynomials in several variables, discussing their fundamental properties and matters regarding the maximum degree these polynomials can have.

Due to the one to one map between Latin squares and bivariate local permutation polynomials, Chapter 3 is dedicated to the study of polynomials in $\mathbb{F}_q[x, y]$. Besides delving into said relation, we provide a family of LPPs in two variables, known as e-Klenian polynomials, and discuss the concept of orthogonality for both polynomials and Latin squares. Also, we show general constructions of sets of MOLS, one of them based on e-Klenian polynomials.

In Chapter 4 we present a couple notable applications of Latin squares, or equivalently bivariate local permutation polynomials, to Coding Theory in the construction of Maximum Distance Separable (MDS) codes and to Cryptography for the design of secret sharing schemes.

We conclude with Appendix A illustrating the more relevant functions of the PermutationPolynomials SageMath package for manipulating permutation and local permutation polynomials, and with Appendix B displaying examples of these polynomials.

CHAPTER 1
Permutation Polynomials in One Variable

The purpose of this introductory chapter is to provide a basis to build upon for the rest of the Thesis. We will study polynomials in one variable over finite fields, focusing on a special family called permutation polynomials.

1.1. Univariate Polynomials over Finite Fields

Let us start by establishing relevant notation. Let $p \in \mathbb{N}$ be a prime number.

- We denote by $\mathbb{F}_p \cong \mathbb{Z}_p$ the field of p elements.
- $\mathbb{F}_p[x]$ is the polynomial ring in the variable x.
- Let $f \in \mathbb{F}_p[x]$ be a polynomial. We denote the degree of f as $\deg(f)$.

Let $f \in \mathbb{F}_p[x]$ be an irreducible polynomial of degree r and (f) the principal ideal generated by f. We construct the quotient ring $\mathbb{F}_p[x]/(f)$, obtaining a finite field of $q = p^r$ elements and characteristic p, denoted by $\mathbb{F}_q$.

$\mathbb{F}_q^*$ is the multiplicative group of nonzero elements of $\mathbb{F}_q$. It is a cyclic group, and a generator α is called a primitive element of $\mathbb{F}_q$.

Proposition 1.1. *Let $c \in \mathbb{F}_q$, where $q = p^r$, $r, p \in \mathbb{N}$, p prime. Then, $c^q = c$ and*

$$x^q - x = \prod_{c \in \mathbb{F}_q} (x - c).$$

Proof. The identity $c^q = c$ is trivial for $c = 0$. Now, let c be a nonzero element of $\mathbb{F}_q$, that is, $c \in \mathbb{F}_q^*$. Since $\mathbb{F}_q^*$ is a cyclic group of order $q-1$ under multiplication, we have $c^{q-1} = 1$, which implies $c^q = c \ \forall c \in \mathbb{F}_q^*$, proving the identity.

Now let us prove the second claim. We have just shown that the q elements of $\mathbb{F}_q$ satisfy $c^q - c = 0$, therefore all of them are roots of the polynomial $x^q - x$, and since

there are at most q roots (because it is of degree q), these are the only ones. So, we can write

$$x^q - x = \prod_{c \in \mathbb{F}_q} (x - c).$$

□

When working with functions over $\mathbb{F}_q$, it suffices to consider polynomials of degree at most $q - 1$ thanks to the Lagrange Interpolation Theorem.

Theorem 1.2 (Lagrange's Interpolation)**.** *For any arbitrary function $\phi : \mathbb{F}_q \to \mathbb{F}_q$ there exists a unique polynomial $f \in \mathbb{F}_q[x]$ with $\deg(f) < q$ and $f(c) = \phi(c)\ \forall c \in \mathbb{F}_q$.*

Proof. To prove the theorem, we are going to show that the polynomial

$$f(x) = \sum_{a \in \mathbb{F}_q} \phi(a)(1 - (x - a)^{q-1})$$

has the desired properties.

First, we need to check that $f(c) = \phi(c)\ \forall c \in \mathbb{F}_q$. Evaluating $f(c)$, we obtain

$$f(c) = \sum_{a \in \mathbb{F}_q} \phi(a)(1 - (c - a)^{q-1}).$$

Since $\mathbb{F}_q^*$ is a cyclic group of order $q - 1$,

$$(c - a)^{q-1} = \begin{cases} 0 & \text{if } c - a = 0 \\ 1 & \text{otherwise.} \end{cases}$$

This implies that $1 - (c - a)^{q-1}$ is equal to 1 if and only if $a = c$, and it is 0 otherwise. Therefore,

$$f(c) = \sum_{a \in \mathbb{F}_q} \phi(a)(1 - (c - a)^{q-1}) = \phi(c) \cdot 1 + \sum_{a \in \mathbb{F}_q, a \neq c} \phi(a) \cdot 0 = \phi(c),$$

which is what we wanted.

Also, f is a polynomial of degree at most $q - 1$ by construction, so $\deg(f) < q$.

Finally, suppose that there is another polynomial $g \in \mathbb{F}_q[x]$ such that $\deg(g) < q$ and $g(c) = \phi(c)\ \forall c \in \mathbb{F}_q$. We know that $f(c) = g(c)\ \forall c \in \mathbb{F}_q$, which implies $f(c) - g(c) = 0\ \forall c \in \mathbb{F}_q$. This means the polynomial $f(x) - g(x)$ has q distinct roots, which is impossible since its degree is at most $q - 1$:

$$\deg(f - g) \leq \max(\deg(f), \deg(g)) < q.$$

We have arrived at a contradiction, thus proving the uniqueness of f and completing the proof. □

1.2. Permutation Polynomials

In this section we will gather some basic results concerning a particular type of polynomials over finite fields, known as permutation polynomials, that we will introduce alongside several examples.

These polynomials were first studied by Charles Hermite (1822-1901), who formulated a criterion to identify them, which will also be presented.

Definition 1.3. Let $f \in \mathbb{F}_q[x]$. f is a **permutation polynomial (PP)** of $\mathbb{F}_q$ if the associated function $f : c \to f(c)$ from $\mathbb{F}_q$ into $\mathbb{F}_q$ is a permutation of $\mathbb{F}_q$.

There are multiple characterizations of permutation polynomials. Some of the most basic and elementary ones are described in this next lemma.

Lemma 1.4. *The polynomial $f \in \mathbb{F}_q[x]$ is a permutation polynomial of $\mathbb{F}_q$ if and only if at least one of the following conditions holds:*

(i). *the function $f : c \to f(c)$ is surjective;*

(ii). *the function $f : c \to f(c)$ is injective;*

(iii). *$f(x) = a$ has a solution in $\mathbb{F}_q$ for each $a \in \mathbb{F}_q$;*

(iv). *$f(x) = a$ has a unique solution in $\mathbb{F}_q$ for each $a \in \mathbb{F}_q$.*

Proof. $\boxed{(i) \text{ and } (ii)}$ A permutation of a finite set of elements defines a bijective function from the set to itself. Considering this and the fact that surjective and injective functions from one finite set to another with the same cardinality are bijective, we get the equivalences with conditions (i) and (ii).

$\boxed{(i) \iff (iii)}$ The function $f : c \to f(c)$ is surjective, which means that $\forall a \in \mathbb{F}_q$ $\exists c \in \mathbb{F}_q$ such that $f(c) = a$. In other words, $f(x) = a$ has a solution in $\mathbb{F}_q$ for each $a \in \mathbb{F}_q$. This reasoning also works in the opposite direction, giving us the equivalence.

$\boxed{(iii) \iff (iv)}$ The left direction of the equivalence is trivial, thus we only need to prove the right direction. We know that $f(x) = a$ has a solution in $\mathbb{F}_q$ for each $a \in \mathbb{F}_q$. The uniqueness of this solution comes from the fact that f is injective, since $(iii) \iff (ii)$. □

We shall now establish a useful and more sophisticated criterion for permutation polynomials. Before doing so, we will need the following two lemmata.

Lemma 1.5. *For $f, g \in \mathbb{F}_q[x]$, we have $f(c) = g(c)$ for all $c \in \mathbb{F}_q$ if and only if $f(x) \equiv g(x) \mod (x^q - x)$.*

Proof. By the division algorithm we can write $f(x) - g(x) = h(x)(x^q - x) + r(x)$ with $h, r \in \mathbb{F}_q[x]$ and $\deg(r) < q$. Since $c^q - c = 0$ for all $c \in \mathbb{F}_q$, $f(c) = g(c)$ for all $c \in \mathbb{F}_q$ if and only if $r(c) = 0$ for all $c \in \mathbb{F}_q$, or in other words, $f(x) \equiv g(x) \mod (x^q - x)$. □

Lemma 1.6. *Let $a_0, a_1, \ldots, a_{q-1}$ be elements of $\mathbb{F}_q$. Then the following two conditions are equivalent:*

(i). $a_0, a_1, \ldots, a_{q-1}$ *are distinct;*

(ii). $\sum_{i=0}^{q-1} a_i^t = \begin{cases} 0 & \text{for } t = 0, 1, \ldots, q-2, \\ -1 & \text{for } t = q-1. \end{cases}$

Proof. For fixed i with $0 \le i \le q-1$, consider the polynomial

$$g_i(x) = 1 - \sum_{j=0}^{q-1} a_i^{q-1-j} x^j.$$

Let us calculate $g_i(b)$ for $b \in \mathbb{F}_q$, which we will do by checking the value of the sum.

If $b = a_i$,

$$\sum_{j=0}^{q-1} a_i^{q-1-j} a_i^j = \sum_{j=0}^{q-1} a_i^{q-1} = \begin{cases} \sum_{j=0}^{q-1} 0 = 0 & \text{if } a_i = 0, \\ \sum_{j=0}^{q-1} 1 = q \equiv 0 & \text{otherwise.} \end{cases}$$

Therefore, $g_i(a_i) = 1 - 0 = 0$.

If $b \neq a_i$, we will be using the equality

$$a^n - b^n = (a-b) \sum_{j=0}^{n-1} a^{n-1-j} x^j \quad \Longrightarrow \quad \sum_{j=0}^{n-1} a^{n-1-j} x^j = (a^n - b^n)(a-b)^{-1} \tag{1.1}$$

considering $a \neq b$, which is our case. Applied to our sum,

$$\sum_{j=0}^{q-1} a_i^{q-1-j} b^j = (a_i^q - b^q)(a_i - b)^{-1} = (a_i - b)(a_i - b)^{-1} = 1.$$

Therefore, $g_i(b) = 1 - 0 = 1$ for all $b \in \mathbb{F}_q$, $b \neq a_i$.

Now consider the polynomial

$$g(x) = \sum_{i=0}^{q-1} g_i(x) = \sum_{i=0}^{q-1} \left(1 - \sum_{j=0}^{q-1} a_i^{q-1-j} x^j \right) = q - \sum_{i=0}^{q-1} \sum_{j=0}^{q-1} a_i^{q-1-j} x^j =$$
$$= - \sum_{j=0}^{q-1} \left(\sum_{i=0}^{q-1} a_i^{q-1-j} \right) x^j.$$

Since $g_i(x) = 1$ if and only if $x = a_i$ and zero otherwise for $i = 0, \ldots, q-1$, g maps each element of $\mathbb{F}_q$ into 1 if and only if $\{a_0, \ldots, a_{q-1}\} = \mathbb{F}_q$, or equivalently, if and only if all a_i are distinct, which is condition (i).

Since $\deg(g) < q$, by Lemma 1.5 g maps each element of $\mathbb{F}_q$ into 1 if and only if $g(x) = 1$. This happens if and only if the independent term of g, $\sum_{i=0}^{q-1} a_i^{q-1}$, equals -1 and the other coefficients, $\sum_{i=0}^{q-1} a_i^{q-1-j}$ with $j = 1, \ldots, q-1$, are equal to 0, which is condition (ii). □

Now, we are prepared to prove the mentioned criterion.

Theorem 1.7 (Hermite's Criterion)**.** *Let $\mathbb{F}_q$ be of characteristic p. Then $f \in \mathbb{F}_q[x]$ is a permutation polynomial of $\mathbb{F}_q$ if and only if the following two conditions hold:*

(i). f has exactly one root in $\mathbb{F}_q$;

(ii). for each integer t with $1 \leq t \leq q-2$ and $t \not\equiv 0 \mod p$, the reduction of $(f(x))^t \mod (x^q - x)$ has degree $\leq q-2$.

Proof. $\boxed{\Longrightarrow}$ Let f be a permutation polynomial of $\mathbb{F}_q$. (i) is true for f since, by Lemma 1.4, $f(x) = 0$ has a unique solution. To see (ii), we find the unique polynomial with degree $< q$ that represents $(f(x))^t$ using the Lagrange Interpolation formula from Theorem 1.2,

$$\sum_{a \in \mathbb{F}_q} (f(a))^t (1 - (x-a)^{q-1}) = \sum_{j=0}^{q-1} b_j^{(t)} x^j, \text{where } b_{q-1}^{(t)} = -\sum_{a \in \mathbb{F}_q} (f(a))^t.$$

As this polynomial is of degree $< q$, it is the reduction of $(f(x))^t \mod (x^q - x)$ by Lemma 1.5. Also, since f is a permutation polynomial, all $f(a)$ with $a \in \mathbb{F}_q$ are distinct, so according to Lemma 1.6, $b_{q-1}^{(t)} = 0$ for $t = 0, 1, \ldots, q-2$, hence (ii) follows.

$\boxed{\Longleftarrow}$ Let (i) and (ii) be satisfied. Since f has exactly one root in $\mathbb{F}_q$ by (i), only one of the summands of the sum $\sum_{a \in \mathbb{F}_q} f(a)^{q-1}$ is equal to 0, while the others are equal to 1. Therefore, $\sum_{a \in \mathbb{F}_q} f(a)^{q-1} = q - 1 \equiv -1$. On the other hand, (ii) implies $\sum_{a \in \mathbb{F}_q} (f(a))^t = 0$ for $1 \leq t \leq q-2$, $t \not\equiv 0 \mod p$ using the representation given above. Thanks to

$$\sum_{a \in \mathbb{F}_q} (f(a))^{tp^j} = \left(\sum_{a \in \mathbb{F}_q} (f(a))^t \right)^{p^j}$$

we get $\sum_{a \in \mathbb{F}_q} (f(a))^t = 0$ for the remaining $1 \leq t \leq q-2$, and this identity holds trivially for $t = 0$. Lemma 1.6 implies that all $f(a)$ with $a \in \mathbb{F}_q$ are distinct, therefore f is a permutation polynomial of $\mathbb{F}_q$. □

Corollary 1.8. *If $d > 1$ is a divisor of $q-1$, then there is no permutation polynomial of $\mathbb{F}_q$ of degree d.*

Proof. Let $f \in \mathbb{F}_q[x]$ be a permutation polynomial such that $\deg(f) = d$. Then, $\deg(f^{(q-1)/d}) = q - 1$, contradicting condition (ii) of Hermite's Criterion 1.7. □

Thanks to the next result, we have a way of producing permutation polynomials from others.

Theorem 1.9. *Let $f \in \mathbb{F}_q[x]$ be a permutation polynomial. Then, $f_1(x) = af(x+b)+c$ for all $a \neq 0, b, c \in \mathbb{F}_q$ is a permutation polynomial.*

Proof. We want to prove that the equation $f_1(x) = d$ has a unique solution for all $d \in \mathbb{F}_q$. Notice that

$$f_1(x) = af(x+b) + c = d \quad \iff \quad f(x+b) = a^{-1}(d-c).$$

Since f is a permutation polynomial, $f(x) = a^{-1}(d-c)$ has a unique solution $x_0 \in \mathbb{F}_q$. Hence, $x_0 - b$ is the only solution to the equation $f_1(x) = d$. □

Definition 1.10. Using the notation from the previous theorem, we say that f_1 is in **normalized form** if a, b, c are chosen so that f_1 is monic, $f_1(0) = 0$ and (provided the characteristic p does not divide the degree n) the coefficient of x^{n-1} is 0.

Via the next theorem and lemmata, different examples of permutation polynomials will be shown.

Theorem 1.11.

(*i*). *Every linear polynomial over $\mathbb{F}_q$ is a permutation polynomial of $\mathbb{F}_q$.*

(*ii*). *x^n is a permutation polynomial of $\mathbb{F}_q$ if and only if* $\gcd(n, q-1) = 1$.

Proof. (*i*) A linear polynomial $f(x) = ax + b$, with $a, b \in \mathbb{F}_q$, $a \neq 0$ represents an injective function, hence it is a permutation polynomial of $\mathbb{F}_q$ by Lemma 1.4.

(*ii*) By Lemma 1.4, x^n is a permutation polynomial of $\mathbb{F}_q$ if and only if the function $c \in \mathbb{F}_q \to c^n$ is surjective, which happens if and only if $\gcd(n, q-1) = 1$. □

Lemma 1.12. *The polynomial*

$$f(x) = x + \sum_{k=0}^{q-2} x^k$$

permutes 1 *and* 0, *and leaves fixed any other element in $\mathbb{F}_q$, considering* $0^0 = 1$. *In general, for any distinct $a, b \in \mathbb{F}_q$,*

$$f_{a,b;q}(x) = a + (b-a)\left(\frac{x-a}{b-a} + \sum_{k=0}^{q-2}\left(\frac{x-a}{b-a}\right)^k\right)$$

is a permutation polynomial representing the transposition (a, b).

Proof. It suffices to prove that $f_{a,b;q}$ is a permutation polynomial representing the transposition (a, b).

First, it is easy to check that

$$f_{a,b;q}(a) = a + (b-a)\left(\frac{a-a}{b-a} + \sum_{k=0}^{q-2}\left(\frac{a-a}{b-a}\right)^k\right) = a + (b-a)\cdot(0+1) = b,$$

$$f_{a,b;q}(b) = a + (b-a)\left(\frac{b-a}{b-a} + \sum_{k=0}^{q-2}\left(\frac{b-a}{b-a}\right)^k\right) = a + (b-a)\left(1 + \sum_{k=0}^{q-2} 1^k\right) =$$
$$= a + (b-a)(1+q-1) = a + (b-a)\cdot q \equiv a.$$

Now we just need to see if $f_{a,b;q}$ fixes all other $c \in \mathbb{F}_q$.

We know $\frac{c-a}{b-a} \neq 0$, then $\left(\frac{c-a}{b-a}\right)^{q-1} = 1$. Using equation (1.1), possible since $\frac{c-a}{b-a} \neq 1$, we see that

$$\sum_{k=0}^{q-2} \left(\frac{c-a}{b-a}\right)^k = \left(1^{q-1} - \left(\frac{c-a}{b-a}\right)^{q-1}\right)\left(1 - \frac{c-a}{b-a}\right)^{-1} = (1-1)\left(1 - \frac{c-a}{b-a}\right)^{-1} = 0.$$

Therefore,

$$f_{a,b;q}(c) = a+(b-a)\left(\frac{c-a}{b-a} + \sum_{k=0}^{q-2}\left(\frac{c-a}{b-a}\right)^k\right) = a+(b-a)\left(\frac{c-a}{b-a} + 0\right) = a+c-a = c.$$

We have shown that $f_{a,b;q}$ fixes all other $c \in \mathbb{F}_q$, completing the proof. □

Lemma 1.13. *If α is a primitive element in $\mathbb{F}_q^*$, then the polynomial*

$$g_q(x) = (\alpha x - 1)^{q-1} - x^{q-1} + \alpha x$$

is a permutation polynomial representing the cycle $(0, 1, \alpha, \ldots, \alpha^{q-2})$ of length q.

Proof. Since α is a primitive element of $\mathbb{F}_q$, all nonzero elements of $\mathbb{F}_q$ can be written as α^r, $r \in \mathbb{N}$.

To prove that g_q represents the cycle $(0, 1, \alpha, \ldots, \alpha^{q-2})$, we need only check

- $g_q(0) = 1$;
- $g_q(\alpha^r) = \alpha^{r+1}$, $r = 0, \ldots, q-3$;
- $g_q(\alpha^{q-2}) = 0$.

Indeed,

$$\begin{aligned} g_q(0) &= (\alpha \cdot 0 - 1)^{q-1} - 0^{q-1} + \alpha \cdot 0 = (-1)^{q-1} = 1 \\ g_q(\alpha^{q-2}) &= (\alpha^{q-1} - 1)^{q-1} - (\alpha^{q-2})^{q-1} + \alpha^{q-1} = (1-1)^{q-1} - 1 + 1 = 0 \\ g_q(\alpha^r) &= (\alpha^{r+1} - 1)^{q-1} - (\alpha^r)^{q-1} + \alpha^{r+1} = 1 - 1 + \alpha^{r+1} = \alpha^{r+1}, \end{aligned}$$

considering in the last calculations that $\alpha^{r+1} \neq 1$ when $r = 0, \ldots, q-3$. □

Let $\mathbb{F}_q$ be of characteristic p. More examples of permutation polynomials are:

- The p-polynomial
$$L(x) = \sum_{i=0}^{m} a_i x^{p^i} \in \mathbb{F}_q[x]$$
is a permutation polynomial of $\mathbb{F}_q$ if and only if $L(x)$ only has the root 0 in $\mathbb{F}_q$.

- Let $r \in \mathbb{N}$ with $\gcd(r, q-1) = 1$ and let s be a divisor of $q-1$. Let $g \in \mathbb{F}_q[x]$ be such that $g(x^s)$ has no nonzero root in $\mathbb{F}_q$. Then $f(x) = x^r(g(x^s))^{(q-1)/s}$ is a permutation polynomial of $\mathbb{F}_q$.
- If $0 \neq a \in \mathbb{F}_q$, then the Dickson polynomial

$$g_k(x,a) = \sum_{j=0}^{\lfloor \frac{k}{2} \rfloor} \frac{k}{k-j} \binom{k-j}{k} (-a)^j x^{k-2j}$$

  permutes $\mathbb{F}_q$ if and only if $\gcd(k, q^2-1) = 1$.

  If we work over the complex numbers, then these polynomials are closely related to the Chebyshev polynomials of the first kind:

$$T_k(x) = \cos(k \arccos x), \quad \text{then} \quad g_k(2x, 1) = 2T_k(x).$$

- If m divides $q-1$, then $x^{(q+m-1)/m} + ax$ is a permutation polynomial of $\mathbb{F}_q$.

The proofs to all of these claims and many other results, such as a table classifing normalized polynomials, can be found in Chapter 7 of the celebrated book [16].

The study of permutation polynomials is a very active research area.

A notable general question is finding non trivial bound for $N_d = N_d(q)$, that is, the number of permutation polynomials of certain degree d. We have seen that $N_1 = q(q-1)$, as all linear polynomials are PPs, and $N_d = 0$ if d is a divisor of $q-1$, consequence of Corollary 1.8.

CHAPTER 2
Permutation Polynomials in Several Variables

Instead of working with polynomials in one variable as in the previous chapter, we are now considering polynomials over finite fields in an arbitrary finite number of variables.

We are going to study the natural generalisation of the concept of Permutation Polynomials to several variables as well as a special type of PP, the so called Local Permutation Polynomials, discussing their fundamental properties and matters regarding the maximum degree these polynomials can have.

2.1. Multivariate Polynomials over Finite Fields

First, we introduce the requierd notation for manipulating polynomials in several variables.

Let $p \in \mathbb{N}$ be a prime number, $n, r \in \mathbb{N}$, $q = p^r$ a power of a prime.

- $\mathbb{F}_q^n$ denotes the cartesian product of n copies of $\mathbb{F}_q$.
- $\bar{x} = (x_1, \ldots, x_n)$, $\bar{x}_i = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$.
- we will be working in the ring of polynomials in n variables over $\mathbb{F}_q$, denoted $\mathbb{F}_q[x_1, \ldots, x_n] = \mathbb{F}_q[\bar{x}]$.
- Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a polynomial. We denote by $\deg(f)$ the total degree of f and by $\deg_{x_i}(f)$ the degree of f as a polynomial in the variable x_i, that is, as a polynomial in $R[x_i]$, where $R = \mathbb{F}_q[\bar{x}_i]$.

Similar to the univariate case, thanks to the Lagrange Interpolation Theorem 2.2 we can identify all functions $\mathbb{F}_q^n \to \mathbb{F}_q$ with polynomials of $\mathbb{F}_q[x_1, \ldots, x_n]$ of degree $< q$ in each variable.

To prove this, we need the following technical result.

Lemma 2.1. *Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$. If f is of degree $< q$ in each indeterminate and satisfies $f(c_1, \ldots, c_n) = 0$ for all $(c_1, \ldots, c_n) \in \mathbb{F}_q^n$, then f is the zero polynomial.*

Proof. This is a proof by induction on n.

The case $n = 1$ is shown as a part of the proof of the Lagrange Interpolation Theorem 1.2, and is also a consequence of Lemma 1.5.

Now, let $n \geq 2$ and suppose the statement is true for polynomials in $n-1$ variables. Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a polynomial of the indicated type.

If f is nonzero, we can write

$$f(x_1, \ldots, x_n) = h_0(x_2, \ldots, x_n) + \sum_{i=1}^{t} h_i(x_2, \ldots, x_n) \cdot x_1^i,$$

where $0 \leq t < q$, each $h_i \in \mathbb{F}_q[x_2, \ldots, x_n]$ is such that $\deg_{x_j}(h_i) < q$, $j = 2, \ldots, n$ and $h_t \neq 0$. By induction hypothesis, $h_t \neq 0$ implies the existence of $(a_2, \ldots, a_n) \in \mathbb{F}_q^{n-1}$ such that $h_t(a_2, \ldots, a_n) \neq 0$.

We know that $f(c, a_2, \ldots, a_n) = 0$ for all $c \in \mathbb{F}_q$, but then

$$f(x_1, a_2, \ldots, a_n) = h_0(a_2, \ldots, a_n) + \sum_{i=1}^{t-1} h_i(a_2, \ldots, a_n) \cdot x_1^i + \underbrace{h_t(a_2, \ldots, a_n)}_{\neq 0} \cdot x_1^t$$

is a polynomial in $\mathbb{F}_q[x_1]$ of degree $t < q$ with q distinct roots, which is a contradiction.

Hence, $f = 0$. □

Theorem 2.2 (Lagrange's Interpolation in several variables)**.** *For any arbitrary function $\tau : \mathbb{F}_q^n \to \mathbb{F}_q$ there exists a unique polynomial g of degree $< q$ in each variable with $g(c_1, \ldots, c_n) = \tau(c_1, \ldots, c_n)$ for all $(c_1, \ldots, c_n) \in \mathbb{F}_q^n$.*

Proof. To prove the theorem, we are going to check that the polynomial

$$g(x_1, \ldots, x_n) = \sum_{(a_1, \ldots, a_n) \in \mathbb{F}_q^n} \tau(a_1, \ldots, a_n)(1 - (x_1 - a_1)^{q-1}) \ldots (1 - (x_n - a_n)^{q-1})$$

has the desired properties.

First, we need to check that $g(\bar{c}) = \tau(\bar{c})$ for all $\bar{c} \in \mathbb{F}_q^n$. Evaluating $g(\bar{c})$, we obtain

$$g(\bar{c}) = \sum_{\bar{a} \in \mathbb{F}_q^n} \tau(\bar{a})(1 - (c_1 - a_1)^{q-1}) \ldots (1 - (c_n - a_n)^{q-1}).$$

If $\bar{c} \neq \bar{a}$, $\exists i \in \{1, \ldots, n\}$ such that $c_i \neq a_i$. Then, $1 - (c_i - a_i)^{q-1} = 1 - 1 = 0$, eliminating the summand associated to $\bar{a}$. By this line of reasoning, the only summand that remains is the one associated with $\bar{c}$, and thus

$$\begin{aligned} g(\bar{c}) &= \sum_{\bar{a} \in \mathbb{F}_q^n} \tau(\bar{a})(1 - (c_1 - a_1)^{q-1}) \ldots (1 - (c_n - a_n)^{q-1}) = \\ &= \tau(\bar{c})(1 - (c_1 - c_1)^{q-1}) \ldots (1 - (c_n - c_n)^{q-1}) = \tau(\bar{c}), \end{aligned}$$

which is what we wanted.

Also, the degree in each variable is at most $q-1$ by construction, so $\deg_{x_i}(g) < q$ for $i = 1, \ldots, n$.

Finally, suppose that there is another polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ such that $\deg_{x_i}(g) < q$ for $i = 1, \ldots, n$ and $f(\bar{c}) = \tau(\bar{c})$ for all $\bar{c} \in \mathbb{F}_q^n$. Then, the polynomial $f - g$ is such that $(f-g)(\bar{c}) = 0$ for all $\bar{c} \in \mathbb{F}_q^n$, and for each $i = 1, \ldots, n$,

$$\deg_{x_i}(f-g) \leq \max(\deg_{x_i}(f), \deg_{x_i}(g)) < q.$$

By Lemma 2.1, $f - g = 0$, hence $f = g$, proving the uniqueness of f and completing the proof. □

Throughout this dissertation, we identify all functions $\mathbb{F}_q^n \to \mathbb{F}_q$ with such polynomials and every polynomial will be of degree $< q$ in each variable, unless otherwise specified.

2.2. Permutation and Local Permutation Polynomials

Now we are going to introduce the main concepts of this chapter: Permutation and Local Permutation Polynomials.

Definition 2.3. A polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ is called a **permutation polynomial** (or PP) in n variables over $\mathbb{F}_q$ if the equation $f(x_1, \ldots, x_n) = a$ has q^{n-1} solutions in $\mathbb{F}_q^n$ for each $a \in \mathbb{F}_q$.

This is a generalisation of the concept of a permutation polynomial to several variables in the sense that, when $n = 1$, we recover the permutation polynomials studied in Chapter 1.

In the case $n > 1$ we cannot use the interpretation that a permutation polynomial $f(x_1, \ldots, x_n)$ over $\mathbb{F}_q$ induces a permutation of $\mathbb{F}_q^n$, because the associated mapping is not a mapping from $\mathbb{F}_q^n$ into itself, but $\mathbb{F}_q$.

Definition 2.4. A polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ is called a **local permutation polynomial** (or LPP) if for each $i \in \{1, \ldots, n\}$, $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a permutation polynomial in $\mathbb{F}_q[x_i]$, for all choices of $\bar{a}_i \in \mathbb{F}_q^{n-1}$.

Clearly if $n = 1$ both concepts are the same, that is, we are talking about univariate permutation polynomials. The situation changes if the number of variables is greater than one.

Proposition 2.5. *Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a local permutation polynomial over $\mathbb{F}_q$. Then, f is a permutation polynomial over $\mathbb{F}_q$.*

Proof. We need to check that for every $a \in \mathbb{F}_q$, the equation $f(x_1, \ldots, x_n) = a$ has q^{n-1} solutions in $\mathbb{F}_q^n$. Since f is an LPP, $f(b_1, \ldots, b_{n-1}, x_n) = a$ has a unique solution, b_n, for each $(b_1, \ldots, b_{n-1}) \in \mathbb{F}_q^{n-1}$, and there are q^{n-1} choices for $(b_1, \ldots, b_{n-1})$. Therefore, there are q^{n-1} solutions in $\mathbb{F}_q^n$ to the equation $f(x_1, \ldots, x_n) = a$, that is, f is a permutation polynomial. □

We have shown that all local permutation polynomials are permutation polynomials, but the opposite is not true in general, as illustrated in the next example.

Example 2.6. *Let us consider the polynomial* $f(\bar{x}) = x_1^{q-1} + x_2 \in \mathbb{F}_q[x_1, \ldots, x_n]$.

(*i*). *f is a permutation polynomial because for each $a \in \mathbb{F}_q$, the equation $f(\bar{x}) = a$ has q^{n-1} solutions: since $x_3, \ldots, x_n$ don't appear in the polynomial, all values for these variables can be a part of the solution, giving us q^{n-2} possible combinations; then, we can choose any value out of the q for x_1 and x_2 is determined by $x_2 = a - x_1^{q-1}$, making the total number of solutions $q \cdot q^{n-2} = q^{n-1}$.*

(*ii*). *For a fixed $(a_2, \ldots, a_n) \in \mathbb{F}_q^{n-1}$, the only two possible values that*

$$f(x_1, a_2, \ldots, a_n) = x_1^{q-1} + a_2$$

can take are a_2 (if $x_1 = 0$) and $1 + a_2$ (otherwise), and therefore it isn't a permutation polynomial in $\mathbb{F}_q[x_1]$. Thus, f isn't a local permutation polynomial.

The above counterexample suggests a couple properties of local permutation polynomials which are not shared with permutation polynomials.

Proposition 2.7. *Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a local permutation polynomial. Then,* $\deg_{x_i}(f) > 0$ *for all $i = 1, \ldots, n$.*

Proof. Suppose that $\exists i \in \{1, \ldots, n\}$ such that $\deg_{x_i}(f) = 0$. Then, the polynomial $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a constant, and therefore cannot be a permutation polynomial in $\mathbb{F}_q[x_i]$. This contradicts that f is an LPP. □

Proposition 2.8. *Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$, $n \geq 2$. f is a local permutation polynomial if and only if for any $c \in \mathbb{F}_q$ and any $i \in \{1, \ldots, n\}$, $f(x_1, \ldots, x_{i-1}, c, x_{i+1}, \ldots, x_n) = f|_{x_i=c}$ is a local permutation polynomial in $\mathbb{F}_q[\bar{x}_i]$.*

Proof. If $n = 2$, for $c \in \mathbb{F}_q$ $f(c, x_2)$ and $f(x_1, c)$ are permutation polynomials (and therefore LPPs) by definition.

If $n > 2$, since f is a local permutation polynomial, for each $j \in \{1, \ldots, n\}$, $j \neq i$ and for all choices of $(a_k)_{k \in I} \in \mathbb{F}_q^{n-2}$, where $I = \{1, \ldots, n\} \setminus \{i, j\}$, $f|_{x_i=c; x_k=a_k, k \in I}$ is a permutation polynomial in $\mathbb{F}_q[x_j]$, hence $f|_{x_i=c}$ is an LPP.

Conversely, suppose f isn't an LPP. Then, there must exist $i \in \{1, \ldots, n\}$ and $\bar{a}_i \in \mathbb{F}_q^{n-1}$ such that $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ isn't a permutation polynomial in $\mathbb{F}_q[x_i]$, and thus $f(a_1, x_2, \ldots, x_n)$ isn't an LPP in $\mathbb{F}_q[\bar{x}_1]$. □

We know how to check if a polynomial is a PP (respectively LPP), but how can we find these polynomials? In the following part of this section, we will delve into the study of several results that will allow us to produce PPs and LPPs from others.

This first one uses polynomial composition to obtain new (local) permutation polynomials.

Theorem 2.9. *Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a non constant polynomial.*

(i). Let $g(z) \in \mathbb{F}_q[z]$ be a permutation polynomial. Then, f is a (local) permutation polynomial if and only if $g(f(x_1, \ldots, x_n))$ is a (local) permutation polynomial.

(ii). Let $h_1(x_1), \ldots, h_n(x_n)$ be permutation polynomials. Then, f is a (local) permutation polynomial if and only if $f(h_1(x_1), \ldots, h_n(x_n))$ is a (local) permutation polynomial.

Proof. $\boxed{(i)}$ g is a permutation polynomial, therefore the equation $g(z) = a$ has a unique solution for each $a \in \mathbb{F}_q$, we will call it c_a. If f is a permutation polynomial, there are q^{n-1} solutions to $f(\bar{x}) = c_a$, thus there are q^{n-1} solutions to the equation $g(f(\bar{x})) = a$ for each $a \in \mathbb{F}_q$. That is, $g(f(\bar{x}))$ is a permutation polynomial.

Now, let us suppose that f is a local permutation polynomial. This implies that the equation $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n) = c_a$ has a unique solution for all $i \in \{1, \ldots, n\}$ and for all choices of $\bar{a}_i \in \mathbb{F}_q^{n-1}$. Then, that is the only solution to the equation $g(f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)) = a$, and consequently $g(f(\bar{x}))$ is an LPP.

Conversely, let $g(f(\bar{x}))$ be a (local) permutation polynomial. Since g is a PP, g^{-1} exists and is a PP as well. Then, $g^{-1}(g(f(\bar{x}))) = f(\bar{x})$ falls into the previous case and thus is a (local) permutation polynomial.

$\boxed{(ii)}$ For $i = 1, \ldots, n$, h_i is a permutation polynomial. Therefore, if we define the function $h(x_1, \ldots, x_n) := (h_1(x_1), \ldots, h_n(x_n))$, h is a bijective function. This implies that its inverse, h^{-1}, exists. Also, if seen as functions, h_i^{-1} exists and is a permutation polynomial for $i = 1, \ldots, n$.

Suppose that f is a permutation polynomial, we want to count the number of solutions to the equation $f(h(\bar{x})) = a$ for each $a \in \mathbb{F}_q$. $f(\bar{x}) = a$ has q^{n-1} solutions, each of those of the form $(c_1, \ldots, c_n)$. Then, $h^{-1}(c_1, \ldots, c_n)$ are the q^{n-1} solutions of the main equation, and consequently $f(h(\bar{x}))$ is a permutation polynomial.

Now, let us assume that f is an LPP, we want to see that, for each $b \in \mathbb{F}_q$, $f(h(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)) = b$ has a unique solution for each $i \in \{1, \ldots, n\}$ and for all choices of $\bar{a}_i \in \mathbb{F}_q^{n-1}$. We know that

$$f(h(a_1, \ldots, a_{i-1}, h_i^{-1}(x_i), a_{i+1}, \ldots, a_n)) =$$

$$= f(h_1(a_1), \ldots, h_{i-1}(a_{i-1}), x_i, h_{i+1}(a_{i+1}), \ldots, h_n(a_n)) = b$$

has a unique solution, c. Then, $h_i^{-1}(c)$ is the only solution to the main equation, making $f(h(\bar{x}))$ an LPP.

Conversely, let $f(h_1(x_1), \ldots, h_n(x_n))$ be a (local) permutation polynomial. Then, $f(h_1(h_1^{-1}(x_1)), \ldots, h_n(h_n^{-1}(x_n))) = f(x_1, \ldots, x_n)$ falls into the previous case and thus is a (local) permutation polynomial. □

The next result is Theorem 7.42 from [16]. In particular, it shows that we can produce a PP by summing a PP and another polynomial in different variables.

Theorem 2.10. *Suppose $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ is of the form*

$$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_m) + h(x_m, \ldots, x_n), \quad 1 \leq m < n.$$

If at least one of g and h is a permutation polynomial over $\mathbb{F}_q$, then f is a permutation polynomial over $\mathbb{F}_q$. If q is prime, then the converse holds as well.

For LPPs we have the following version of this theorem.

Theorem 2.11. *Suppose $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ is of the form*

$$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_m) + h(x_{m+1}, \ldots, x_n), \quad 1 \leq m < n.$$

Then f is an LPP if and only if g and h are local permutation polynomials.

Proof. Let us fix $i \in \{1, \ldots, n\}$. Suppose that $1 \leq i \leq m$. Then, when we evaluate

$$f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n) = g(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_m) + h(a_{m+1}, \ldots, a_n),$$

with $\bar{a}_i \in \mathbb{F}_q^{n-1}$, h becomes a constant. $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a permutation polynomial if and only if $g(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_m)$ is a permutation polynomial by Theorem 2.9 - *(i)*, taking $g(z) = z + h(a_{m+1}, \ldots, a_n)$.

Using the same argument for $m < i \leq n$, we get that $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a PP if and only if $h(a_{m+1}, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a PP.

Combining everything, we see that $f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a PP for any $i \in \{1, \ldots, n\}$ and all choices of $\bar{a}_i \in \mathbb{F}_q^{n-1}$ (that is, f is an LPP) if and only if $g(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a PP for any $i \in \{1, \ldots, m\}$ and all choices of $(a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_m) \in \mathbb{F}_q^{m-1}$ and $h(a_{m+1}, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n)$ is a PP for any $i \in \{m+1, \ldots, n\}$ and all choices of $(a_{m+1}, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n) \in \mathbb{F}_q^{m-n}$ (that is, g and h are LPPs). □

We conclude this section discussing the number of permutation polynomials and local permutation polynomials in $\mathbb{F}_q[x_1, \ldots, x_n]$. In the case of permutation polynomials there is a satisfying answer, given by the following theorem.

Theorem 2.12. *The number of permutation polynomials in $\mathbb{F}_q[x_1, \ldots, x_n]$ is*

$$N_n(q) = \frac{(q^n)!}{((q^{n-1})!)^q}.$$

Proof. Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a permutation polynomial, and $\mathbb{F}_q = \{c_0, \ldots, c_{q-1}\}$. Then, for each $c_i \in \mathbb{F}_q$, $i = 0, \ldots, q-1$, we define the set

$$A_i = \{(a_1, \ldots, a_n) \in \mathbb{F}_q^n : f(a_1, \ldots, a_n) = c_i\}.$$

It is clear to see that $\{A_i : i = 0, \ldots, q-1\}$ is a partition of $\mathbb{F}_q^n$, and since f is a permutation polynomial, $|A_i| = q^{n-1}$ for $i = 0, \ldots, q-1$.

To count the number of permutation polynomials, it suffices to count the number of partitions with these characteristics.

To build such a partition, we first need to choose q^{n-1} elements out of the q^n that $\mathbb{F}_q^n$ has, these will form A_0. Then, to construct A_1 we again pick q^{n-1} elements out of the $q^n - q^{n-1}$ remaining. We keep doing this until there are only q^{n-1} elements left, which will form A_{q-1}. The number of ways of doing this is

$$\binom{q^n}{q^{n-1}}\binom{q^n-q^{n-1}}{q^{n-1}}\cdots\binom{q^n-(q-2)q^{n-1}}{q^{n-1}}\binom{q^{n-1}}{q^{n-1}} = \prod_{k=0}^{q-1}\binom{q^n-k\cdot q^{n-1}}{q^{n-1}}$$

If we work out this expression, we get

$$\begin{aligned}\prod_{k=0}^{q-1}\binom{q^n-k\cdot q^{n-1}}{q^{n-1}} &= \prod_{k=0}^{q-1}\frac{(q^n-k\cdot q^{n-1})!}{(q^{n-1})!(q^n-(k+1)\cdot q^{n-1})!} = \\ &= \frac{1}{((q^{n-1})!)^q}\prod_{k=0}^{q-1}\frac{(q^n-k\cdot q^{n-1})!}{(q^n-(k+1)\cdot q^{n-1})!} = \\ &= \frac{1}{((q^{n-1})!)^q}\cdot\frac{(q^n-0\cdot q^{n-1})!}{(q^n-(q-1+1)\cdot q^{n-1})!} = \frac{(q^n)!}{((q^{n-1})!)^q}.\end{aligned}$$

Remark: Actually, to count the number of partitions of $\mathbb{F}_q^n$ with sets of cardinality q^{n-1} we would have to divide this number by $q!$, since it doesn't matter the order in which we build the sets. But, in this case, each A_i is associated with c_i, and changing the order of the sets would result in a different polynomial.

Hence, the number of permutation polynomials in $\mathbb{F}_q[x_1,\ldots,x_n]$ is

$$N_n(q) = \frac{(q^n)!}{((q^{n-1})!)^q}.$$

□

The situation changes dramatically for local permutation polynomials, as there isn't a concrete formula for this: the number of LPPs in $\mathbb{F}_q[x_1,\ldots,x_n]$ has to be calculated independently for every choice of q, n. We will make a couple of remarks on this open problem in Subsection 3.4.2.

However, it is easy to count the number of linear LPPs in $\mathbb{F}_q[x_1,\ldots,x_n]$. A linear polynomial $f \in \mathbb{F}_q[x_1,\ldots,x_n]$ is of the form

$$f(x_1,\ldots,x_n) = a_0 + \sum_{i=1}^{n} a_i x_i, \quad a_i \in \mathbb{F}_q,\ i = 0,\ldots,n.$$

By Theorem 2.7, all variables must appear in an LPP, so $a_i \neq 0$ for $i = 1,\ldots,n$. That leaves $q-1$ possibilities for each a_i, $i = 1,\ldots,n$ and q for a_0. All of these polynomials are LPPs by Theorems 2.11 and 1.11, giving us a total of $q(q-1)^n$ linear LPPs in $\mathbb{F}_q[x_1,\ldots,x_n]$.

2.3. Permutation and Local Permutation Polynomials of Maximum Degree

As part of our study of the different properties permutation and local permutation polynomials have, we will dedicate this section to finding non trivial bounds on their degree and determining whether or not those bounds are sharp.

There is a natural bound to the degree of all polynomials in $\mathbb{F}_q[x_1, \ldots, x_n]$.

Theorem 2.13. *Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a non constant polynomial. Then, f is of degree at most $n(q-1)$.*

Proof. All polynomials we consider have degree $< q$ in each variable thanks to the Lagrange Interpolation Theorem 2.2. This implies that the monomial of greatest degree that can appear in f is $x_1^{q-1} \ldots x_n^{q-1} = \prod_{i=1}^n x_i^{q-1}$, and therefore establishing a bound of $n(q-1)$ to its degree. □

For permutation polynomials, this bound is slightly smaller, as we will show in the next proposition.

Proposition 2.14. *Any permutation polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ has degree at most $n(q-1)-1$.*

Proof. By the Lagrange Interpolation Theorem 2.2, we have

$$f(x_1, \ldots, x_n) = \sum_{(c_1,\ldots,c_n)\in\mathbb{F}_q^n} f(c_1, \ldots, c_n)(1-(x_1-c_1)^{q-1}) \ldots (1-(x_n-c_n)^{q-1}).$$

The coefficient of the monomial $x_1^{q-1} \ldots x_n^{q-1} = \prod_{i=1}^n x_i^{q-1}$ in the above polynomial identity is

$$\sum_{(c_1,\ldots,c_n)\in\mathbb{F}_q^n} (-1)^n f(c_1, \ldots, c_n).$$

Since f is a permutation polynomial, for any $a \in \mathbb{F}_q$ the cardinality of the set

$$C_a = \{(c_1, \ldots, c_n) \in \mathbb{F}_q^n : f(c_1, \ldots, c_n) = a\}$$

is q^{n-1}. Moreover, the set $\Delta = \{C_a : a \in \mathbb{F}_q\}$ forms a partition of $\mathbb{F}_q^n$. Therefore, the coefficient previously mentioned is

$$\sum_{(c_1,\ldots,c_n)\in\mathbb{F}_q^n} (-1)^n f(c_1, \ldots, c_n) = (-1)^n \sum_{a\in\mathbb{F}_q} q^{n-1} \cdot a = 0.$$

Hence, f has degree at most $n(q-1)-1$. □

In the case of local permutation polynomials, we can refine that bound even more.

Theorem 2.15. *Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a non constant polynomial. If f is an LPP, then f is linear if $q = 2$ and $q = 3$, and has degree at most $n(q-2)$ otherwise.*

Proof. $\boxed{q = 2}$ This will be a proof by induction on n.

For $n = 1$, we are looking at polynomials in $\mathbb{F}_2[x]$. There are only two polynomials to consider, x and $x + 1$, since they are the only ones with degree < 2. Both of these are, in fact, linear, and PPs (which is the same as LPP for $n = 1$) by Theorem 1.11.

Now, suppose the claim is true for n and lwe shall prove it for $n + 1$. Let $f \in \mathbb{F}_2[x_1, \ldots, x_n, y]$ be an LPP. Since $\deg_y(f) < 2$, we can write it as

$$f(x_1, \ldots, x_n, y) = g(x_1, \ldots, x_n) \cdot y + h(x_1, \ldots, x_n), \quad g, h \in \mathbb{F}_2[x_1, \ldots, x_n].$$

Our goal is to show that $g(\bar{x}) = 1$.

Substituting y with 0 or 1, we obtain two LPPs in $\mathbb{F}_2[x_1, \ldots, x_n]$ by Proposition 2.8. By induction hypothesis, $f(\bar{x}, 0)$ and $f(\bar{x}, 1)$ are linear.

- $f(\bar{x}, 0) = h(\bar{x})$ is an LPP and linear, therefore $\deg_{x_i}(h) \leq 1$ for $i = 1, \ldots, n$.
- $f(\bar{x}, 1) = g(\bar{x}) + h(\bar{x})$ is linear. Knowing that h is also linear, this implies that g is linear, and therefore $\deg_{x_i}(h) \leq 1$ for $i = 1, \ldots, n$.

  Since $h(\bar{x})$ is an LPP, $\deg_{x_i}(h) \geq 1$ for $i = 1, \ldots, n$ by Proposition 2.7, hence $\deg_{x_i}(h) = 1$ for $i = 1, \ldots, n$. If there exists $i \in \{1, \ldots, n\}$ such that $\deg_{x_i}(g) = 1$, that variable would disappear in $g(\bar{x}) + h(\bar{x})$, contradicting Proposition 2.7 as it is an LPP. Thus, $\deg_{x_i}(g) = 0$ for all $i = 1, \ldots, n$, that is, g is a constant.

  If $g(\bar{x}) = 0$, that would mean that y doesn't appear in f, again contradicting Proposition 2.7. Hence, $g(\bar{x}) = 1$.

Combining everything together, we get that f is of the form

$$f(x_1, \ldots, x_n) = y + h(x_1, \ldots, x_n)$$

where h is a linear polynomial. Then, f is a linear polynomial.

$\boxed{q = 3}$ This will be a proof by induction on n.

For $n = 1$, we are looking at polynomials in $\mathbb{F}_3[x]$. The only polynomials we have to consider are those of degree < 3, that is, $ax^2 + bx + c$, $a, b, c \in \mathbb{F}_3$. Thanks to Theorem 1.11, we know that all linear polynomials ($a = 0$,$b \in \mathbb{F}_3^*$) are PPs. By Corollary 1.8, there is no PP of degree 2 ($a \neq 0$), and constants ($a, b = 0$) aren't PPs since they arent injective functions. Thus, all the PPs in $\mathbb{F}_3[x]$ are the linear polynomials.

Now, suppose the claim is true for n and we shall prove it for $n + 1$.

Let $h \in \mathbb{F}_3[x_1, \ldots, x_n, y]$ be an LPP. Since $\deg_y(h) < 3$, we can write it as

$$h(x_1, \ldots, x_n, y) = f_2(x_1, \ldots, x_n) \cdot y^2 + f_1(x_1, \ldots, x_n) \cdot y + f_0(x_1, \ldots, x_n),$$

where $f_i \in \mathbb{F}_2[x_1, \ldots, x_n]$, $i = 0, 1, 2$. Our goal is to show that $f_2(\bar{x}) = 0$ and $f_1(\bar{x})$ is a nonzero constant.

Substituting y with 0, 1 or 2, we obtain three LPPs in $\mathbb{F}_3[x_1, \ldots, x_n]$ by Proposition 2.8. By induction hypothesis, $h(\bar{x}, 0)$, $h(\bar{x}, 1)$ and $h(\bar{x}, 2)$ are linear.

- $h(\bar{x}, 0) = f_0(\bar{x})$ is an LPP and linear, therefore $\deg_{x_i}(h) \leq 1$ for $i = 1, \ldots, n$, or in other words, all variables x_i appear on f_0.
- $h(\bar{x}, 1) = f_2(\bar{x}) + f_1(\bar{x}) + f_0(\bar{x})$ is an LPP and linear.
- $h(\bar{x}, 2) = f_2(\bar{x}) + 2f_1(\bar{x}) + f_0(\bar{x})$ is an LPP and linear.

Since $f_0, h(\bar{x}, 1)$ and $h(\bar{x}, 2)$ are linear polynomials, $2f_0(\bar{x}) + h(\bar{x}, 1) = f_2(\bar{x}) + f_1(\bar{x})$ and $2f_0(\bar{x}) + h(\bar{x}, 2) = f_2(\bar{x}) + 2f_1(\bar{x})$ are linear as well. But at the same time, this implies that

$$(f_2(\bar{x}) + 2f_1(\bar{x})) - (f_2(\bar{x}) + f_1(\bar{x})) = f_1(\bar{x})$$

is also linear, and

$$(f_2(\bar{x}) + f_1(\bar{x})) - f_1(\bar{x}) = f_2(\bar{x})$$

is linear too. To summarise, f_i is a linear polynomial for $i = 0, 1, 2$.

f_2 is a linear polynomial. If $f_2(\bar{x}) \neq 0$, there exists $\bar{a} \in \mathbb{F}_q^n$ such that $f_2(\bar{a}) \neq 0$. As h is an LPP, $h(\bar{a}, y)$ is a PP in $\mathbb{F}_3[y]$, but

$$h(\bar{a}, y) = \underbrace{f_2(\bar{a})}_{\neq 0} \cdot y^2 + f_1(\bar{a}) \cdot y + f_0(\bar{a}),$$

which isn't a linear polynomial, and thus contradicting the induction hypothesis. Therefore, $f_2(\bar{x}) = 0$.

We also know that f_0 and f_1 are linear, and all variables appear on f_0. So, we can write

$$f_1(\bar{x}) = a_0 + \sum_{i=1}^{n} a_i x_i, \quad f_0(\bar{x}) = b_0 + \sum_{i=1}^{n} b_i x_i; \quad a_i, b_i \in \mathbb{F}_3, i = 0, \ldots, n,\ b_i \neq 0 \text{ if } i \neq 0.$$

This leaves us with

$$h(\bar{x}, 1) = f_1(\bar{x}) + f_0(\bar{x}) = (a_0 + b_0) + \sum_{i=1}^{n} (a_i + b_i) x_i,$$
$$h(\bar{x}, 2) = 2f_1(\bar{x}) + f_0(\bar{x}) = (2a_0 + b_0) + \sum_{i=1}^{n} (2a_i + b_i) x_i.$$

Both of these polynomials are LPPs, and therefore all variables appear in them, that is, $a_i + b_i$ and $2a_i + b_i$ are nonzero for $i = 1, \ldots, n$. Let $i \in \{1, \ldots, n\}$. b_i cannot be zero. Then,

- If $b_i = 1$, $a_i + b_i = a_i + 1$ and $2a_i + b_i = 2a_i + 1$. Necessarily $a_i = 0$ for these to be nonzero.
- If $b_i = 2$, $a_i + b_i = a_i + 2$ and $2a_i + b_i = 2a_i + 2$. Necessarily $a_i = 0$ for these to be nonzero.

Hence, $a_i = 0$ for $i = 1, \ldots, n$, which implies $f_1(\bar{x}) = a_0$, that is, f_1 is a constant. If $f_1(\bar{x}) = 0$, that would mean that y doesn't appear in h, contradicting Proposition 2.7. Therefore, $f_1(\bar{x}) = 1$.

Combining everything together, we get that h is of the form

$$h(x_1, \ldots, x_n) = y + f_0(x_1, \ldots, x_n)$$

where f_0 is a linear polynomial. Then, h is a linear polynomial.

$\boxed{q > 3}$ We know that $\deg_{x_i} < q$ $i = 1, \ldots, n$. Here we will show that $\deg_{x_i}(f) < q - 1$ for $i = 1, \ldots, n$. It suffices to prove it for $i = 1$, as the other cases are analogous.

Since $\deg_{x_1}(f) < q$, we can write

$$f = M_{q-1}x_1^{q-1} + M_{q-2}x_1^{q-2} + \cdots + M_1 x_1 + M_0, \quad M_i \in \mathbb{F}_q[x_2, \ldots, x_n].$$

Suppose that M_{q-1} is a nonzero polynomial. Then, there exists $(a_2, \ldots, a_n) \in \mathbb{F}_q^{n-1}$ such that $M_{q-1}(a_2, \ldots, a_n) \neq 0$. As f is an LPP, $f(x_1, a_2, \ldots, a_n)$ is a PP in $\mathbb{F}_q[x_1]$, but it is of degree $q - 1$, contradicting Corollary 1.8. Hence, $M_{q-1}(x_2, \ldots, x_n) = 0$ and $\deg_{x_1}(f) < q - 1$.

The same reasoning can be applied for the other variables, so $\deg_{x_i}(f) < q - 1$ for $i = 1, \ldots, n$. This implies that the monomial of greatest degree that can appear in f is $\prod_{i=1}^{n} x_i^{q-2}$, and therefore establishing a bound of $n(q-2)$ to its degree.

Remark: This part of the proof works for $q > 2$, but, as we have seen, the bound obtained can be refined. □

Once these bounds have been established, we wonder whether or not there are polynomials that actually reach them. To study this, we can rely on the results proven in the previous sections for providing useful examples of PPs and LPPs and tools to construct them from others.

This question has a fulfilling resolution in the case of permutation polynomials.

Theorem 2.16. *There exists a permutation polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ of maximum degree $n(q-1) - 1$.*

Proof. Let $q = p^r$. Consider the polynomial

$$g(x) = x + \sum_{k=0}^{q-2} x^k \in \mathbb{F}_q[x],$$

which is a PP that represents the permutation $(0, 1)$ as seen in Lemma 1.12. We will construct the polynomial

$$h_n(x_1, \ldots, x_n) = x_1^{q-1} \ldots x_{n-1}^{q-1}(g(x_n) - x_n) + x_n.$$

This polynomial has degree $n(q-1) - 1$, since

$$h_n(\bar{x}) = x_1^{q-1} \ldots x_{n-1}^{q-1} \left(x_n + \sum_{k=0}^{q-2} x_n^k - x_n \right) + x_n = x_1^{q-1} \ldots x_{n-1}^{q-1} \cdot x_n^{q-2} + \begin{matrix}\text{smaller} \\ \text{degree terms}\end{matrix}.$$

Let $a \in \mathbb{F}_q$. For any choice of $(c_1, \ldots, c_{n-1}) \in \mathbb{F}_q^{n-1}$, we are going to show that there exists a unique $c_n \in \mathbb{F}_q$ such that $h(c_1, \ldots, c_n) = a$. We have

$$h_n(c_1, \ldots, c_{n-1}, x_n) = c_1^{q-1} \ldots c_{n-1}^{q-1}(g(x_n) - x_n) + x_n.$$

There are two possibilities:

- If $c_i \neq 0$ for all $i = 1, \ldots, n-1$, $h_n(c_1, \ldots, c_{n-1}, x_n) = (g(x_n) - x_n) + x_n = g(x_n)$, which is a univariate permutation polynomial.
- If there exists $i \in \{1, \ldots, n\}$ such that $c_i = 0$, then $h_n(c_1, \ldots, c_{n-1}, x_n) = x_n$, which is a univariate permutation polynomial, since it is linear.

We have shown that $h_n(c_1, \ldots, c_{n-1}, x_n)$ is a permutation polynomial in $\mathbb{F}_q[x_n]$ for all choices of $(c_1, \ldots, c_{n-1}) \in \mathbb{F}_q^{n-1}$, therefore for each $a \in \mathbb{F}_q$ there exists a unique $c_n \in \mathbb{F}_q$ such that $h_n(c_1, \ldots, c_{n-1}, c_n) = a$. This implies that there are q^{n-1} solutions of $h_n(\bar{x}) = a$ for each $a \in \mathbb{F}_q$, or in other words, h_n is a PP in $\mathbb{F}_q[x_1, \ldots, x_n]$. □

Now we will study local permutation polynomials of maximum degree in $\mathbb{F}_q[x_1, \ldots, x_n]$, separating the case where $\mathbb{F}_q$ is a field of characteristic 2 and 3 from the rest.

Let us start by analysing the latter case. To get to the final result, we first need to prove a series of preliminary statements.

Theorem 2.17. *Let $p \geq 5$ be a prime number and $n < p-1$ a positive integer. There exists a local permutation polynomial in $\mathbb{F}_q[x_1, \ldots, x_n]$ defined over $\mathbb{F}_q$, of maximum degree for every $q = p^r$, $r \geq 1$.*

Proof. According to Theorem 1.11, x_i is a permutation polynomial for all $i = 1, \ldots, n$. Therefore, $S(\bar{x}) = x_1 + \cdots + x_n$ is a PP by Theorem 2.11. We will again consider the polynomial

$$g(x) = x + \sum_{k=0}^{q-2} x^k \in \mathbb{F}_q[x],$$

which is a PP that represents the permutation $(0, 1)$ as seen in Lemma 1.12. Then, we can construct the polynomial

$$f(x_1, \ldots, x_n) = g\left(S(x_1^{q-2}, \ldots, x_n^{q-2})\right) = x_1^{q-2} + \cdots + x_n^{q-2} + \sum_{k=0}^{q-2}(x_1^{q-2} + \cdots + x_n^{q-2})^k.$$

Since $\gcd(q-2, q-1) = 1$, $h_i(x_i) = x_i^{q-2}$ is a PP for all $i = 1, \ldots, n$ by Theorem 1.11. Then, $S(x_1^{q-2}, \ldots, x_n^{q-2})$ and consequentely f are LPPs by Theorem 2.9.

Remember that $\deg_{x_i}(f) \leq q-2$. Thus, f is of degree $n(q-2)$ if and only if f has the monomial $x_1^{q-2} \ldots x_n^{q-2}$ with nonzero coefficient, which happens if and only if

$$g(S(\bar{x})) = S(\bar{x}) + \sum_{k=0}^{q-2}(S(\bar{x}))^k$$

has the monomial $x_1 \dots x_n$ with nonzero coefficient, since if $k \in \{0, \dots, q-2\}$, $x^{k(q-2)} = x^{q-2}$ in $\mathbb{F}_q$ if and only if $k = 1$.

Now, for any $0 \leq k \leq q-2$, the monomials of S^k are of the form $x_1^{i_1} \dots x_n^{i_n}$, where $i_1 + \cdots + i_n = k$. Then, the only one of these where $x_1 \dots x_n$ appears is S^n, with coefficient $n!$. Therefore,

$$f = n! x_1^{q-2} \dots x_n^{q-2} + \text{ terms in less variables},$$

so f is an LPP with degree $n(q-2)$. □

Lemma 2.18. *If there is an LPP $f \in \mathbb{F}_q[x_1, \dots, x_n]$ of maximum degree, then there is an LPP of maximum degree for any $m \leq n$.*

Proof. It suffices to prove that there is an LPP of maximum degree in $\mathbb{F}_q[x_1, \dots, x_{n-1}]$.

Let $f \in \mathbb{F}_q[x_1, \dots, x_n]$ be an LPP of maximum degree. Then, f is of the form

$$f(x_1, \dots, x_n) = a x_1^{q-2} \dots x_n^{q-2} + P$$

for some $a \in \mathbb{F}_q$ and P of smaller degree. By Proposition 2.8, $f(x_1, \dots, x_{n-1}, \alpha) = a\alpha^{q-2} x_1^{q-2} \dots x_{n-1}^{q-2} + P|_{x_n=\alpha}$ is an LPP in $\mathbb{F}_q[x_1, \dots, x_{n-1}]$ for any $\alpha \in \mathbb{F}_q$.

So, we choose an $\alpha \in \mathbb{F}_q^*$. Then, we have the maximum degree LPP we were looking for unless $\deg(f(x_1, \dots, x_{n-1}, \alpha)) < (n-1)(q-2)$. That can only happen if $P|_{x_n=\alpha}$ has the monomial $-a\alpha^{q-2} x_1^{q-2} \dots x_{n-1}^{q-2}$. But then, $f(x_1, \dots, x_{n-1}, \beta)$ is an LPP that has the monomial $(a\beta^{q-2} - a\alpha^{q-2}) x_1^{q-2} \dots x_n^{q-2} \neq 0$, with $\beta \in \mathbb{F}_q$.

This way, we have found an LPP of maximum degree in $\mathbb{F}_q[x_1, \dots, x_{n-1}]$. □

Theorem 2.19. *Let $n = st$ with $s < p$, $\gcd(s, q-1) = 1$. Suppose there is an LPP over $\mathbb{F}_q[x_1, \dots, x_t]$ of maximum degree defined over $\mathbb{F}_p$. Then, there is an LPP over $\mathbb{F}_q[x_1, \dots, x_n]$ of maximum degree defined over $\mathbb{F}_p$.*

Proof. Let $F_t \in \mathbb{F}_q[x_1, \dots, x_t]$ be an LPP of maximum degree. Then,

$$F_t = a x_1^{q-2} \dots x_t^{q-2} + P_t,$$

where $a \in \mathbb{F}_q$, $\deg(P_t) < t(q-2)$ and $\deg_{x_i}(P_t) \leq q-2$ for $i = 1, \dots, t$. Consider the polynomials $S(y_1, \dots, y_s) = y_1 + \cdots + y_s$ and $h(z) = z^s$, which are LPPs. We define the polynomial

$$\begin{aligned} F &= \big(F_t(x_1, \dots, x_t) + F_t(x_{t+1}, \dots, x_{2t}) + \cdots + F_t(x_{(s-1)t+1}, \dots, x_{st})\big)^s = \\ &= h(S(F_t(x_1, \dots, x_t), \dots, F_t(x_{(s-1)t+1}, \dots, x_{st}))), \end{aligned}$$

an LPP by Theorem 2.9. Then,

$$F = \left(a x_1^{q-2} \dots x_t^{q-2} + \cdots + a x_{(s-1)t+1}^{q-2} \dots x_{st}^{q-2} + P\right)^s,$$

where $\deg(P) < t(q-2)$ and $\deg_{x_i}(P) \leq q-2$ for $i = 1, \dots, t$. Hence,

$$F(x_1, \dots, x_n) = c^s s! x_1^{q-2} \dots x_n^{q-2} + G(x_1, \dots, x_n),$$

where $\deg(G) < n(q-2)$ and $\deg_{x_i}(P) \leq q-2$ for $i = 1, \dots, t$.

Thus, F is an LPP of maximum degree in $\mathbb{F}_q[x_1, \dots, x_n]$. □

With all of these results proven, we can finally show what we were aiming for.

Corollary 2.20. *Let $n, p \in \mathbb{N}$, with $p \geq 5$ a prime number and $q = p^r$. If there exists $1 < b < p-1$ such that $\gcd(b, q-1) = 1$, then there is a local permutation polynomial over $\mathbb{F}_q[x_1, \ldots, x_n]$ of maximum degree, $n(q-2)$.*

Proof. By Theorem 2.17 there exists an LPP of maximum degree in $\mathbb{F}_q[x_1, \ldots, x_b]$ defined over $\mathbb{F}_q$. Let $k \in \mathbb{N}$ be such that $n \leq b^k$. Considering $s = b$, we can apply recursively Theorem 2.19 to get an LPP in $\mathbb{F}_q[x_1, \ldots, x_{b^k}]$. Then, by Lemma 2.18, there is an LPP of maximum degree in n variables over $\mathbb{F}_q$. □

Observe that such b doesn't always exist. For example, if $p = 5$ and $q = 5^2 = 25$, the only choices for b are 2 and 3, but both divide $q - 1 = 24$. However, there are infinitely many q for which Corollary 2.20 applies if we restrict our search to b's such that $\gcd(b, p-1) = 1$.

Lemma 2.21. *For any prime number p and integer $\gcd(b, p-1) = 1$ there exist infinitely many $r \geq 1$ such that $\gcd(b, p^r - 1) = 1$.*

Proof. Let b be an integer. If $p \,|\, b$, we can write $b = p^a l$ for $p \nmid l$, which implies $\gcd(b, p^r - 1) = \gcd(l, p^r - 1)$. Thus, we can suppose that $p \nmid b$, that is, $\gcd(p, b) = 1$.

Now, we define $r_m = m\varphi(b)$ for any $m \in \mathbb{N}$. By the Fermat–Euler Theorem we have $p^{r_m} \equiv (p^{\varphi(b)})^m \equiv 1^m \equiv 1 \mod b$ and hence

$$p^{r_m+1} - 1 \equiv p^{r_m+1} - p^{r_m} + p^{r_m} - 1 \equiv (p^{r_m+1} - p^{r_m}) + (1-1) \equiv p^{r_m}(p-1) \mod b.$$

It is known that $\gcd(\alpha, \beta) = \gcd(\alpha, \beta \mod \alpha)$ for $\alpha, \beta \in \mathbb{Z}$. Therefore,

$$\gcd(b, p^{r_m+1} - 1) = \gcd(b, p^{r_m}(p-1)) = \gcd(b, p-1) = 1.$$

There is an infinite number of these r_m, thus completing the proof. □

Now we will briefly discuss the case where $\mathbb{F}_q$ is a field of characteristic $p = 2$ or $p = 3$, $q = p^r$, $r \in \mathbb{N}$.

When $r = 1$ ($q = 2, 3$) all LPPs are linear, as seen in Theorem 2.15, which there are $q(q-1)^n$ of in $\mathbb{F}_q[x_1, \ldots, x_n]$. For example, the polynomial $S(x_1, \ldots, x_n) = x_1 + \cdots + x_n$ which appears in the proof of Theorem 2.17 is a linear LPP.

When $r > 1$, in [9] and [10] there are results showing LPPs of maximum degree in two and three variables over $\mathbb{F}_q$.

In fact, in [10] there is a general proof of existence of local permutation polynomials of maximum degree in $\mathbb{F}_q[x_1, \ldots, x_n]$ for any $q > 3$ and any $n \in \mathbb{N}$.

CHAPTER 3
Bivariate Local Permutation Polynomials

In this chapter we will focus on local permutation polynomials in $\mathbb{F}_q[x, y]$. We will provide a family of bivariate LPPs, the so called e-Klenian polynomials, based on a class of symmetric subgroups without fixed points. Because of their relation with LPPs, we will also talk about Latin squares and their generalisation to higher dimensions, Latin hypercubes, as well as study the concept of orthogonality for both polynomials and Latin squares.

3.1. Permutation Polynomial Tuples

Here we are going to show an alternative representation of bivariate local permutation polynomials: q-tuples of permutations of $\mathbb{F}_q$, thus translating their study to discussing these tuples.

Let Σ_q be the permutation group with q elements and $\mathbb{F}_q = \{c_0, \ldots, c_{q-1}\}$ the field with $q = p^r$ elements.

Lemma 3.1. *There is a bijective map between the set of local permutation polynomials $f \in \mathbb{F}_q[x, y]$ and the set of q-tuples $\underline{\beta}_f = (\beta_0, \ldots, \beta_{q-1})$ such that $\beta_i \in \Sigma_q$ $(i = 0, \ldots, q-1)$ and for $i \neq j$, $\beta_i^{-1}\beta_j$ has no fixed points.*

Proof. First we will show how to associate a q-tuple of permutations to a given LPP.

Let $f \in \mathbb{F}_q[x, y]$ be a local permutation polynomial. We remember the sets we defined for each $i = 0, \ldots, q-1$ in the proof of Theorem 2.12,

$$A_i = \{(a, b) \in \mathbb{F}_q^2 : f(a, b) = c_i\}.$$

Since f is an LPP, it is also a PP, and thus $\{A_i : i = 0, \ldots, q-1\}$ is a partition of $\mathbb{F}_q^2$ and $|A_i| = q$.

If we take into account that f is an LPP, we notice that $f(a, y)$ is a permutation polynomial in $\mathbb{F}_q[y]$ for all $a \in \mathbb{F}_q$. This means that for each $i = 0, \ldots, q-1$, the

equation $f(a,y) = c_i$ has a unique solution $b_i \in \mathbb{F}_q$. Therefore, for each $i = 0, \ldots, q-1$ we can define a permutation $\beta_i \in \Sigma_q$ such that $\beta_i(a) = b_i$ for each $a \in \mathbb{F}_q$. Then,

$$A_i = \{(a, \beta_i(a)) : a \in \mathbb{F}_q\}.$$

Since $A_i \cap A_j = \emptyset$ when $i \neq j$, $\beta_i(a) \neq \beta_j(a)$ for all $a \in \mathbb{F}_q$. In other words, the permutation $\beta_i^{-1}\beta_j$ has no fixed points when $i \neq j$.

Hence, we have associated the LPP f with the q-tuple $\underline{\beta}_f = (\beta_0, \ldots, \beta_{q-1})$.

Conversely, let $\underline{\beta} = (\beta_0, \ldots, \beta_{q-1})$ be a q-tuple such that $\beta_i \in \Sigma_q$ $(i = 0, \ldots, q-1)$ and for $i \neq j$, $\beta_i^{-1}\beta_j$ has no fixed points. Then, for each $i = 0, \ldots, q-1$ we can construct the set

$$A_i = \{(a, \beta_i(a)) : a \in \mathbb{F}_q\}.$$

Note that $A_i \cap A_j = \emptyset$ when $i \neq j$ because $\beta_i^{-1}\beta_j$ has no fixed points, and $|A_i| = q$ for $i = 0, \ldots, q-1$.

Then, thanks to the Lagrange Interpolation Theorem 2.2 we can define a polynomial f_β such that

$$f_\beta(a, \beta_i(a)) = c_i, \quad a \in \mathbb{F}_q, i = 0, \ldots, q-1.$$

As the A_i are disjoint, f_β is well defined. Also, $|A_i| = q$, making f_β a permutation polynomial. All that is left to check is if f_β is an LPP.

Let $a \in \mathbb{F}_q$. The equation $f_\beta(a,y) = c_i$ has a unique solution, $\beta_i(a)$, for each $c_i \in \mathbb{F}_q$. On the other hand, $f_\beta(x,a) = c_i$ also has a unique solution, $\beta_i^{-1}(a)$, for each $c_i \in \mathbb{F}_q$. Thus, f_β is an LPP.

Hence, we have associated the q-tuple $\underline{\beta}$ with the LPP f_β. □

We denote by $\underline{\beta}_f$ the q-tuple associated with the LPP f.

This relation leads us to the next definition.

Definition 3.2. We say that $(\beta_0, \ldots, \beta_{q-1}) \in \Sigma_q^q$ is a **permutation polynomial tuple** is it satisfies that $\beta_i^{-1}\beta_j$ has no fixed points for $i, j = 0, \ldots, q-1$, $i \neq j$.

Similar to the remark made at the end of the proof of Theorem 2.12, we note that changing the order of the A_i's would result in a different permutation polynomial. In particular, if f is the (local) permutation polynomial associated to the A_i's and $\sigma \in \Sigma_q$, if we define the polynomial h as

$$h(a,b) = c_i, \quad (a,b) \in A_{\sigma(i)}, \quad i = 0, \ldots, q-1,$$

then h is a (local) permutation polynomial and $h(x,y) = g(f(x,y))$, where $g(z) \in \mathbb{F}_q[z]$ is the permutation polynomial associated to the permutation σ.

If f is an LPP associated with the q-tuple of permutations $\underline{\beta}_f = (\beta_0, \ldots, \beta_{q-1})$, the polynomial h that results from this permutation is the LPP associated with the q-tuple $\underline{\beta}_h = (\beta_{\sigma(0)}, \ldots, \beta_{\sigma(q-1)})$.

Using this method, we can obtain up to $q!$ permutation polynomials from one permutation polynomial tuple.

However, there are many other LPPs we can construct, as shown in the next result.

Proposition 3.3. *Let $\Omega = (\beta_0, \ldots, \beta_{q-1}) \in \Sigma_q^q$ be a permutation polynomial tuple. Let $\sigma, \delta \in \Sigma_q$. Then, $\sigma\Omega\delta = (\sigma\beta_0\delta, \ldots, \sigma\beta_{q-1}\delta) \in \Sigma_q^q$ is also a permutation polynomial tuple.*

Proof. Suppose that $\sigma\Omega\delta$ isn't a permutation polynomial tuple. Then, there exist $i, j \in \{0, \ldots, q-1\}$, $i \neq j$, and a $c \in \mathbb{F}_q$ such that c is a fixed point of $(\sigma\beta_i\delta)^{-1}(\sigma\beta_j\delta)$, that is, $(\sigma\beta_i\delta)^{-1}(\sigma\beta_j\delta)(c) = c$. But $(\sigma\beta_i\delta)^{-1}(\sigma\beta_j\delta) = \delta^{-1}\beta_i^{-1}\beta_j\delta$, and thus

$$\delta^{-1}\beta_i^{-1}\beta_j\delta(c) = c \quad \Longrightarrow \quad \beta_i^{-1}\beta_j(\delta(c)) = \delta(c).$$

$\delta(c)$ is a fixed point of $\beta_i^{-1}\beta_j$ and $i \neq j$, which is a contradiction with the fact that Ω is a permutation polynomial tuple.

Hence, $\sigma\Omega\delta$ is a permutation polynomial tuple. □

This proposition motivates the following concept.

Definition 3.4. Two permutation polynomial tuples Ω and Γ are **similar** if there exist $\sigma, \delta \in \Sigma_q$ such that $\sigma\Omega\delta = \Gamma$.

Analogously, we say that two local permutation polynomials f and g are **similar** if the corresponding permutation polynomial tuples $\underline{\beta}_f$ and $\underline{\beta}_g$ are similar.

Proposition 3.5. *The relation described in Definition 3.4 is an equivalence relation defined in the set of local permutation polynomials.*

Proof. Let $f, g, h \in \mathbb{F}_q[x, y]$ be local permutation polynomials.

- **Reflexivity:** f is similar to f.
  $id^{-1}\underline{\beta}_f id = \underline{\beta}_f$, where $id \in \Sigma_q$ is the identity.
- **Symmetry:** f is similar to g if and only if g is similar to f.
  Let $\sigma, \delta \in \Sigma_q$. Then, $\sigma\underline{\beta}_f\delta = \underline{\beta}_g \iff \underline{\beta}_f = \sigma^{-1}\underline{\beta}_g\delta^{-1}$.
- **Transitivity:** If f is similar to g and g is similar to h, then f is similar to h.
  Let $\sigma, \delta, \hat{\sigma}, \hat{\delta} \in \Sigma_q$. Then,

$$\begin{cases} \sigma\underline{\beta}_f\delta = \underline{\beta}_g \\ \hat{\sigma}\underline{\beta}_g\hat{\delta} = \underline{\beta}_h \end{cases} \quad \Longrightarrow \quad (\hat{\sigma}\sigma)\underline{\beta}_f(\delta\hat{\delta}) = \underline{\beta}_h.$$

□

Every class of permutation polynomial tuples has a representative containing the identity: the permutation polynomial tuple $\underline{\beta}_f = (\beta_0, \ldots, \beta_{q-1})$ is similar to

$$\beta_i^{-1}\underline{\beta}_f = (\beta_i^{-1}\beta_0, \ldots, \beta_i^{-1}\beta_{i-1}, id, \beta_i^{-1}\beta_{i+1}, \ldots, \beta_i^{-1}\beta_{q-1}), \quad i = 0, \ldots, q-1.$$

Note that $\beta_i^{-1}\beta_j \neq id$ when $i \neq j$, as they don't have fixed points by definition.

If needed, we will use one of these representatives.

3.2. e-Klenian Polynomials

In this section we will present a new class of bivariate local permutation polynomials, e-Klenian polynomials, based on a particular family of permutation polynomial tuples.

Definition 3.6. Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a local permutation polynomial and $\underline{\beta}_f = (\beta_0, \ldots, \beta_{q-1})$. f is a **permutation group polynomial** if $\{\beta_0, \ldots, \beta_{q-1}\}$ is a subgroup of Σ_q. We denote this subgroup by $G_{\underline{\beta}_f}$.

Proposition 3.7. *Let $G = \{\beta_0, \ldots, \beta_{q-1}\}$ be a subgroup of Σ_q. The elements of G can form a permutation polynomial tuple if and only if none of them have fixed points, apart from the identity.*

Proof. $\boxed{\Longrightarrow}$ $G = \{\beta_0, \ldots, \beta_{q-1}\}$ is a subgroup of Σ_q, therefore $id \in G$. Without loss of generality, we can suppose that $\beta_0 = id$.

$(\beta_0, \ldots, \beta_{q-1})$ form a permutation polynomial tuple, thus $\beta_i^{-1}\beta_j$ have no fixed points when $i \neq j$. In particular, $\beta_0^{-1}\beta_j = id^{-1}\beta_j = \beta_j$ has no fixed points for $j \neq 0$.

$\boxed{\Longleftarrow}$ Since G is a subgroup of Σ_q, $\beta_i^{-1}\beta_j = \beta_k \in G$ for some k. If $i \neq j$, $\beta_k \neq id$ and thus has no fixed points. Hence $\beta_i^{-1}\beta_j$ has no fixed points when $i \neq j$.

Therefore, the elements of G can form a permutation polynomial tuple. □

Proposition 3.8. *Let $C \in \Sigma_q$ be a cycle of maximum length q. Then, the cycle subgroup $\langle C \rangle$ generated by C is a group without fixed points.*

Proof. As C is a cycle of length q, $\langle C \rangle = \{C^s : s = 1, \ldots, q\}$ has q distinct elements and we can write $C = (a_0, \ldots, a_{q-1})$, where $a_i \in \mathbb{F}_q$ for $i = 0, \ldots, q-1$. For each $c \in \mathbb{F}_q$ there exists $i \in \{0, \ldots, q-1\}$ such that $c = a_i$, and

$$C^s(c) = C^s(a_i) = a_{i+s \mod q}, \quad s = 1, \ldots, q.$$

$C^q = id$, and for $s \neq q$ we see that C^s has no fixed points, as $i \not\equiv i + s \mod q$. □

Combining these last two propositions, we gather that the elements of $\langle C \rangle$ can form a permutation polynomial tuple when C is a cycle of maximum length q. This is gives us yet another easy way to find local permutation polynomials in $\mathbb{F}_q[x, y]$.

However, these aren't the only subgroups without fixed points. We will now work to describe a family of such subgroups.

We denote by $|C|$ the length of a cycle $C \in \Sigma_q$.

Lemma 3.9. *Let $q = p^r$, $G \subset \Sigma_q$ be a nontrivial subgroup without fixed points, and $\alpha \in G$. Then, there exists $0 < e \leq r$ such that for $t = p^e$ and $k = p^{r-e}$ we have $\alpha = C_1 \dots C_k$ where $|C_i| = t$ for all $i = 1, \dots, k$.*

Proof. If $\alpha = id$, then $e = 0$ $(t = 1, k = p^r)$.

Let $\alpha \neq id$. Suppose that $\alpha = C_1 \dots C_k$ is the representation of α as a product of disjoint cycles.

If $|C_1| = t_1 < t_2 = |C_2|$, then

$$\alpha^{t_1} = (C_1 \dots C_k)^{t_1} = \underbrace{C_1^{t_1}}_{=id} \underbrace{C_2^{t_1}}_{\neq id} \dots C_k^{t_1}.$$

α^{t_1} is an element of G that isn't the identity since $C_2^{t_1} \neq id$, but fixes all the elements in C_1. This contradicts that G is a subgroup without fixed points, which implies that $|C_1| \geq |C_2|$.

Using the same reasoning for every possible pairing results in $|C_1| = |C_2| = \dots = |C_k| = t$, that is, all the cycles have the same length t, and therefore α is of order t.

By the Lagrange Theorem from Group Theory, we know that the order of α has to divide $|\Sigma_q| = q = p^r$. Thus, $t = p^e$ for some $0 \leq e \leq r$. Since α isn't the identity, $0 < e \leq r$.

Also, $\alpha \in G$. G is a group without fixed points, hence α has no fixed points. This means that each element of $\mathbb{F}_q$ appears in exactly one of the C_i, $i = 1, \dots, k$, as they are disjoint. Then,

$$p^r = q = \sum_{i=1}^{k} |C_i| = \sum_{i=1}^{k} t = k \cdot t \quad \Longrightarrow \quad k = \frac{p^r}{t} = \frac{p^r}{p^e} = p^{r-e}.$$

□

As stated earlier, we want to find new subgroups without fixed points. By Lemma 3.9, the permutations in these subgroups will be products of cycles of the same length.

Lemma 3.10. *Let $q = p^r$, $F_q = \{c_0, \dots, c_{q-1}\}$. Let $1 \leq e \leq r$, $l = p^e$, $t = \frac{q}{l}$. We define the permutations $\alpha = C_{0,\alpha} \dots C_{t-1,\alpha}$ and $\beta = C_{0,\beta} \dots C_{l-1,\beta}$, where*

$$C_{i,\alpha} = (c_{il}, c_{1+il}, \dots, c_{(l-1)+il}) = \{c_{j+il} : j = 0, \dots, l-1\}, \quad i = 0, \dots, t-1,$$
$$C_{j,\beta} = (c_j, c_{j+l}, \dots, c_{j+(t-1)l}) = \{c_{j+il} : j = 0, \dots, l-1\}, \quad i = 0, \dots, l-1.$$

Then for any $0 \leq a \leq l-1$ and $0 \leq b \leq t-1$, $\beta^b \alpha^a$ has no fixed points and $\alpha^a \beta^b = \beta^b \alpha^a$.

Proof. We write the elements of $\mathbb{F}_q$ as c_{j+il} for some $0 \leq j \leq l-1$ and $0 \leq i \leq t-1$.

In order to see the action of $\beta^b \alpha^a$, we will first study $C_{i,\alpha}^a$ and $C_{j,\beta}^b$ for each $i = 0, \dots, t-1$, $j = 0, \dots, l-1$.

Fixed $i \in \{0, \ldots, t-1\}$, the cycle $C_{i,\alpha}$ leaves fixed all $c_{J+Il} \in \mathbb{F}_q$ where $I \neq i$, thus $C_{i,\alpha}^a$ does too. If $I = i$,

$$C_{i,\alpha}^a(c_{J+il}) = c_{(J+a \mod l)+il}.$$

Therefore,

$$\alpha^a(c_{j+il}) = C_{i,\alpha}^a(c_{j+il}) = c_{(j+a \mod l)+il}.$$

Fixed $j \in \{0, \ldots, l-1\}$, the cycle $C_{j,\beta}$ leaves fixed all $c_{J+Il} \in \mathbb{F}_q$ where $J \neq j$, thus $C_{j,\beta}^b$ does too. If $J = j$,

$$C_{j,\beta}^b(c_{j+Il}) = c_{j+(I+b \mod t)l}.$$

Therefore,

$$\beta^b(c_{j+il}) = C_{j,\beta}^b(c_{j+il}) = c_{j+(i+b \mod t)l}.$$

Then,

$$\beta^b\alpha^a(c_{j+il}) = \beta^b(c_{(j+a \mod l)+il}) = c_{(j+a \mod l)+(i+b \mod t)l}.$$

If $\beta^b\alpha^a$ had a fixed point c_{j+il},

$$\beta^b\alpha^a(c_{j+il}) = c_{(j+a \mod l)+(i+b \mod t)l} \implies \begin{cases} j \equiv j+a \mod l \Longrightarrow a \equiv 0 \mod l \\ i \equiv i+b \mod t \Longrightarrow b \equiv 0 \mod t \end{cases}$$

Since $0 \leq a \leq l-1$ and $0 \leq b \leq t-1$, this implies that $a = b = 0$. In other words, $\beta^b\alpha^a$ is the identity.

Hence, $\beta^b\alpha^a$ has no fixed points unless it is the identity.

Moreover,

$$\alpha^a\beta^b(c_{j+il}) = \alpha^a(c_{j+(i+b \mod t)l}) = c_{(j+a \mod l)+(i+b \mod t)l} = \beta^b\alpha^a(c_{j+il}),$$

proving the commutativity. □

An alternative way of defining the permutations α, β is the following.

With the above notations and definitions, we define

$$C_\alpha = \begin{pmatrix} C_{0,\alpha} \\ C_{1,\alpha} \\ \vdots \\ C_{t-1,\alpha} \end{pmatrix} = \begin{pmatrix} c_0 & c_1 & \cdots & c_{l-1} \\ c_l & c_{l+1} & \cdots & c_{2l-1} \\ \vdots & \vdots & \ddots & \vdots \\ c_{(t-1)l} & c_{(t-1)l+1} & \cdots & c_{q-1} \end{pmatrix},$$

$$C_\beta = \begin{pmatrix} C_{0,\beta} \\ C_{1,\beta} \\ \vdots \\ C_{l-1,\beta} \end{pmatrix} = \begin{pmatrix} c_0 & c_l & \cdots & c_{(t-1)l} \\ c_1 & c_{l+1} & \cdots & c_{(t-1)l+1} \\ \vdots & \vdots & \ddots & \vdots \\ c_{l-1} & c_{2l-1} & \cdots & c_{q-1} \end{pmatrix}.$$

C_α is a $t \times l$ matrix whose rows are $C_{i,\alpha}$, $i = 0, \ldots, t-1$. C_β is an $l \times t$ matrix whose rows are $C_{i,\beta}$, $i = 0, \ldots, l-1$. Notice how C_β is the transpose matrix of C_α.

Corollary 3.11. *Let α, β be as in the previous Lemma 3.10. Then, the set defined by*

$$G = \{\alpha^i \beta^j : 0 \leq i \leq l-1, 0 \leq j \leq t-1\}$$

is a subgroup of Σ_q without fixed points and order $|G| = q$.

Proof. First we will show that G is a subgroup of Σ_q. It suffices to prove that, for any $\alpha^i\beta^j, \alpha^k\beta^m \in G$, $(\alpha^i\beta^j)^{-1}(\alpha^k\beta^m) \in G$.

$$(\alpha^i\beta^j)^{-1}(\alpha^k\beta^m) = \beta^{-j}\alpha^{-i}\alpha^k\beta^m = \beta^{-j}\alpha^{(k-i \mod l)}\beta^m = \alpha^{(k-i \mod l)}\beta^{-j}\beta^m = \\ = \alpha^{(k-i \mod l)}\beta^{(m-j \mod t)} \in G,$$

using in these calculations the commutativity of α and β proven in Lemma 3.10.

In Lemma 3.10 we have already shown that G has no fixed points.

The only thing left to see is that $|G| = q$. First, we need to prove that the $\alpha^i\beta^j$ are distinct. Suppose that $\alpha^i\beta^j = \alpha^k\beta^m$. Then,

$$\alpha^0\beta^0 = id = (\alpha^k\beta^m)^{-1}\alpha^i\beta^j = \alpha^{(k-i \mod l)}\beta^{(m-j \mod t)}$$

Then,

$$\begin{cases} 0 \equiv k-i \mod l \implies i \equiv k \mod l \\ 0 \equiv m-j \mod t \implies j \equiv m \mod t \end{cases}$$

Since $0 \leq i, k \leq l-1, 0 \leq j, m \leq t-1$, this implies that $i = k$ and $j = m$. Hence, all $\alpha^i\beta^j$ are distinct.

As all $\alpha^i\beta^j$ are distinct, there are $l \cdot t = q$ elements in G, thus $|G| = q$. □

The above result suggests the following definition.

Definition 3.12. We will call **e-Klenian subgroup** to any group of the form given in Corollary 3.11. Also, we say that a polynomial $f \in \mathbb{F}_q[x, y]$ is an **e-Klenian polynomial** if f is a permutation group polynomial and the associated group $G_{\underline{\beta}_f}$ is an e-Klenian group.

As stated earlier, this is just a family of subgroups of Σ_q without fixed points, not all subgroups with these characteristics are e-Klenian, and therefore not all permutation group polynomials are e-Klenian polynomials.

One might ask how many e-Klenian polynomials are there. No significant results can be found in the literature for $e \geq 1$. However, for $e = 0$ this has a simple answer.

Proposition 3.13. *The number of 0-Klenian polynomials in $\mathbb{F}_q[x, y]$ is*

$$\frac{q!(q-1)!}{\varphi(q)} = \frac{p^r!(p^r-1)!}{p^{r-1}(p-1)}.$$

Proof. 0-Klenian subgroups are those of the form

$$G = \{\beta^j : 0 \leq j \leq q-1\} = \langle \beta \rangle,$$

where $\beta = C_{0,\beta}$ is a cycle of length q.

The number of cycles of maximal length in Σ_q is $(q-1)!$, and a subgroup generated by a cycle of length q contains exactly $\varphi(q)$ generators, β^i where $\gcd(i,q) = 1$. Therefore, the number of 0-Klenian groups of Σ_q is

$$\frac{(q-1)!}{\varphi(q)}.$$

Now, the number of permutation group polynomials that we can construct from a given set of permutations is $q!$, as seen in the previous section. Hence, the number of 0-Klenian polynomials is

$$q! \cdot \frac{(q-1)!}{\varphi(q)} = \frac{p^r!(p^r-1)!}{p^{r-1}(p-1)}.$$

□

The case $e = 0$ is of special relevance, as it is the only case appearing when we restrict to prime fields $\mathbb{F}_p$.

In Appendix B.2 we describe all e-Klenian polynomials in $\mathbb{F}_q[x,y]$ for $q = 2, 3, 4$.

3.3. Orthogonal Polynomial Systems

For a brief moment we are going to go back to $\mathbb{F}_q[x_1, \ldots, x_n]$ to explore a new concept closely related to permutation polynomials.

Up until now we have only worked with permutation polynomials on their own, functions from $\mathbb{F}_q^n$ to $\mathbb{F}_q$. The next definition, however, enables us to consider functions from $\mathbb{F}_q^n$ into $\mathbb{F}_q^m$.

Definition 3.14. A system of polynomials $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$, $1 \leq m \leq n$, is said to be **orthogonal** in $\mathbb{F}_q$ if the system of equations

$$f_1(x_1, \ldots, x_n) = a, \ldots, f_m(x_1, \ldots, x_n) = a_m$$

has q^{n-m} solutions in $\mathbb{F}_q^n$ for each $(a_1, \ldots, a_m) \in \mathbb{F}_q^m$.

In the special case $m = n$ this means that the orthogonal system $f_1, \ldots, f_n$ induces a permutation of $\mathbb{F}_q^n$.

We could as well say that f is a permutation polynomial if and only if f alone forms an orthogonal system.

Proposition 3.15. *Every nonempty subsystem of an orthogonal system of polynomials is again orthogonal.*

Proof. Let $f_1, \ldots, f_{m+1} \in \mathbb{F}_q[x_1, \ldots, x_n]$ be an orthogonal system. To prove the proposition, it suffices to show that $f_1, \ldots, f_m$ are an orthogonal system.

Let $(a_1, \ldots, a_m) \in \mathbb{F}_q^m$. We want to count the number of solutions of the system

$$f_1(x_1, \ldots, x_n) = a_1, \ldots, f_m(x_1, \ldots, x_n) = a_m,$$

that is, $|S|$ where

$$S = \{\bar{y} \in \mathbb{F}_q^n : f_i(\bar{y}) = a_i, i = 1, \ldots, m\}.$$

Let $F_q = \{c_0, \ldots, c_{q-1}\}$. Since $f_1, \ldots, f_{m+1}$ form an orthogonal system,

$$f_1(x_1, \ldots, x_n) = a_1, \ldots, f_m(x_1, \ldots, x_n) = a_m, f_{m+1}(x_1, \ldots, x_n) = c_j$$

has $q^{n-(m+1)}$ solutions for any choice of $c_j \in \mathbb{F}_q$. For each $j \in \{0, \ldots, q-1\}$ we define the set

$$A_j = \{\bar{y} \in \mathbb{F}_q^n : f_{m+1}(\bar{y}) = c_j, f_i(\bar{y}) = a_i, i = 1, \ldots, m\}.$$

Notice that $|A_j| = q^{n-(m+1)}$ for $j = 0, \ldots, q-1$ and $\cup_{j=0}^{q-1} A_j = S$. Also, $A_j \cap A_k = \emptyset$ if $j \neq k$. Then,

$$|S| = \left| \bigcup_{j=0}^{q-1} A_j \right| = \sum_{j=0}^{q-1} |A_j| = \sum_{j=0}^{q-1} q^{n-(m+1)} = q \cdot q^{n-(m+1)} q^{n-m}.$$

Hence, $f_1, \ldots, f_m$ form an orthogonal system. □

Corollary 3.16. *Every polynomial ocurring in an orthogonal system is a permutation polynomial.*

Proof. Let $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ be an orthogonal system of polynomials. Then, each polynonial f_i forms a subsystem, and thus is ortogonal by Proposition 3.15. Hence, f_i is a permutation polynomial for $i = 1, \ldots, m$. □

Given an orthogonal system where $n = m$, we can construct new ones using the following proposition.

Proposition 3.17. *If $f_1, \ldots, f_n \in \mathbb{F}_q[x_1, \ldots, x_n]$ form an orthogonal system, then the polynomials*

$$g_i(x_1, \ldots, x_n) = \sum_{j=1}^{n} b_{ij} f_i(x_1, \ldots, x_n), \quad i = 1, \ldots, n$$

where $b_{ij} \in \mathbb{F}_q$ for $i, j \in \{1, \ldots, n\}$ and $\det((b_{ij})_{i,j=1}^n) \neq 0$ also form an orthogonal system.

Proof. Let $(a_1, \ldots, a_n) \in \mathbb{F}_q^n$. We have the system of equations

$$\begin{aligned} b_{1,1} f_1(\bar{x}) + b_{1,2} f_2(\bar{x}) + \cdots + b_{1,n} f_n(\bar{x}) &= a_1, \\ b_{2,1} f_1(\bar{x}) + b_{2,2} f_2(\bar{x}) + \cdots + b_{2,n} f_n(\bar{x}) &= a_2, \\ &\vdots \\ b_{n,1} f_1(\bar{x}) + b_{n,2} f_2(\bar{x}) + \cdots + b_{n,n} f_n(\bar{x}) &= a_n. \end{aligned}$$

The matrix $B = (b_{ij})_{i,j=1}^{n}$ is invertible by hypothesis. Then, if we write the system in matrix form,

$$\underbrace{\begin{pmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,n} \\ b_{2,1} & b_{2,2} & \dots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n,1} & b_{n,2} & \dots & b_{n,n} \end{pmatrix}}_{=B} \begin{pmatrix} f_1(\bar{x}) \\ f_2(\bar{x}) \\ \vdots \\ f_n(\bar{x}) \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \iff \begin{pmatrix} f_1(\bar{x}) \\ f_2(\bar{x}) \\ \vdots \\ f_n(\bar{x}) \end{pmatrix} = B^{-1} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}.$$

The system on the right has a unique solution since the f_i form an orthogonal system. Hence, the system on the left has a unique solution as well, and thus the g_i form an orthogonal system. □

However, to apply this result we need an orthogonal system to begin with, which isn't trivial to obtain. Here's an example of a family of orthogonal systems where $m = n$, provided by univariate permutation polynomials.

Proposition 3.18. *Let $f_i(z), h_j(z) \in \mathbb{F}_q[z]$ be permutation polynomials for $i, j = 1, \dots, n$. Then, the polynomials*

$$g_i(x_1, \dots, x_n) = f_i\left(\sum_{j=1}^{n} b_{ij} h_j(x_j)\right), \quad i = 1, \dots, n$$

where $b_{ij} \in \mathbb{F}_q$ for $i, j \in \{1, \dots, n\}$ and $\det((b_{ij})_{i,j=1}^{n}) \neq 0$ also form an orthogonal system.

Proof. Let $(a_1, \dots, a_n) \in \mathbb{F}_q^n$. We have the system of equations

$$\begin{aligned} f_1(b_{1,1}h_1(x_1) + b_{1,2}h_2(x_2) + \dots + b_{1,n}h_n(x_n)) &= a_1, \\ f_2(b_{2,1}h_1(x_1) + b_{2,2}h_2(x_2) + \dots + b_{2,n}h_n(x_n)) &= a_2, \\ &\vdots \\ f_n(b_{n,1}h_1(x_1) + b_{n,2}h_2(x_2) + \dots + b_{n,n}h_n(x_n)) &= a_n. \end{aligned}$$

All f_i are permutation polynomials in one variable and hence invertible functions. Then, this system is equivalent to

$$\begin{aligned} b_{1,1}h_1(x_1) + b_{1,2}h_2(x_2) + \dots + b_{1,n}h_n(x_n) &= f_1^{-1}(a_1), \\ b_{2,1}h_1(x_1) + b_{2,2}h_2(x_2) + \dots + b_{2,n}h_n(x_n) &= f_2^{-1}(a_2), \\ &\vdots \\ b_{n,1}h_1(x_1) + b_{n,2}h_2(x_2) + \dots + b_{n,n}h_n(x_n) &= f_n^{-1}(a_n). \end{aligned}$$

The matrix $B = (b_{ij})_{i,j=1}^{n}$ is invertible by hypothesis. Then, if we write the system in matrix form,

$$\underbrace{\begin{pmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,n} \\ b_{2,1} & b_{2,2} & \dots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n,1} & b_{n,2} & \dots & b_{n,n} \end{pmatrix}}_{=B} \begin{pmatrix} h_1(x_1) \\ h_2(x_2) \\ \vdots \\ h_n(x_n) \end{pmatrix} = \begin{pmatrix} f_1^{-1}(a_1) \\ f_2^{-2}(a_2) \\ \vdots \\ f_n^{-1}(a_n) \end{pmatrix} \iff \begin{pmatrix} h_1(x_1) \\ h_2(x_2) \\ \vdots \\ h_n(x_n) \end{pmatrix} = B^{-1} \begin{pmatrix} f_1^{-1}(a_1) \\ f_2^{-2}(a_2) \\ \vdots \\ f_n^{-1}(a_n) \end{pmatrix}.$$

Each equation determines one variable. Since the h_i are permutation polynomials, there is a unique solution for each x_i, and therefore the system on the right has a unique solution in $\mathbb{F}_q^n$. Hence, the original has a unique solution as well, and thus the g_i form an orthogonal system. □

Due to the nature of this chapter, the orthogonal systems we will be working with consist of one or two polynomials in $\mathbb{F}_q[x, y]$.

We already knew how to construct PPs (orthogonal systems with one polynomial) from others thanks to the results seen in Section 2.2, and now using Proposition 3.17 we can do the same thing for orthogonal systems with two polynomials, covering every possible case. Also, an example of a family of orthogonal systems consisting of two polynomials is given by Proposition 3.18.

3.4. Latin Squares

Latin squares occur in many structures such as group multiplication tables and Cayley tables. To be precise Latin squares are referred to as the multiplication tables of an algebraic structure called a quasigroup.

Definition 3.19. A **Latin square of order** q is a $q \times q$ matrix L with entries from a set T of size q such that each element of T occurs exactly once in every row and every column of L.

We will work with Latin squares of order a prime power $q = p^r$ with entries from $\mathbb{F}_q$.

In the following lemma we show the relation between Latin squares and bivariate local permutation polynomials.

Lemma 3.20. *There is a bijective map between Latin squares of order q and local permutation polynomials of $\mathbb{F}_q$.*

Proof. Given a Latin square L over $\mathbb{F}_q = \{c_0, \dots, c_{q-1}\}$ with entries $a_{ij} \in \mathbb{F}_q$, thanks to the Lagrange Interpolation Theorem 2.2 we can construct a polynomial $f \in \mathbb{F}_q[x, y]$ such that

$$f(c_i, c_j) = a_{ij}, \quad 0 \le i, j \le q - 1$$

and $\deg_x(f) < q$, $\deg_y(f) < q$. All that's left to see is if f is an LPP.

Let $a, b \in \mathbb{F}_q$, we want to count the number of solutions of the equation $f(a, y) = b$. There exists $i \in \{0, \ldots, q-1\}$ such that $a = c_i$. Each element of $\mathbb{F}_q$ appears exactly once in every row, hence there exists a unique $j \in \{0, \ldots, q-1\}$ such that $b = a_{ij}$, and thus c_j is the only solution to the equation.

The same reasoning can be applied to find that the equation $f(x, a) = b$ has a unique solution, making f an LPP.

Conversely, let $f \in \mathbb{F}_q[x, y]$ be an LPP. We can construct the matrix

$$L = (a_{ij})_{i,j=0}^{q-1} = (f(c_i, c_j))_{i,j=0}^{q-1}.$$

All that remains to see is if L is a Latin square.

The q elements on row i form the set $L_i = \{f(c_i, c_j) : j = 0, \ldots, q-1\} = \{f(c_i, a) : a \in \mathbb{F}_q\}$. This is the image of the function $f(c_i, x)$, which represents a permutation since f is an LPP. This means that $L_i = \mathbb{F}_q$, and thus each element of $\mathbb{F}_q$ must appear exactly once per row.

The same argument can be applied to the columns of L, and hence it is Latin square. □

If we take the concept of orthogonal polynomial systems over to Latin squares, we get the following definition.

Definition 3.21. Let L_1, L_2 be Latin squares of order q. We say that L_1 and L_2 are **orthogonal Latin squares** if

$$(L_1(i_1, j_1), L_2(i_1, j_1)) \neq (L_1(i_2, j_2), L_2(i_2, j_2))$$

for all distinct pairs of coordinates $(i_1, j_1), (i_2, j_2) \in \mathbb{N}^2$.

Equivalently, two Latin squares of the same order are said to be orthogonal if, when superimposed, each position has a different pair of ordered entries.

Corollary 3.22. *Two Latin squares L_1 and L_2 are orthogonal if and only if their associated local permutation polynomials form an orthogonal system.*

Proof. Let f_i be the LPP associated with L_i, $i = 1, 2$. Then,

$$\begin{gathered}(L_1(i_1, j_1), L_2(i_1, j_1)) \neq (L_1(i_2, j_2), L_2(i_2, j_2)) \\ \Updownarrow \\ (f_1(c_{i_1}, c_{j_1}), f_2(c_{i_1}, c_{j_1})) \neq (f_1(c_{i_2}, c_{j_2}), f_2(c_{i_2}, c_{j_2})).\end{gathered}$$

Therefore, $(L_1(i_1, j_1), L_2(i_1, j_1)) \neq (L_1(i_2, j_2), L_2(i_2, j_2))$ for all distinct pairs of coordinates $(i_1, j_1), (i_2, j_2) \in \mathbb{N}^2$ if and only if all q^2 pairs $(f_1(a, b), f_2(a, b))$ are distinct, or equivalently, if the system

$$f_1(x, y) = a_1, f_2(x, y) = a_2$$

has a unique solution for each $(a_1, a_2) \in \mathbb{F}_q^2$. □

Now, we want to construct families of orthogonal Latin squares. This problem is equivalent to constructing families of orthogonal systems consisting of two polynomials in $\mathbb{F}_q[x, y]$, which brings us to the next definition.

Definition 3.23. Given a permutation polynomial $f \in \mathbb{F}_q[x, y]$, we say that g is a **companion** of f if $(f, g) : \mathbb{F}_q^2 \to \mathbb{F}_q^2$ defines a permutation, that is, f, g form an orthogonal system.

By Corollary 3.16, any companion must be a permutation polynomial.

One may wonder how many companions a permutation polynomial has. The following result answers that question.

Theorem 3.24. *A permutation polynomial f has exactly $(q!)^q$ companions.*

Proof. Let $\mathbb{F}_q = \{c_0, \ldots, c_{q-1}\}$. In the proof of Theorem 2.12 we defined a partition of $\mathbb{F}_q^2$ given by

$$A_i = \{(a, b) \in \mathbb{F}_q^2 : f(a, b) = c_i\}, \quad i = 0, \ldots, q-1,$$

which is such that $|A_i| = q$ because f is a permutation polynomial.

If we order the q elements of each A_i, we can express every $(a, b) \in \mathbb{F}_q^2$ as $(a, b) = (a_{ij}, b_{ij})$, where $i \in \{0, \ldots, q-1\}$ indicates the A_i the pair belongs to and $j \in \{0, \ldots, q-1\}$ indicates the position of the pair in the order we have established.

Now, consider a q-tuple $(\sigma_0, \ldots, \sigma_{q-1}) \in \Sigma_q^q$. Thanks to the Lagrange Interpolation Theorem 2.2 we can define a polynomial g such that

$$g(a_{ij}, b_{ij}) = \sigma_i(c_j), \quad i, j \in \{0, \ldots, q-1\}.$$

We are going to show that g is a companion of f.

Let $(c_i, c_k) \in \mathbb{F}_q^2$. Then, consider the system

$$\begin{aligned} f(x, y) &= c_i \\ g(x, y) &= c_k \end{aligned}$$

The first equation tells us that all solutions (a, b) are in A_i. Then, restricting to that set, we want to find $j \in \{0, \ldots, q-1\}$ such that

$$c_k = g(a_{ij}, b_{ij}) = \sigma_i(c_j) \quad \Longrightarrow \quad c_j = \sigma_i^{-1}(c_k).$$

Therefore, for each $(c_i, c_k) \in \mathbb{F}_q^2$ the system has a unique solution: the pair $(a_{ij}, b_{ij}) \in \mathbb{F}_q^2$ such that $(a_{ij}, b_{ij}) \in A_i$ and $c_j = \sigma_i^{-1}(c_k)$. Thus, f and g are companions.

Each selection of a q-tuple of σ_i gives us a different g. We have $q!$ ways of choosing each σ_i, so f has at least $(q!)^q$ companions.

Finally, we will show that every companion of f can be obtained by the previously explained process.

If g is a companion of f, $g(A_i) = \mathbb{F}_q$, since there has to be a solution to the system

$$\begin{aligned} f(x,y) &= c_i \\ g(x,y) &= a \end{aligned}$$

for every $a \in \mathbb{F}_q$. $|A_i| = q = |\mathbb{F}_q|$, so this implies that $g(a,b)$ is different for each $(a,b) \in A_i$. Also, we can then define for each $i = 0, \ldots, q-1$ the bijection

$$\begin{aligned} h_i : \mathbb{F}_q &\longrightarrow A_i \\ c_j &\longrightarrow (a_{ij}, b_{ij}) \end{aligned}$$

which reflect the ordering on A_i.

Then, there is a q-tuple of permutations $\sigma_i = g \circ h_i$ associated to g.

Since all companions of f can be obtained this way, f has exactly $(q!)^q$ companions. □

Given a permutation polynomial f, Theorem 3.24 not only lets us count its companions, but tells us how to construct them.

When we restrict this to local permutation polynomials (that is, Latin squares), it isn't as straightforward: not all LPPs have LPPs as companions.

Example 3.25. *When $q = 2$, we are working with polynomials in $\mathbb{F}_2[x,y]$. The only LPPs are $f(x,y) = x + y$ and $g(x,y) = x + y + 1$, which don't form an orthogonal system, as the system*

$$\begin{aligned} x + y &= 0 \\ x + y + 1 &= 0 \end{aligned}$$

has no solution in $\mathbb{F}_2^2$. Hence, nor f nor g have a companion that is an LPP.

However, LPPs that have LPPs as companions do exist. We will see an example of this in the next theorem.

Theorem 3.26. *For $q \geq 3$, every linear LPP has at least one companion which is also a linear LPPs.*

Proof. Let $f(x,y) = ax + by + c$ be an LPP ($a, b \neq 0$). Now, consider the LPP $g = ux + vy + w$, where $u, v, w \in \mathbb{F}_q$ are such that $u, v \neq 0$ and $av - bu \neq 0$.

Now, let $(a_1, a_2) \in \mathbb{F}_q^2$. Then, we have the system

$$\begin{aligned} ax + by + c &= a_1 \\ ux + vy + w &= a_2 \end{aligned} \iff \begin{aligned} ax + by &= a_1 - c \\ ux + vy &= a_2 - w \end{aligned}$$

If we write it in matrix form,

$$\underbrace{\begin{pmatrix} a & b \\ u & v \end{pmatrix}}_{=A} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_1 - c \\ a_2 - w \end{pmatrix} \iff \begin{pmatrix} x \\ y \end{pmatrix} = A^{-1} \begin{pmatrix} a_1 - c \\ a_2 - w \end{pmatrix}$$

Remember that A^{-1} exists because $\det(A) = av - bu \neq 0$. Therefore, the system has a unique solution.

This means that f and g form an orthogonal system, and hence they are companions. □

Finally, we will see what happens when we consider e-Klenian polynomials.

Theorem 3.27. *Let $2 \nmid q$. Let $f \in \mathbb{F}_q[x, y]$ be an e-Klenian polynomial. Then, f has a companion which is an LPP.*

Proof. Let $f(x, y)$ be an e-Klenian polynomial and for each $m = 0, \ldots, q-1$ written as $m = a + bl$ where $0 \leq a \leq l-1$ and $0 \leq b \leq t-1$, consider the set

$$A_m = \{(c_j, \alpha^a \beta^b(c_j)) : j = 0, \ldots, q-1\}.$$

f is the LPP associated to this partition.

We will see that the polynomial g associated with the partition

$$B_m = \{(c_k, \alpha^{a+i}\beta^{b+j}(c_k) : k = i + jl, 0 \leq i \leq l-1, 0 \leq j \leq t-1\}, \quad m = 0, \ldots, q-1$$

is an LPP which is companion of f.

First we will show that g is an LPP. We start by proving that for any $c_k, c_m \in \mathbb{F}_q$, there exists a unique $y \in \mathbb{F}_q$ such that $g(c_k, y) = c_m$. As in the definitions before, let $k = u + lv$, with $0 \leq u \leq l-1$, $0 \leq v \leq t-1$, and $m = a + bl$, with $0 \leq a \leq l-1$, $0 \leq b \leq t-1$. We want $(c_k, y) \in B_m$, and thus the only possible value for y is

$$y = \alpha^{a+u}\beta^{b+v}(c_k) = c_{(a+2u \mod l)+(b+2v \mod t)l}.$$

Now, we want to prove that g is also a permutation polynomial in the first variable, or in other words, that given $c_k, c_m \in \mathbb{F}_q$ as before, there exists a unique x such that $g(x, c_k) = c_m$. In particular, we need to find i, j such that $c_k = \alpha^{a+i}\beta^{b+j}(c_{i+jl})$, and in this case $x = c_{i+jl}$ is a solution, since by definition $(x, c_k) \in B_m$. Then,

$$c_k = c_{u+lv} = \alpha^{a+i}\beta^{b+j}(c_{i+jl}) = c_{(a+2i \mod l)+(b+2j \mod t)l}.$$

This implies that

$$\begin{cases} u \equiv a + 2i \mod l \implies i \equiv 2^{-1}(u-a) \mod l, \\ v \equiv b + 2j \mod t \implies j \equiv 2^{-1}(v-b) \mod t. \end{cases}$$

These i, j are unique since $0 \leq i \leq l-1, 0 \leq j \leq t-1$, and thus $x = c_{i+jl}$ is the unique solution we were looking for.

Finally, we need to check that f, g form an orthogonal system. Let $c_m, c_k \in \mathbb{F}_q$ as before. We want to see that

$$f(x, y) = c_m = c_{a+bl}$$
$$g(x, y) = c_k = c_{u+vl}$$

has exactly one solution. This happens if and only if there exist unique $0 \leq i \leq l-1$, $0 \leq j \leq t-1$ such that

$$(c_{i+jl}, \alpha^a \beta^b(c_{i+jl})) = (c_{i+jl}, \alpha^{u+i}\beta^{v+j}(c_{i+jl})).$$

Remember that each element of $\{\alpha^i\beta^j : 0 \leq i \leq l-1, 0 \leq j \leq t-1\}$ is distint, so this can happen if and only if

$$\begin{cases} a \equiv u+i \mod l \implies i \equiv a-u \mod l, \\ b \equiv v+j \mod t \implies j \equiv b-v \mod t. \end{cases}$$

These i, j are unique since $0 \leq i \leq l-1, 0 \leq j \leq t-1$, and thus the system has a unique solution.

Hence, f and g are companions. □

3.4.1. Mutually Orthogonal Latin Squares

We can generalise the concept of orthogonality in Latin squares to more than two.

Definition 3.28. A set of Latin squares, all of the same order, such that all pairs of Latin squares are orthogonal is called a set of **Mutually Orthogonal Latin Squares (MOLS)**.

The following are well known results regarding MOLS, proven in [14].

Theorem 3.29. *Let $N(n)$ be the size of the largest collection of MOLS of order n. Then, we have*

(i). $N(n) \leq n-1$.

(ii). *If q is a power of a prime, then* $N(q) = q-1$.

Definition 3.30. A set of $t > 1$ MOLS of order n is called a **complete** set if $t = N(n)$.

Using Propositions 3.17 and 3.18 we can find examples of complete sets of MOLS.

Theorem 3.31. *With the above notations and definitions:*

- *If $f(x,y)$ is a local permutation polynomial and $g(x,y)$ is any LPP companion of $f(x,y)$, then the set $\{f(x,y) + ag(x,y) : a \in \mathbb{F}_q^*\}$ is a complete set of MOLS.*
- *If $f(x), h(y)$ are permutation polynomials, then the set $\{f(x) + ah(y) : a \in \mathbb{F}_q^*\}$ is a complete set of MOLS.*

In the next chapter we will explore some applications that these sets of Latin squares have in cryptography.

3.4.2. Hypercubes

The concept of Latin squares as well as their relation with LPPs can be generalised to higher dimensions.

Definition 3.32.

(*i*). Let $n, q \in \mathbb{N}$ and T a set of q elements (symbols). A **n-dimensional hypercube** H of order q is a $q \times \cdots \times q$ array with q^n symbols based on the q elements of T. Such a hypercube is **of type** j, $0 \leq j \leq n-1$, if whenever any j of the coordinates are fixed each of the q elements of T appears q^{n-j-1} in that subarray.

If H is of type $n-1$ it is called a **Latin hypercube**. If $n = 2$, H is a **Latin square**.

(*ii*). Let $q = p^r$ and $F = \mathbb{F}_q$. Let $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ and $0 \leq j \leq n-1$. f is a **j-permutation polynomial** (or j-PP) if for all choices of j variables $x_{i_1}, \ldots, x_{i_j}$ and for all choices of points $(a_1, \ldots, a_j) \in \mathbb{F}_q^j$, the equation

$$f|_{x_{i_k}=a_k, k=1,\ldots,j} = a$$

has q^{n-j-1} solutions in $\mathbb{F}_q^{n-j}$ for each $a \in \mathbb{F}_q$.

- f is an $(n-1)$-PP $\iff$ f is a **local permutation polynomial**,
- f is a 0-PP $\iff$ f is a **permutation polynomial**.

Remark: Any hypercube H of type j si also of type k and any j-PP is also a k-PP, for $k = 0, \ldots, j-1$.

Theorem 3.33.

(*i*). *There is a bijective map between n-dimensional hypercubes H of order a prime power p and polynomials $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ such that* $\deg_{x_i}(f) < q$.

(*ii*). *There is a bijective map between n-dimensional hypercubes H of type j and order a prime power p and j-PPs $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ such that* $\deg_{x_i}(f) < q$.

Proof. Given an n-dimensional hypercube H, we can identify the symbols with the elements of $\mathbb{F}_q = \{c_0, \ldots, c_{q-1}\}$. Then, thanks to the Lagrange Interpolation Theorem 2.2 we can construct a polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ such that

$$f(c_{i_1}, \ldots, c_{i_n}) = H(i_1, \ldots, i_n), \quad 0 \leq i_j \leq q-1,$$

and $\deg_{x_i}(f) < q$ for each $i = 1, \ldots, n$.

If H is a hypercube of type j, it is easy to see that f is a j-PP.

Conversely, given the polynomial f we can construct an n-dimensional hypercube H as follows: given any cell indexed by $(i_1, \ldots, i_n) \in \{0, \ldots, q-1\}^n$,

$$H(i_1, \ldots, i_n) = f(c_{i_1}, \ldots, c_{i_n}).$$

If f is a j-PP, it is easy to see that H is a hypercube of type j. □

In particular, this relation, together with the fact that an $(n-1)$-PP is the same thing as an LPP, is what allows us to count the number of LPPs in $\mathbb{F}_q[x_1, \ldots, x_n]$.

Latin hypercubes is a much more explored topic, and thus we can find documentation dedicated to counting Latin hypercubes for small values of q and n, see [18].

For instance, for $q = 2, 3, 4, 5$, we have:

- $\mathbb{F}_q[x_1, x_2] : 2, 12, 576, 161280$;
- $\mathbb{F}_q[x_1, x_2, x_3] : 2, 24, 55296, 2782803520$;
- $\mathbb{F}_q[x_1, x_2, x_3, x_4] : 2, 48, 36972288, 52260618977280$.

CHAPTER 4
Applications of Latin Squares

In this chapter we illustrate two important applications of the bivariate local permutation polynomials: in Coding Theory, to construct Maximum Distance Separable (MDS) codes, and in Cryptography, to design secret sharing schemes.

4.1. Coding Theory

Coding Theory is the study of methods of accurately transferring data across noisy channels and recovering corrupted messages. We will begin this section by introducing definitions and basic results regarding this area.

Definition 4.1. Let $\mathcal{A}$ be a finite set called **alphabet**.

- A **word** is a finite list of elements of $\mathcal{A}$. The number of elements in this list is the **length** of the word. We denote by $\mathcal{A}^n$ the set of words over $\mathcal{A}$ of length n, and by $\mathcal{A}^*$ the set of all words over $\mathcal{A}$.
- A **code** C is a subset of $\mathcal{A}$. If $|\mathcal{A}| = q$, C is a **q-ary code**.
- If all words in C have the same length n, C is a **block code** of length n.

Usually, $\mathcal{A} = \mathbb{F}_q$ for some q power of a prime p. In this case, a block code $C \subset \mathbb{F}_q^n$ is **linear** if it is a linear subspace of $\mathbb{F}_q^n$.

Definition 4.2. Let $\mathcal{A}$ be an alphabet, $x, y \in \mathcal{A}^n$. We define the **Hamming distance** $d(x, y)$ as

$$d(x, y) = \text{ number of coordinates in which } x \text{ and } y \text{ differ.}$$

Proposition 4.3. *The Hamming distance is a metric on $\mathcal{A}^n$.*

Definition 4.4. Let C be a q-ary code of length n.

- We define the **distance of** C, denoted by $d(C)$, as

$$d(C) = \min\{d(x, y) : x, y \in C, x \neq y\},$$

  where $d(\cdot\,, \cdot)$ is the Hamming distance.

- If $d(C) = d$ and $|C| = M$, we say that C is a $(n, M, d)_q$-code.
- $A_q(n, d)$ denotes the largest value of M for which exists an $(n, M, d)_q$-code.

Calculating $A_q(n, d)$ is a very difficult problem, and in fact there is no known formula for this value. However, several bounds have been proven, among which we find the Singleton bound.

Theorem 4.5 (Singleton bound)**.** *For all $q, n, d \in \mathbb{N}$,*

$$A_q(n, d) \leq q^{n-d+1}.$$

Definition 4.6. A **Maximum Distance Separable (MDS) code** is a $(n, M, d)_q$-code C such that $M = q^{n-d+1}$.

These are part of a family of codes known as **error-correcting codes**, which can detect and correct a certain number of errors that may occur during the transmission of a message.

There is a rather interesting connection between MDS codes and sets of MOLS.

By Definition 4.6, if $d = n-1$, all $(n, q^2, n-1)_q$-codes are MDS codes. In Chapter 13 of [14], we find the following Theorem.

Theorem 4.7. *There exists a $(n, q^2, n-1)_q$-code if and only if there exist $n-2$ MOLS of order q.*

The proof consists in showing a way that we can construct an $(n, q^2, n-1)_q$-code from a family of $n-2$ MOLS of order q and viceversa.

We will show this process through an example.

Example 4.8. *Consider the following pair of MOLS of order 3:*

$$L_1 = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 2 & 0 \\ 2 & 0 & 1 \end{bmatrix}, \quad L_2 = \begin{bmatrix} 0 & 1 & 2 \\ 2 & 0 & 1 \\ 1 & 2 & 0 \end{bmatrix}$$

From these MOLS of order 3, we can construct the code

$$C = \{(i, j, L_1(i, j), L_2(i, j)) : 0 \leq i, j \leq 2\}.$$

We just need to show that C is, in fact, a $(4, 9, 3)_3$-code.

- *$C \subset \mathbb{F}_3^4$, so C is a ternary (3-ary) block code of length 4.*
- *It is trivial that $|C| = 9$.*
- *$d(C) \neq 4$ since there are three words with the symbol 0 in the first coordinate. The first two coordinates of each word in C are different, so $d(C) \geq 2$. If $d(C) = 2$, that would mean that there exist $x, y \in C$, where*

  $$x = (i_1, j_1, L_1(i_1, j_1), L_2(i_1, j_1)), \quad y = (i_2, j_2, L_1(i_2, j_2), L_2(i_2, j_2)),$$

  such that x and y differ in exactly two coordinates. There are two possibilities:

– $i_1 \neq i_2$ *and* $j_1 \neq j_2$.
This would imply that

$$(L_1(i_1, j_1), L_2(i_1, j_1)) = (L_1(i_2, j_2), L_2(i_2, j_2)),$$

contradicting the fact that L_1 *and* L_2 *are orthogonal.*

– $i_1 = i_2$ *or* $j_1 = j_2$, *but not both at the same time.*
Without loss of generality, let us suppose that $i_1 = i_2 = i$.
Since $d(C) = 2$, *either*

$$L_1(i, j_1) = L_1(i, j_2) \quad or \quad L_2(i, j_1) = L_2(i, j_2).$$

Then, one of the Latin squares would have a repeated symbol in row i *(because* $j_1 \neq j_2$*), which is a contradiction.*

Thus, $d(C) \neq 2$, *and hence* $d(C) = 3$.

4.2. Cryptography

Cryptography is the study of how to design secure systems of communication, known as **cryptosystems**. The necessary components to set up a cryptosystem are:

- an **encryption key** K_E and a **decryption key** K_D. We will only be considering symmetric cryptosystems, where $K_E = K_D = K$;
- a **message** M;
- an **encrypting scheme** E_K to encrypt or encipher the message M to form a **ciphertext** C;
- a **decrypting scheme** D_K to decrypt or decipher received ciphertexts.

Given a message M, E_K and D_K are transformations such that

$$D_K(E_K(M)) = M.$$

They are also dependent on the key K, which must be kept secret by the users of the system.

A Latin square is a good candidate to be the key in a cryptosystem due to the huge number of Latin squares for a large order and the computationally difficult problems related to them. In this section we will explore how Latin squares can be used to build cryptosystems.

Additionally, in Section 14.4 of [14], versions of classic cryptosystems using Latin squares, such as RSA or Diffie-Hellman key exchange, are shown.

4.2.1. Encryption

We will first present a simple encryption method based upon the theory of sets of MOLS.

Our starting point will be a set $\{L_1, \ldots, L_k\}$ of MOLS of order n, and suppose that $T = \{1, \ldots, n\}$.

The key K in this cryptosystem will be a pair of Latin squares from the set, L_c and L_d such that $c \neq d$. Thus, there is a total of $\binom{k}{2}$ possible keys.

The messages we can transmit using this cryptosystem are those of the form $(i, j) \in \{1, \ldots, n\}^2$, giving us a total of n^2 possible messages. Any of these pairs can be interpreted as a position in a matrix, or in this case, a Latin square. The way this message is encrypted is by transmitting the pair of elements (α, β) that occur in position (i, j) in L_c and L_d, that is,

$$E_K((i, j)) = (L_c(i, j), L_d(i, j)) = (\alpha, \beta).$$

Note that E_K is injective, as L_c and L_d are orthogonal. Since E_K is a function from $\{1, \ldots, n\}^2$ to itself, this implies that E_K is bijective, and hence D_K exists and is bijective as well.

To decrypt the ciphertext, the receiver just has to look for the unique coordinate in which the pair (α, β) occurs, and thus deciphering the message.

$$D_K((\alpha, \beta)) = (i, j) \quad \text{such that} \quad L_1(i, j) = \alpha,\ L_2(i, j) = \beta.$$

4.2.2. Secret Sharing Schemes

For the proper functioning of a cryptosystem, it is of vital importance to keep the key secret from outsiders. A way we can ensure this is by using a secret sharing scheme.

Definition 4.9. A (t, k)**-secret sharing scheme** is a system where k pieces of information called **shares** or **shadows** of a key K are distributed so that each participant has a share such that

(i). the key K can be reconstructed from knowledge of any t or more shares;

(ii). the key K cannot be reconstructed from knowledge of fewer than t shares.

The goal of this part of the dissertation is to show examples of secret sharing schemes where the secret in question is a Latin square.

First, we need to define a couple new concepts.

Definition 4.10. A **partial Latin square of order** n is an $n \times n$ matrix L with entries from a set T of size n such that no element of T occurs twice in any row or column.

The difference between a Latin square and a partial Latin square is that the latter can have empty cells, but both conform with the Latin property of the array. Notice how all Latin squares are partial Latin squares, but the converse isn't usually true.

Some partial Latin squares can be extended to Latin squares by filling the empty cells. In fact, it was conjectured in [8] and later proven in [25] that any partial Latin square of order n with at most $n-1$ cells filled can be completed to a Latin square of order n. However, this isn't always possible.

Example 4.11. *The partial Latin square of order 3*

$$\begin{bmatrix} 1 & * & 3 \\ * & 2 & * \\ * & * & * \end{bmatrix}$$

cannot be completed to a Latin square.

Given a Latin square, there is a very special family of partial Latin squares associated to it: its critical sets.

Definition 4.12. Let L be a Latin square of order n. A **critical set** C of L is a set

$$C = \{(i, j; k) : i, j, k \in \{1, \ldots, n\}\}$$

with the following two properties:

(i). L is the only Latin square of order n which has symbol k in cell (i, j) for each $(i, j; k) \in C$; and

(ii). no proper subset of C has property (i).

Basically, a critical set C of a Latin square L is a partial Latin square which can only be extended to L, and if we remove any entry from C, the unique completion property does not hold anymore.

Note how a Latin square can have many different critical sets of various sizes.

Definition 4.13. Let C be a critical set of a Latin square L. C is called **minimal** if its cardinality is the smallest possible for L.

Even though we know we can complete a Latin square from one of its critical sets, to do so might be a very time-consumming process. As a matter of fact, deciding whether a partial Latin square can be completed or not is an NP-complete problem, as shown in [4]. This concern leads us to the next definition.

Definition 4.14. Let L be a Latin square of order n and C one of its critical sets. Let $|C|$ be the size of C, that is, the number of non empty cells in C. C is called a **strong critical set** if there exists a sequence of partial Latin squares $\{P_0, \ldots, P_m\}$ such that

- $C = P_0 \subset P_1 \subset \cdots \subset P_m = L$, where $m = n^2 - |C|$;

- for any ℓ, $0 \leq \ell < m-1$, $P_\ell \cup \{(i_\ell, j_\ell; k_\ell)\} = P_{\ell+1}$ and $P_\ell \cup \{(i_\ell, j_\ell; k)\}$ is not a partial Latin square if $k \neq k_\ell$.

Completing a strong critical set to a Latin square is a much easier job, since everytime we get a new partial square P_ℓ, $0 \leq \ell < m-1$, there always exists a position (i_ℓ, j_ℓ) in P_ℓ that can only be filled by a particular k_ℓ because all other elements of $\{1, \ldots, n\} \setminus \{k_\ell\}$ already appear either on row i_ℓ or in column j_ℓ.

Example 4.15. *Here we will show an example of a critical set that isn't strong. Let C, L be the following partial Latin square and Latin square of order 6.*

$$C = \begin{bmatrix} * & 2 & 3 & 4 & * & * \\ 3 & * & * & * & * & 4 \\ * & * & * & 5 & * & * \\ * & 4 & 6 & * & * & 1 \\ * & 6 & * & * & 2 & * \\ 6 & * & * & * & 1 & * \end{bmatrix}, \quad L = \begin{bmatrix} 1 & 2 & 3 & 4 & 6 & 5 \\ 3 & 1 & 2 & 6 & 5 & 4 \\ 2 & 3 & 1 & 5 & 4 & 6 \\ 5 & 4 & 6 & 2 & 3 & 1 \\ 4 & 6 & 5 & 1 & 2 & 3 \\ 6 & 5 & 4 & 3 & 1 & 2 \end{bmatrix}.$$

It is easy to check that C is a critical set of L, but when trying to complete it there are at least two choices for each of its empty squares, meaning that it isn't strong.

Now, we will illustrate some examples of secret sharing schemes involving Latin squares. In all of these, the very secret we want to keep is a particular Latin square L of order n, which can be used as a key in a cryptosytem as shown in 4.2.1.

The first scheme is fairly simple. Let $C_1, \ldots, C_m$ be critical sets of L. We define the set S as the union of these C_i, $i = 1, \ldots, m$. Then, we can distribute a share in S to each participant of the scheme. Whenever a group of participants joins together to form one of the critical sets C_i, they can reconstruct the Latin square L, and thus recovering the secret.

Example 4.16. *Let L be the following Latin square of order 3, and S a union of critical sets of L.*

$$L = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix}, \quad S = \{(1,1;1), (2,2;3), (3,3;2)\}.$$

Any subset of S with two elements is a critical set of L, and thus we have a $(2,3)$-secret sharing scheme.

If we choose carefully the critical sets that form S, we can construct secret sharing schemes with special properties.

We might want a system where some participants' shares carry more weight than others, that is, a **multilevel scheme**. In a multilevel scheme, a share from a participant in a higher rank equates to multiple shares from lower ranked participants.

To get this using this system, we can give the elements of a critical set C_1 to the people from higher ranks, and choose the other critical sets in a way that at least one element of C_1 is in them.

Example 4.17. *Let L be the Latin square of order 3 from Example 4.16. The following are all critical sets of L:*

$$C_1 = \{(3,2;1),(2,1;2)\},$$
$$C_2 = \{(3,2;1),(1,1;1),(1,2;2)\},$$
$$C_3 = \{(2,1;2),(1,1;1),(1,2;2)\}.$$

Then,

$$S = \{(3,2;1),(2,1;2),(1,1;1),(1,2;2)\}.$$

The high rank participants would be given the shares from C_1, while the lower level participants would receive the remaining shares in S. This way, the high rank participants can recover L if they join their shares, but the low rank participants need at least one high ranked person to uncover the secret.

Finally, we will present a variation of a multilevel system through an example.

Example 4.18. *Imagine a company with three departments, each of which use a secret sharing scheme to keep a common key, L, secret. The president of the company would need to be part of all three schemes, while the rest of the employees are only participants in the scheme of their respective department. For the sake of commodity, the president wants to have the same share in all three schemes.*

A possible way to set this system up is the following. The secret will be the Latin square L, where

$$L = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 1 & 4 & 5 & 3 \\ 3 & 5 & 1 & 2 & 4 \\ 4 & 3 & 5 & 1 & 2 \\ 5 & 4 & 2 & 3 & 1 \end{bmatrix}.$$

This Latin square has 41 minimal critical sets, three of which are

$$C_1 = \{(1,1;1),(2,5;3),(3,5;4),(4,2;3),(4,3;5),(5,1;5),(5,3;2)\},$$
$$C_2 = \{(1,1;1),(1,5;5),(3,2;5),(3,5;4),(4,2;3),(5,3;2),(5,4;3)\},$$
$$C_3 = \{(1,1;1),(1,5;5),(3,4;2),(4,2;3),(4,5;2),(5,2;4),(5,4;3)\}.$$

The only common element to all three critical sets is $(1,1;1)$, which would be the president's share. The rest of the shares would be distributed in a way such that when all employees from a department join together their shares with the president, they form one of the three critical sets.

Also, note how $S \setminus \{(1,1;1)\}$ isn't a critical set of L, there are 5 completions of $S \setminus \{(1,1;1)\}$ to a Latin square. Therefore L can't be recovered if the president isn't there, even if all departments pool together their shares.

However, if using this type of model to build a secret sharing scheme, precaution is needed when selecting the critical sets that form S, because otherwise it can happen that subsets of S we haven't accounted for form critical sets, giving an unauthorized group of people the chance to obtain the secret key. Note how in Example 4.18 we had to verify this situation wasn't possible.

Another problem with this system and its variations is that even though a group of people whose shares don't form a critical set can't recover the Latin square L we are hiding, they do have partial information if they pool all of their shares: a partial Latin square of L. It can happen that, by trial and error, this unauthorized group finds L. As mentioned in Example 4.18, if all employees join together, they reduced the number of possible keys from 161.280, which is the number of Latin squares of order 5, to 5.

To solve this issue, we will introduce one last secret sharing scheme.

This will be a (t,t)-secret sharing scheme. As before, the secret will be a Latin square L of order n. Let C be a critical set of L with m elements. Each participant will receive a m-tuple P_ℓ, where

$$P_\ell \in \{(i,j;k) : i,j,k \in \{0,\dots,n-1\}\}^m, \quad \ell = 1,\dots,t.$$

The first $t-1$ tuples will be randomly generated, and the last one will be calculated in such a way that, when summing all the tuples $\mod n$, the resulting tuple will be formed by the elements in C.

Using this system, the only way L can be recovered is if all participants pool their shares. If one person is missing, the rest of the group has as much information about L as a complete outsider.

Let us show an example of this system. Notice how we have shifted from considering $(i,j;k) \in \{1,\dots,n\}^3$ to $(i,j;k) \in \{0,\dots,n-1\}^3$, so modular arithmetic could be applied.

Example 4.19. *Let $C = \{(0,0;0),(1,1;1)\}$ be a critical set of the Latin square*

$$L = \begin{bmatrix} 0 & 2 & 1 \\ 2 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix}.$$

Suppose that $t = 3$. Then, we need pairs (2-tuples) P_1, P_2 and P_3. P_1 and P_2 will be randomly generated. For example,

$$P_1 = ((0,1;2),(2,0;0)), \quad P_2 = ((1,2;1),(0,2;1)).$$

Now, we calculate P_3. All calculations are done $\mod 3$.

$$\begin{aligned} P_3 &= C - (P_1 + P_2) = ((0,0;0),(1,1;1)) - ((0,1;2),(2,0;0)) - ((1,2;1),(0,2;1)) = \\ &= ((0-0-1, 0-1-2; 0-2-1),(1-2-0, 1-0-2; 1-0-1)) = \\ &= ((2,0;0),(2,2;0)). \end{aligned}$$

This way, $P_1 + P_2 + P_3$ is a tuple with the elements of C, and thus the participants can recover L when they pool all their shares together.

Bibliography

[1] N. Anbar, C. Kasikci, A. Topuzoglu. *On components of vectorial permutations of $\mathbb{F}_q^n$, Finite Fields and their applications*, 58(2019) 124-132.

[2] J. Bate, G. H. van Rees, *The Size of the Smallest Strong Critical Set in a Latin Square.* Ars Comb. 53 (1999).

[3] C. Chum, X. Zhang, *The Latin squares and the secret sharing schemes.* Groups, Complexity, Cryptology. 2. (2010).

[4] C.J. Colbourn, *The Complexity of Completing Partial Latin Squares*, Discrete Aplied mathematics 8 (1984), 25-30.

[5] . L.E. Dickson, *History of the Theory of Numbers*, vol. 3, Carnegie Institute, Washington, D.C., 1923, Dover, New York, 2005.

[6] W.S. Diestelkamp, S.G. Hartke, R.H. Kenney, *On the degree of local permutation polynomials*, J.Comb. Math. Comb. Comput. 50 (2004) 129–140.

[7] J. Egan, I.M. Wanless, *Enumeration of MOLS of small order.* Math. Comput. 85, 799–824 (2016).

[8] T. Evans, *Embedding incomplete Latin squares*, The American Mathematical Monthly 67 (1960), 958-961.

[9] J. Gutierrez, J. J. Urroz, *Local permutation polynomials and the action of e-Klenian groups*, Finite Fields and Their Applications 91 (2023) 102261 https://doi.org/10.1016/j.ffa.2023.102261.

[10] J. Gutierrez, J. J. Urroz, *Permutation and local permutation polynomials of maximum degree*, (2023) arXiv:2308.01258v2

[11] X. Hou, *Permutation polynomials over finite fields — A survey of recent advances*, Finite Fields and Their Applications 32(2015)82–119.

[12] K.H. Hicks, G.L. Mullen, L. Storme, J. Vanpoucke, *The number of different reduced complete sets of MOLS corresponding to PG (2,q)*, J. Geometry. 109:5 (2018).

[13] A.D. Keedwell, J. Dénes: *Latin Squares and their Applications.* Elsevier, Amsterdam (2015).

[14] C. F. Laywine, G. L. Mullen. *Discrete Mathematics Using Latin Squares*, John Wiley & Sons, 1998.

[15] R. Lidl, G. L. Mullen, *When Does a Polynomial Over a Finite Field Permute the Elements of the Field?* The American Mathematical Monthly, 95(3), 243-246, (1988).

[16] R. Lidl, H. Niederreiter, *Finite Fields*, 2nd edn., Encyclopedia Math. Appl., vol.20, Cambridge University Press, Cambridge, 1997.

[17] L. Mariot, M. Gadouleau, E. Formenti, A. Leporati, *Mutually orthogonal Latin squares based on cellular automata.* Des. Codes Cryptogr. 88(2): 391-411 (2020).

[18] B. D. McKay, I. M. Wanless, *A census of small Latin hypercubes*, SIAM Journal on Discrete Mathematics 22 (2008) 719-736.

[19] D.C. Montgomery: *Design and Analysis of Experiments.* Wiley, Hoboken (2017).

[20] G.L. Mullen, *Local permutation polynomials over* $\mathbb{Z}_p$, Fibonacci Q. 18 (1980) 104–108.

[21] G.L. Mullen, *Local permutation polynomials in three variables over* $\mathbb{Z}_p$, Fibonacci Q. 18 (1980) 208–214.

[22] G.L. Mullen, D. Panario, *Handbook of Finite Fields*, Taylor & Francis, Boca Raton, 2013.

[23] H. Niederreiter, *Permutation polynomials in several variables over finite fields*, Proc. Jpn. Acad. 46 (1970) 1001–1005.

[24] H. Niederreiter, *Orthogonal systems of polynomials in finite fields*, Proc. Am. Math. Soc. 28 (1971) 415–422.

[25] B. Smetaniuk, *A new construction on Latin squares. I. A proof of the Evans conjecture.* Ars Combin, 11, 155-172.

[26] D. R. Stinson, *Combinatorial characterizations of authentication codes.* Des. Codes Cryptogr. 2(2), 175–187 (1992).

[27] A. Winterhof, *Generalizations of complete mappings of finite fields and some applications.* J. Symb. Comput. 64: 42-52 (2014)

APPENDIX A
SageMath Package

In this appendix we present the SageMath package PermutationPolynomials. The keys for its access are

- IP address : 193.146.75.191:8080
- Login: PermutationPolynomial
- Password: AMAC

This package, still in development, is designed for manipulating Permutation and Local Permutation Polynomials, so the main objects that PermutationPolynomials deals with are multivariate polynomials in $\mathbb{F}_q[x_1, \ldots, x_n]$ over any finite field $\mathbb{F}_q$. Also, there are functions for the concrete case $n = 2$, that is, $\mathbb{F}_q[x, y]$, as we have seen more results regarding it in Chapter 3.

One can define $\mathbb{F}_q$ as follows.

```
if is_prime(q):
    K=GF(q)
else:
    K.<u>=FiniteField(q)
```

Now, we show two ways of defining $\mathbb{F}_q[x_1, \ldots, x_n]$. If we want to give a particular name to each of the variables, we can do it like so.

```
S.<x,y>=PolynomialRing(K)
```

Otherwise, the next command can be used and the variables will be $x0, x1, \ldots, xn$. It is especially useful when $n \geq 4$.

```
S=PolynomialRing(K,'x',n)
```

The package contains about 30 functions. Now, we are going to illustrate a couple of such functions.

The first one is an implementation of Theorem 3.27, where the argument is a cycle of maximum length that generates a 0-Klenian subgroup.

```
def e_Klenian_OS(cycle,Kx):
    '''
    This function computes a companion polynomial of a 0-Klenian polynomial
    associated to the subgroup generated by the input cycle.
    Input: - A cycle of maximum length q;
           - The polynomial ring Kx.
    Output: A companion polynomial of a 0-Klenian polynomial associated to the
            subgroup generated by the input cycle.
    '''
    K=Kx.base_ring()
    q=K.cardinality()

    D={i:cycle[i] for i in range(q)}
    A=cycle_set(cycle,K)
    B=[[] for j in range(q)]
    for j in range(q):
        for i in range(q):
            B[j].append((D[i], A[(j+i+1)%q][i][1]))
    return interpol(B,Kx)
```

`e_Klenian_OS` calls two other functions:

- `cycle_set`, for computing the partition

$$A_i = \{(a, \beta^i(a)) : a \in \mathbb{F}_q\}, \quad i = 1, \ldots, q$$

  where β is the input cycle.

- `interpol`, for computing the corresponding associated polynomial, that is, the implementation of the Lagrange Interpolation Theorem 2.2. This function works in $\mathbb{F}_q[x_1, \ldots, x_n]$.

```
def cycle_set(C,K):
    '''
    This function computes the following partition given a cycle permutation C.
    Input: - A list [a,b,c,..,x] of elements of the field K representing a cycle
           permutation, that is, C: a-->b-->c-------...>x---a;
           - A field K.
    Output: The partition {A_i:i=1,...,q}, where
            A_i=[(c_j, C^i(c_j)), c_j in K={c_1,...,c_q}], and |A_i|=q.
            That is, a list of q sublists of tuples in K^2 with q elements.
```

```
    '''
    q=K.cardinality()

    A=[[] for i in range(q)]
    for i in range(q):
        for j in range(q):
            A[i].append((K(C[j]), K(C[(j+i+1)%q])))
    return A
```

```
def interpol(points,Kx):
    '''
    This function computes the Lagrange Interpolation polynomial.
    Input: - A list of q sublists, where the i-th list contains the points
           (a_1,...,a_n) such that f(a_1,...,a_n)=c_i, K={c_0,...,c_{q-1}}.
           - A polynomial ring Kx.
    Output: The Lagrange Interpolation polynomial that interpolates the list.
            If len(points)!=q**n, the function returns 0.
    '''
    gens=Kx.gens()
    K=Kx.base_ring()
    q=K.cardinality()
    n=len(gens)

    f=0
    S=set()
    listK=list(K)
    for i in range(len(points)):
        S = S.union(set(points[i]))
    if len(S)==q**n:
        for i in range(q):
            list_i=points[i]
            for point in list_i:
                if n==1:
                    point=[point]
                f+=listK[i]*prod([1-(gens[j]-K(point[j]))**(q-1) for j in range(n)])
    return f
```

Also, to check if the polynomial obtained with `e_Klenian_OS` is in fact a companion of the polynomial defined by the input cycle, we may use the function `is_OP`, that determines whether or not two given bivariate polynomials form an orthogonal system.

```
def is_OP(f,g):
    '''
```

```
    This function determines whether or not two given bivariate polynomials form an
    orthogonal system.
    Input: Two bivariate polynomials f,g.
    Output: Boolean True/False.
    '''
    Kx=f.parent()
    K=Kx.base_ring()
    q=K.cardinality()
    A=cartesian_product([K,K])

    C=set()
    for a in A:
        C.add((f(a[0],a[1]),g(a[0],a[1])))
    if len(C)==q**2:
        return True
    return False
```

These last two functions compute the bivariate LPP associated to a Latin square with entries in $\mathbb{F}_q$ and viceversa.

```
def LPPoly(H,Kx):
    '''
    This function computes the bivariate LPP associated to a given Latin square.
    Input: - A Latin square H with entries in F_q (matrix or list of sublists);
           - The polynomial ring Kx=F_q[x,y].
    Output: The associated bivariate LPP
    '''
    K=Kx.base_ring()
    q=K.cardinality()
    u=K.primitive_element()

    L={k:[] for k in K}
    listK=list(K)
    for i in range(q):
        for j in range(q):
            L[K(H[i][j])].append((listK[i],listK[j]))
    return(interpol(list(L[i] for i in K),Kx))

def Latin_square(f):
    '''
    This function computes the Latin square associated to a given bivariate LPP.
    Input: A bivariate polynomial f.
    Output: The Latin square associated to f if f is an LPP, a zero matrix otherwise.
    '''
    Kx=f.parent()
```

```
    K=Kx.base_ring()
    q=K.cardinality()

    if is_LPP(f):
        return matrix(K,[[f(i,j) for j in list(K)] for i in list(K)])
    else:
        return zero_matrix(q,q)
```

`Latin_square` calls the function `is_LPP`, that determines whether or not a given bivariate polynomial is an LPP.

```
def is_LPP(f):
    '''
    This function determines whether or not a given bivariate polynomial is an LPP
    Input: A bivariate polynomial f.
    Output: Boolean True/False.
    '''
    Kx=f.parent()
    K=Kx.base_ring()
    q=K.cardinality()

    for i in K:
        image1 = []
        image2 = []
        for j in K:
            temp1 = f(i,j)
            temp2 = f(j,i)
            if temp1 in image1 or temp2 in image2:
                return False
            else:
                image1.append(temp1)
                image2.append(temp2)
    return True
```

APPENDIX B
Examples

B.1. The Package PermutationPolynomials

We will be using the SageMath package PermutationPolynomials introduced in Appendix A to produce examples of Local Permutation Polynomials. In particular, we will showcase examples of usage of the functions we commented on.

We begin by defining

$$\mathbb{F}_9 = \{u^i : i = 1, \ldots, 8\} \cup \{0\}, \quad \text{where } u^2 + 2u + 2 = 0$$

and $\mathbb{F}_9[x, y]$.

```
q=9
if is_prime(q):
    K=GF(q)
else:
    K.<u>=FiniteField(q)

S.<x,y>=PolynomialRing(K)
```

Now, we will find the e-Klenian polynomial f associated to the cycle

$$cc = (u, 0, u^2, u^3, u^6, u^8, u^7, u^5, u^4)$$

using the functions `cycle_set` and `interpol`, as well as a companion polynomial g of f with `e_Klenian_OS`.

```
cc=[u,0,u**2,u**3,u**6,u**8,u**7,u**5,u**4]
f=interpol(cycle_set(cc,K),S)
g=e_Klenian_OS(cc,S)
```

This results in the following polynomials:

$f = (u+1)\,x^7y^7+2ux^7y^6+(2u+1)\,x^6y^6+(u+2)\,x^5y^7+(u+1)\,x^7y^4+(2u+1)\,x^6y^5-x^5y^6+(u+1)\,x^4y^7-x^7y^3+(u+1)\,x^5y^5+ux^4y^6+x^3y^7+(u+1)\,x^7y^2+x^6y^3+ux^5y^4+2ux^4y^5+x^3y^6+(2u+1)\,x^2y^7+(2u+2)\,x^7y+(u+2)\,x^5y^3+x^4y^4+(u+2)\,x^3y^5+(2u+2)\,xy^7+ux^7+(u+2)\,x^6y+x^5y^2+2ux^4y^3+2ux^3y^4-x^2y^5+y^7+(u+2)\,x^6+(u+1)\,x^5y+(2u+2)\,x^3y^3+(u+2)\,x^2y^4+(u+2)\,xy^5+(2u+2)\,y^6+(2u+2)\,x^5-x^4y+x^3y^2+(2u+2)\,x^2y^3+xy^4+y^5+ux^4+(2u+1)\,x^3y+(u+2)\,x^2y^2+2uy^4+ux^3+(u+1)\,x^2y+xy^2-y^3+x^2+xy+uy^2-x+1,$

$g = (u+2)\,x^7y^7+(u+1)\,x^7y^6+(2u+2)\,x^6y^7+ux^7y^5+2ux^6y^6+2ux^5y^7+2ux^7y^4-x^6y^5+2ux^5y^6+2ux^4y^7-x^7y^3+(u+1)\,x^6y^4+ux^5y^5+x^4y^6+ux^3y^7+x^7y^2+(2u+1)\,x^6y^3+2ux^5y^4+(2u+2)\,x^4y^5+(2u+2)\,x^3y^6+(2u+1)\,x^2y^7+ux^7y+(u+2)\,x^6y^2+ux^4y^4+(2u+1)\,x^3y^5+(2u+1)\,x^2y^6+(2u+2)\,xy^7+(u+2)\,x^7+(2u+1)\,x^6y-x^5y^2+(u+2)\,x^4y^3+(2u+2)\,x^3y^4-x^2y^5+(2u+1)\,y^7+2ux^6+ux^5y+2ux^4y^2-x^3y^3+(2u+2)\,x^2y^4+xy^5+(u+1)\,y^6+(2u+2)\,x^5+(u+1)\,x^3y^2-x^2y^3+(2u+2)\,xy^4+y^5-x^4+(2u+1)\,x^2y^2+(2u+2)\,xy^3-y^4+(u+2)\,x^3+(u+1)\,x^2y+2uxy^2+(u+2)\,xy+(2u+2)\,y^2+(2u+1)\,x+2u+2.$

Now, using the function `is_OP` we check that f and g are, in fact, companions, since they form an orthogonal system, as well as show they are LPPs with `is_LPP`.

```
is_OP(f,g), is_LPP(f), is_LPP(g)
```

All of these return `True`.

We may also compute the Latin squares associated to f and g, since they are both LPPs, using `Latin_square`.

```
Lf=Latin_square(f)
Lg=Latin_square(g)
```

This results in the following matrices:

$$Lf = \begin{pmatrix} 1 & u+2 & 0 & u & 2u+2 & 2u & u+1 & 2 & 2u+1 \\ 0 & 1 & u & u+1 & u+2 & 2u+2 & 2u+1 & 2u & 2 \\ u+2 & 2u+2 & 1 & 0 & 2u & 2 & u & 2u+1 & u+1 \\ 2u+2 & 2u & u+2 & 1 & 2 & 2u+1 & 0 & u+1 & u \\ u & 0 & u+1 & 2u+1 & 1 & u+2 & 2 & 2u+2 & 2u \\ u+1 & u & 2u+1 & 2 & 0 & 1 & 2u & u+2 & 2u+2 \\ 2u & 2 & 2u+2 & u+2 & 2u+1 & u+1 & 1 & u & 0 \\ 2u+1 & u+1 & 2 & 2u & u & 0 & 2u+2 & 1 & u+2 \\ 2 & 2u+1 & 2u & 2u+2 & u+1 & u & u+2 & 0 & 1 \end{pmatrix}$$

$$Lg = \begin{pmatrix} 2u+2 & 2u & u+2 & 1 & 2 & 2u+1 & 0 & u+1 & u \\ 1 & u+2 & 0 & u & 2u+2 & 2u & u+1 & 2 & 2u+1 \\ 2 & 2u+1 & 2u & 2u+2 & u+1 & u & u+2 & 0 & 1 \\ u+1 & u & 2u+1 & 2 & 0 & 1 & 2u & u+2 & 2u+2 \\ u & 0 & u+1 & 2u+1 & 1 & u+2 & 2 & 2u+2 & 2u \\ 2u+1 & u+1 & 2 & 2u & u & 0 & 2u+2 & 1 & u+2 \\ 0 & 1 & u & u+1 & u+2 & 2u+2 & 2u+1 & 2u & 2 \\ 2u & 2 & 2u+2 & u+2 & 2u+1 & u+1 & 1 & u & 0 \\ u+2 & 2u+2 & 1 & 0 & 2u & 2 & u & 2u+1 & u+1 \end{pmatrix}$$

Finally, to check the proper functioning of `LPPoly` and `Latin_square`, we may carry out the following verification, which returns `True` in both cases.

```
1 LPPoly(Lf,S)==f, LPPoly(Lg,S)==g
```

B.2. e-Klenian Polynomials in $\mathbb{F}_2$, $\mathbb{F}_3$ and $\mathbb{F}_4$

To better understand the concept of e-Klenian polynomials, we are going to describe them in $\mathbb{F}_q[x,y]$ for $q = 2,3,4$. Actually, all local permutation polynomials in these rings are equivalent to e-Klenian polynomials, which isn't true when $q \leq 5$, see [9].

B.2.1. The Finite Field $\mathbb{F}_2$

As 2 is prime, there are only 0-Klenian polynomials over $\mathbb{F}_2 = \{0,1\}$. By Proposition 3.13, there are $\frac{2!(2-1)!}{\varphi(2)} = 2$ of them, and there is only $\frac{(2-1)!}{\varphi(2)} = 1$ 0-Klenian group.

The 0-Klenian group we are looking for is $\langle\beta\rangle = \{id, \beta\}$ where $\beta = (0,1)$, the only cycle of length 2. Thus, we get two 0-Klenian polynomials, f and g, where $\underline{\beta}_f = (id, \beta)$ and $\underline{\beta}_g = (\beta, id)$.

These are actually the only two local permutation polynomials in $\mathbb{F}_2[x,y]$,

$$f(x,y) = x + y, \quad g(x,y) = x + y + 1.$$

We know these are the only ones because, as LPPs, they have to be linear by Theorem 2.15 and both x and y have to appear in them by Proposition 2.7.

B.2.2. The Finite Field $\mathbb{F}_3$

Similar to the case $q = 2$, as 3 is prime, there are only 0-Klenian polynomials over $\mathbb{F}_3 = \{0,1,2\}$. By Proposition 3.13, there are $\frac{3!(3-1)!}{\varphi(3)} = 6$ of them, and there is only $\frac{(3-1)!}{\varphi(3)} = 1$ 0-Klenian group.

The 0-Klenian group we are looking for is $\langle\beta\rangle = \{id, \beta, \beta^2\}$ where $\beta = (0, 1, 2)$. Thus, we get 6 0-Klenian polynomials depending on how we order these 3 permutations.

The LPPs in $\mathbb{F}_3[x, y]$ are linear by 2.15, and all linear polynomials are LPPs. Therefore, they are

$$ax + by + c, \quad a, b \in \mathbb{F}_3^*,\ c \in \mathbb{F}_3.$$

This results in a total of $2 \cdot 2 \cdot 3 = 12$ local permutation polynomials in $\mathbb{F}_3[x, y]$, 6 of those being 0-Klenian polynomials.

B.2.3. The Finite Field $\mathbb{F}_4$

We will use the following description for $\mathbb{F}_4$:

$$\mathbb{F}_4 = \{0, u, u^2, u^3\} = \{0, u, u + 1, 1\}, \quad \text{where } u^2 + u + 1 = 0.$$

Since $q = 4 = 2^2$, e can be 0 or 1.

With $e = 0$, there are $\frac{(4-1)!}{\varphi(4)} = 3$ 0-Klenian groups:

- $K_1 = \langle\beta_1\rangle$, where $\beta_1 = (0, u, u^2, u^3)$.
- $K_2 = \langle\beta_2\rangle$, where $\beta_2 = (0, u^2, u, u^3)$.
- $K_3 = \langle\beta_3\rangle$, where $\beta_3 = (0, u^2, u^3, u)$.

By Proposition 3.13, these give $\frac{4!(4-1)!}{\varphi(4)} = 72$ 0-Klenian polynomials.

With $e = 1$ we have a group generated by $\alpha = (0, u)(u^2, u^3)$ and $\beta = (0, u^2)(u, u^3)$:

$$K_4 = \{id, \alpha, \beta, \alpha\beta\},$$

giving $4! = 24$ 1-Klenian polynomials.