CHAPTER 1

Introduction

Over the last two decades, several applications of the theory of algebraic curves over finite fields have been developed: for example, algebraic-geometric codes introduced by Goppa [Gop81], elliptic curve cryptography by Koblitz [Kob87] and Miller [Mil86], factorizations of large numbers by Lenstra Jr. [LJ87], generating pseudorandom sequences by Xing and Lam [XL99], and so on. In this thesis, we present new applications of algebraic curves to the constructions of combinatorial designs and combinatorial arrays, which in turn are useful in statistical experiments.

In this chapter, an introduction to algebraic curves will be given, and their precedent applications will be described. Definitions of balanced arrays and balanced n-ary designs will be also presented together with the sketches of our new constructions of them. Several practical usages of these arrays and designs will be discussed briefly at the end of this chapter.

1.1. Algebraic curves

An *algebraic curve* is the set of zeroes (roots) of a polynomial in two variables. Each element of a curve is called a *point* on the curve. For example, the polynomial $F(x, y) = y^2 - x^3 - x - 1$ defines a curve, and $(1, \sqrt{3})$ is a point on the curve. The points at infinity, that is, those with at least one of the coordinates being $+\infty$ or $-\infty$, are often considered as

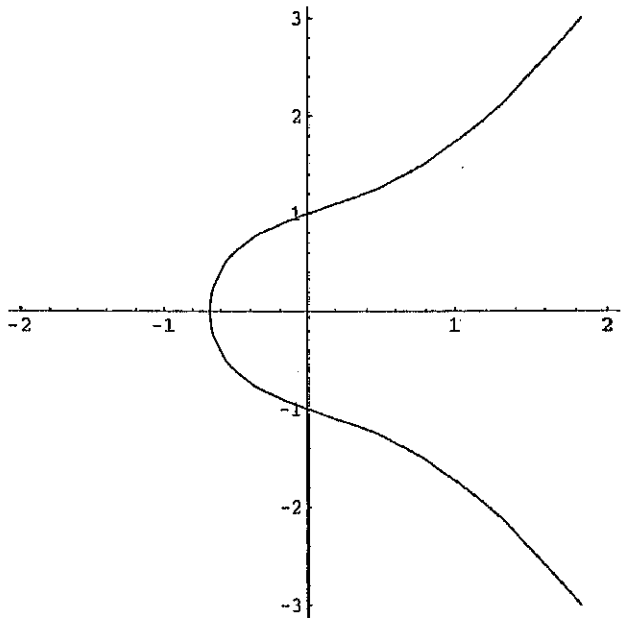

Figure 1. $y^2 = x^3 + x + 1$

points on an algebraic curve defined by $F = 0$ together with the points (x_0, y_0) satisfying $F(x_0, y_0) = 0$. However, in this chapter, a curve will mean the set of zeroes of a polynomial unless otherwise specified. We will show in Chapter 2 how the points at infinity can be included in the set of points on a curve.

A polynomial is said to be *defined over* a field K if all of its coefficients are from K. The set of polynomials with two variables x and y defined over K is denoted by $K[x, y]$. The polynomial F in the above example can be regarded as a polynomial defined over the real number field $\mathbb{R}$. On the curve C defined by this polynomial F, the points with x-coordinate being -1 have no value on y-coordinate in $\mathbb{R}$. However, if we consider the imaginary unit i such that $i^2 = -1$, then the point $(-1, i)$ is on the curve C. From these observations, it is useful if we consider the points on curves over the complex number field $\mathbb{C}$ rather than the real number field $\mathbb{R}$ or the rational number field $\mathbb{Q}$, even if the curves are defined over $\mathbb{R}$, $\mathbb{Q}$, and so on. In general, a curve is

considered over an *algebraic closure* $\bar{K}$ of a field K, that is, the coordinates of the points on the curve are in $\bar{K}$ when the curve is defined by a polynomial over K. An algebraic closure $\bar{K}$ of a field K is a field in which the roots of any polynomial with one variable defined over K always lie. Similarly, an *algebraically closed field* is a field in which any root of any polynomial with one variable defined over the field always lie. For example, the complex number field $\mathbb{C}$ is algebraically closed, while the real number field $\mathbb{R}$ is not.

An *intersection point* is a common root of two polynomials, and the *multiplicity* of the point is defined as the multiplicity of the root. By considering an algebraically closed field, we have a number of important results on algebraic curves, such as the following on the number of intersection points of two curves defined by polynomials of degree d_1 and d_2 respectively: the two curves intersect at $d_1 d_2$ points, counting multiplicities. Consider the system of equations $y = x^2$ and $y = 2x - 1$. Since $x^2 - 2x + 1 = 0$ has a multiple root $x = 1$, the zero $(1, 1)$ is a solution of the system with multiplicity 2, and there is no other intersection point.

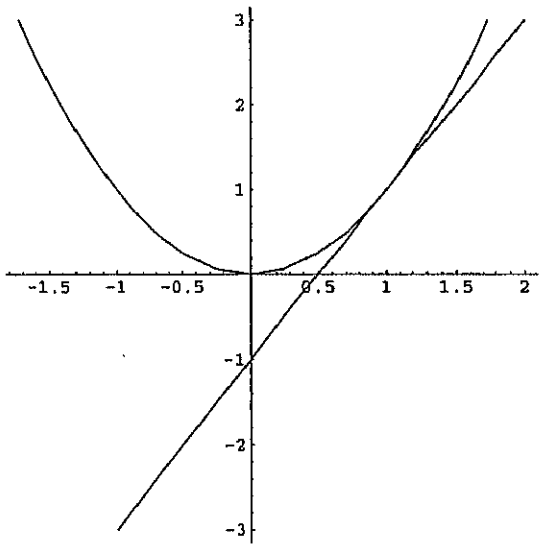

Figure 2. $y = x^2$ and $y = 2x - 1$

A *finite field* $\mathbb{F}_q$ is a field with q elements, q being a prime power. The number q of elements is called the *order* of $\mathbb{F}_q$. When the order of a finite field is a prime number p, the field can be regarded as $\{0, 1, \ldots, p-1 \pmod p\}$. Let us consider the curve $C : F = y^2 - x^3 - x - 1 = 0$ over $\mathbb{F}_5$. The points on C with coordinates in $\mathbb{F}_5$ are $(0, \pm 1)$, $(2, \pm 1)$, $(3, \pm 1)$ and $(4, \pm 2)$. When $y = 0$ there is no zero of F whose x-coordinate is in $\mathbb{F}_5$. This means that the line $y = 0$ does not meet the curve C. Let α be an element satisfying $\alpha^3 + \alpha + 1 = 0$, but not in $\mathbb{F}_5$. If the points on C are considered over a field which includes α and the elements of $\mathbb{F}_5$, then the point $(\alpha, 0)$ also becomes a point on the curve C. From these observations, we can say that any line intersects with any curve over a finite field $\mathbb{F}_q$ at some points by considering the points over an extension $\mathbb{F}_{q^m}$ of $\mathbb{F}_q$. Two finite fields K_1 and K_2 are isomorphic if they have the same number of elements. Hence the union $\bar{\mathbb{F}}_q = \mathbb{F}_q \cup \mathbb{F}_{q^2} \cup \mathbb{F}_{q^3} \cup \cdots$ of its all finite dimensional extensions of a field $\mathbb{F}_q$ is a closed field containing $\mathbb{F}_q$.

In the theory of algebraic curves, the complex number field $\mathbb{C}$ is often used as the ground field of curves. However, in its application to information sciences, an algebraically closed field containing a finite field is more preferable.

A point is said to be *K-rational* if the coordinates of the point lie in the field K. Suppose here that polynomials F and G are over a field K. A *divisor* on a curve is a formal sum $\sum_P m_P P$ of points on the curve with integral coefficients. A *rational function* is a fractional expression $f = F/G$ of two polynomials F and G. It is well known (see for example, [Mor91]) that a certain set of rational functions determined by a divisor forms a linear space over K. The interested reader is referred to Chapter 2 for details.

Let $\mathbb{F}_q$ be a finite field in which the order q is not a power of 2 or 3. An *elliptic curve* E is an algebraic curve defined by an equation

$$y^2 = x^3 + a_4 x + a_6,$$

where $a_4, a_6 \in \mathbb{F}_q$ and (a_4, a_6) satisfies $4a_4^3 + 27a_6^2 \neq 0$ in $\mathbb{F}_q$. Recall that an $\mathbb{F}_q$-rational point is a point (x, y) such that both x and y lie in $\mathbb{F}_q$. Any elliptic curve passes through exactly one point at infinity (see page 23). Let $E(\mathbb{F}_q)$ be the set of $\mathbb{F}_q$-rational points together with the point at infinity. It is well known (see for example, [**ST92**]) that $E(\mathbb{F}_q)$ forms a group under the following addition. For P_1 and $P_2 \in E(\mathbb{F}_q)$, we consider the line passing through the two points. The line always intersects the elliptic curve at a third point, say Q. Consider the vertical line through the point Q; this line passes through the point at infinity. The addition $P_1 + P_2$ is defined as the third intersection point of the vertical line with the elliptic curve. These steps for defining

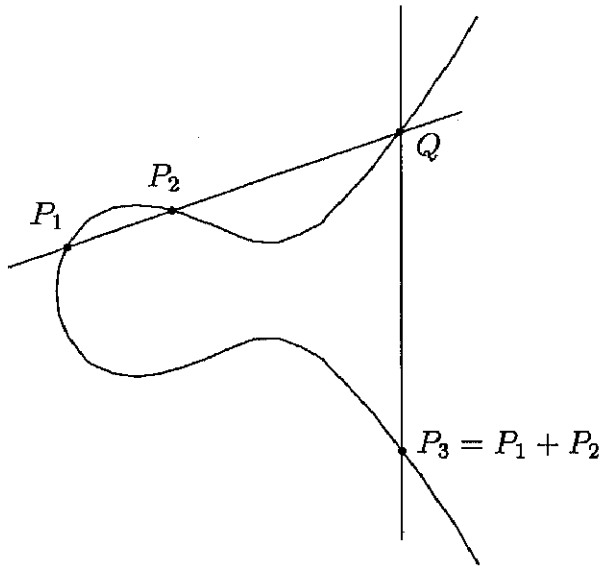

Figure 3. Addition on an elliptic curve

addition on points are illustrated in Figure 3.

In Chapter 2, the theory of algebraic curves will be described more precisely in connection with our new applications. The next section is devoted to some precedent applications of algebraic curves.

1.2. Precedent applications of algebraic curves

The theory of algebraic curves over finite fields has been applied to the theory of information sciences over the last two decades. In this section we mention some of such applications: algebraic-geometric codes, elliptic curve cryptography, factorizations of large numbers, and generating pseudorandom sequences for stream ciphers.

1.2.1. Algebraic-geometric codes

In communication through a digital channel, a sender transmits encoded data to a receiver, and then the receiver decodes the data for restoring the original message. Error-correcting codes are systems which can detect and correct errors caused by noises in transmission channels. An algebraic-geometric code is one of error-correcting codes introduced by Goppa [Gop81] as a possible generalization of Reed-Solomon codes, BCH-codes and "classical" Goppa codes.

Let C be a subset of an n-dimensional vector space. C is called a *code* of *length* n and its elements are *codewords*. An *encoding* is a transformation of the set of original messages, called *source*, to a code C. When a sender transmits codewords to a receiver, some errors may occur, so that the receiver receives data different from the original codewords. The detection and correction of errors are made by the maximum likelihood principle, that is, the receiver picks the codeword nearest to the received data in terms of some distance. The most commonly used is the *Hamming distance* $d(x, y)$ which is defined to be the number of coordinates in which $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$ differ, that is, $d(x, y) = |\{i : x_i \neq y_i\}|$. A *linear code* C, which is one of the most popular error-correcting codes, is a k-dimensional subspace of an n-dimensional linear space. Algebraic-geometric codes are linear codes generated from linear spaces of rational functions. Recall that a rational function over a finite field $\mathbb{F}_q$ is a fractional expression of

two polynomials over $\mathbb{F}_q$. If a rational function over $\mathbb{F}_q$ has a representation $f = F/G$ with $G(P) \neq 0$ for a $\mathbb{F}_q$-rational point P, then the function f has a value $f(P) = F(P)/G(P)$ in $\mathbb{F}_q$. We can therefore define a map from a set of n $\mathbb{F}_q$-rational points $P_1, \ldots, P_n$ to n-tuples $(f(P_1), \ldots, f(P_n))$ in $\mathbb{F}_q^n$. It is well known (see for example, [Mor91]) that a certain set of rational functions is a linear space, which leads to a linear code C consisting of n-tuples $(f(P_1), \ldots, f(P_n))$ for all rational functions of the aforementioned subset. This is fundamental in constructing algebraic-geometric codes.

The most important result on algebraic-geometric codes is that a family of the codes asymptotically achieves the Varshamow-Gilbert bound, which is a lower bound of information rate k/n for the series of optimal codes with the same relative minimum distance d/n [TVZ82]. Another important result is that all linear codes are algebraic-geometric codes. There are several books on algebraic-geometric codes, see for example, [Gop91], [vLvdG88] and [TV91].

1.2.2. Elliptic curve cryptography

In communication through insecure open networks such as the Internet, cryptosystem is indispensable to defend transmitted secret information against impersonation and substitution by strangers.

Before we review how the cryptography works by using elliptic curves, we redefine the notion of elliptic curves together with the formulae of addition in the group of points on the curve. Let $\mathbb{F}_q$ be a finite field of order $q \neq 2, 3$, and E the elliptic curve defined by

$$y^2 = x^3 + a_4 x + a_6,$$

where $a_4, a_6 \in \mathbb{F}_q$. The addition of the group $E(\mathbb{F}_q)$ is formulated as follows: for $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ in $E(\mathbb{F}_q)$, $P_3 = P_1 + P_2 =$

(x_3, y_3) is defined as

$$\begin{cases} x_3 = \lambda^2 - x_1 - x_2, \\ y_3 = \lambda(x_1 - x_3) - y_3, \end{cases}$$

where

$$\lambda = \begin{cases} (y_2 - y_1)/(x_2 - x_1) \text{ for } x_1 \neq x_2 \\ (3x_1^2 + a_6)/2y_1 \text{ for } P_1 = P_2. \end{cases}$$

The inverse of $P_1 = (x_1, y_1)$ is $-P_1 = (x_1, -y_1)$. Note that these formulae represent the addition defined geometrically in Section 1.1 and illustrated in Figure 3.

Let us consider a situation that Alice wants to send a message $M \in E(\mathbb{F}_q)$ to Bob through an insecure channel. Secure communication is established by the following steps.

1. Bob opens $P = \alpha Q$ and Q to the public, where α is an integer randomly chosen, and holds α secret.

2. Alice sends $(B_1, B_2) = (M + \beta P, \beta Q)$, where β is an integer randomly chosen, and holds β secret.

3. Bob decrypts the received message by calculating $B_1 - \alpha B_2 = M$.

The information $P = \alpha Q$ and Q opened by Bob are called *public keys*, and the information α and β held secretly by each of Alice and Bob are *secret keys*. The security of this system depends on the difficulty of finding the secret keys α and β from the public information Q, αQ and βQ. The problem of finding an integer α from αQ over a finite group is called the *discrete logarithm* problem. The elliptic curve cryptography is a modification of a cryptography introduced by ElGamal [ElG85] based on the discrete logarithm over a group of integers. This modification can obtain the ability of defence against known attacks.

1.2.3. Factorization of large numbers

There are a number of factoring algorithms. In this section we review the method of factorization by means of the theory on elliptic curves, which was introduced by Lenstra Jr. [LJ87] in 1987.

Suppose that we want to factor the composite number n which has a prime factor $p > 3$. Let E be an elliptic curve defined by the equation $y^2 = x^3 + ax + b$, where $a, b \in \mathbb{Z}$ and $\gcd(4a^3 + 27b^2, n) = 1$. In the same way as that in the previous section, we consider the addition of two points on E, that is, if $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$, then $P_1 + P_2 = (x_3, y_3) = (\lambda^2 - x_1 - x_2, \lambda(x_1 - x_3) - y_3)$, where

$$\lambda = \begin{cases} (y_2 - y_1)/(x_2 - x_1) & \text{if } x_1 \neq x_2, \\ (3x_1^2 + b)/2y_1 & \text{if } P_1 = P_2, \end{cases}$$

and the inverse of P_1 is $-P_1 = (x_1, -y_1)$. Let $E(\mathbb{Q})$ be the set of points on E whose coordinates lie in the set of rational numbers $\mathbb{Q}$ and the point at infinity. By taking the point at infinity as the identity, $E(\mathbb{Q})$ forms a group under the above addition. Let $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2) \neq -P_1$ be two points on E whose coordinates have denominators relatively prime to n. Consider the elliptic curve E as the elliptic curve E' defined over a finite field $\mathbb{F}_p$ of order p, that is, E' is defined by the equation $y^2 = x^3 + a'x + b'$, where $a \equiv a' \pmod{p}$ and $b \equiv b' \pmod{p}$. The points P_1 and P_2 can be replaced by $\mathbb{F}_p$-rational points P_1' and P_2' by reducing their coordinates modulo p, since the denominators of the coordinates are relatively prime to n, and since p is a prime factor of n.

Theorem 1.2.1 ([Kob94]). *The point $P_1 + P_2 \in E(\mathbb{Q})$ has coordinates with denominator prime to n if and only if there is no prime p dividing n such that the point $P_1' + P_2' \in E'(\mathbb{F}_p)$ is not the point at infinity.*

This theorem plays an important role in Lenstra's algorithm to find a factor of a composite number n. Suppose that, for a prime factor p of n, an integer k is divisible by the number of points in $E'(\mathbb{F}_p)$. Then any element of $E'(\mathbb{F}_p)$ has an order which is a factor of $|E'(\mathbb{F}_p)|$, and kP' is the point at infinity, where P' is the point in $E'(\mathbb{F}_p)$ corresponding to a point $P \in E(\mathbb{Q})$. From Theorem 1.2.1, the denominator d of $kP \in E(\mathbb{Q})$ has common factors with n, that is, $\gcd(d, n)$ is a nontrivial factor of n if d is not divisible by n. From these observations, we have a method of factorization called Lenstra's algorithm.

Lenstra's elliptic curve algorithm. Let $n > 2$ be a composite integer for which we are to find a factor.

Step 0: Check that the greatest common divisor $\gcd(n, 6)$ is 1 and that n does not have the form m^r for any $r \geq 2$.

Step 1: Choose random integers x_1, y_1, a between 1 and n.

Step 2: Let $b \equiv y_1^2 - x_1^3 - ax_1 \pmod{n}$, let E be the cubic curve $E : y^2 = x^3 + ax + b$, and let $P = (x_1, y_1) \in E$.

Step 3: Check that $c = \gcd(4a^3 + 27b^2, n) = 1$. If $c = n$ then go back to Step 1. If $1 < c < n$ then c is a non-trivial factor of n, so we are done.

Step 4: Choose an integer k which is divisible by powers of small primes, for example the least common multiple

$$k = \mathrm{lcm}(1, 2, \ldots, M)$$

for an integer M.

Step 5: Compute

$$kP = \left(\frac{x_k}{d_k^2}, \frac{y_k}{d_k^3}\right)$$

in the group $E(\mathbb{Q})$. If $\gcd(d_k, n)$ is strictly between 1 and n, then D is a non-trivial factor of n and we are done. If $\gcd(d_k, n)$ is

equal to 1, then either go back to Step 4 and increase k, or go back to Step 1 and choose a new elliptic curve. If $\gcd(d, n)$ is equal to n, then go back to Step 4 and decrease k.

Generally speaking (see for example, [Sti95]), Lenstra's method is faster than other factorization algorithms when the prime factors of n are of differing size. However, when n is a composite of large primes, this method is not so fast.

1.2.4. Sequences for stream ciphers

Generating long pseudorandom sequences from short seeds is very important for the theory of stream ciphers [Rue86, Rue92].

Let $x = x_1x_2\ldots$ be a string of plaintext and $y = y_1y_2\ldots$ the string of the encrypted ciphertext of x. The basic idea of stream ciphers is to generate a keystream $z = z_1z_2\ldots$, and use it to encrypt the plaintext string x according to the rule

$$y = y_1y_2\ldots = e_{z_1}(x_1)e_{z_2}(x_2)\ldots,$$

where e_{z_i}'s are encryption functions. A pseudorandom sequence generated from a 'seed' key is used as the key stream z. A stream cipher is often described in terms of binary alphabet. In this situation, the encryption operation is the addition modulo 2, i.e.,

$$e_{z_i} = x_i + z_i \pmod 2.$$

Decrypting the ciphertext string y can be accomplished by computing the key stream from the seed key shared by the sender and the receiver. Hence, the corresponding decryption operation is

$$d_{z_i} = y_i + z_i \pmod 2.$$

Clearly, we need long sequences with unpredictability and randomness generated from short seeds. A sequence $(a_1, a_2, \ldots, a_n)$ in $\mathbb{F}_q$ is

called a *k-th order linear feedback shift-register sequence* if there exist constants $\lambda_0, \ldots, \lambda_k \in \mathbb{F}_q$ with $\lambda_k \neq 0$ such that $\lambda_0 a_i + \lambda_1 a_{i+1} + \cdots + \lambda_k a_{i+k} = 0$ for all $1 \leq i \leq n-k$. The *linear complexity* of a sequence of length n, which is a well-known measure of unpredictability and randomness, is defined to be the least k such that the sequence is a k-th order linear feedback shift-register sequence. An infinite sequence $(a_1, a_2, \ldots)$ is called *d-perfect* if for any $n \geq 1$ the linear complexity of the subsequence $(a_1, a_2, \ldots, a_n)$ of length n is greater than or equal to $(n+1-d)/2$. Let $P = (x, y)$ be an $\mathbb{F}_q$-rational point on a curve C defined over $\mathbb{F}_q$. The point P can be represented by a parameter t as $x = \sum_i^\infty x_i t^i$ and $y = \sum_i^\infty y_i t^i$, where $x_i, y_i \in \mathbb{F}_q$. The expansion of a rational function $f(x, y)$ at P has the form $f = \sum_i^\infty \alpha_i t^i$. Consider the sequence $\boldsymbol{\alpha}(f)$ of the coefficients of the above expansion except α_0, that is, $\boldsymbol{\alpha}(f) = (\alpha_1, \alpha_2, \ldots)$. Xing and Lam [XL99] showed that the sequence $\boldsymbol{\alpha}(f)$ is d-perfect if P and f satisfy certain conditions. This is one of the latest applications of algebraic curves to information sciences.

1.3. A new application to balanced arrays

Unlike those mentioned in the preceding section, the applications in this section and the next section use intersection multiplicities. In this section, we first give the definition of balanced array.

Let S be a set $\{0, 1, \cdots, s-1\}$ of s elements. A *balanced array*, denoted by $\mathrm{BA}(v, b, s)$ or simply BA, is a $v \times b$ matrix $\boldsymbol{A}$ with entries from S satisfying the following conditions:

(A1): for any 2-rowed submatrix $\boldsymbol{A_0}$ of $\boldsymbol{A}$, any ordered pair (x, y) of elements from S occurs exactly $\mu(x, y)$ times as columns in $\boldsymbol{A_0}$, and

(A2): $\mu(x, y) = \mu(y, x)$ for every pair of x and y.

The $\mu(x,y)$'s are called the *indices*, and v the number of *constraints*. When $\mu(x,y) = \mu$ for every pair (x,y), the balanced array becomes an *orthogonal array*, which is denoted by $\mathrm{OA}(v,b,s)$.

The notion of balanced array was first introduced by Chakravarti [Cha56] in connection with statistical designs. Later on, many people have contributed to the theory and construction of balanced arrays. A historical note on balanced arrays will be mentioned in Chapter 3.

The fundamental idea of our new constructions in this thesis is to consider the intersection multiplicities of curves as entries in some array. Let C_0 be a curve defined over $\mathbb{F}_q$, V a set of points on the curve C_0, and $\mathcal{C}$ a set of curves over $\mathbb{F}_q$. We assign the points of V to the rows, and the curves of $\mathcal{C}$ to the columns to construct a $|V| \times |\mathcal{C}|$ array. Each entry of the array is the intersection multiplicity at the corresponding point of the corresponding curve with the curve C_0. The resultant array, however, is not always a balanced array. Therefore we require the triple $(C_0, V, \mathcal{C})$ to satisfy the following conditions:

- for any point $P \in V$, the number of curves of $\mathcal{C}$ having intersection multiplicity α is exactly λ_α, and
- for any ordered pair (P,Q) of distinct points of V, the number of curves $C \in \mathcal{C}$ satisfying $I_P(C,C_0) = \alpha$ and $I_Q(C,C_0) = \beta$ is equal to $\lambda_{\alpha,\beta}$,

where $I_P(C,C_0)$ is the intersection multiplicity at the point P of the curves C and C_0. The triple $(C_0, V, \mathcal{C})$ satisfying the above conditions is called a *symmetric set of curves*, and it yields a balanced array. In Chapter 3, we will discuss the construction methods of symmetric sets of curves by using linear spaces of rational functions. When the degree of the polynomial which defines the curve C_0 is greater than or equal to 3, it is difficult to construct such symmetric sets of curves by this method. For the case that C_0 is an elliptic curve, i.e. the degree is 3, one method to obtain the point set V and the curve set $\mathcal{C}$ satisfying the conditions of symmetric sets of curves will be also proposed in Chapter 3.

1.4. A new application to balanced n-ary designs

A balanced n-ary design is similar to the well-known balanced incomplete block design (BIBD) except that its blocks are multisets with multiplicities $0, 1, \ldots,$ and $n-1$. In this section, we give the definition of balanced n-ary designs and describe how algebraic curves can be applied to construct the designs.

Let V be a set of v elements and $\mathcal{B}$ a collection of b multi-subsets of V. The elements of V and $\mathcal{B}$ are called *treatments* and *blocks*, respectively. A *block design* $(V, \mathcal{B})$ is an arrangement of v treatments of V into b blocks of $\mathcal{B}$. A *balanced n-ary design*, denoted by BnD, is a pair $(V, \mathcal{B})$ satisfying

(B1): each block is of a constant size k,

(B2): each treatment occurs at most $n-1$ times in any block $B \in \mathcal{B}$, and

(B3): each unordered pair of distinct treatments occurs exactly λ times in the blocks of $\mathcal{B}$.

Note that, for example, the block size of $B = \{x, x, x, y, y, z\}$ is 6 since the treatments x, y and z occur 3 times, twice and once, respectively, and the pairs $\{x, y\}$, $\{y, z\}$ and $\{x, z\}$ are counted 6, 2 and 3 times, respectively, in the block B.

Remark. To distinguish balanced n-ary designs from binary designs, capital letters have been traditionally used for the parameters of n-ary designs. In this thesis, however, the parameters are denoted by small letters since we will mainly discuss designs of n-ary types.

Equivalently to the above definition, balanced n-ary designs could be defined by a matrix. Let $\boldsymbol{N} = (n_{ij})$ be a $v \times b$ matrix such that n_{ij} is the number of occurrences of the i-th treatment in the j-th block. The matrix $\boldsymbol{N}$ is called the *incidence matrix* of a balanced n-ary design. Using the incidence matrix, the conditions in the definition of a balanced n-ary design is rewritten as follows:

(B1′): $\sum_i n_{ij} = k$ for any j,

(B2′): $0 \leq n_{ij} \leq n-1$ for any i, j, and

(B3′): $\sum_j n_{ij} n_{i'j} = \lambda$ for any unordered pair $\{i, i'\}$, $i \neq i'$.

There is a close connection between finite geometries and block designs. For example, in an affine plane over a finite field, the point set V and the line set $\mathcal{B}$ form a balanced incomplete block design $(V, \mathcal{B})$. In traditional methods of constructions using finite geometries, we have regarded geometric objects as treatments and sets of objects as blocks of designs. Since the resultant blocks are not multisets, designs with repeated elements in blocks cannot be directly produced in traditional ways. The fundamental idea in this thesis for constructing balanced n-ary designs is that the intersection multiplicities of algebraic curves are considered as the multiplicities of treatments in blocks. This can be thought of as a generalization of the traditional correspondence between binary designs and finite geometries, since the intersection multiplicities can be regarded as the multiplicities of points on algebraic varieties, which are sets of points defined as common roots of some equations. In Chapter 4, we will present some construction methods of balanced n-ary designs by applying the theory of algebraic curves.

Similarly to the construction methods of balanced arrays in Chapter 3, the incidence matrix of a balanced n-ary design can be obtained from a linear space of rational functions. One of the differences between a balanced n-ary design and a balanced array is that the incidence matrix of a balanced n-ary design must have constant column sum so that the design have a constant block size, while a balanced array does not. For a curve C_0, we can define the divisor of a curve C on C_0 as $\sum_P m_P P$, where m_P is the intersection multiplicity at the point P of the curves C_0 and C. This means that a curve can be represented by its unique divisor. If two divisors on a curve C_0 are *linearly equivalent* (see Chapter 4 for its formal definition), then the corresponding curves have the same number of intersection points with C_0. By applying this

equivalence, we can obtain designs with constant block size from sets of curves.

1.5. Practical usages of balanced n-ary designs and balanced arrays

Balanced n-ary designs and balanced arrays have applications in the design of experiments. In this section, we briefly mention some statistical designs based on the balanced n-ary designs and balanced arrays, such as block designs, balanced fractional factorial designs and weighing designs.

1.5.1. Block designs

Block designs were originally used in agricultural experiments with the aim of allowing all treatments to be compared within similar conditions. First, we introduce some terminologies used in the theory of experimental designs and review elementary properties of block designs.

A *block* is a set of experimental units, and treatments are assigned to the units. For example, in an experiment to compare three varieties of wheat in five different farms, the blocks are the five farms which are divided into several plots. A *design* is an allocation of v treatments to N plots grouped in b blocks. A balanced n-ary design can be used as a design whose blocks may have some experimental units receiving the same treatments.

A $v \times b$ matrix $\boldsymbol{N} = (n_{ij})$ is called the *incidence matrix* of a balanced n-ary design if each entry n_{ij} is the number of units in the j-th block that receive the i-th treatment. Let $\boldsymbol{D}$ be a $b \times N$ matrix with elements 0 and 1, the (i, j)-entry being 1 if the j-th unit is in the i-th block, and 0 otherwise. The transposed matrix $\boldsymbol{D}^T$ is called the *design matrix for blocks*. Similarly, let Δ be a $v \times N$ matrix in which the (i, j)-entry is 1 if the j-th unit receives the i-th treatment, and 0 otherwise. The

transposed matrix Δ is called the *design matrix for treatments*. It can be easily seen that the relation $\Delta D^T = N$ holds.

Now we consider an experiment using a balanced n-ary design. Let y_l be the yield of the l-th plot in the j-th block sown with the i-th variety, τ_i the effect of the i-th variety, and β_j the effect of the j-th block. Then we have the linear statistical model

$$y_l = \tau_i + \beta_j + \epsilon_l,$$

where ϵ_l is a random error. We can write this in vector notation as

$$y = \Delta^T \tau + D^T \beta + \epsilon,$$

where $y = (y_1, \ldots, y_N)^T$,$\tau = (\tau_1, \ldots, \tau_v)^T$, $\beta = (\beta_1, \ldots, \beta_b)^T$, and $\epsilon = (\epsilon_1, \ldots, \epsilon_N)^T$. By putting $X = (\Delta^T D^T)$ and $\gamma = (\tau^T \beta^T)^T$, we have

$$y = X\gamma + \epsilon.$$

Suppose that the expected value of the error term ϵ_l is zero and that the variance is σ^2. When we choose b satisfying the *normal equation* $X^T X b = X^T y$, we have a least squares estimator $\hat{\gamma}$ of γ from $X\hat{\gamma} = Xb$. (See, for example, [SS87]).

When a design satisfies the conditions (B1) and (B3), we can easily obtain an unbiased estimator. This is one of the reasons why balanced n-ary designs are suitable for the experiments discussed above. For more details on statistical analysis, the interested reader is referred to [CK96].

1.5.2. Balanced fractional factorial designs

A *factor* in an experiment is an attribute of the experimental units which may affect the response observed in the experiment. Any factor may take one of several values which are called the *levels* of the factor.

For example, in an experiment to compare three varieties of wheat in five different farms, a factor is 'wheat' with three levels. Block designs can be used for experiments which have two factors, one is the treatment factor and the other is the blocking factor. In an experiment with more factors, balanced fractional factorial designs are very useful. Such designs can be constructed from balanced arrays.

Let us consider the linear model

$$y = X\tau + \epsilon,$$

where y is an $N \times 1$ vector of observations, X^T a $v \times N$ design matrix, τ a $v \times 1$ vector of treatments, and ϵ an $N \times 1$ vector of errors. Each treatment is assigned to a factor with s levels, that is, the design matrix X is s-ary. The aim of this experiment is to estimate the treatment effects which have up to l factors. A design based on this model is a balanced fractional factorial design. More precisely, if the effects involving up to l factors are estimable under a design, then the design is called a *fractional factorial design of resolution* $2l + 1$; if, moreover, the matrix $(X^T X)^{-1}$ is invariant with respect to any permutation of factors, then the design is called a *balanced fractional factorial design of resolution* $2l + 1$.

The design matrix X of a balanced fractional factorial design can be obtained from a balanced array. Many people have studied the connections between balanced fractional factorial designs and balanced arrays: Srivastava [**Sri70**], Srivastava and Anderson [**SA70**], Yamamoto, Shirakura and Kuwada [**YSK75**], Shirakura and Kuwada [**SK75, SK76**], Shirakura [**Shi75, Shi76, Shi77**], Kuwada [**Kuw79**], Kuwada and Nishii [**KN79**], and Hyodo [**Hyo92**]. In particular, Kuwada and Nishii [**KN79**] established a connection between balanced fractional factorial designs of resolution $2l + 1$ with m factors of s levels and balanced arrays with s symbols.

1.5.3. Weighing designs

A weighing design is another application of balanced n-ary designs to the theory of experimental designs. Let us consider the problem of weighing v objects with a balance which has no bias. An easy way of weighing is to put objects in one pan, and then known weights in the other pan. Hotelling's [Hot44] improvement of this method is to put certain objects in one pan (left hand), next the remaining objects in the other (right hand), and then balance these pans on the scale by adding known weights. Let $x_{ui} = 1$ or -1 if the i-th object is included in the u-th weighing by being placed in the left or right hand pan, and let $x_{ui} = 0$ if the i-th object is not included in the u-th weighing, where $i = 1, 2, \ldots, v$ and $u = 1, 2, \ldots, N$. Let y_u be the result recorded for the u-th weighing, that is, y_u is the total weight of the known weights placed in the right hand pan, and let ϵ_u be the error in y_u. If τ_i is the true weight of the i-th object, we have

$$y = \Delta\tau + \epsilon,$$

where

$$\begin{aligned}
y^T &= (y_1, y_2, \ldots, y_N) \text{ is the vector of observations,} \\
\Delta^T &= (x_{iu}) \text{ the } v \times N \text{ design matrix,} \\
\tau^T &= (\tau_1, \tau_2, \ldots, \tau_v), \text{ and} \\
\epsilon^T &= (\epsilon_1, \epsilon_2, \ldots, \epsilon_N) \text{ the vector of errors.}
\end{aligned}$$

Then we know that the least squares estimator of τ is obtained from

$$\hat{\tau} = (\Delta^T\Delta)^{-1}\Delta^T y,$$

and the variance-covariance matrix is given by $\sigma^2(\Delta^T\Delta)^{-1}$, σ^2 being the variance of the experimental error. The weighing problem therefore

reduces to the investigation of the matrix $\Delta^T\Delta$ so that the true weights can be estimated with minimum variance.

Balanced n-ary designs were applied by Murty and Das [**MD67**] and Saha and Dey [**SD73**] to produce the design matrices Δ of weighing designs by replacing the entries $\{0, 1, \ldots, n-1\}$ of the incidence matrices with $\{-1, 0, 1\}$.