Universidad del Atlántico

MATUA

Revista MATUA Páginas: 32–49

**Facultad de Ciencias Básicas**
©Programa de Matemáticas
Vol. VI , Nº 2, (2019)

Received Dec 1, 2019; accepted Dec 30, 2019.

# Bernstein polynomials and stochastic computing

# Polinomios de Bernstein y codificación estocástica de la información

Yamilet Quintana

*Departamento de Matemáticas Puras y Aplicadas, Edificio Matemáticas y Sistemas (MYS), Apartado Postal: 89000, Caracas 1080 A, Universidad Simón Bolívar, Venezuela*

*yquintana@usb.ve*

**Abstract**

Among the multiples applications of Bernstein polynomials there is one related to the processing of random signals, originally introduced by John von Neumann in 1956. Thanks to advances in technology, some ideas from the late sixties of the last century have been retaken in order to design implementations which allow -in certain cases- a simpler and more efficient processing than the traditional one. In this descriptive review article we will illustrate the use and importance of Bernstein polynomials in solving problems associated with stochastic computing, taking as a starting point the notion of stochastic logic in the sense of Qian-Riedel-Rosenberg.

*Keywords:* stochastic computing, stochastic logic, Bernstein polynomials.

**Resumen**

Entre las aplicaciones de los polinomios de Bernstein se encuentra el procesamiento de señales aleatorias, originalmente presentado por John von Neumann en 1956. Gracias a los avances de la tecnología se han podido retomar algunas ideas -de finales de los años sesenta del siglo pasado- para diseñar implementaciones que permiten un procesamiento más simple y eficiente que el tradicional en determinados casos. En este artículo de revisión descriptiva ilustraremos el uso e importancia de los polinomios de Bernstein en la resolución de problemas asociados a la codificación estocástica de la información, tomando como punto de partida la noción de lógica estocástica en el sentido de Qian-Riedel-Rosenberg.

*Palabras claves:* codificación estocástica de la información, lógica estocástica, polinomios de Bernstein.

## 1. Introducción

As B.R. Gaines pointed out in [9], the invention of the steam engine in the late eighteenth century allowed the replacement of the muscle-power of men and animals by the motive power of machines. While, the invention of the stored-program digital computer during the second world war made it possible to replace the lower-level mental processes of the human being (such as arithmetic computation and information storage), by electronic data-processing in machines [36, 37].

In the mid-twentieth century, Boolean algebras were of great practical importance, which was increasing until today, mainly in the management of digital information. The first computers were composed of decimal numbering system, but in 1930 John von Neumann proposed replacing the decimal numbering system with a binary numbering system, which as we know, is nothing more than a system based on symbolic logic introduced by Boole [11, 12]. With this binary numbering system, von Neumann was able to state the architecture model that defines the internal structure of computers since the first generation and with that, the era of digital computing began. In 1936, Alan Turing used Boolean algebras theoretically, in his design of the Turing [35] machine.

By the end of the sixties of twentieth century we had reached the stage in which it was reasonable to contemplate replacing some of the higher mental processes of the human being, such as the ability to recognize patterns and to learn, with similar capabilities in machines. However, the technology needed to 'infuse the gift of learning and recognition of patterns to a machine'was lacking. Nowadays, and with well-defined levels or stages of development, we are still immersed in a real technological transformation [3, 4, 18, 19, 20, 34, 38, 39].

In the summer of 1965 some research teams working on these topics, independently discovered a new form of computer (or coding structure) which seemed to provide the necessary technology to introduce learning and pattern recognition in machines [6, 7, 8, 9, 24, 31]. The *stochastic computer*, as this coding structure has come to be called, was not the final answer to the technological problems of learning and pattern recognition. Although, it did allow progress in the direction of processing by parallel structures similar to those of the human brain, and is of great interest in itself as a novel addition to the family of basic computing techniques, and as a system which uses what is generally considered as a disposable product: the random noise (cf. [1, 2, 3, 10, 15, 25, 27, 34] and the references therein).


Figura 1: Stochastic Computing.

Thus stochastic computing (SC) arose, as a collection of techniques to represent analog quantities by probabilities of discrete events, or represent continuous values by means of random bit-streams, so that complex operations can be performed by simple bitwise operations on random pulse trains[1]. In Spanish, the term 'stochastic computing'might present some confusion with the term 'stochastic calculus', so it

[1]The bits are closely linked to the processing of random signals, originally presented by J. von Neumann in [37]. Bit is the

is worthy to mention these terms are very different. Some authors also use the terms 'stochastic arithmeticór 'stochastic coding of the information'to refer to stochastic computing, as illustrated in Figure 1. Also, despite the similarity of their names stochastic computing is different from the study of random algorithms.

The analogy between probability algebras and Boolean algebras [11, 12, 13] is used to obtain very simple processing units and an adequate arithmetic. The basic operations described in the literature are the addition and the multiplication since these are the fundamental operations involved in neural networks and in the design of stochastic circuitry (fields in which fertile ground has been found for applications of SC). Fortunately, the advancement of technology in relation to programmable devices has allowed to retake those ideas from the sixties of the last century for reaching implementations. These implementations, being totally digital, allow a stochastic processing, which is much simpler and more efficient than the traditional calculation in certain cases [1, 2, 3, 15, 18, 25, 27, 34, 38, 39].

In this descriptive review article we will illustrate the use and importance of Bernstein polynomials in solving problems associated with stochastic computing, taking as a starting point the notion of stochastic logic in the sense of Qian-Riedel-Rosenberg [25, 28].

## 2. Stochastic numbers

The information stored on a computer is measured, encoded and transmitted as a finite sequence of switches ('on'represents one and 'off'represents zero) which are represented by bits. For instance, file sizes and transfer rates are measured in bits. In general, all information entered in the language of the user is converted into bits for the computer 'understands it'. Bits are also used to classify the colors in an image. For example, a monochrome image has 1 bit at each point (white or black), while a 8 bit image supports up to $2^8 = 256$ colors. For systems of 32 or 64 bits, these numbers indicate the capacity of the computer for processing the size of the data types that it handles and the size of its registry all at once (i.e., in a single cycle of the processor). Also, they usually can mean the amount of bits used to represent an address in the computer memory.

A *stochastic number* can be defined as a pair $(x, p_x)$, where $x$ a finite binary sequence, i.e., $x \in \{0,1\}^N$, for some $N \in \mathbb{N}$ and $p_x \in [0,1]$ is the probability of observing a 1 at an arbitrary position of $x$, [1, 8, 9, 20]. So, a stochastic number is represented by a finite binary sequence (or bit-stream) in such a way that the probability (ratio) of '1ín the binary sequence is interpreted as the number itself. Some authors call to the probability $p_x$ *value of the stochastic number* (see, e.g., [20]).

For example, if $N = 16$ the following pairs of finite binary sequences and probabilities represent stochastic numbers:

$$\begin{aligned} x &= (0,1,1,1,0,1,1,1,0,1,1,1,0,1,1,1) \quad \longmapsto p_x = \frac{12}{16} = \frac{3}{4}, \\ y &= (0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1) \quad \longmapsto p_y = \frac{12}{16} = \frac{3}{4}. \end{aligned}$$

Note that although $x \neq y$, their associated probabilities coincide. Hence, the pairs $(x, p_x)$ and $(y, p_y)$ represent the same stochastic number. Neither the length nor the structure of a finite binary sequence $x$ need be fixed; for example, the probability $p = \frac{1}{4}$ is associated with the binary sequences $x_p = (1,0,0,0)$, $y_p = (0,1,0,0)$ and $z_p = (0,1,0,0,0,1,0,0)$, so the stochastic numbers $(x_p, p)$, $(y_p, p)$ and $(z_p, p)$ have the same value, that is,

$$(x_p, p) = (y_p, p) = (z_p, p).$$

If $(x, p_x)$ is a stochastic number whose binary sequence $x$ has $N$ components, of which $m$ are equal to 1 and $N-m$ are equal to 0, then $p_x = \frac{m}{N}$ and, clearly, the representation of the pair $(x, p_x)$ is not unique. SC uses a redundant number system in which there are $\binom{N}{m}$ possible representations for each value $p_x = \frac{m}{N}$. Furthermore, a binary sequence $x$ can only has associated probabilities in the set $\{0, \frac{1}{N}, \frac{2}{N}, \ldots, \frac{N-1}{N}, 1\}$, so only a small subset of the real numbers in $[0,1]$ can be expressed exactly in SC.

---

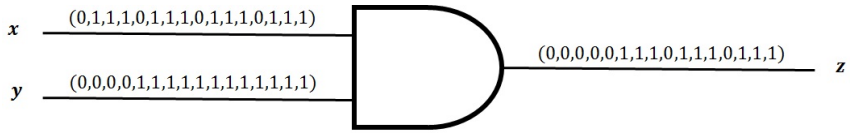
acronym for the English expression 'Binary Digit'. In 1948 the term bit was used by the engineer Claude Shannon in his article [33] to designate the binary digit. Several bits combined with each other, give rise to the other information units of a computer: byte, megabyte, gigabyte, terabyte, etc.

The construction of arithmetic operations between stochastic numbers depends on the arithmetic operations that can be defined on the Boole algebras $\{0,1\}$ ($\mathbb{Z}_2$) or $\{0,1\}^N$ ($\mathbb{Z}_2^N$), where $\mathbb{Z}_2$ and $N$ denote the integers modulo 2 and the length of the finite binary sequence, respectively [12, 13, 21]. For example, the multiplication of elements in $\{0,1\}$ (an also in $\mathbb{Z}_2$) can be expressed by tabulating their values as follows

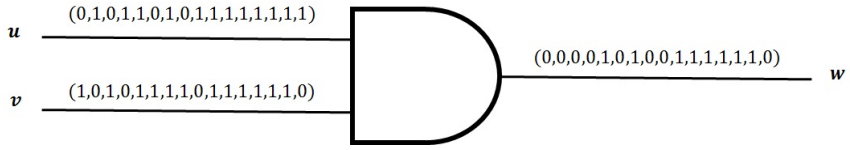
| $*$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |

Tabla 1: Multiplication of elements in $\{0,1\}$.

This multiplication also represents the logical conjunction on two logical values (typically the values of two propositions: True=1 and False=0). In advanced programming and digital electronics the logical conjunction is usually represent by the AND gate[2]. In this case, the multiplication in Table 1 is extended componentwise (i.e., bitwise) giving rise to the multiplication of binary sequences as follows.




(a) Multiplication of the sequences $x$ and $y$.




(b) Multiplication of the sequences $u$ and $v$.

Figura 2: AND gate used as a stochastic multiplier.

Note that the multiplication of binary sequences $x$ and $y$ of Figure 2(a) is compatible with the product of the probabilities associated with them, that is,

$$p_x p_y = \left(\frac{3}{4}\right)^2 = \frac{9}{16} = 0{,}5625 = p_z = p_{x*y}.$$

For this reason, the operation above is usually called *mutiplication of the stochastic numbers* $(x, p_x)$ and $(y, p_y)$. Figure 2(b) shows another two possible alternatives for representing the binary sequences corresponding to the stochastic numbers $(x, p_x)$ and $(y, p_y)$, i.e., $(x, p_x) = (u, p_u)$ and $(y, p_y) = (v, p_v)$, respectively. In this case, the result $w = u$ AND $v = u * v$ has associated probability $p_w = \frac{1}{2} = 0{,}5$, which can be interpreted as an approximation to the value $p_z = 0{,}5625$.

On the other hand, variations in the representations of the processed binary sequences could yield some inaccuracy. For instance, an extreme case occurs when the AND gate is used as a stochastic multiplier of the same pair $(x, p_x)$: the compatibility between the multiplication of the binary sequences and the product of their associated probabilities is lost, since $z = x$ AND $x = x * x = x$, and hence, $p_z = p_x = \frac{3}{4}$, rather than the numerically correct product $p_z = (p_x)^2 = \frac{9}{16}$. In order to avoid inaccurate results in [22] was introduced the notion of correlation: for fixed $N \in \mathbb{N}$, two binary sequences $u, v \in \{0,1\}^N$ are said independent (or uncorrelated) if and only if

$$\langle u, v \rangle = \frac{\|u\|_1 \|v\|_1}{N}, \tag{1}$$

[2]A logic gate is an idealized or physical device performing a logical operation on one or more binary inputs and produces a single binary output.

where $\langle \cdot, \cdot \rangle$ y $\| \cdot \|_1$ denote the usual inner product and the $l_1$ norm on $\mathbb{R}^N$, respectively. Two binary sequences $u, v \in \{0,1\}^N$ not satisfying (1) are called correlated. In Figure 2(a) the binary sequences $x, y$ are independent, hence $p_{x*y} = p_x p_y$. While the sequence $x$ is correlated with itself, and the binary sequences $u, v$ of Figure 2(b) also are correlated.

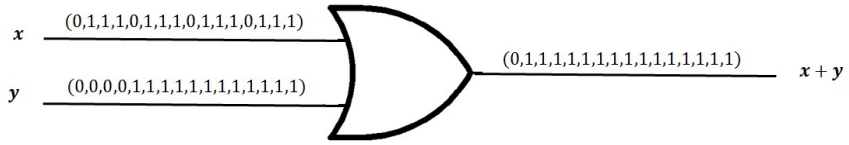
For the definition of the addition the logical disjunction (OR gate) and the exclusive logical disjunction (XOR gate) are commonly used. For example, the OR and XOR gates allow us to implement the following sums of elements in $\{0, 1\}$:

| + | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 1 | 1 |

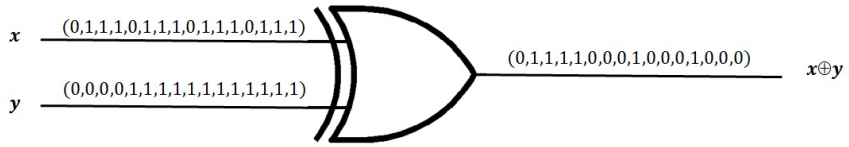
| ⊕ | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 1 | 0 |

Tabla 2: Additions of elements in $\{0, 1\}$ induced by OR and XOR gates.

Proceeding as before, the additions in Table 2 can be extended bitwise giving rise to two additions of binary sequences as follows.




(a) Addition $x + y$ of the sequences $x$ and $y$.




(b) Addition $x \oplus y$ of the sequences $x$ and $y$.

Figura 3: OR and XOR gates used as stochastic adders.

Another usual operation is the logic negation, which is implement by the NOT gate:

| $A$ | $\neg A$ |
|---|---|
| $F$ | $V$ |
| $V$ | $F$ |

| Input | ¬ |
|---|---|
| 0 | 1 |
| 1 | 0 |

Tabla 3: Logic negation ($\neg$) of the proposition $A$ and the NOT gate.

The logic negation of Table 3 can be extended bitwise as is illustrated in the next figure.

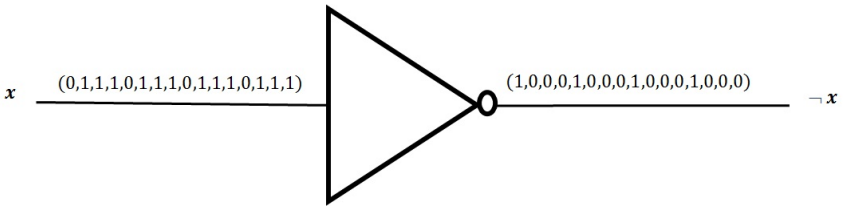



Figura 4: NOT gate used as a stochastic inverter.

All logic gates, except the NOT gate, can have more than two inputs. The interested reader may consult [32] for a detailed description of the logic gates AND, OR, XOR, NOT, NAND and NOR. Again, the compatibility between the addition of the binary sequences and the sum of their associated probabilities is lost. For example, in Figure 3(b) the addition $x + y$ has an associated probability $p_{x+y} = \frac{15}{16}$, while

$p_x + p_y = \frac{3}{2} \notin [0,1]$. As we know, this incompatibility comes from the fact that the sum of two numbers in $[0,1]$ belongs to $[0,2]$. To attempt to correct this incompatibility, SC introduces a special sum called *scaled addition*, as follows: Let us consider $(u, p_u)$ and $(v, p_v)$ stochastic numbers with $u, v \in \{0,1\}^N$. Let $(w, p_w)$ be a fixed stochastic number with $w \in \{0,1\}^N$, which we call control input. The scaled addition of $(u, p_u)$ and $(v, p_v)$ is defined as the stochastic number whose binary sequence and associated probability are given by

$$u \boxplus_w v := (u * w) + (v * (\neg w)), \qquad p_{u \boxplus_w v} := p_u p_w + p_v(1 - p_w). \tag{2}$$

For instance, if $N = 16$ let us consider the following stochastic numbers:

$$\begin{aligned} r &= (1,0,1,0,1,0,1,0,1,0,0,0,1,1,0,0) \quad \longmapsto p_r = \frac{7}{16}, \\ y &= (0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1) \quad \longmapsto p_y = \frac{12}{16} = \frac{3}{4}, \end{aligned}$$

and the control input

$$w = (1,0,0,1,0,1,0,1,1,0,1,0,1,0,1,0) \quad \longmapsto p_w = \frac{8}{16} = \frac{1}{2}.$$

Using (2) we obtain that the binary sequence and associated probability of the scaled addition of $(r, p_r)$ e $(y, p_y)$ are

$$\begin{aligned} z = r \boxplus_w y &= (1,0,0,0,1,0,1,0,1,1,0,1,1,1,0,1), \\ p_{r \boxplus_w y} &= p_r p_w + p_y(1 - p_w) = \frac{1}{2}\left(\frac{7}{16} + \frac{3}{4}\right) = \frac{19}{32} = 0{,}5937. \end{aligned}$$

It is important to point out that the binary sequence $z = (1,0,0,0,1,0,1,0,1,1,0,1,1,1,0,1)$ has nine components equal to 1, so that its associated probability is

$$p_z = \frac{9}{16} = 0{,}5625.$$

Consequently, the probability $p_{r \boxplus_w y} = 0{,}5937$ can be interpreted as an approximation to $p_z$. Figure 5 uses a multiplexer[3] to illustrate the representation of the scaled addition in digital electronics.

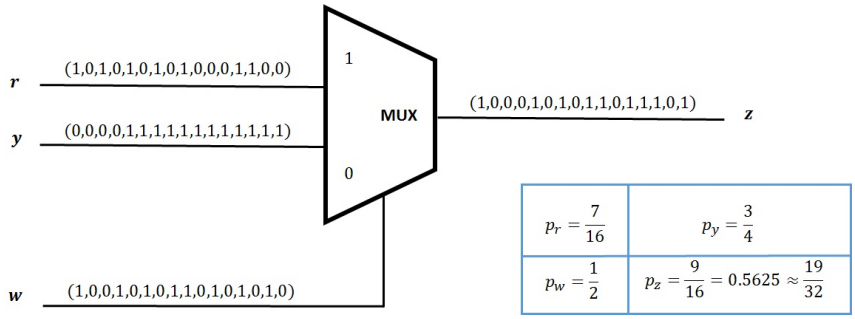



Figura 5: Implementation of scaled addition using a multiplexer.

For $N = 8$ the next example given in [1] shows that $p_z$ and $p_{r \boxplus_w y}$ coincides. Consider the stochastic numbers:

$$\begin{aligned} r &= (1,1,1,1,1,0,1,1) \quad \longmapsto p_r = \frac{7}{8}, \\ y &= (0,0,1,0,0,1,1,0) \quad \longmapsto p_y = \frac{3}{8}, \end{aligned}$$

[3] A multiplexer is a combinational circuit with several inputs and a unique output. It is endowed with one or several fixed inputs (called control inputs) capable to choose one and only one of the data inputs in order to allow the transmission from the chosen input to such a output [32].

and the control input

$$w = (1,0,0,1,0,1,0,1) \longmapsto p_w = \frac{4}{8} = \frac{1}{2}.$$

By (2) the binary sequence and associated probability of the scaled addition of $(r, p_r)$ and $(y, p_y)$ are

$$\begin{aligned} z = r \boxplus_w y &= (1,0,1,1,0,0,1,1), \\ p_{r\boxplus_w y} &= p_r p_w + p_y(1-p_w) = \frac{1}{2}\left(\frac{7}{8}+\frac{3}{8}\right) = \frac{5}{8} = p_z. \end{aligned}$$

The previous examples show a key problem in SC: How do we generate 'good'stochastic numbers for a particular application? One manner used for solving this problem has been of design of circuits that convert binary numbers to stochastic numbers, and vice versa. These number conversion circuits are called *stochastic number generators (SNG)*. The main function of SNGs is to produce stochastic numbers that are sufficiently random and independent, but they are not exempt to yield some inaccuracies (see e.g., [1, 3, 9, 20, 32, 38]).

In addition to the basic operations of multiplication and addition, SC has been applied to division and square-rooting [9, 34], matrix operations and decoding of low-density parity check (LDPC) codes [10, 14, 17], and polynomial arithmetic [25, 28].

## 3. Stochastic logic of Qian-Riedel-Rosenberg and Bernstein polynomials

The main idea behind the combinational circuits design with polynomial arithmetic of Qian et al. [25, 28] consist of:

(1) Take advantage -in a suitable way- of the redundancy provided by SC for choosing binary sequences $x \in \{0,1\}^N$ corresponding to the value $p_x$, in order to make an association between $x$ and a certain $N$-tuple of independent random variables $X = (X_1, \ldots, X_N)$, where each component $X_k$ has Bernoulli distribution with some parameter $p_k \in [0,1]$.

(2) Given a boolean function $y = f(x_1, \ldots, x_N)$ implementing a combinational circuit, use the association aforementioned for inducing a stochastic circuit implemented by a function of the form $Y = F(X_1, \ldots, X_N)$ (see for instance, [21]).

Given $N \in \mathbb{N}$ and $(x, p_x)$ a stochastic number with $x \in \{0,1\}^N$. For each $k = 1, 2, \ldots, N$ we choose $p_k \in [0,1]$ and consider discrete and independent random variables $X_k$ having Bernoulli distribution with parameter $p_k$, i.e., $X_k \sim Be(p_k)$ (cfr. [21, 31]). Since $x_k \in \{0,1\}$, each probability density function is given by

$$P\{X_k = x_k\} = p_k^{x_k}(1-p_k)^{1-x_k}. \tag{3}$$

We define

$$p_{X_k} := P\{X_k = 1\} = p_k \quad \text{and} \quad 1 - p_{X_k} := P\{X_k = 0\} = 1 - p_k, \quad k = 1, 2, \ldots, N.$$

For example, for the stochastic numbers $(x, p_x)$ and $(y, p_y)$ of Figure 3(a) we can choose discrete and independent random variables $X_k \sim Be(p_x)$ and $Y_k \sim Be(p_y)$, for all $k = 1, \ldots, 16$.

Let $f : \{0,1\}^N \to \{0,1\}$ be any boolean function. Given $(x, p_x)$ a stochastic number with $x = (x_1, \ldots, x_N) \in \{0,1\}^N$, choose an $N$-tuple of discrete and independent random variables $X = (X_1, \ldots, X_N)$ such that $X_k \sim Be(p_k)$ for some $p_k \in [0,1]$ and satisfying (3). We can associate to each $y = f(x) = f(x_1, \ldots, x_N) \in \{0,1\}$ a discrete random variable $Y$ using that its probability density function is uniquely determined by the given $N$-tuple $X = (X_1, \ldots, X_N)$. More precisely, for determining $p_Y := P\{Y = 1\}$ we proceed as follows (cf. [21]):

$$\begin{aligned} p_Y &= P\{Y=1\} = \sum_{x_1,\ldots,x_n:\, f(x_1,\ldots,x_N)=1} P\{X_1 = x_1, X_2 = x_2, \ldots, X_N = x_N\} \\ &= \sum_{x_1,\ldots,x_N:\, f(x_1,\ldots,x_N)=1} \left(\prod_{k=1}^{N} P\{X_k = x_k\}\right) \qquad (4) \\ &= \sum_{x_1,\ldots,x_N:\, f(x_1,\ldots,x_N)=1} \prod_{k=1}^{N} p_{X_k}, \qquad (5) \end{aligned}$$

The identity (4) is consequence of the independence of $X_k$, and (5) comes from

$$P\{X_k = x_k\} = \begin{cases} p_{X_k}, \text{ if } x_k = 1, \\ 1 - p_{X_k}, \text{ if } x_k = 0. \end{cases}$$

Furthermore, the random variable $Y$ has Bernoulli distribution with parameter $p_Y$. So, the boolean function $f : \{0,1\}^N \to \{0,1\}$ induces a function $F$ acting on the discrete and independent random variables $X_1, \ldots, X_N$ such that for each $Y = F(X_1, \ldots, X_N)$ we have

$$p_Y = \sum_{x_1,\ldots,x_N:\, f(x_1,\ldots,x_N)=1} \prod_{k=1}^{N} p_{X_k}.$$

Following [25, 28], we call *stochastic logic or stochastic logic of Qian-Riedel-Rosenberg* to the passage of the boolean function $y = f(x_1, \ldots, x_N)$ to the function $Y = F(X_1, \ldots, X_N)$.

EXAMPLE 3.1. *For $N = 3$, let us consider the boolean function $f : \{0,1\}^3 \to \{0,1\}$ given by*

$$f(x, y, z) = (x \wedge z) \vee (y \wedge (\neg z)) = (x * z) + (y * (\neg z)).$$

*Choose $p_1, p_2, p_3 \in [0,1]$ and let $X, Y, Z$ be three discrete and independent random variables such that $X \sim Be(p_1)$, $Y \sim Be(p_2)$, $Z \sim Be(p_3)$ whose probability density functions satisfy (3).*

*Let $w = f(x, y, z) \in \{0,1\}$, since*

$$\{x, y, z \in \{0,1\} : \ f(x, y, z) = 1\} \quad = \quad \{x, y, z \in \{0,1\} : (x, y, z) \in \{(0,1,0), (1,1,0), (1,0,1), (1,1,1)\}\},$$

*then*

$$\begin{aligned} p_W &= P\{W = 1\} = \sum_{\{x,y,z:\, f(x,y,z)=1\}} P\{s : X(s) = x, Y(s) = y, Z(s) = z\} \\ &= P\{X = 0, Y = 1, Z = 0\} + P\{X = 1, Y = 1, Z = 0\} + P\{X = 1, Y = 0, Z = 1\} \\ &\quad + P\{X = 1, Y = 1, Z = 1\} \\ &= (1 - p_X)p_Y(1 - p_Z) + p_X p_Y(1 - p_Z) + p_X(1 - p_Y)p_Z + p_X p_Y p_Z \\ &= p_X p_Z + p_Y(1 - p_Z). \end{aligned}$$

*Hence,*

$$p_W = p_X p_Z + p_Y(1 - p_Z), \tag{6}$$

*and the random variable $W$ is given by*

$$W = F(X, Y, Z) = XZ + Y(1 - Z). \tag{7}$$

*Note that both (6) and (7) induce the following polynomial in the variables $(a, b, c)$ with integer coefficients:*

$$\hat{F}(a, b, c) = ac + b(1 - c).$$

Example 3.1 illustrates the next result.

THEOREM 1. *(cf. [25, Theorem 1]). Given a boolean function $f : \{0,1\}^n \to \{0,1\}$. Stochastic logic yields a polynomial in $n$ variables $\hat{F}$ given by*

$$\hat{F}(a_1, \ldots, a_n) = \sum_{i_1=0}^{1} \cdots \sum_{i_n=0}^{1} \left( \alpha_{i_1 \ldots i_n} \prod_{k=1}^{n} a_k^{i_k} \right), \tag{8}$$

*where the coefficients $\alpha_{i_1 \ldots i_n}$ are integers. Moreover, for each $y = f(x_1, \ldots, x_n)$ we have*

$$p_Y = \hat{F}(p_{X_1}, p_{X_2}, \ldots, p_{X_n}) = \sum_{i_1=0}^{1} \cdots \sum_{i_n=0}^{1} \left( \alpha_{i_1 \ldots i_n} \prod_{k=1}^{n} p_{X_k}^{i_k} \right). \tag{9}$$

If we preassign some variables of the polynomial $\hat{F}(a_1, \ldots, a_n)$ given in (8) as constant values in $[0, 1]$ and the rest of variables is taken equal to one variable $t$, then $\hat{F}$ becomes a polynomial in one variable and real coefficients $g(t)$. Let us consider the polynomial of Example 3.1 with $a = 0{,}3$, $b = 0{,}7$ y $c = t$, then:

$$g(t) = 0{,}7 - 0{,}4t.$$

Thus, different boolean functions $f$ and preassigned variables will give rise to different polynomials $g(t)$. For particular combinational circuits whose stochastic logic yields a multivariate polynomial as in (8) and for their corresponding associated polynomials $g(t)$, the authors of [25] propose representations of $g(t)$ in terms of certain families of Bernstein polynomials. Before we look at these representations, we will recall the definition and some algebraic and analytic properties of the Bernstein polynomials (cf. [16, 28]).

For $f : [0, 1] \to \mathbb{R}$ continuous function, $n \geq 1$ and $x \in [0, 1]$, the $n$th Bernstein polynomial of $f$ is given by

$$B_n(t) = B_n(f; t) := \sum_{k=0}^{n} f\left(\frac{k}{n}\right)\binom{n}{k} t^k (1-t)^{n-k}. \tag{10}$$

It is clear that $B_n(t) \in \mathbb{P}_n$ and its definition also holds when $f : [0, 1] \to \mathbb{R}$ is only a bounded function. The polynomials $B_n(t)$ converge uniformly to $f$ on $[0, 1]$ and this fact is the key piece for the Bernstein constructive demonstration of Weierstrass approximation theorem [16, 23, 29].

The polynomials appearing in the formula on the right hand side of (10), namely;

$$b_k(t) = b_{k,n}(t) := \binom{n}{k} t^k (1-t)^{n-k}, \tag{11}$$

are the probability mass functions or Newton probabilities, that is, the probabilities of $k$ successes in $n$ trials of random process with individual probability of success $t$ in each trial, $k = 0, \ldots, n$. Also, it is clear that $\deg(b_{k,n}(t)) = n$, for each $k = 0, \ldots, n$.

A fact apparently trivial, but tremendously important for storing a polynomial into the memory of a computer is that the probabilities of Newton (11) form a basis for the polynomial space $\mathbb{P}_n$. That is,

$$\mathbb{P}_n = \operatorname{span}\{b_{0,n}(t), b_{1,n}(t), \ldots, b_{n,n}(t)\}.$$

We call Bernstein polynomial to the representation of any polynomial $P(t) \in \mathbb{P}_n$ in terms of the basis $\{b_{0,n}(t), b_{1,n}(t), \ldots, b_{n,n}(t)\}$. So, for each $P(t) \in \mathbb{P}_n$ there exists a unique vector $(\beta_{0,0}, \beta_{1,n}, \ldots, \beta_{n,n}) \in \mathbb{R}^{n+1}$ such that

$$P(t) = \underbrace{\sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t)}_{\text{Bernstein polynomial}}. \tag{12}$$

The name Bernstein polynomial for the expression on the right hand side of (12) was coined by Qian et al. (cf. [2, 25, 26, 28, 27]), although Farouki and Goodman [5] have preferred to use the term *Bernstein form* to refer to the same expression. By (12) we have that the $n$th Bernstein polynomial do the function $f \in C[0, 1]$ given by (10) becomes in a particular case of Bernstein polynomials, for which $\beta_{k,n} = f\left(\frac{k}{n}\right)$, $k = 0, 1, \ldots, n$.

The next result is a straightforward consequence of the definition (11).

PROPOSITION 3.1. *The Newton probabilities $\{b_{0,n}(t), b_{1,n}(t), \ldots, b_{n,n}(t)\}$ satisfy the following algebraic and analytic properties:*

*(i) Partition of unity property.*

$$\sum_{k=0}^{n} b_{k,n}(t) = 1, \quad \text{for all } t \in \mathbb{R}. \tag{13}$$

*(ii) Non-negativity property.*

$$b_{k,n}(t) \geq 0, \quad \text{for all } t \in [0, 1]. \tag{14}$$

*(iii) Symmetry property.*

$$b_{k,n}(t) = b_{n-k,n}(1-t), \quad \text{for all } t \in [0,1]. \tag{15}$$

*(iv) Recurrence formula.*

$$b_{k,n+1}(t) = tb_{k-1,n}(t) + (1-t)b_{k,n}(t), \quad \text{for all } t \in [0,1]. \tag{16}$$

*(v) Unimodality or extremal property. For $n \geq 1$, $b_{k,n}(t)$ attains a relative maximum at $t = \frac{k}{n}$, $k = 0, \ldots, n$.*

*(vi) Degree elevation property. For $k = 0, \ldots, n$, we have*

$$b_{k,n}(t) = \frac{n+1-k}{n+1} b_{k,n+1}(t) + \frac{k+1}{m+1} b_{k+1,n+1}(t), \tag{17}$$

*for all $t \in [0,1]$.*

*(vii) Representation in terms of the canonical basis of $\mathbb{P}_n$.*

$$b_{k,n}(t) = \sum_{j=k}^{n} (-1)^{j-k} \binom{n}{j}\binom{j}{k} t^j. \tag{18}$$

For Bernstein polynomials we have the following result.

PROPOSITION 3.2. *Let $P(t) = \sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t)$ be a Bernstein polynomial. Then the following properties hold.*

*(i) $P(0) = \beta_{0,n}$ and $P(1) = \beta_{n,n}$.*

*(ii) Inversion formula. For each $0 \leq j \leq n$, we have*

$$t^j = \sum_{k=j}^{n} \frac{\binom{k}{j}}{\binom{n}{j}} b_{k,n}(t). \tag{19}$$

*(iii) Change of basis. If $P(t)$ has the following representation in terms of the canonical basis for $\mathbb{P}_n$:*

$$P(t) = \sum_{k=0}^{n} a_{k,n} t^k,$$

*then*

$$\beta_{k,n} = \sum_{j=0}^{k} \frac{\binom{k}{j}}{\binom{n}{j}} a_{j,n}, \quad k = 0, \ldots, n. \tag{20}$$

*(iv) Lower and upper bounds.*

$$\min_{0 \leq k \leq n} \beta_{k,n} \leq P(t) \leq \max_{0 \leq k \leq n} \beta_{k,n}. \tag{21}$$

*(v) Differential relation and difference operator of coefficients. For each $j = 0, \ldots, n$, we have*

$$P^{(j)}(t) = \frac{n!}{(n-j)!} \sum_{k=0}^{n-j} \Delta^j(\beta_{k,n})\, b_{k,n-j}(t), \tag{22}$$

*where the finite difference of order $j$ of the coefficient $\beta_{k,n}$ is given by*

$$\begin{aligned} \Delta^j(\beta_{k,n}) &:= \Delta\left(\Delta^{j-1}(\beta_{k,n})\right) \\ &= \beta_{k+j,n} - \binom{j}{1}\beta_{k+j-1,n} + \cdots + (-1)^{j-1}\binom{j}{j-1}\beta_{k+1,n} + (-1)^j \beta_{k,n} \\ &= \sum_{r=0}^{j} (-1)^r \binom{j}{r} \beta_{k+j-r,n}. \end{aligned}$$

*(vi) Integration and partial sums.*

$$\int_0^t P(s)ds = \sum_{k=0}^{n} \beta_{k,n} c_{n-k}(t), \tag{23}$$

*where*

$$c_{n-k}(t) = \binom{n}{k} \sum_{r=0}^{n-k} \frac{(-1)^r \binom{n-k}{r}}{k+r+1} t^{k+r+1}.$$

According to the inversion formula (19) to verify that the Newton probabilities (11) form a basis for $\mathbb{P}_n$, it suffices to show that the set $\{b_{0,n}(t), b_{1,n}(t), \ldots, b_{n,n}(t)\}$ is linearly independent. Assume that

$$c_0 b_{0,n}(t) + c_1 b_{1,n}(t) + \cdots + c_n b_{n,n}(t) = 0,$$

for some $c_k \in \mathbb{R}$, $k = 0, \ldots, n$. By (18) we have

$$\begin{aligned} 0 &= c_0 b_{0,n}(t) + c_1 b_{1,n}(t) + \cdots + c_n b_{n,n}(t) \\ &= c_0 \sum_{j=0}^{n} (-1)^j \binom{n}{j}\binom{j}{0} t^j + c_1 \sum_{j=1}^{n} (-1)^{j-1} \binom{n}{j}\binom{j}{1} t^j + \cdots + c_n \sum_{j=n}^{n} (-1)^{j-n} \binom{n}{j}\binom{j}{n} t^j \\ &= c_0 + \left[\sum_{j=0}^{1} c_j (-1)^{1-j} \binom{n}{1}\binom{1}{j}\right] t + \left[\sum_{j=0}^{2} c_j (-1)^{2-j} \binom{n}{2}\binom{2}{j}\right] t^2 + \cdots + \left[\sum_{j=0}^{n} c_j (-1)^{n-j} \binom{n}{n}\binom{n}{n}\right] t^n. \end{aligned}$$

Next, using that $\{1, t, \ldots, t^n\}$ is basis for $\mathbb{P}_n$, we obtain

$$\begin{aligned} c_0 &= 0, \\ \sum_{j=0}^{1} c_j (-1)^{1-j} \binom{n}{1}\binom{1}{j} &= 0, \\ &\cdots \\ \sum_{j=0}^{n} c_j (-1)^{n-j} \binom{n}{n}\binom{n}{n} &= 0, \end{aligned}$$

which implies that $c_0 = c_1 = \cdots = c_n = 0$.

PROPOSITION 3.3. *(cf. [28, Corollary 1]). Let $P(t)$ a polynomial of degree $n$. For any $\epsilon > 0$ there exists a positive integer $M \geq n$, such that for all $t \in [0, 1]$ and integer $m \geq M$, we have*

$$\left| \sum_{k=0}^{m} \left( \beta_{k,m} - P\left(\frac{k}{m}\right) \right) b_{k,m}(t) \right| < \epsilon, \tag{24}$$

*where $\beta_{0,m}, \beta_{1,m}, \ldots, \beta_{m,m}$ satisfy $P(t) = \sum_{k=0}^{m} \beta_{k,m} b_{k,m}(t)$.*

*Demostración.* Let $n$ be the degree of $P(t)$. Since for any $m \geq n$ the Newton probabilities form a basis for $\mathbb{P}_m$, we have

$$P(t) \in \operatorname{span}\{b_{0,m}(t), b_{1,m}(t), \ldots, b_{m,m}(t)\}, \text{ for all } m \geq n.$$

On the other hand, $P(t)$, $t \in [0, 1]$ is a continuous function on $[0, 1]$. Then, the Bernstein demonstration of the Weierstrass approximation theorem [16, 23, 29] guarantees that

$$\lim_{m \to \infty} |P(t) - B_m(P; t)| < \epsilon, \text{ for all } t \in [0, 1].$$

Or equivalently, there exists $M \in \mathbb{N}$ with $M \geq n$ such that

$$\left| \sum_{k=0}^{m} \left( \beta_{k,m} - P\left(\frac{k}{m}\right) \right) b_{k,m}(t) \right| < \epsilon, \text{ whenever } m \geq M, \text{ for all } t \in [0, 1],$$

where $\beta_{0,m}, \beta_{1,m}, \ldots, \beta_{m,m}$ satisfy $P(t) = \sum_{k=0}^{m} \beta_{k,m} b_{k,m}(t)$.

□

Now we ready for showing the connection between stochastic logic and Bernstein polynomials. Suppose that we have a combinational circuit $y = f(x_1, x_2, \ldots, x_n)$ consisting of a decoding block[4] and a

[4] In general, a decoding block is a combinatorial circuit which has $n$ inputs and $m$ outputs, with $m \leq 2^n$. A typical application of the decoding blocks is to generate keyboard codes for introducing data into the computer from a keyboard [32].

multiplexing block[5], which transform the $n$ inputs $\{x_1, \ldots, x_n\} \in \{0, 1\}$ as follows: If $k$ out of the inputs $\{x_1, \ldots, x_n\}$ of the decoding block are logical 1, then $s_k$ is set to 1 and the other outputs are set to 0, $(0 \leq k \leq n)$. So, the output of the decoding block is $(s_0, \ldots, s_n)$. The outputs of the decoding block are fed into the multiplexing block, as shown in Figure 6 and act as the selecting signals (control inputs). The data signals (inputs) of the multiplexing block consist of $n + 1$ inputs $z_0, \ldots, z_n \in \{0, 1\}$.

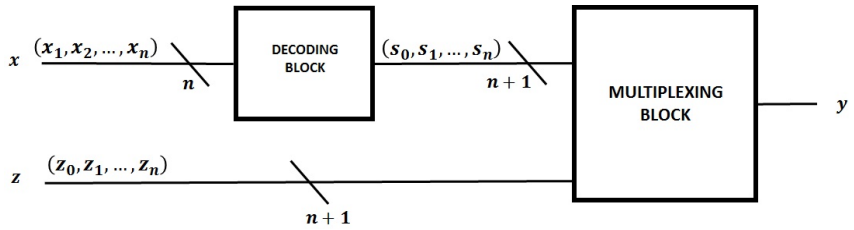



Figura 6: Combinational circuit associated to a Bernstein polynomial with coefficients in $[0, 1]$ (cf. [1, 25, 26, 28]).

Since the circuit contains a multiplexing block, once it decodes the inputs $x_1, \ldots, x_n$ then the boolean function $y = f(x_1, \ldots, x_n)$ takes the form

$$y = \bigvee_{k=0}^{n} (z_k \wedge s_k), \tag{25}$$

which means that the output of the multiplexing block $y$ is set to be the input $z_k$ if $s_k = 1$. Using the association (3) for $(x_1, \ldots, x_n)$, $(s_0, \ldots, s_n)$ and $(z_1, \ldots, z_n)$ we can choose discrete and independent random variables $(X_1, \ldots, X_n)$, $(S_0, \ldots, S_n)$ and $(Z_0, \ldots, Z_n)$, such that $X_k \sim Be(p_k)$, $k = 1, \ldots, n$, $S_j \sim Be(\hat{p}_j)$ and $Z_j \sim Be(\hat{\hat{p}}_j)$, $j = 0, \ldots, n$. Similarly, we define

$$\begin{aligned}
p_{X_k} &:= P\{X_k = 1\} = p_k \quad \text{and} \quad 1 - p_{X_k} := P\{X_k = 0\} = 1 - p_k, \quad k = 1, 2, \ldots, N, \\
p_{S_j} &:= P\{S_j = 1\} = \hat{p}_j \quad \text{and} \quad 1 - p_{S_j} := P\{S_j = 0\} = 1 - \hat{p}_j, \\
p_{Z_j} &:= P\{Z_j = 1\} = \hat{\hat{p}}_j \quad \text{and} \quad 1 - p_{S_j} := P\{S_j = 0\} = 1 - \hat{\hat{p}}_j, \quad j = 0, 1, \ldots, N.
\end{aligned}$$

Applying Theorem 1 to the function $y = f(x_1, \ldots, x_n)$, we have that the stochastic logic yields a multivariate polynomial as in (8), such that $p_Y = \hat{F}(p_{X_1}, \ldots, p_{X_n})$.

Let $g(t)$ be the polynomial associated to $\hat{F}$ for $a_k = t$, $k = 1, \ldots, n$. Assume that $p_{X_1} = \cdots = p_{X_n} = t_0$, since $s_j$ is set to 1 if and only if $j$ out of $n$ inputs of the decoding block are 1, the probability that $S_j$ is 1 is (see e.g., [1, pp. 10-11.]):

$$p_{S_j} = P\{S_j = 1\} = \binom{n}{j} t_0^j (1 - t_0)^{n-j} = b_{j,n}(t_0), \quad j = 0, \ldots, n. \tag{26}$$

Now, assume that $p_{Z_j} = \beta_{j,n}$, $j = 0, \ldots, n$. Then

$$p_Y = P\{Y = 1\} = \sum_{k=0}^{n} P\{Y = 1 | S_k = 1\} P\{S_k = 1\}, \tag{27}$$

but from (25) is deduced that $S_j = 1$ implies $Y = Z_j$, so

$$P\{Y = 1 | S_j = 1\} = P\{Z_j = 1\} = p_{Z_j} = \beta_{j,n}. \tag{28}$$

By (12), (9), (27) and (28) we obtain

$$g(t_0) = p_Y = P\{Y = 1\} = \sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t_0). \tag{29}$$

[5] A multiplexing block is a combinatorial circuit which has $n$ inputs, $m$ control inputs with $m \leq n$, and a unique output. Essentially, multiplexing blocks generalized to a multiplexer circuit [32].

Hence, under the constrains imposed by us, the combinational circuit associated to the function $y = f(x_1, \ldots, x_n)$ would require that $g(t)$ be a Bernstein polynomial whose coefficients $\beta_{j,n}$ belong to $[0, 1]$.

The next result summarizes the ideas above.

THEOREM 2. *[25, Theorem 2]. Let $P(t) = \sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t)$ be any Bernstein polynomial. If the coefficients $\beta_{k,n} \in [0, 1]$, $k = 0, \ldots, n$, then we can design stochastic logic to compute the Bernstein polynomial. That is, there exists an multivariate polynomial $\hat{F}$ satisfying (9) such that its associated polynomial is $P(t)$.*

Given $g(t) \in \mathbb{P}_n$, we say that $g(t)$ *can be implemented by stochastic logic*, if there exists an multivariate polynomial $\hat{F}$ as in (8) such that if $P_{X_1} = P_{X_2} = \cdots = P_{X_n} = t_0$ then $g(t_0)$ coincides with (9), that is

$$g(t_0) = \hat{F}(\underbrace{t_0, t_0, \ldots, t_0}_{n-\text{ times}}) = \hat{F}(P_{X_1}, P_{X_2}, \ldots, P_{X_n}) = p_Y. \tag{30}$$

In [25, 28] Qian et al. focus their attention on the study and characterization of the polynomials $g(t) \in \mathbb{P}_n$ which can be implemented by stochastic logic. More precisely, they consider $U, V, W \subset \mathbb{P}$ subsets of polynomials given by:

$$W = \left\{ g(t) \in \mathbb{P} : g(t) \text{ can be implemented by stochastic logic } \right\}, \tag{31}$$

$$U = \left\{ P(t) \in \mathbb{P} : \exists\, n \geq 1, 0 \leq \beta_{0,n}, \beta_{1,n}, \ldots, \beta_{n,n} \leq 1, \text{ such that } P(t) = \textstyle\sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t) \right\}, \tag{32}$$

$$V = \left\{ \begin{array}{c} P(t) \in \mathbb{P} : P(t) \equiv 0, \text{ or } P(t) \equiv 1, \text{ or there exists } n \geq 1, \text{ such that } \deg(P(t)) = n, \\ 0 < P(t) < 1,\ t \in (0, 1), \text{ and } 0 \leq P(0),\ P(1) \leq 1 \end{array} \right\}. \tag{33}$$

Note that Theorem 2 says us that $U \subseteq W$.

THEOREM 3. *[25]. Given $W, V \subset \mathbb{P}$ as in (31) and (33), respectively, then $W \subseteq V$.*

In order to obtain a characterization of $W$ we need the following result.

THEOREM 4. *[28, Theorem 1]. Let $P(t) \in \mathbb{P}_n$ with $\deg(P(t)) = n$, $n \geq 0$. For any $\epsilon > 0$ there exists a positive integer $M \geq n$, such that for all integers $m \geq M$ and $k = 0, \ldots, m$, we have*

$$\left| \beta_{k,m} - P\left(\frac{k}{m}\right) \right| < \epsilon, \tag{34}$$

*where $\beta_{0,m}, \beta_{1,m}, \ldots, \beta_{m,m}$ satisfy $P(t) = \sum_{k=0}^{m} \beta_{k,m} b_{k,m}(t)$.*

Theorem 4 is a result stronger than Proposition 3.3, since $P(t) \in \mathbb{P}_n$ satisfies the hypothesis of Theorem 4, using (13) and (14) we obtain (24). We refer the interested reader to [28] for a detailed proof of this theorem. For the purpose of our discussion here, we only give a sketch of it.

Given $\epsilon > 0$. If $n = 0$, then $P(t) = c$, with $c$ some constant. Taking $M = 1$, we have $P\left(\frac{k}{m}\right) = c$, $k = 0, \ldots, m$, whenever $m \geq M$. Thus, Theorem 4 holds. If $n > 0$, we choose $M \in \mathbb{N}$ such that

$$M > \text{máx}\left\{ \frac{n^2}{\epsilon} \sum_{i=0}^{n} \left| \binom{n}{i} \beta_{i,n} \right|, 2n \right\},$$

where the real numbers $\beta_{0,n}, \beta_{1,n}, \ldots, \beta_{n,n}$ satisfy $P(t) = \sum_{i=0}^{n} \beta_{i,n} b_{i,n}(t)$. Now consider any $m \geq M$. Since

$$2n \leq \text{máx}\left\{ \frac{n^2}{\epsilon} \sum_{i=0}^{n} \left| \binom{n}{i} \beta_{i,n} \right|, 2n \right\} < M \leq m,$$

we have $m - n > n$.

Using the following inequality as the main tool of the proof (cf. [28, Lemma 2]):

$$\left| \left(\frac{k}{m}\right)^i \left(1 - \frac{k}{m}\right)^{n-i} - \frac{\binom{m-n}{k-i}}{\binom{m}{k}} \right| \leq \frac{n^2}{m},$$

holds whenever $n > 0$, $m > n$, $0 \leq k \leq m$ and máx$\{0, k-m+n\} \leq i \leq$ mín$\{k, n\}$, the authors of [28] obtain

$$\left|\beta_{k,m} - P\left(\frac{k}{m}\right)\right| < \epsilon, \text{ in each one of the following cases: } \begin{cases} n \leq k \leq m-n, \\ 0 \leq k < n, \\ m-n < k \leq m. \end{cases}$$

THEOREM 5. *[28, Theorem 2]. Given $U, V \subset \mathbb{P}$ as in (32) and (33), respectively, then $U = V$.*

Again, we just show the main ideas involved in the proof of Theorem 5. We strongly recommend to the interested reader see [28] for the detailed proof of this theorem.

**Case 1:** $U \subseteq V$, (cf. [28, Theorem 3]). Let $n \geq 1$ and $\beta_{k,n} = 0$, for all $k = 0, \ldots, n$, then

$$P(t) = \sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t) = 0,$$

thus $P(t) \equiv 0 \in U$, and by (33) $P(t) \in V$. Analogously, if we take $\beta_{k,n} = 1$, for all $k = 0, \ldots, n$, then by (13) we have

$$P(t) = \sum_{k=0}^{n} b_{k,n}(t) = 1,$$

thus $P(t) \equiv 1 \in U$, and by (33) $P(t) \in V$.

Now consider any polynomial $P(t) \in U$ such that $P(t) \not\equiv 0$ and $P(t) \not\equiv 1$, then there exist $n \geq 1$ and $0 \leq \beta_{0,n}, \beta_{1,n}, \ldots, \beta_{n,n} \leq 1$ such that

$$P(t) = \sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t).$$

By part (ii) of Proposition 3.2 it is cleat that $P(0) \leq 0$ and $P(1) \leq 1$. Since $0 \leq \beta_{k,n} \leq 1$, for $k = 0, \ldots, n$, using (13) and (14) we get $0 \leq \sum_{k=0}^{n} \beta_{k,n} b_{k,n}(t) \leq 1$. That is, $0 \leq P(t) \leq 1$.

Now, we assume that there exists a $t_0 \in (0,1)$ such that $P(t_0) \leq 0$, or $P(t_0) \geq 1$. Again, (13) and (14) imply $0 \leq P(t_0) \leq 1$, thus $P(t_0) = 0$ or $P(t_0) = 1$. If $P(t_0) = 0$, since $t_0 \in (0,1)$ then $0 < t_0^k (1-t_0)^{n-k}$, $k = 0, \ldots, n$, and hence, $b_{k,n}(t_0) > 0$ for all $k = 0, \ldots, n$. Then, $P(t_0) = 0$ implies $\beta_{k,n} = 0$, for all $k = 0, \ldots, n$, i.e., $P(t) \equiv 0$, which is a contradiction.

A similar reasoning allows to show that $P(t_0) = 1$ implies $P(t) \equiv 1$, and we get a contradiction again. Therefore, $P(t)$ satisfies $0 < P(t) < 1$ for $t \in (0,1)$ and $0 \leq P(0), P(1) \leq 1$, that is, $P(t) \in V$.

**Case 2:** $V \subseteq U$, (cf. [28, Theorems 4,5, Corollaries 2,3]). Let us consider the following subsets of polynomials:

$$V_1 = \left\{ P(t) \in \mathbb{P} : \exists n \geq 1, \text{ such that } \deg(P(t)) = n,\ 0 < P(t) < 1, \text{ for all } t \in (0,1), \text{ and } 0 \leq P(0),\ P(1) < 1 \right\},$$

$$V_2 = \left\{ P(t) \in \mathbb{P} : \exists n \geq 1, \text{ such that } \deg(P(t)) = n,\ 0 < P(t) < 1, \text{ for all } t \in (0,1), \text{ and } P(0) = 0,\ P(1) = 1 \right\},$$

$$V_3 = \left\{ P(t) \in \mathbb{P} : \exists n \geq 1, \text{ such that } \deg(P(t)) = n,\ 0 < P(t) < 1, \text{ for all } t \in (0,1), \text{ and } 0 < P(0),\ P(1) \leq 1 \right\},$$

$$V_4 = \left\{ P(t) \in \mathbb{P} : \exists n \geq 1, \text{ such that } \deg(P(t)) = n,\ 0 < P(t) < 1, \text{ for all } t \in (0,1), \text{ and } P(0) = 1, P(1) = 0 \right\}.$$

Then

$$V = \{P(t) \in \mathbb{P} : P(t) \equiv 0, \text{ or } P(t) \equiv 1\} \cup V_1 \cup V_2 \cup V_3 \cup V_4.$$

Since $\{P(t) \in \mathbb{P} : P(t) \equiv 0, \text{ or } P(t) \equiv 1\} \subset U$, our problem is reduced to show that $V_j \subseteq U$, for all $j = 1, 2, 3, 4$.

If $P(t) \in V_1$, then there exists $n \geq 1$ such that $\deg(P(t)) = n$ and

$$0 \leq P(t) < 1, \text{ for } t \in [0,1]. \tag{35}$$

Since $P(t)$ is continuous on $[0,1]$, it attains its maximum value $M_P$ on $[0,1]$ and using (35) we have $M_P < 1$. Let $\epsilon_1 = 1 - M_P > 0$, by Theorem 4, there exists a positive integer $M_1 \geq n$ such that for all integers $m \geq M_1$ and $k = 0, \ldots, m$, we have

$$\left|\beta_{k,m} - P\left(\frac{k}{m}\right)\right| < \epsilon_1,$$

where $\beta_{0,m}, \beta_{1,m}, \ldots, \beta_{m,m}$ satisfy that $P(t) = \sum_{k=0}^{m} \beta_{k,m} b_{k,m}(t)$.

Note that for all $m \geq M_1$ and $k = 0, \ldots, m : \beta_{k,m} < \epsilon_1 + P\left(\frac{k}{m}\right) \leq \epsilon_1 + M_P = 1$.

Denote by $r$ the multiplicity of 0 as a root of $P(t)$ (where $r$=0 if $P(0) \neq 0$) and by $s$ the multiplicity of 1 as a root of $P(t)$ (where $s$=0 if $P(1) \neq 0$). Then, $P(t)$ admits the following factorization:

$$P(t) = t^r (1-t)^s h(t), \tag{36}$$

where $h(t) \in \mathbb{P}$ satisfies $h(0) \neq 0$ and $h(1) \neq 0$.

Suppose that $h(0) < 0$, by the continuity of $h(t)$, there exists $t_0 \in (0,1)$ such that $h(t_0) < 0$. Then,

$$P(t_0) = t_0^r (1-t_0)^s h(t_0) < 0,$$

which is a contradiction, since $P(t) \in V_1$. Hence, $h(0) > 0$. Similarly, we have $h(1) > 0$.

Since $P(t) > 0$ for $t \in (0,1)$, then $h(t) = \frac{P(t)}{t^r(1-t)^s} > 0$ for $t \in (0,1)$. Hence, $h(t) > 0$, for all $t \in [0,1]$. Again, using the continuity of $h(t)$ on $[0,1]$, we have $h(t)$ attains its maximum value $M_h > 0$ on $[0,1]$. Applying Theorem 4 with $\epsilon_2 = M_h$, there exists a positive integer $M_2 \geq n - r - s$, such that for all integers $d \geq M_2$ and $k = 0, \ldots, d$, we have

$$\left| \gamma_{k,d} - h\left(\frac{k}{d}\right) \right| < \epsilon_2,$$

where $\gamma_{0,d}, \gamma_{1,d}, \ldots, \gamma_{d,d}$ satisfy $h(t) = \sum_{k=0}^{d} \gamma_{k,d} b_{k,d}(t)$.

Notice that for all $d \geq M_2$ and $k = 0, \ldots, d : \gamma_{k,d} > h\left(\frac{k}{d}\right) - \epsilon_2 \geq M_h - M_h = 0$. Next, taking $m_0 \geq$ máx$\{M_1, M_2 + r + s\}$, $P(t)$ can be expressed as a Bernstein polynomial of degree $m_0$: $P(t) = \sum_{k=0}^{m_0} \alpha_{k,m_0} b_{k,m_0}(t)$, with $0 \leq \alpha_{k,m_0} \leq 1$, for $k = 0, \ldots, m_0$. Therefore, $P(t) \in U$.

If $P(t) \in V_3$, let us consider the polynomial $h(t) = 1 - P(t)$, $t \in [0,1]$. Then $h(t) \in V_1 \subseteq U$. Since $g(t) \in U$ implies $1 - g(t) \in U$, then we have $P(t) = 1 - h(t) \in U$. Hence, $V_3 \subseteq U$.

Now, if $P(t) \in V_2$, let $r$ be the multiplicity of 0 as a root of $P(t)$. So, $P(t)$ admits the following factorization:

$$P(t) = t^r h(t), \tag{37}$$

where $h(t) \in \mathbb{P}$ satisfies $h(0) \neq 0$. Proceeding similarly as the case $V_1 \subseteq U$, we obtain $h(0) > 0$. Since $h(t) = \frac{P(t)}{t^r} > 0$ for $t \in (0,1]$, then $h(t) > 0$ for $t \in [0,1]$. By continuity of $h(t)$ on $[0,1]$, it attains its maximum value $M_h > 0$ on $[0,1]$. Applying Theorem 4 with $\epsilon_1 = M_h$, there exists a positive integer $M_1 \geq n - r$ such that for all integers $d \geq M_1$ and $k = 0, \ldots, d$, we have

$$\left| \gamma_{k,d} - h\left(\frac{k}{d}\right) \right| < \epsilon_1,$$

where $\gamma_{0,d}, \gamma_{1,d}, \ldots, \gamma_{d,d}$ satisfy $h(t) = \sum_{k=0}^{d} \gamma_{k,d} b_{k,d}(t)$.

Moreover, for all $d \geq M_1$ and $k = 0, \ldots, d : \gamma_{k,d} > h\left(\frac{k}{d}\right) - \epsilon_1 \geq M_h - M_h = 0$.

Consider $Q(t) = 1 - P(t)$, $t \in [0,1]$. Then $0 < Q(t) < 1$ for $t \in (0,1)$ and $Q(0) = 1$, $Q(1) = 0$. Let $s$ be the multiplicity of 1 as a root of $Q(t)$. Then $Q(t)$ admits the following factorization:

$$Q(t) = (1-t)^s q(t), \tag{38}$$

where $q(t) \in \mathbb{P}$ satisfies $q(1) \neq 0$. Proceeding similarly as the case $V_1 \subseteq U$, we obtain $q(1) > 0$. Since $q(t) = \frac{Q(t)}{(1-t)^s} > 0$ for $t \in [0,1)$, then $q(t) > 0$ for all $t \in [0,1]$. The continuity of $q(t)$ on $[0,1]$ guarantees that $q(t)$ attains its maximum value $M_q > 0$ on $[0,1]$. Applying Theorem 4 with $\epsilon_2 = M_q$, there exists a positive integer $M_2 \geq n - s$ such that for all integers $m \geq M_2$ and $k = 0, \ldots, m$, we have

$$\left| \beta_{k,m} - q\left(\frac{k}{m}\right) \right| < \epsilon_2,$$

where $\beta_{0,m}, \beta_{1,m}, \ldots, \beta_{m,m}$ satisfy $q(t) = \sum_{k=0}^{m} \beta_{k,m} b_{k,m}(t)$. Since for all integers $m \geq M_2$ and $k = 0, \ldots, m$ we have $\beta_{k,d} > q\left(\frac{k}{m}\right) - \epsilon_2 \geq M_q - M_q = 0$, taking $m_0 \geq$ máx$\{M_1 + r, M_2 + s\}$, $P(t)$ can be expressed as a Bernstein polynomial of degree $m_0$: $P(t) = \sum_{k=0}^{m_0} \alpha_{k,m_0} b_{k,m_0}(t)$, with $0 \leq \alpha_{k,m_0} \leq 1$, for $k = 0, \ldots, m_0$. Therefore, $P(t) \in U$.

Finally, if $P(t) \in V_4$, let us consider the polynomial $h(t) = 1 - P(t)$, $t \in [0,1]$. Then $h(t) \in V_2 \subseteq U$. Since $g(t) \in U$ implies $1 - g(t) \in U$, we have $P(t) = 1 - h(t) \in U$. Hence, $V_4 \subseteq U$, and we can conclude that $V \subseteq U$.

Theorems 3, 5 and 2 allow to deduce that $W \subseteq V = U \subseteq W$. That is;

THEOREM 6. *[28]. Given $W, U, V \subset \mathbb{P}$ as in (31)-(33), respectively, then $W = U = V$.*

For concluding, it is worthy to mention that to the best of our knowledge, the treatment or implementation by use of some stochastic logic of Qian-Riedel-Rosenberg type has not been considered for boolean functions of the form $f : \{0,1\}^N \to \{0,1\}^N$. Thus, the following questions related to Theorem 1 arise: Can Theorem 1 be extended in this setting? In negative case, what is the difficult for finding such a extension? In affirmative case, how do we characterize such a extension? Some answers to these questions constitute part of the work in progress [30].

**Referencias**


[1] Alaghi, A., Hayes, J. P.: *"Survey of stochastic computing"*. ACM Trans. Embed. Comput. Syst. 12, 2s, Article 92, 19 pages, (2013).

[2] Alaghi, A., Li, Ch., Hayes, J. P.: *"Stochastic circuits for real-time image-processing applications"*, Design Automation Conference, DAC'13, Austin, TX, EEUU, May 29 - June 07, 2013.

[3] Alaghi, A., Qian, W., Hayes, J. P.: *"The promise and challenge of stochastic computing"*. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 37 (8), 1515-1531, (2018).

[4] Arora, S.: *"The mathematics of machine learning deep learning"* [Slides] Plenary Lecture presented at International Congress of Mathematicians 2018. Disponible en: http://www.cs.princeton.edu/∼arora/.

[5] Farouki, R. T., Goodman, T. N. T.: *"On the optimal stability of the Bernstein basis"*. Math. Comp. 65(216), 1553-1566, (1996).

[6] Gaines, B. R.: *"Stochastic computing"*. En: *Proc. American Federation of Information Processing Societies. Spring Joint Computer Conference*, Vol. 30, 149-156. Books, Inc., New York, (1967).

[7] Gaines, B. R.: *"Techniques of identification with the stochastic computer"*. Proceedings IFAC Symposium on The Problems of Identification in Automatic Control Systems, Section 6 Special Identification Instruments. Prague, Czech Republic, 1967. Disponible en https://www.catalog.hathitrust.org.

[8] Gaines, B. R.: *"Stochastic computing"*. En: *Encyclopaedia of Information, Linguistics and Control*, 766-781. Pergamon Press, New York and London, (1968).

[9] Gaines, B. R.: *"Stochastic computing systems"*. En: *Advances in Information Systems Science*. Tou, J. F. (ed), Vol. 2, Chapter 2, 37-172. Springer Science + Business Media, New York. First edition published by Plenum Press, New York, (1969).

[10] Gallager, R. G.: *"Discrete Stochastic Processes"*. The Kluwer International Series in Engineering and Computer Science. Communications and Information Theory. Springer Science + Business Media, New York, EEUU. Originally published by Kluwer Academic Publishers, (1996).

[11] Halmos, P.: *"The Legend of John Von Neumann"*. Amer. Math. Montly, 80 (4), 382-394, (1973).

[12] Halmos, P.: *"Lectures on Boolean Algebras"*. Springer-Verlag, New York, (1974).

[13] Hernández, S., Quintana, Y.: *"Estructuras booleanas y el código genético: algunos comentarios"*. Revista de Matemática de la Universidad del Atlántico - MATUA, 2 (2), 12-36, (2015).

[14] Kschischang, F. R., Frey, B. J., Loeliger, H. A.: *"Factors graphs and the sum-product algorithm"*. IEEE Trans. Inform. Theory, 47, 498-519, (2001).

[15] Li, X., Qian, W., Riedel, M. D., Bazargan, K., Lilja, D.: *"A reconfigurable stochastic architecture for reliable computing"*. Proceedings of the Great Lakes Symposium on VLSI, 315-320, (2009).

[16] Lorentz, G. G.: *"Bernstein polynomials"*. Second edition. Chelsea Publishing Company, New York, EEUU. (1986).

[17] MacKay, D. J. C., Neal, M. R.: *"Near Shannon limit performance of low density parity check codes"*. Electron. Lett. 33 (6), 457-458, (1997).

[18] Mądry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: *"Towards deep learning models resistant to adversarial attacks"*. ArXiv:1706.06083v3[stat.ML], 2017. Disponible en https://arxiv.org/pdf/1706.06083.pdf.

[19] Maldonado, C. E.: *"La web profunda y las dinámicas de la información"*. En: *Le Monde diplomatique, edición Colombia*, Nro. 179, 34-35, 2018. Disponible en https://www.academia.edu/37039302.

[20] Muguruma, R., Yamashita, S.: *"Stochastic number generation with the minimum inputs"*. IEICE Transactions on Fundamentals of Electronics, Communications and Computers Sciences, E100.A, 128-133, (2017).

[21] Parker, K. P., McCluskey, E. J.: *"Probabilistic treatment of general combinational networks"*. IEEE Trans. Comput. 24 (6), 668-670, (1975).

[22] Peavons, J., Cohen, D. A., Shawe-Taylor, J.: *"Generating binary sequences for stochastic computing"*. IEEE Trans. Inform. Theory, 40, 716-720, (1994).

[23] Pérez, D., Quintana, Y.: *"A survey on the Weierstrass approximation theorem"*. Divulg. Mat. 16 (1), 231-247, (2008).

[24] Poppelbaum, W. J., Afuso, C., Esch, J. W.: *"Stochastic computing elements and systems"*. En: *Proc. American Federation of Information Processing Societies, Fall Joint Computer Conference*, Vol. 31, 635-644. Books, Inc., New York, (1967).

[25] Qian, W., Riedel, M. D.: *"The synthesis of robust polynomial arithmetic with stochastic logic"*. En: *Design Automation Conference*, 648-653, (2008).

[26] Qian, W., Riedel, M. D.: *"Synthesizing logical computation on stochastic bit streams"*. Library of the Semiconductor Research Corporation (SRC). Publication ID:P059675, 8 pages, 2011. Disponible en https://www.src.org/library/publication/p059675/.

[27] Qian, W., Li, X., Riedel, M. D., Bazargan, K., Lilja, D.: *"An architecture for fault-tolerant computation with stochastic logic"*. IEEE Trans. Comput. 60, 93-105, (2011).

[28] Qian, W., Riedel, M. D., Rosenberg, I.: *"Uniform approximation and Bernstein polynomials with coefficients in the unit interval"*. European J. Combin. 32, 448-463, (2011).

[29] Quintana, Y.: *"Aproximación polinomial y ortogonalidad estándar sobre la recta"*. Escuela Matemática de América Latina y El Caribe. EMALCA-Colombia. Editorial: Universidad del Atlántico-Uniatlántico, Barranquilla, Colombia. (2013).

[30] Quintana, Y.: *"Concerning multivariate Bernstein polynomials and stochastic logic"*. Manuscript in progress, (n.d.).

[31] Ribeiro, S.: *"Random-pulse machines"*. IEEE Trans. Electron. Comput. V EC-16 No. 3, 261-276, (1967).

[32] Saha, A., Manna, N.: *"Digital Principles and Logic Design"*. Infinity Science Press LLC., Hingham, MA, EEUU. (2007).

[33] Shannon, C. E.: *"A Mathematical Theory of Communication"*. Reprinted with corrections from The Bell System Technical Journal, 27, 379-423, 623-656, July, October, (1948).

[34] Toral, S. L.: *"Análisis y síntesis de circuitos digitales estocásticos para la realización de sistemas analógicos*. Tesis Doctoral, Universidad de Sevilla, España, 1999.

[35] Turing, A.: *"On computable numbers, with an application to the Entscheidungsproblem"*. Proc. Lond. Math. Soc. (2), 42, 230-265, (1936). Errata appeared in 43, 544-546, (1937).

[36] von Neumann, J.: *"Various techniques used in connection with random digits"*. National Bureau of Standards Applied Mathematics Series, 12, 36-38, (1951).

[37] von Neumann, J.: *"Probabilistic logics and synthesis of reliable organisms from unreliable components"*. Automata Studies, 43-48. Princeton University Press, New Jersey, (1956).

[38] Zhakatayev, A., Kim, K., Choi K., Lee, J.: *"An efficient and accurate stochastic number generator using even-distribution coding"*. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 11 pages, (2018). DOI: 10.1109/TCAD.2018.2789732.

[39] Zhang, Ch., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: *"Understanding deep learning requires re-thinking generalization"*. ArXiv:1611.03530v2[cs.LG], 2017. Disponible en https://arxiv.org/pdf/1611.03530.pdf.