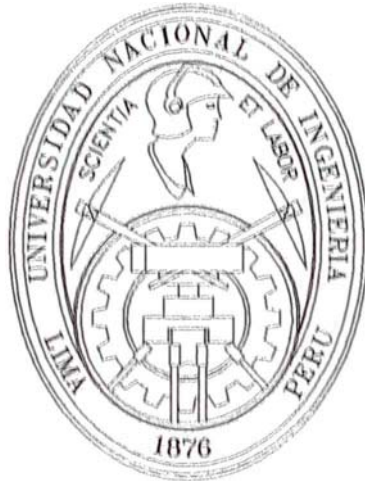


**UNIVERSIDAD NACIONAL DE INGENIERIA  
FACULTAD DE CIENCIAS**

*Escuela Profesional de Matemática*



**TESIS PARA OPTAR EL TITULO PROFESIONAL  
DE LICENCIATURA EN MATEMATICA**

**TITULADA:**

**ESTABILIDAD NUMERICA DE ALGORITMOS PARA  
EL DISEÑO GEOMETRICO**

**PRESENTADA POR:**

**PAREDES SORIA , Leopoldo**

**LIMA - PERU  
1998**

A mis queridos padres Leopoldo  
y Julia, a mis hermanos Irene,  
Gladys y Roberto.

# AGRADECIMIENTOS

Mi más sincero agradecimiento al Profesor Rolf Schroeder por su asesoría y dirección en la realización de este trabajo, y por brindarme la oportunidad y los medios para desarrollarlo.

Asimismo a las personas que indirectamente me brindaron todo su apoyo y me motivaron para seguir adelante hasta culminar este trabajo.

No puedo dejar de mencionar a mis padres y hermanos, quienes me motivaron y apoyaron en todo momento, brindándome todo su comprensión y amor.

# RESUMEN

En el presente trabajo se realiza el análisis de la estabilidad numérica de varios algoritmos del área del CAGD (Diseño Gráfico Asistido por Computadora), para la evaluación de polinomios en cualquier base. En particular se analiza en la base de potencia y de Bernstein, considerando los errores de redondeo que son obtenidos de los cálculos aritméticos. Se consiguen algoritmos que mejoran los números de condición de las raíces sin considerar el error de cálculo.

# INDICE

RESUMEN . . . . .	v
LISTA DE FIGURAS . . . . .	viii
<b>1 ANÁLISIS DE ERRORES</b>	<b>2</b>
1.1 Conceptos Básicos . . . . .	2
1.2 Errores de la Evaluación de un Polinomio por el Algoritmo de Horner . . . . .	16
1.3 Errores en el Cálculo de los Ceros de Polinomios	24
<b>2 TEOREMAS BÁSICOS DEL DISEÑO GEOMÉTRICO</b>	<b>28</b>
2.1 Características de la Base de Bernstein	29
2.2 Superficies . . . . .	39
2.2.1 Algoritmo de De Casteljaou para Superficies .	40
2.3 Tipos de Evaluación . . . . .	40
2.4 Operaciones Aritméticas .	41
2.5 Uso de los Coeficientes de Bernstein Escalados .	45
<b>3 ERRORES EN LA EVALUACIÓN DE UN POLINOMIO POR EL ALGORITMO DE DE CASTELJAU, ELEVACIÓN DE GRADO Y SUBDIVISIÓN . . . . .</b>	<b>47</b>
3.1 Determinación del Error del Polinomio	51

<b>3.2</b>	<b>Condición de Bernstein en Elevación de Grado y Subdivisión</b>	<b>53</b>
<b>4</b>	<b>TRANSFORMACIONES ENTRE LA BASE MONOMIAL A LA BASE DE BERNSTEIN</b>	<b>62</b>
<b>4.1</b>	<b>Conversión Básica . . . . .</b>	<b>62</b>
<b>4.2</b>	<b>Conversión Básica Matricial</b>	<b>65</b>
4.2.1	Los Números de Condición . . .	66
4.2.2	Una Expresión Aproximada de $\text{Cond}(M)$	72
4.2.3	Las Condiciones de Conversión entre la Base Monomial y de Bernstein	73
<b>4.3</b>	<b>Número de Condición de las Raíces . . . . .</b>	<b>74</b>
<b>4.4</b>	<b>Propiedades de Transformación de los Números de Condición</b>	<b>76</b>
<b>4.5</b>	<b>Condiciones de las Raices en las Bases de Potencia y de Bernstein</b>	<b>77</b>
	<b>BIBLIOGRAFIA . . . . .</b>	<b>91</b>

# LISTA DE FIGURAS

Figura 1.1	Representación del Punto Flotante . . . . .	3
Figura 2.1	Un polinomio cúbico $P(x)$ de Bernstein sobre $[0, 1]$ y su poligono de control (la región es la cápsula convexa del poligono de control). La barra vertical indica las cotas locales determinado por la cápsula convexa en un valor particular $x = x_0$ . . . .	30

# INTRODUCCIÓN

El presente trabajo tiene como objetivo principal garantizar la estabilidad de los algoritmos que manejan polinomios para la representación y manipulación de diseños de objetos a través del computador, tales como curvas y superficies.

En sus inicios los sistemas CAGD (Diseño Gráfico Asistido por Computadora) contaban con algoritmos que no consideraban los errores de cálculo en los procesos de evaluación de polinomios. Estos errores deben tomarse en cuenta además de los errores propios que tiene cada algoritmo. Así podemos comparar con mayor certeza la eficiencia de los algoritmos. Se desarrollan un diferente y largamente inexplorado aspecto favorable de los métodos de Bernstein-Bézier, su estabilidad numérica inherente ante la influencia de las perturbaciones de los datos de entrada.

En el capítulo uno se dan conceptos básicos sobre el error de redondeo, la influencia en los polinomios de base potencial, y también en las matrices debido a perturbación de los datos de entrada.

En el capítulo dos se dan teoremas básicos del CAGD y a su vez se estudia los polinomios en la base de Bernstein, así como las operaciones aritméticas que se realizan en la base de Bernstein y potencial.

En los capítulos tres y cuatro se evalúan y se analizan los números de condición de los algoritmos de Casteljaou, elevación de grado, subdivisión y la transformación de la base de potencia a Bernstein.



# 1 ANÁLISIS DE ERRORES

## 1.1 Conceptos Básicos

En este capítulo se da a conocer las nociones fundamentales de los cálculos internos que realizan las máquinas, para minimizar en lo posible los errores, en particular cuando se realiza cálculos recursivos.

Esta sección esta basado en los trabajos de Kincaid y Cheney [7], Burden y Faires [1], Nieves y Domínguez [9] y Stoer y Bulirsch [10].

En algunos casos los pueden ser más grandes que los números de máquina. Se habla de una CAJA NEGRA, si no se tiene conocimiento de los calculos internos que realiza la máquina, y si se tiene conocimiento se habla de una CAJA BLANCA o CAJA TRANSPARENTE.

### Características del Manejo de una Computadora

Las computadoras trabajan con las bases con las que fueron diseñados. En adelante trabajaremos con máquinas que usan el sistema binario con 32 bits de precisión, donde los números reales son representados por:

signo del número real $x$ (S)	:	1 bit
signo del exponente $E$ (s)	:	1 bit
exponente (entero $E$ )	:	7 bits
mantisa (número real $q$ )	:	23 bits

Un número distinto de cero tiene la forma  $x = \pm q \times 2^{\pm E}$ . Su mantisa se encuentra en el rango  $\frac{1}{2} \leq q < 1$ . El primer bit distinto de cero es 1 y por lo tanto no requiere almacenamiento. La mantisa  $q$  está representado por  $q = (0.1F)_2$ , donde  $F$  es de 23 bits. El bit menos significativo en la mantisa representa unidades de  $2^{-24}$

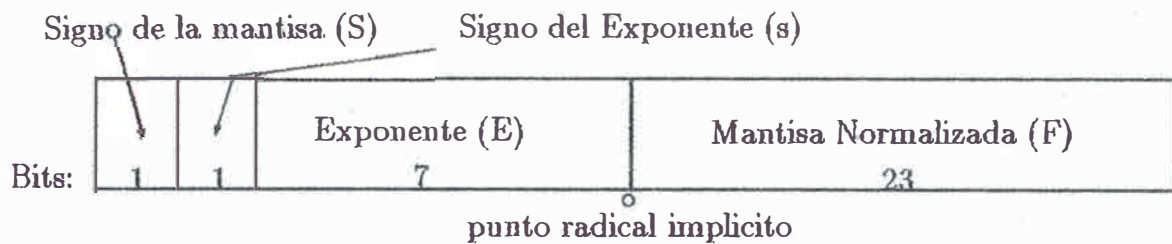


Figura 1.1: Representación del Punto Flotante

(aproximadamente  $10^{-7}$ ). Con el asignamiento de 7 bits para el exponente ( $E$ ), se cumple  $0 \leq E \leq (1111111)_2 = 2^7 - 1 = 127$ . Luego la máquina puede manejar números tan pequeños como  $10^{-38} \approx 2^{-127}$  y tan grandes como  $10^{38} \approx 2^{127}$ . La representación del punto flotante para un número real de precisión sencilla en la computadora de 32 bits se divide en 3 campos, como se muestra en la figura (1.1). Por consiguiente, los números máquinas normalizados no nulos se decodifican como sigue:  $x = (-1)^S \times q \times 2^{(-1)^S E}$ .

## Notación Científica Normalizada

Los números reales son expresados mediante la notación científica normalizada. Eso significa que el punto decimal se desplaza de tal manera que todos los dígitos aparecen a la derecha del punto decimal y con el primer dígito distinto de cero. Sea  $x = \pm q \times \beta^{\pm E}$ ,  $\frac{1}{\beta} \leq q < 1$ , donde  $x \neq 0$  y  $E$  es un entero (positivo, negativo, o cero). El número  $q$  se llama mantisa y el entero  $E$  exponente. Consideremos que en una computadora binaria tanto  $q$  y  $E$  están representados como números en base 2.

## Números de Máquina Próximos

Suponemos que se tiene:  $x = q \times 2^{\pm E}$ ,  $\frac{1}{2} \leq q < 1$ ,  $E \leq 127$ , donde  $x = (.a_1 a_2 \cdots a_{23} a_{24} a_{25} \cdots)_2 \times 2^{\pm E}$ . Descartando los dígitos  $a_{25}, a_{26}, a_{27}, \dots$ , se

obtiene el número de máquina próximo  $x' = (.a_1a_2 \cdots a_{24})_2 \times 2^{\pm E}$ , donde  $x'$  se encuentra a la izquierda de  $x$  en la recta real. Este procedimiento se llama TRUNCAMIENTO, el cual puede ser un redondeo exacto (redondeo correcto, cuando el bit  $a_{25} = a_{26} = \cdots = 0$ ). Otro número de máquina próximo  $x''$  queda a la derecha de  $x$ . El cual se obtiene mediante el proceso de redondeo por EXCESO, aumentamos en una unidad el último bit en la mantisa de  $x^{-1}$  (referente a  $a_{24}$ ), resultando  $x'' = ((.a_1a_2 \cdots a_{24})_2 + 2^{-24}) \times 2^{\pm E}$ .

Si  $x'$  representa mejor a  $x$ , entonces resulta:

$$|x - x'| \leq \frac{1}{2}|x'' - x'| = \frac{1}{2} \times 2^{-24} \times 2^{\pm E} = 2^{\pm E - 25}$$

El error relativo está acotado como sigue:

$$\left| \frac{x - x'}{x} \right| \leq \frac{2^{\pm E - 25}}{q \times 2^{\pm E}} = \frac{2^{-25}}{q} \leq \frac{2^{-25}}{1/2} = 2^{-24}$$

En el segundo caso, si  $x$  está más cercana a  $x''$  que a  $x'$ , cumple:

$$\left| \frac{x - x'}{x} \right| \leq 2^{-24}$$

Se concluye que el error relativo no excede de  $2^{-24}$ . Si  $E$  es demasiado grande se dice que ha ocurrido un DESBORDAMIENTO POR EXCESO. En muchas computadoras la ejecución del programa se interrumpirá de manera automática, cuando se use en el cálculo, la variable que contiene la indeterminación. Si  $E$  es demasiado pequeño hablamos de un DESBORDAMIENTO POR DEFECTO. Muchas computadoras simplemente asignan a la variable el valor 0 y el cálculo continúa. Haciendo  $\delta = \left( \begin{smallmatrix} x^* \\ x \end{smallmatrix} \right)$ , donde  $x^* = x(1 + \delta)$ ,  $|\delta| \leq 2^{-24}$ , se utiliza la notación  $fl(x)$  para representar el número de máquina de punto flotante  $x^*$  más cercano a  $x$ . El número  $2^{-24}$  se denomina error de redondeo unitario para la computadora.

## Análisis del Error de Punto Flotante

El diseño interno de la máquina realiza los siguientes pasos para combinar aritméticamente dos números:

Se forma correctamente la combinación, se normaliza, se redondea, y se almacena en la memoria en la forma de una palabra de máquina. Las operaciones (+, -, /, ×) son representadas por  $\oplus$ :

- Si  $x$  y  $y$  son números de máquina,  $fl(x \oplus y) = (x \oplus y)(1 + \delta)$ ,  $|\delta| \leq 2^{-24}$ .

- Si  $x$  y  $y$  no son números de máquina, el resultado correspondiente es:

$fl(fl(x) \oplus fl(y)) = (x(1 + \delta_1) \oplus y(1 + \delta_2))(1 + \delta_3)$ ,  $|\delta_i| \leq 2^{-24}$ ,  $i = 1, 2, 3$ .

## Epsilon de la Máquina

Una máquina funciona con una base  $\beta$  (puede ser binario, octal ó hexadecimal) y utiliza  $n$  posiciones en la mantisa de sus números de punto flotante, entonces:  $fl(x) = x(1 + \delta)$ ,  $|\delta| \leq \eta$ , donde:

$$\begin{aligned} \eta &= \frac{1}{2}\beta^{1-n} && \text{en caso de redondeo correcto,} \\ \eta &= \beta^{1-n} && \text{en caso de truncamiento.} \end{aligned} \tag{1.1}$$

El error de redondeo unitario  $\eta$  varía ampliamente entre las computadoras modernas, debido a las diferencias en las longitudes de una palabra de máquinas, diferencias en las bases utilizadas para la aritmética, el tipo de redondeo utilizado, etc. Las longitudes de palabra varían: Entre 64 y 60 para computadoras científicas (grandes), 32 y 36 para máquinas de tamaño mediano, y 16 para algunas computadoras personales. Los diferentes tipos de redondeo utilizados son: El redondeo perfecto, y el truncamiento.

Algunos compiladores poseen un switch, el cual permite al usuario seleccionar el tipo de redondeo para realizar los cálculos.

## Pérdida de Dígitos Significativos

Los errores de redondeo son inevitables y a su vez difíciles de controlar. Sin embargo, algunos tipos de errores son fácilmente controlados. En el estudio del Análisis Numérico es importante entender y controlar varios tipos de errores. Presentamos un tipo de error que con frecuencia aparece como consecuencia de descuidos durante el proceso de programación.

**Ejemplo 1.1.** *Se deben tener mucho cuidado en la diferencia de dos números muy próximos. Para  $x = .6532186521$   $y = .6531032022$  se tiene  $x - y = .0001144499$ .*

*Para una computadora de base decimal con una mantisa de cinco dígitos, se tiene:  $fl(x) = .65322$   $fl(y) = .65310 \implies fl(x - y) = .00012$ .*

*El error relativo es muy grande:*

$$\frac{x - y - [fl(x) - fl(y)]}{x - y} = \frac{.0001144499 - .00012}{.0001144499} \approx 0.04, \text{ es decir } 4\%.$$

*Al analizar se añaden ceros a la derecha para tener 5 dígitos en la mantisa, los cuales carecen de sentido y no representan una exactitud adicional. Por lo tanto  $fl(x) - fl(y)$  se representa en la computadora como  $0.12000 \times 10^{-3}$  aunque los ceros en la mantisa sólo sirven para denotar el lugar decimal.  $\square$*

## Inestabilidad Numérica

Se habla de INESTABILIDAD, cuando los pequeños errores en los procesos del cálculo en alguna de sus etapas se agrandan en etapas posteriores y degradan seriamente la exactitud de dichos cálculos. En base a los errores relativos, se puede establecer si un proceso es numéricamente ESTABLE o INESTABLE.

## Condicionamiento

Si un pequeño cambio en los datos de entrada da lugar a grandes cambios en las salidas (respuestas), se dice que dicho problema está MAL CONDICIONADO. Por consiguiente las palabras CONDICION y CONDICIONAMIENTO se usan para indicar cuál sensible es la solución de un problema respecto de pequeños cambios relativos en los datos de entrada. Para ciertos problemas se puede definir un número de condición. Si el número es grande eso significa que se tiene un problema mal condicionado.

**Ejemplo 1.2.** Dado el polinomio de distribución lineal de las raíces reales  $x_0^* = \frac{k}{n}$ ,  $k = 1, 2, \dots, n$ , en el intervalo unitario  $[0, 1]$  de  $n = 20$ :

$$P(x) = \prod_{k=1}^{20} \left( x - \frac{k}{20} \right)$$

El desarrollo expresado en forma potencial es el siguiente:

$$P(x) = \left(x - \frac{1}{20}\right)\left(x - \frac{2}{20}\right) \cdots \left(x - \frac{19}{20}\right)\left(x - \frac{20}{20}\right) = \left[\left(x - \frac{210}{20}\right)x + \cdots\right]x + \frac{1 \times 2 \times \cdots \times 20}{20^{20}}$$

Perturbando el coeficiente  $a_{19}$  con  $\delta = \frac{2^{-23}}{210} \approx 5.7 \times 10^{-10}$ , el resultado de las nuevas raíces son:

$$\begin{aligned}x_1 &= .05000000, & x_2 &= .10000000, & x_3 &= .15000000, & x_4 &= .20000000, & x_5 &= .25000000, \\x_6 &= .30000035, & x_7 &= .34998486, & x_8 &= .40036338, & x_9 &= .44586251, & x_{10} &= .50476331 - .03217504i, \\x_{11} &= .50476331 + .03217504i, & x_{12} &= .58968169 - .08261649i, & x_{13} &= .58968169 + .08261649i, \\x_{14} &= .69961791 - .12594150i, & x_{15} &= .69961791 + .12594150i, & x_{16} &= .83653687 - .14063124i, \\x_{17} &= .83653687 + .97512197i, & x_{18} &= .97512197 - .09701652i, & x_{19} &= .97512197 + .09701652i, \\x_{20} &= 1.04234541.\end{aligned}$$

Los resultados nos indican que se trata de un problema mal condicionado, por ejemplo:  $x_{20}^* - x_{20} = 1 - 1.04234541 = 0.04234541$  y también aparecen valores complejos de  $x_k$  con partes imaginarias muy grandes, ver Farouki y Rajan '87, (p. 209).  $\square$

## Número de Condición de una Raíz de un Polinomio

El número de condición es una medida de la sensibilidad de sus soluciones a perturbaciones en sus parámetros de entrada. Los números de condición pueden también servir como una indicación de la sensibilidad de una solución a los errores de redondeo durante la ejecución del algoritmo.

La condición de polinomios con respecto a la evaluación y determinación de raíces ha sido extensamente estudiado por Wilkinson (en 1959). En este trabajo nos referimos a Farouki y Rajan '87, (p. 199).

Consideremos el desplazamiento  $\delta x$  de una raíz real  $x_s$  de un polinomio en la base  $\{\phi_k(x)\}$ :

$$P(x) = \sum_{k=0}^n \lambda_k \phi_k(x), \quad (1.2)$$

debido a una perturbación  $\delta \lambda_r$  de un coeficiente simple  $\lambda_r$ . Si  $x_s + \delta x$  es una raíz del polinomio perturbado, tenemos:

$$P(x_s + \delta x) = -\delta \lambda_r \phi_r(x_s + \delta x). \quad (1.3)$$

Desarrollando una expansión de la serie de Taylor alrededor de  $x_s$  en ambos lados de (1.3) y teniendo en cuenta que  $P(x_s) = 0$ , obtenemos:

$$P(x_s + \delta x) = P(x_s) + \frac{P'(x_s)}{1!} \delta x + \frac{P''(x_s)}{2!} (\delta x)^2 + \dots = \sum_{k=1}^n \frac{(\delta x)^k}{k!} P^{(k)}(x_s)$$

En forma análoga para  $\phi_r(x_s + \delta x)$ , y luego reemplazando en (1.3) se tiene:

$$\begin{aligned} \phi_r(x_s + \delta x) &= \sum_{k=0}^n \frac{(\delta x)^k}{k!} \phi_r^{(k)}(x_s) \\ \Rightarrow \sum_{k=1}^n \frac{(\delta x)^k}{k!} P^{(k)}(x_s) &= -\delta \lambda_r \sum_{k=0}^n \frac{(\delta x)^k}{k!} \phi_r^{(k)}(x_s) \end{aligned} \quad (1.4)$$

Si  $x_s$  es una raíz simple de  $P(x)$ , se tiene  $P'(x_s) \neq 0$ . Como  $(\delta x)^k \rightarrow 0$  para  $k = 2, 3, \dots, n$  y  $\delta \lambda_r (\delta x)^k \rightarrow 0$  para  $k = 1, 2, \dots, n$ , de (1.4) se obtiene:

$$\begin{aligned} \delta x \left\{ P'(x_s) + \sum_{k=2}^n \frac{(\delta x)^{k-1}}{k!} P^{(k)}(x_s) \right\} &= -\delta \lambda_r \left\{ \phi_r(x_s) + \sum_{k=1}^n \frac{(\delta x)^k}{k!} \phi_r^{(k)}(x_s) \right\} \\ \implies \lim_{\delta \lambda_r \rightarrow 0} \frac{\delta x}{\delta \lambda_r / \lambda_r} &= - \lim_{\delta \lambda_r \rightarrow 0} \frac{\lambda_r \phi_r(x_s)}{P'(x_s)} = - \frac{\lambda_r \phi_r(x_s)}{P'(x_s)} \quad (1.5) \end{aligned}$$

Para raíces simples, la ecuación (1.5) predice un desplazamiento  $\delta x$  de la raíz real la cual crece linealmente con la perturbación relativa  $\frac{\delta \lambda_r}{\lambda_r}$  en el  $r$ -ésimo coeficiente, en el límite de las perturbaciones infinitesimales. El valor absoluto del lado derecho de (1.5), es llamado **número de condición de la raíz  $x_s$  con respecto al coeficiente  $\lambda_r$** .

Si ahora permitimos perturbaciones aleatorias de magnitud relativa fija  $\frac{\varepsilon}{n+1} = \left| \frac{\delta \lambda_r}{\lambda_r} \right|$ ,  $r = 0, 1, \dots, n$ , pequeño en cada coeficiente, la magnitud del desplazamiento máximo de una raíz simple puede sufrir cambios, en el límite cuando  $\varepsilon \rightarrow 0$ .

$$\frac{|\delta x|}{\varepsilon / (n+1)} = \left| \frac{\lambda_r \phi_r(x_s)}{P'(x_s)} \right| = |P'(x_s)|^{-1} |\lambda_r \phi_r(x_s)|$$

Se tiene:

$$|\delta x| = |P'(x_s)|^{-1} \sum_{r=0}^n |\lambda_r \phi_r(x_s)| \frac{\varepsilon}{n+1} \leq \underbrace{|P'(x_s)|^{-1} \sum_{r=0}^n |\lambda_r \phi_r(x_s)|}_{C_\phi} \varepsilon.$$

De esto se obtiene:

$$|\delta x| \leq C_\phi \varepsilon \quad \text{con} \quad C_\phi = |P'(x_s)|^{-1} \sum_{r=0}^n |\lambda_r \phi_r(x_s)|, \quad (1.6)$$

donde  $C_\phi$  es el **número de condición total de la raíz  $x_s$** . Nos referimos a esta cantidad como el número de condición simple de la raíz del polinomio en la base dada.



Note que el número de condición dado por (1.6) está definido en términos de las permutaciones absolutas de una raíz inducido por permutaciones relativas de los coeficientes. En aplicaciones de modelo geométrico, estamos usualmente involucrados con aproximaciones de raíces  $x_s$  en  $[0, 1]$  con un error absoluto fijo  $\delta x$ . El crecimiento no acotado de los errores relativos  $\frac{\delta x}{x_s}$  en las raíces del origen ( $x_s \rightarrow 0$ ), no tiene significado especial, dado que es fácilmente eliminado por un cambio de origen. Para los coeficientes, consideramos solo perturbaciones relativas, como significativos; las raíces son inalterables al multiplicarse por un escalar arbitrario (no-nulo) a todos los coeficientes del polinomio.

Ahora, si los coeficientes son derivados de un cálculo uniforme del punto flotante de los datos de entrada, ellos tenderán a sufrir similares errores relativos de redondeo. Las raíces multiples son inherentemente mal condicionadas, el desplazamiento de las raíces tienen un rápido crecimiento lineal con la perturbación de los coeficientes. Si  $x_s$  es una raíz de  $m$  - cortes,  $m \geq 2$ , entonces  $P(x_s) = P'(x_s) = P''(x_s) = \dots = P^{(m-1)}(x_s) = 0$  y la ecuación (1.4) predice un desplazamiento típico de la raíz  $\delta x$  satisfaciéndose:

$$\begin{aligned} \Rightarrow \frac{(\delta x)^m}{m!} P^{(m)}(x_s) + \sum_{k=m+1}^n \frac{(\delta x)^k}{k!} P^{(k)}(x_s) &= -\delta \lambda_r \phi_r(x_s) - \delta \lambda_r \sum_{k=1}^n \frac{(\delta x)^k}{k!} \phi_r^{(k)}(x_s) \\ \lim_{\delta \lambda_r \rightarrow 0} \frac{(\delta x)^m}{\delta \lambda_r / \lambda_r} &= -m! \frac{\lambda_r \phi_r(x_s)}{P^{(m)}(x_s)} \end{aligned} \quad (1.7)$$

para una perturbación  $\delta \lambda_r$  en un coeficiente único  $\lambda_r$ . Si permitimos perturbaciones aleatorias de magnitudes relativamente fijas  $\frac{\varepsilon}{n+1} = \left| \frac{\delta \lambda_r}{\lambda_r} \right|$ ,  $r = 0, 1, \dots, n$  en cada coeficiente, sus contribuciones para  $(\delta x)^m$  se adicionan para definir un número de condición de raíz multiple  $C_\phi^{(m)}$ . Trabajando en el límite cuando  $\varepsilon \rightarrow 0$ :

$$\frac{|\delta x|^m}{\varepsilon / (n+1)} = m! \frac{|\lambda_r \phi_r(x_s)|}{|P^{(m)}(x_s)|},$$

obtenemos con  $\frac{\varepsilon}{n+1} = \left| \frac{\delta \lambda_r}{\lambda_r} \right|$ ,  $r = 0, 1, \dots, n$ :

$$\begin{aligned} |\delta x|^m &= m! |P^{(m)}(x_s)|^{-1} \sum_{r=0}^n |\lambda_r \phi_r(x_s)| \frac{\varepsilon}{n+1} \\ \Rightarrow |\delta x| &\leq \underbrace{\left( m! |P^{(m)}(x_s)|^{-1} \sum_{r=0}^n |\lambda_r \phi_r(x_s)| \right)^{1/m}}_{C_\phi^{(m)}} \varepsilon^{1/m} \end{aligned}$$

De esto se obtiene:

$$|\delta x| \leq C_\phi^{(m)} \varepsilon^{1/m} \quad \text{con} \quad C_\phi^{(m)} = \left[ \frac{m!}{|P^{(m)}(x_s)|} \sum_{r=0}^n |\lambda_r \phi_r(x_s)| \right]^{1/m}. \quad (1.8)$$

Una perturbación causará generalmente una raíz de  $m$ -cortes para multiplicarse en  $m$  distintas raíces, y algunas de estas o todas pueden ser complejas. La inecuación dado por (1.8) implica que estas raíces distintas, estarán contenidas dentro de un disco de radio  $R = C_\phi^{(m)} \varepsilon^{1/m}$  en el plano complejo centrado en  $(x_s, 0)$ .

Finalmente notamos que si el polinomio  $P(x)$  es el múltiplo de una de las funciones de la base  $\{\phi_k(x)\}$ , donde  $\phi(x_s) = 0$  ó  $\phi(x_s) = \phi'(x_s) = \phi''(x_s) = \dots = \phi^{(m-1)}(x_s) = 0$ , entonces el número de condición de sus raíces dados por (1.6) ó (1.8) son idénticamente igual a cero.

**Ejemplo 1.3.** Usando el ejemplo anterior, determinamos los números de condición de las raíces, con la perturbación realizada al coeficiente  $a_{19}$ , para dicho calculo se usa (1.6), para la base de potencia.

Los resultados han sido calculados con 4 dígitos significativos, y las evaluaciones han sido realizado en una vecindad de  $x_s = \frac{1}{2}$ . Estos resultados son malos (ver los resultados del ejemplo (1.2)), esto explica, porque el número de condición es muy grande, con lo cual se tiene un problema mal condicionado. En la siguiente tabla se muestran los números de condicionamiento de cada una de las raíces  $x_k$ .

k	$C_p(x_s)$	k	$C_p(x_s)$
1	$2.100 \times 10^1$	11	$7.822 \times 10^{12}$
2	$4.389 \times 10^3$	12	$1.707 \times 10^{13}$
3	$3.028 \times 10^5$	13	$2.888 \times 10^{13}$
4	$1.030 \times 10^7$	14	$3.777 \times 10^{13}$
5	$2.059 \times 10^8$	15	$3.777 \times 10^{13}$
6	$2.677 \times 10^9$	16	$2.833 \times 10^{13}$
7	$2.409 \times 10^{10}$	17	$1.541 \times 10^{13}$
8	$1.566 \times 10^{11}$	18	$5.742 \times 10^{12}$
9	$7.570 \times 10^{11}$	19	$1.310 \times 10^{12}$
10	$2.775 \times 10^{12}$	20	$1.378 \times 10^{11}$

Ver Farouki y Rajan '87 (p. 209).  $\square$

**Ejemplo 1.4.** Consideramos ahora el polinomio de progresión geométrica de las raíces reales  $x_s = \frac{2}{2^k}$ ,  $k = 1, 2, \dots, n$ , en el intervalo unitario  $[0, 1]$  de  $n = 20$ :

$$P(x) = \prod_{k=1}^{20} \left( x - \frac{2}{2^k} \right)$$

El número de condición de (1.6), en la base monomial sobre  $[0, 1]$  es calculado con aritmética exacta para cada raíz, y sus puntos flotantes son formados con 4 dígitos significativos, como se muestra a continuación:

k	$C_p(x_s)$	k	$C_p(x_s)$
1	$1.651 \times 10^1$	11	$1.323 \times 10^{-1}$
2	$2.477 \times 10^1$	12	$6.598 \times 10^{-2}$
3	$2.064 \times 10^1$	13	$3.275 \times 10^{-2}$
4	$1.327 \times 10^1$	14	$1.613 \times 10^{-2}$
5	$7.518 \times 10^0$	15	$7.815 \times 10^{-3}$
6	$4.001 \times 10^0$	16	$3.671 \times 10^{-3}$
7	$2.064 \times 10^0$	17	$1.620 \times 10^{-3}$
8	$1.048 \times 10^0$	18	$6.299 \times 10^{-4}$
9	$5.279 \times 10^{-1}$	19	$1.890 \times 10^{-4}$
10	$2.647 \times 10^{-1}$	20	$3.149 \times 10^{-5}$

En este ejemplo las raíces son razonablemente bien-condicionadas en la forma potencial, porque el número de condición es pequeño, ver Farouki y Rajan '87 (p. 211).  $\square$

## Número de Condición para la Evaluación del Polinomio

Otra consideración importante en la evaluación polinomial es la estabilidad con respecto a la perturbación en los coeficientes. Esto depende de las bases del polinomio escogido, más no de la evaluación del algoritmo (el cual, se asume, opera en aritmética de precisión finita). Consideremos la magnitud  $|\delta P|$  de los cambios en  $P(x)$  inducido por la perturbación de la magnitud relativa  $\varepsilon$  en cada coeficiente. Se verifican las siguientes cotas:

$$\begin{aligned} \delta P &= \sum_{k=0}^n \delta \lambda_k \phi_k(x_s) \\ |\delta P| &= \left| \sum_{k=0}^n \frac{\delta \lambda_k}{\lambda_k} \lambda_k \phi_k(x_s) \right| \\ |\delta P| &\leq S_\phi(x_s) \varepsilon, \quad \text{donde } S_\phi(x_s) = \sum_{k=0}^n |\lambda_k \phi_k(x_s)|. \end{aligned} \quad (1.9)$$

La suma  $S_\phi(x_s)$  es el número de condición para la evaluación del polinomio  $P(x)$  en  $x_s$  en la base  $\phi$ . El cual juega un rol muy importante en los condicionamientos de las raíces de los polinomios.

## Condición Matricial

Sean  $M = (m_{ij}) \in \mathbb{R}^{n+1} \times \mathbb{R}^{n+1}$  ó  $\mathbb{C}^{n+1} \times \mathbb{C}^{n+1}$   $i, j = 0, 1, \dots, n$ , y  
 $p = (p_i) \in \mathbb{R}^{n+1}$  ó  $\mathbb{C}^{n+1}$   $i = 0, 1, \dots, n$   
 $M^{-1} = (m_{ij}^{-1}) \in \mathbb{R}^{n+1} \times \mathbb{R}^{n+1}$  ó  $\mathbb{C}^{n+1} \times \mathbb{C}^{n+1}$   $i, j = 0, 1, \dots, n$ , y  
 $q = (q_i) \in \mathbb{R}^{n+1}$  ó  $\mathbb{C}^{n+1}$   $i = 0, 1, \dots, n$ .

Sea  $\|\cdot\|$  una norma de vectores en  $\mathbb{R}^{n+1}$  ó  $\mathbb{C}^{n+1}$ .

La norma matricial inducida por la norma  $\|\cdot\|$  se define por:

$$\|M\| = \sup_{x \neq 0} \frac{\|Mx\|}{\|x\|}. \quad (1.10)$$

Para las normas matriciales más usadas se cumplen:

$$\|M\|_1 = \max_{0 \leq j \leq n} \sum_{i=0}^n |m_{ij}|, \quad \|M\|_\infty = \max_{0 \leq i \leq n} \sum_{j=0}^n |m_{ij}| \quad (1.11)$$

y

$$\|M\|_2 = \max_{0 \leq i \leq n} \sqrt{\lambda_i}, \quad (1.12)$$

donde los  $\lambda_0, \lambda_1, \dots$  son los  $(n+1)$  valores propios (reales, no negativos) de la matriz simétrica  $M^T M$ .

## Transformación de $q$ a $p$ por $M$

Dado el sistema  $Mp = q$ . Si queremos transformar el vector  $p$  por  $M$  al valor  $q$  nos interesa como influye el error relativo  $\frac{\delta p}{p}$ .

Sea  $\delta q$  el error absoluto de  $q$  y  $\delta p$  el error absoluto de  $p$ . De  $q = Mp$  se tiene  $(q + \delta q) = M(p + \delta p)$ , donde  $\delta q = M\delta p$ . También se cumple  $\|\delta q\| \leq \|M\| \|\delta p\|$ .

Como  $p = M^{-1}q$ , resulta  $\|p\| \leq \|M^{-1}\| \|q\|$ . Luego obtenemos:

$$\frac{\|\delta q\|}{\|q\|} \leq \|M\| \|M^{-1}\| \frac{\|\delta p\|}{\|p\|}. \quad (1.13)$$

El número  $Cond(M) := \|M\| \|M^{-1}\|$  es llamado la condición de  $M$ .

Definiendo  $\epsilon_p = \frac{\|\delta p\|}{\|p\|}$  y  $\epsilon_q = \frac{\|\delta q\|}{\|q\|}$ , se tiene:

$$\epsilon_q \leq Cond(M) \epsilon_p. \quad (1.14)$$

Se calcula la condición y se tiene el error de  $p$ , entonces se puede determinar el error de  $q$ , ver en Farouki '91 (p. 31).

## Solucionar un Sistema Lineal

Un sistema lineal de  $n \times n$  es un sistema de  $n$  ecuaciones lineales con  $n$  incógnitas. Nuestra notación para un sistema lineal general es  $Ax = b$ , donde  $A$  no singular y  $b \neq 0$ . Deseamos obtener una descripción cuantitativa de cómo pequeños cambios en  $A$  o  $b$  pueden afectar la solución  $x = A^{-1}b$ . Supongamos que  $A$  es exacta pero  $b$  es alterada por  $\delta b$ . Si designamos a la solución exacta del sistema alterado por  $x + \delta x$ , entonces  $A(x + \delta x) = b + \delta b$ ; de donde,  $A\delta x = \delta b$ , esto es,  $\delta x = A^{-1}\delta b$ . Aplicando norma a  $b = Ax$  y  $\delta x = A^{-1}\delta b$  se obtiene  $\|b\| \leq \|A\|\|x\|$  y  $\|\delta x\| \leq \|A^{-1}\|\|\delta b\|$ , de lo cual

$$\frac{\|\delta x\|}{\|x\|} \leq \text{Cond}(A) \frac{\|\delta b\|}{\|b\|}.$$

Puesto que las razones  $\frac{\|\delta x\|}{\|x\|}$  y  $\frac{\|\delta b\|}{\|b\|}$  pueden interpretarse como los cambios relativos respectivos de  $x$  y  $b$  muestra que el cambio relativo en la norma del vector solución no puede ser mayor que  $\text{Cond}(A)$  multiplicado por el vector de términos independientes. De manera similar, si  $A$  tiene un error  $\delta A$  pero  $b$  es exacta, entonces la desigualdad

$$\frac{\|\delta x\|}{\|x\|} \leq \text{Cond}(A) \frac{\|\delta A\|}{\|A\|}, \text{ donde } \text{Cond}(A) = \|A\| \cdot \|A^{-1}\|,$$

puede obtenerse e interpretarse de modo análogo, ver Melvin [8].

**Ejemplo 1.5.** Sea

$$A = \begin{pmatrix} 1 & 0 \\ 0 & \epsilon \end{pmatrix} \quad y \quad A^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\epsilon} \end{pmatrix}$$

Para  $\epsilon > 0$  pequeño se cumple  $\text{Cond}(A) = \frac{1}{\epsilon} \gg 1$  para las normas  $\|x\|_2$ ,  $\|x\|_1$  o  $\|x\|_\infty$ . Supongamos que  $b = (1 \ \epsilon)^T$  se altera por  $\delta b = (0 \ 0.01\epsilon)^T$  convirtiéndose en  $b_1 = (1 \ 1.01\epsilon)^T$ , y que la solución exacta del sistema lineal  $Ax = b$  es  $x = A^{-1}b = (b_1 \ \frac{b_2}{\epsilon})^T$ ,  $x = (1 \ 1)^T$  se altera por  $\delta x = (0 \ 0.01)^T$  para convertirse en  $x_1 = (1 \ 1.01)^T$ . Usando  $\|\cdot\|_\infty$ , obtenemos  $\frac{\|\delta b\|}{\|b\|} = \frac{0.01\epsilon}{1} = 0.01\epsilon$  y  $\frac{\|\delta x\|}{\|x\|} = 0.01\epsilon$ , así que  $\frac{\|\delta x\|}{\|x\|} = \frac{1}{\epsilon} \frac{\|\delta b\|}{\|b\|}$ , de modo que el valor grande de  $\text{Cond}(A) = \frac{1}{\epsilon}$  indica en forma correcta que un pequeño cambio  $\epsilon\%$  en  $\|b\|$  puede causar un cambio mucho mayor de  $1\%$  en  $\|x\|$ .

## 1.2 Errores de la Evaluación de un Polinomio por el Algoritmo de Horner

En este capítulo se muestra el funcionamiento del algoritmo de Horner, y sus respectivos errores que se generan debido a los cálculos realizados por las computadoras. Este algoritmo fue considerado eficiente por mucho tiempo, pero actualmente existen otros algoritmos que son más convenientes de utilizar.

El Algoritmo de Horner (también conocido como multiplicador de nodos) se viene usando desde el tiempo de Newton para la evaluación de polinomios en forma potencial, donde  $P(x) = \sum_{i=0}^n a_i x^i$ ,  $a_i \in \mathbb{R}$  ó  $\mathbb{C}$ .

Se calcula el polinomio por la recurrencia:

$$\begin{aligned} P_0 &:= a_n \\ P_r &:= xP_{r-1} + a_{n-r} \text{ para } r = 1, 2, \dots, n \end{aligned} \quad (1.15)$$

Mediante (1.15) se evalúa el polinomio  $P(x)$  en un punto  $x = x_s$ . Se obtiene finalmente  $P(x_s) \equiv P_n$ . Asociados a la multiplicación y adición en cada estado  $r$  de (1.15) están los errores aleatorios  $|\pi_r|, |\sigma_r| \leq \eta$ , donde  $\eta$  está dado en (1.1). El método de Horner es eficiente. Se requiere  $\frac{n(n+1)}{2}$  multiplicaciones y

$n$  adiciones para la evaluación usual, mientras que por el método de Horner, se requiere  $n$  multiplicaciones y  $n$  adiciones, reduciéndose en un 60% el número de multiplicaciones requeridas.

Consideremos el comportamiento del algoritmo con una ejecución en aritmética de punto flotante de precisión finita.

Sean  $a_i$ ,  $i = 1, 2, \dots, n$  los números de máquina. Si el error  $e_r^*$  denota la magnitud de error máximo en la cantidad calculado  $P_r$ , entonces tenemos manteniendo solo términos de primer orden en pequeñas cantidades.

Definimos  $e_r^* := fl[P_r] - P_r$ , para  $r = 1, 2, \dots, n$ , además resulta  $e_{r-1}^* \eta = 0$  porque  $e_r^* = fl[P_r] - P_r = P_r(1 + \eta) - P_r = P_r \eta$ , para  $r = 1, 2, \dots, n$ , y se está trabaja en punto flotante, ver Farouki y Rajan '87, (p. 196).

$$\begin{aligned}
 e_r^* &= fl[P_r] - P_r \\
 &= fl[xP_{r-1} + a_{n-r}] - P_r \\
 &= \{x fl[P_{r-1}](1 + \eta) + a_{n-r}\}(1 + \eta) - P_r \\
 &= \{x(fl[P_{r-1}] - P_{r-1} + P_{r-1})(1 + \eta) + a_{n-r}\}(1 + \eta) - P_r \\
 &= \{x(e_{r-1}^* + P_{r-1})(1 + \eta) + a_{n-r}\}(1 + \eta) - P_r \\
 &= \{xe_{r-1}^* + xP_{r-1}\eta + P_r\}(1 + \eta) - P_r \\
 &= xe_{r-1}^* + xP_{r-1}\eta + P_r\eta.
 \end{aligned}$$

La estimación siguiente nos permite obtener la cota superior sobre la magnitud del error de redondeo  $|\delta P|$  en el valor calculado  $P(x_s)$  del polinomio para  $x_s \in [0, 1]$ .

$$e_r \leq |x|e_{r-1} + \{|x||P_{r-1}| + |P_r|\}\eta.$$

Inicializando  $e_0 = 0$  tenemos:

$$\begin{aligned}
 e_1 = |e_1^*| &< |x|e_0 + \{|x||P_0| + |P_1|\}\eta \\
 &\quad \{ |x||P_0| + |P_1| \} \eta
 \end{aligned}$$



$$\begin{aligned}
e_2 &\leq |x|e_1 + \{|x||P_1| + |P_2|\}\eta \\
&= |x|\{|x||P_0| + |P_1|\}\eta + \{|x||P_1| + |P_2|\}\eta \\
&= \{|x|^2|P_0| + 2|x||P_1| + |P_2|\}\eta \\
e_3 &\leq |x|e_2 + \{|x||P_2| + |P_3|\}\eta \\
&= |x|\{|x|^2|P_0| + 2|x||P_1| + |P_2|\}\eta + \{|x||P_2| + |P_3|\}\eta \\
&= \{|x|^3|P_0| + 2|x|^2|P_1| + 2|x||P_2| + |P_3|\}\eta \\
&\vdots \\
e_r &\leq \{|x|^r|P_0| + 2|x|^{r-1}|P_1| + \dots + 2|x||P_{r-1}| + |P_r|\}\eta
\end{aligned}$$

**Ejemplo 1.6.** Consideremos el polinomio  $P(x) = x^2 - 2x + 1$  y su perturbación  $\tilde{P}(x) = x^2 - 2.00001x + 1$ . Por el algoritmo de Horner, obtenemos:

$$\tilde{P}(x) = (x - 2.00001)x + 1.$$

Los coeficientes del polinomio  $\tilde{P}(x)$  son  $\{a_0, a_1, a_2\} = \{1, -2.00001, 1\}$ , y las raíces son  $x_1 = 1.0032$  y  $x_2 = 0.9968$ , donde seleccionamos  $x_p = x_2$ . Las raíces exactas son  $x_1 = x_2 = 1$ .

Sabemos que el error relativo para este polinomio de segundo orden es:

$$e_2 \leq \{|x|^2|P_0| + 2|x||P_1| + |P_2|\}\eta$$

Determinemos los  $\tilde{P}_0, \tilde{P}_1, \tilde{P}_2$  y  $P_0, P_1, P_2$ , según la recurrencia del polinomio:

$$\tilde{P}_2 = (1(0.9968) - 2.00001)(0.9968) + 1 = 0.0000003 \quad P_2 = 0$$

$$\tilde{P}_1 = 1(0.9968) - 2.00001 = -1.003, \quad P_1 = -1$$

$$\tilde{P}_0 = 1, \quad P_0 = 1.$$

Reemplazando se tiene:  $\tilde{e}_2 = 2.994\eta$ .

En el caso del error absoluto del polinomio no perturbado resulta:

$$e_2 = \{|1|^2|1| + 2|1||-1| + |0|\}\eta = 3\eta.$$

Si consideramos  $\eta = 0.5 \times 10^{-3}$  con una mantisa de 4 dígitos, se obtiene  $\tilde{e}_2 \leq 1.497 \times 10^{-3}$  cuya cota es mayor que  $\eta$ , y para el otro caso  $e_2 = 1.5 \times 10^{-3}$ , que también es mayor que  $\eta$ . Por lo cual concluimos que el método aplicado no es el adecuado. Nuestro objetivo es encontrar que el error relativo total sea menor que  $\eta$ .  $\square$

Denotando

$$P_M := \max\{|x|^r|P_0|, |x|^{r-1}|P_1|, \dots, |P_r|\},$$

y poniendo  $|\delta P| = |e_n|$ , obtenemos:

$$\begin{aligned} |\delta P| &\leq \{P_M + 2P_M + \dots + 2P_M + P_M\}\eta \\ &= \{2P_M + 2P_M + \dots + 2P_M\}\eta = 2nP_M\eta \end{aligned}$$

Otra consideración importante en la evaluación polinomial es la estabilidad con respecto a la perturbación en los coeficientes.

## Determinación del Error del Polinomio

Se debe tener en cuenta los coeficientes del polinomio. Dependiendo si son o no son números de máquina se presentan dos casos:

**Primer Caso :** Sean  $a_i$ ,  $i = 1, \dots, n$  y  $x$  números de máquina, y  $\sigma_i$  y  $\pi_i$ ,  $i = 1, 2, \dots, n$  son los errores de la adición y multiplicación, respectivamente. Para determinar el polinomio de grado  $n$  con su respectivo error relativo, se usará la ecuación (1.15). Veamos por inducción:

Para  $n = 1$ , se tiene

$$\begin{aligned}
fl(P_1) &= fl(xP_0 + a_{n-1}) \\
&= [xa_n(1 + \pi_1) + a_{n-1}](1 + \sigma_1) \\
&= (xa_n + a_{n-1}) + [xa_n(\pi_1 + \sigma_1) + a_{n-1}\sigma_1].
\end{aligned}$$

Para  $n = 2$ , se tiene

$$\begin{aligned}
fl(P_2) &= fl[x(xa_n + a_{n-1}) + a_{n-2}] \\
&= \{x[xa_n(1 + \pi_1) + a_{n-1}](1 + \sigma_1)(1 + \pi_2) + a_{n-2}\}(1 + \sigma_2) \\
&= (x^2a_n + xa_{n-1} + a_{n-2}) + [x^2a_n(\pi_1 + \pi_2 + \sigma_1 + \sigma_2) + \\
&\quad xa_{n-1}(\pi_2 + \sigma_1 + \sigma_2) + a_{n-2}\sigma_2].
\end{aligned}$$

Supongamos que para  $n = m$  se cumple

$$\begin{aligned}
fl(P_m) &= (x^m a_m + \dots + a_0) + [x^m a_m \{ \sum_{r=1}^m \pi_r + \sum_{r=1}^m \sigma_r \} + \\
&\quad \sum_{k=1}^{m-1} [ \sum_{r=k+1}^m \pi_r + \sum_{r=k}^m \sigma_r ] x^{m-k} a_{m-k} + a_0 \sigma_m].
\end{aligned}$$

Veamos el caso  $n = m + 1$ , entonces

$$\begin{aligned}
fl(P_{m+1}) &= fl[xP_m + a_0] \\
&= [x\{P_m + x^m a_{m+1} [x^m a_m \{ \sum_{r=1}^m \pi_r + \sum_{r=1}^m \sigma_r \} + \sum_{k=1}^{m-1} [ \sum_{r=k+1}^m \pi_r + \\
&\quad \sum_{r=k}^m \sigma_r ] x^{m-k} a_{m-k+1} + a_1 \sigma_m \} \} (1 + \pi_{n+1}) + a_0] (1 + \sigma_{m+1}) \\
&= (x^{m+1} a_{m+1} + \dots + a_0) + [x^{m+1} a_{m+1} \{ \sum_{r=1}^{m+1} \pi_r + \sum_{r=1}^{m+1} \sigma_r \} + \\
&\quad \sum_{k=1}^m [ \sum_{r=k+1}^{m+1} \pi_r + \sum_{r=k}^{m+1} \sigma_r ] x^{m-k+1} a_{m-k+1} + a_0 \sigma_{m+1}].
\end{aligned}$$

De tal manera se cumple:

$$\begin{aligned}
 fl(P_n) &= (x^n a_n + \cdots + a_0) + [x^n a_n \{ \sum_{r=1}^n \pi_r + \sum_{r=1}^n \sigma_r \} + \\
 &\quad \sum_{k=1}^{n-1} [ \sum_{r=k+1}^n \pi_r + \sum_{r=k}^n \sigma_r ] x^{n-k} a_{n-k} + a_0 \sigma_n] \quad (1.16)
 \end{aligned}$$

donde  $|\pi_r|, |\sigma_r| \leq \eta$  para  $r = 1, 2, \dots, n$ .

El error absoluto  $\delta P = |fl[P(x)] - P(x)|$  en el valor calculado (1.16) es entonces:

$$\delta P = x^n a_n \{ \sum_{r=1}^n \pi_r + \sum_{r=1}^n \sigma_r \} + \sum_{k=1}^{n-1} [ \sum_{r=k+1}^n \pi_r + \sum_{r=k}^n \sigma_r ] x^{n-k} a_{n-k} + a_0 \sigma_n \quad (1.17)$$

Estamos interesados en los valores de las variables aleatorias  $\{\pi_r\}$  y  $\{\sigma_r\}$ , sujeto a las restricciones  $|\pi_r|, |\sigma_r| \leq \eta$  para  $r = 1, 2, \dots, n$ , si se considera  $|\pi_r|, |\sigma_r| = \eta$  el cual a (1.17) determinar una cota de error máximo para el valor calculado. Esto corresponde a un problema de programación lineal en  $2n$  variables, y para sacar posteriores conclusiones teniendo en cuenta su solución, necesitamos información concreta guardando sus signos y la magnitud de los coeficientes  $a_0, a_1, \dots, a_n$ .

Por consiguiente, se debe tener presente la asistencia de un error, el cual no puede ser despreciado, que es arrastrado en el cálculo de las raíces del polinomio o en cualquier otra operación con dicho polinomio. Debemos recordar que los errores se incrementan en cada operación que se realiza, ver Farouki y Rajan '88, (p. 11, 12).

**Ejemplo 1.7.** Analizaremos para este caso, el Ejemplo 1.6 con  $n = 2$ . Resulta:

$$fl(P_2) = P_2 + [(x_s)^2 a_2] \{ \sum_{r=1}^2 \pi_r + \sum_{r=1}^2 \sigma_r \} + \sum_{k=1}^{2-1} [ \sum_{r=k+1}^2 \pi_r + \sum_{r=k}^2 \sigma_r ] x^{2-k} a_{2-k} + a_0 \sigma_2$$

Los coeficientes potenciales son  $\{a_0, a_1, a_2\} = \{1, -2.00001, 1\}$ .

Sabemos que el error absoluto para este polinomio de segundo orden es:

$$|\delta P| = fl(P_2) - P_2 = [\pi_1 + \pi_2 + \sigma_1 + \sigma_2] a_2 x^2 + [\pi_2 + \sigma_1 + \sigma_2] a_1 x + a_0 \sigma_2.$$

Procedemos a determinar los  $\pi_i, \sigma_i$ :

$$\begin{aligned}\pi_1 &= \left| \frac{0.9968 - 0.9968}{0.9968} \right| = 0 \\ \sigma_1 &= \left| \frac{-1.003 + 1.00321}{-1.00321} \right| = 2 \times 10^{-4} \\ \pi_2 &= \left| \frac{0.9998 - 0.9997904}{0.9997904} \right| = 10^{-5} \\ \sigma_2 &= \left| \frac{0.0002 - 0.0002}{0.0002} \right| = 0.\end{aligned}$$

Se obtiene  $|\pi_i|, |\sigma_i| \leq 0.5 \times 10^{-3}$ . Para  $i = 1, 2$ , resulta:

$$\begin{aligned}|\delta P| &= |[0 + 20 \times 10^{-5} + 10^{-5} + 0](1)(0.9968)^2 + \\ &\quad [20 \times 10^{-5} + 10^{-5} + 0](-2.00001)(0.9968) - (1)(0)| \\ &= |21 \times 10^{-5}[0.9936 - 1.994]| \\ &= 0.2 \times 10^{-3}\end{aligned}$$

El resultado es menor que  $\eta = 0.5 \times 10^{-3}$  según (1.1) donde estamos considerando 4 dígitos significativos, lo cual nos indica que se ha obtenido una mejor aproximación utilizando este método en lugar que el ejemplo anterior.  $\square$

**Segundo Caso :** Los  $a_i, i = 1, 2, \dots, n$  y  $x$  no sean números de máquina, y por consiguiente los errores aleatorios de redondeo sea  $|\delta_i| \leq \eta, i = 1, 2, \dots, n$ . Se procede a convertir a números de máquina:  $fl(a_i)$ , para  $i = 1, \dots, n$  y  $fl(x)$ . Procedemos a determinar el polinomio de grado  $n$  con su respectivo error relativo usando el método de HORNER.

Veamos por inducción:

Para  $n = 1$ , se tiene

$$\begin{aligned}fl(P_1) &= fl(xP_0 + a_{n-1}) \\ &= [x(1 + \eta_x)a_n(1 + \delta_n)(1 + \pi_1) + a_{n-1}(1 + \delta_{n-1})](1 + \sigma_1) \\ &= (xa_n + a_{n-1}) + [xa_n(\delta_x + \delta_n + \pi_1 + \sigma_1) + a_{n-1}(\delta_{n-1} + \sigma_1)].\end{aligned}$$

Para  $n = 2$ , resulta

$$\begin{aligned}
fl(P_2) &= fl[x(xa_n + a_{n-1}) + a_{n-2}] \\
&= \{x[fl(xa_n) + a_{n-1}(1 + \delta_{n-1})](1 + \sigma_1)(1 + \pi_2) + a_{n-2}(1 + \delta_{n-2})\}(1 + \sigma_2) \\
&= \{x[x(1 + \delta_x)a_n(1 + \delta_n)(1 + \pi_1) + a_{n-1}(1 + \delta_{n-1})](1 + \sigma_1)(1 + \pi_2) + \\
&\quad a_{n-2}(1 + \delta_{n-2})\}(1 + \sigma_2) \\
&= (x^2a_n + xa_{n-1} + a_{n-2}) + [x^2a_n(2\delta_x + \delta_n + \pi_1 + \pi_2 + \sigma_1 + \sigma_2) + \\
&\quad xa_{n-1}(\delta_x + \delta_{n-1} + \pi_2 + \sigma_1 + \sigma_2) + a_{n-2}(\delta_{n-2} + \sigma_2)].
\end{aligned}$$

Supongamos que para  $n = m$  se cumple

$$\begin{aligned}
fl(P_m) &= (x^m a_m + \dots + a_0) + [x^m a_m \{m\delta_x + \delta_m + \sum_{r=1}^m \pi_r + \sum_{r=1}^m \sigma_r\} + \\
&\quad \sum_{k=1}^{m-1} [k\delta_x + \sum_{r=k}^{m-1} \delta_{m-r} + \sum_{r=k+1}^m \pi_r + \sum_{r=k}^m \sigma_r] a_{m-r} x^{m-k} + a_0(\delta_0 + \sigma_m)].
\end{aligned}$$

Para  $n = m + 1$ . Se tiene entonces

$$\begin{aligned}
fl(P_{m+1}) &= fl[xP_m + a_0] \\
&= P_{m+1} + xP_{m+1}\pi_{m+1} + xP_{m+1}\delta_x + [x^m a_m \{m\delta_x + \delta_m + \sum_{r=1}^m \pi_r + \sum_{r=1}^m \sigma_r\} \\
&\quad + \sum_{k=1}^{m-1} [k\delta_x + \sum_{r=k}^{m-1} \delta_{m-r} + \sum_{r=k+1}^m \pi_r + \sum_{r=k}^m \sigma_r] a_{m-r} x^{m-k} + a_0(\delta_0 + \sigma_m)]
\end{aligned}$$

$$\begin{aligned}
fl(P_{m+1}) &= P_{m+1} + [x^{m+1} a_{m+1} \{(m+1)\delta_x + \delta_{m+1} + \sum_{r=1}^{m+1} \pi_r + \sum_{r=1}^{m+1} \sigma_r\} + \sum_{k=1}^m [k\delta_x + \\
&\quad \sum_{r=k}^m \delta_{m+1-r} + \sum_{r=k+1}^{m+1} \pi_r + \sum_{r=k}^{m+1} \sigma_r] a_{m+1-r} x^{m+1-k} + a_0(\delta_0 + \sigma_{m+1})],
\end{aligned}$$

Por consiguiente se cumple

$$fl(P_n) = (x^n a_n + \dots + a_0) + [x^n a_n \{n\delta_x + \delta_n + \sum_{r=1}^n \pi_r + \sum_{r=1}^n \sigma_r\} + \sum_{k=1}^{n-1} [k\delta_x + \sum_{r=k}^{n-1} \delta_{n-r} + \sum_{r=k+1}^n \pi_r + \sum_{r=k}^n \sigma_r] a_{n-r} x^{n-k} + a_0(\delta_0 + \sigma_n)]. \quad (1.18)$$

Se observa que cuando los coeficientes ingresados, no son números de máquina, el error relativo aumenta, lo cual nos indica que tenemos que tener bastante cuidado al trabajar con dichos números. Podemos encontrar casos en donde el error relativo resultaría mucho mayor que el número y se debe evitar tales casos con los métodos conocidos.

### 1.3 Errores en el Cálculo de los Ceros de Polinomios

Como los errores relativos son extendidos a través de los cálculos que se realicen en la determinación de los ceros del polinomio, se presenta el Método de Horner Iterado para el cálculo de una derivada, que se usa para la aplicación del método de Newton-Raphson. Los siguientes teoremas son útiles para la evaluación del polinomio, para determinar a su vez la primera derivada y encontrar sus raíces.

**Teorema 1.1.** *Sea  $P(x) = a_n x^n + \dots + a_1 x + a_0$  y  $b_n = a_n$ . Si  $b_k = a_k + b_{k+1} x_s$  para  $k = n - 1, n - 2, \dots, 1, 0$ , entonces  $b_0 = P(x_s)$ , donde  $a_i, b_i \in \mathbb{R}$ . Además, si*

$$Q(x) = b_n x^{n-1} + b_{n-1} x^{n-2} + \dots + b_2 x + b_1$$

entonces:

$$P(x) = (x - x_s)Q(x) + b_0.$$

Se obtiene:

$$P'(x) = Q(x) + (x - x_s)Q'(x) \implies P'(x_s) = Q(x_s)$$

**Demostración:** Por la definición de  $Q(x)$ .

$$\begin{aligned} (x - x_s)Q(x) + b_0 &= (x - x_s)(b_n x^{n-1} + b_{n-1} x^{n-2} + \dots + b_2 x + b_1) + b_0 \\ &= b_n x^n + (b_{n-1} - b_n x_s) x^{n-1} + \dots + (b_1 - b_2 x_s) x + (b_0 - b_1 x_s) \end{aligned}$$

Por las hipótesis del teorema  $b_n = a_n$  y  $b_k - b_{k+1} x_s = a_k$ , así que:

$$(x - x_s)Q(x) + b_0 = P(x) \text{ y } b_0 = P(x_s). \blacksquare$$

El proceso de evaluación de  $P(x_s)$  es mostrado en la siguiente tabla:

$x_0 = x_s$	$a_n$	$a_{n-1}$	$\dots$	$a_1$	$a_0$
	$b_n \times x_0$	$\dots$	$b_1 \times x_0$	$b_0 \times x_0$	
	$b_n$	$b_{n-1}$	$\dots$	$b_1$	$b_0 = P(x_0)$

En forma analoga se procede para evaluar  $P'(x_s)$ . Ahora determinamos las raíces de una ecuación polinomial  $P(x) = 0$ .

Sea  $P(x) = a_n x^n + \dots + a_1 x + a_0$  con  $x_0 = x_s$ , usando el Teorema 1.1 se obtienen las evaluaciones de  $P(x_s)$  y  $P'(x_s)$ . Para hallar un cero de  $P(x)$  con el método de Newton-Raphson se ejecutan las iteraciones  $x_k := x_{k-1} - \frac{P(x_{k-1})}{P'(x_{k-1})}$ , hasta que  $|x_k - x_{k-1}| \leq \eta$ , donde  $\eta$  esta dado por (1.1).

**Ejemplo 1.8.** Sea  $P(x) = 4x^4 + 3x^3 - 2x^2 + 4x - 8$  con  $x_0 = 1.5$ , consideramos 5 dígitos significativos y en base 10.

Sean  $a_i$ , para  $i = 0, 1, 2, 3$  números de máquina:  $a_0 = -0.80000 \times 10$ ,



$a_1 = 0.40000 \times 10$ ,  $a_2 = -0.20000 \times 10$ ,  $a_3 = 0.30000 \times 10$  y  $a_4 = 0.40000 \times 10$ . Por los teoremas 1.1 y 1.2 se obtiene:

$x_0 = 1.5000$	$0.40000 \times 10$	$0.30000 \times 10$	$-0.20000 \times 10$	$0.40000 \times 10$	$-0.80000 \times 10$
		$0.60000 \times 10$	$0.13500 \times 10^2$	$0.17250 \times 10^2$	$0.31875 \times 10^2$
	$0.40000 \times 10$	$0.90000 \times 10$	$0.11500 \times 10^2$	$0.21250 \times 10^2$	$0.23875 \times 10^2$ $=P(x_0)$
		$0.60000 \times 10$	$0.22500 \times 10^2$	$0.51000 \times 10^2$	
	$0.40000 \times 10$	$0.15000 \times 10^2$	$0.34000 \times 10^2$	$0.72250 \times 10^2$ $=P'(x_0)$	

$$x_1 = 0.15000 \times 10 - \frac{0.23875 \times 10^2}{0.72250 \times 10^2} = 0.11696 \times 10$$

$$e_1 = |x_1 - x_0| = |1.1696 - 1.5| = 0.3304$$

$x_1 = 1.1696$	$0.40000 \times 10$	$0.30000 \times 10$	$-0.20000 \times 10$	$0.40000 \times 10$	$-0.80000 \times 10$
		$0.46784 \times 10$	$0.89807 \times 10$	$0.81646 \times 10$	$0.14228 \times 10^2$
	$0.40000 \times 10$	$0.76784 \times 10$	$0.69807 \times 10$	$0.12165 \times 10^2$	$0.62282 \times 10$ $=P(x_1)$
		$0.46784 \times 10$	$0.14453 \times 10^2$	$0.25069 \times 10^2$	
	$0.40000 \times 10$	$0.12357 \times 10^2$	$0.21434 \times 10^2$	$0.37234 \times 10^2$ $=P'(x_1)$	

$$x_2 = 0.11696 \times 10 - \frac{0.62282 \times 10}{0.37234 \times 10^2} = 0.10023 \times 10$$

$$e_2 = |x_2 - x_1| = |1.0023 - 1.1696| = 0.1673$$

$x_2 = 1.0023$	$0.40000 \times 10$	$0.30000 \times 10$	$-0.20000 \times 10$	$0.40000 \times 10$	$-0.80000 \times 10$
		$0.40092 \times 10$	$0.70253 \times 10$	$0.50369 \times 10$	$0.90577 \times 10$
	$0.40000 \times 10$	$0.70092 \times 10$	$0.50253 \times 10$	$0.90369 \times 10$	$0.10577 \times 10$ $=P(x_2)$
		$0.40092 \times 10$	$0.11043 \times 10^2$	$0.16105 \times 10^2$	
	$0.40000 \times 10$	$0.11018 \times 10^2$	$0.16068 \times 10^2$	$0.25142 \times 10^2$ $=P'(x_2)$	

$$x_3 = 0.10023 \times 10 - \frac{0.10577 \times 10}{0.25142 \times 10^2} = 0.96023$$

$$e_3 = |x_3 - x_2| = |0.96023 - 1.0023| = 0.04207$$

$x_3 = 0.96023$	$0.40000 \times 10$	$0.30000 \times 10$	$-0.20000 \times 10$	$0.40000 \times 10$	$-0.80000 \times 10$
		$0.38409 \times 10$	$0.65688 \times 10$	$0.43871 \times 10$	$0.80535 \times 10$
	$0.40000 \times 10$	$0.68409 \times 10$	$0.45688 \times 10$	$0.83871 \times 10$	$\underbrace{0.53500 \times 10^{-1}}_{=P(x_3)}$
		$0.38409 \times 10$	$0.10257 \times 10^2$	$0.14236 \times 10^2$	
	$0.40000 \times 10$	$0.10682 \times 10^2$	$0.14826 \times 10^2$	$\underbrace{0.22623 \times 10^2}_{=P'(x_3)}$	

$$x_4 = 0.96023 - \frac{0.50500 \times 10^{-1}}{0.22623 \times 10} = 0.95787$$

$$e_4 = |x_4 - x_3| = |0.95787 - 0.96023| = 0.00236$$

$x_4 = 0.95787$	$0.40000 \times 10$	$0.30000 \times 10$	$-0.20000 \times 10$	$0.40000 \times 10$	$-0.80000 \times 10$
		$0.38315 \times 10$	$0.65437 \times 10$	$0.43523 \times 10$	$0.80004 \times 10$
	$0.40000 \times 10$	$0.68315 \times 10$	$0.45437 \times 10$	$0.83523 \times 10$	$\underbrace{0.40000 \times 10^{-3}}_{=P(x_4)}$
		$0.38315 \times 10$	$0.10214 \times 10^2$	$0.14136 \times 10^2$	
	$0.40000 \times 10$	$0.10663 \times 10^2$	$0.14758 \times 10^2$	$\underbrace{0.22488 \times 10^2}_{=P'(x_4)}$	

$$x_5 = 0.95787 - \frac{0.40000 \times 10^{-3}}{0.22488 \times 10^2} = 0.95785$$

$$e_5 = |x_5 - x_4| = |0.95785 - 0.95787| = 0.00002$$

*El valor aproximado obtenido nos garantiza una mejor precisión a la raíz del polinomio. Hay que tener presente que en cada calculo que se realiza se tiene un error de redondeo. La suma de todos ellos da el error redondeo total, el cual tiene que ser considerado también. □*

## 2 TEOREMAS BÁSICOS DEL DISEÑO GEOMÉTRICO

Los polinomios de Bernstein fueron originalmente introducidos por Bernstein (inició su trabajo en 1912), en la aproximación de funciones continuas  $f(x)$  en un intervalo  $[\alpha, \beta]$ . El Teorema de Weierstrass dice: Si  $f$  está definida y es continua en  $[0, 1]$ , dado  $\epsilon > 0$ , entonces existe un polinomio  $P_n$ , definido en  $[0, 1]$ , con la propiedad de que

$$|P_n(x) - f(x)| < \epsilon, \forall x \in [0, 1]. \quad (2.1)$$

Esto es, decimos que la sucesión de polinomios  $P_n$  converge uniformemente a  $f$ ,

$$\lim_{n \rightarrow \infty} P_n(x) = f(x), \quad (2.2)$$

sobre el intervalo  $[0, 1]$ . A través de los polinomios de Bernstein se puede probar este teorema de una manera elegante.

**Definición 2.1.** Para  $n \in \mathbb{N}$ , se definen los polinomios de Bernstein

$$B_k^n(x) = \binom{n}{k} (1-x)^{n-k} x^k, \quad k = 0, 1, \dots, n, \quad x \in [0, 1], \quad (2.3)$$

donde:

$$\binom{n}{k} := \begin{cases} \frac{n!}{k!(n-k)!} & , k \in \{0, 1, \dots, n\} \\ 0 & , \text{ caso contrario} \end{cases}$$

## 2.1 Características de la Base de Bernstein

Las características geométricas atractivas de las formulaciones de Bernstein nacen de ciertas características elementales de las funciones bases de Bernstein: Las propiedades de la positividad y de la partición de la unidad sobre  $[0, 1]$  son:

$$B_k^n(x) \geq 0, \quad k = 0, 1, \dots, n \quad \text{y} \quad \sum_{k=0}^n B_k^n(x) = 1. \quad (2.4)$$

**Definición 2.2.** *Dados los puntos  $c_0, c_1, \dots, c_n \in \mathbb{R}^2$  o  $\mathbb{R}^3$ . Entonces se denomina a*

$$P(x) = \sum_{k=0}^n c_k B_k^n(x), \quad P : [0, 1] \longrightarrow \mathbb{R}^2, \quad (2.5)$$

el **polinomio de Bézier** y  $C := P([0, 1])$  la *curva de Bézier*. Los coeficientes  $c_k$ ,  $k = 0, 1, \dots, n$ , son llamados *puntos de Bézier* y el polígono que pasa por la secuencia ordenada de estos puntos es llamado **polígono de Bézier**.

En un valor particular  $x_0$ , sobre  $[0, 1]$ , se puede imponer cotas más finas por la propiedad de la cápsula convexa (ver figura 2.1). La cápsula convexa de  $n+1$  puntos  $p_0, p_1, \dots, p_n$  en el plano, es la región definida por el conjunto de puntos dados por la combinación afin ponderado de los  $\{p_k\}$ ,  $p = u_0 p_0 + u_1 p_1 + \dots + u_n p_n$  de tal manera que los pesos  $\{u_k\}$  son no-negativos y suman la unidad.

Definiendo un polígono de control para el polinomio de Bézier (2.5) como la secuencia ordenada de puntos  $p_k = (x_k, y_k)$ , donde  $x_k = \frac{k}{n}$  y  $y_k = c_k$ , consideremos la curva plana:

$$r(x) := \{x, P(x)\} = \sum_{k=0}^n p_k B_k^n(x). \quad (2.6)$$

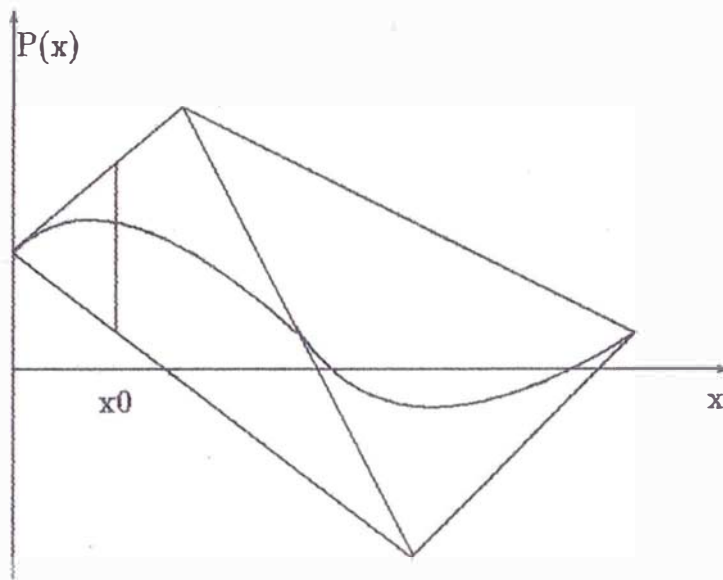


Figura 2.1: Un polinomio cúbico  $P(x)$  de Bernstein sobre  $[0, 1]$  y su polígono de control (la región es la cápsula convexa del polígono de control). La barra vertical indica las cotas locales determinado por la cápsula convexa en un valor particular  $x = x_0$ .

Por las propiedades de la positividad y partición de la unidad de la propiedad (2.4) de los  $\{B_k^n(x)\}$ , es evidente que la cápsula convexa del polígono de control contiene cada punto  $\{x, P(x)\}$  de la curva (2.6). El confinamiento de la cápsula convexa proporciona una alternativa a la propiedad de la variación decreciente para raíces aisladas y para la aproximación de raíces de polinomios de Bézier.

**Lema 2.1.** (ver Farin 1992) *Los polinomios y curvas de Bézier tienen las siguientes propiedades:*

- 1)  $P(0) = c_0$  y  $P(1) = c_n$ , es decir, la curva empieza en  $c_0$  y termina en  $c_n$ .
- 2) La curva de Bézier esta dentro de la capsula convexa

$$Co\{c_0, c_1, \dots, c_n\} := \left\{ \sum_{k=0}^n \lambda_k c_k : \lambda_k \geq 0, k = 0, 1, \dots, n, \sum_{k=0}^n \lambda_k = 1 \right\}$$

3)

$$P^{(r)}(x) = \frac{n!}{(n-r)!} \sum_{k=0}^{n-r} \Delta^r c_k B_k^{n-r}(x), \quad r = 0, 1, \dots, n$$

con las diferencias

$$\Delta^0 c_k := c_k, \quad \Delta^1 c_k = c_{k+1} - c_k, \quad \Delta^r c_k = \Delta^1(\Delta^{r-1} c_k).$$

4) El polinomio queda invariante bajo transformaciones afines del intervalo del parámetro.

El siguiente teorema sirve para evaluar al polinomio por el algoritmo de De Casteljaou.

**Teorema 2.1 (El Algoritmo de De Casteljaou).** *Considerando el polinomio  $P(x)$  de Bézier definido por (2.5). Definiendo  $P_k^0(x) := c_k$ ,  $k = 0, 1, \dots, n$  y calculando*

$$P_k^{(r)}(x) = (1-x)P_{k-1}^{(r-1)}(x) + xP_k^{(r-1)}(x), \quad (2.7)$$

con  $r = 1, 2, \dots, n$ , y  $k = r, r+1, \dots, n$ , resulta que  $P(x) = P_n^n(x)$ , es el punto con parámetro  $x$  sobre la curva de Bézier de grado  $n$ .

**Prueba :** *Veamos por inducción: Para  $r = 1$ , se tiene:*

$$P_1^{(1)} = (1-x_s)P_0^{(0)} + x_s P_1^{(0)}.$$

*Para  $r = n$ , se cumple:*

$$P_n^{(n)} = \binom{n}{0} (1-x_s)^n P_0^{(0)} + \binom{n}{1} (1-x_s)^{n-1} x_s P_1^{(0)} + \dots + \binom{n}{n} x_s^n P_n^{(0)}$$

Para  $r = n + 1$ , se tiene:

$$\begin{aligned}
P_{n+1}^{(n+1)} &= (1 - x_s)P_n^{(n)} + x_s P_{n+1}^{(n)} \\
&= (1 - x_s) \left\{ \binom{n}{0} (1 - x_s)^n P_0^{(0)} + \dots + \binom{n}{n} x_s^n P_n^{(0)} \right\} + \\
&\quad x_s \left\{ \binom{n}{0} (1 - x_s)^n P_1^{(0)} + \dots + \binom{n}{n} x_s^n P_{n+1}^{(0)} \right\}
\end{aligned}$$

$$\begin{aligned}
P_{n+1}^{(n+1)} &= \binom{n+1}{0} (1 - x_s)^{n+1} P_0^{(0)} + \binom{n+1}{1} (1 - x_s)^n x_s P_1^{(0)} + \dots + \\
&\quad \binom{n+1}{n} (1 - x_s) x_s^n P_n^{(0)} + \binom{n+1}{n+1} x_s^{n+1} P_{n+1}^{(0)}
\end{aligned}$$

Entonces se ha probado:

$$\begin{aligned}
P_n^{(n)} &= \binom{n}{0} (1 - x_s)^n P_0^{(0)} + \binom{n}{1} (1 - x_s)^{n-1} x_s P_1^{(0)} + \dots + \binom{n}{n} x_s^n P_n^{(0)} \\
&= P(x_s).
\end{aligned} \tag{2.8}$$

Una técnica frecuentemente empleado en manipulación de curvas de Bézier es la elevación de grado, que puede ser usado para introducir nuevo vértices sobre el polinomio de control de una curva sin alterar su forma. Un polinomio  $P(x)$  de grado  $n$  siempre tiene una representación no-trivial en una base de Bernstein  $\{B_k^{n+r}(x)\}$  de grado superior ( $r \geq 1$ ), i.e., los  $n + r + 1$  coeficientes  $\{c_k^{n+r}\}$  en esta base son, en general, todos no-nulos.

**Teorema 2.2 (Elevación de Grado).** *Un polinomio  $P(x)$  de grado  $n$ , siempre tiene una presentación no-trivial en una base de Bernstein  $\{B_k^{n+r}(x)\}$  de grado superior ( $r \geq 1$ ), es decir, los  $n + r + 1$  coeficientes  $c_k^{(n+r)}$  en esta base son, en general, todos no-nulos y están dadas por*

$$c_k^{n+r} = \sum_{j=\max(0, k-r)}^{\min(n, k)} \frac{\binom{r}{k-j} \binom{n}{j}}{\binom{n+r}{k}} c_j^n \tag{2.9}$$

para  $k = 0, 1, \dots, n + r$ .

**Prueba :** Consideramos primero un estado simple de elevación de grado ( $r = 1$ ).

Mostramos que los nuevos coeficientes son obtenidos de:

$$c_k^{n+1} = \frac{k}{n+1} c_{k-1}^n + \left(1 - \frac{k}{n+1}\right) c_k^n$$

donde:

$$\binom{n+1}{j} = \frac{(n+1)!}{j!(n+1-j)!} = \frac{(n+1)n!}{j!(n+1-j)(n-j)!} = \frac{n+1}{n+1-j} \binom{n}{j}.$$

Reemplazando se tiene:

$$\begin{aligned} c_k^{n+1} &= \sum_{j=0}^k \frac{\binom{k}{j}}{\frac{n+1}{n+1-j} \binom{n}{j}} c_j^n = \sum_{j=0}^k \left(1 - \frac{j}{n+1}\right) \frac{\binom{k}{j}}{\binom{n}{j}} c_j^n \\ &= c_k^n - \frac{1}{n+1} \sum_{j=0}^k \frac{k \binom{k-1}{j-1}}{\binom{n}{j}} c_j^n \text{ porque } j \binom{k}{j} = k \binom{k-1}{j-1} \\ c_k^{n+1} &= \frac{k}{n+1} c_{k-1}^n + \left(1 - \frac{k}{n+1}\right) c_k^n \end{aligned} \quad (2.10)$$

para  $k = 1, 2, \dots, n$ , y  $c_0^{n+1} = c_0^n$ ,  $c_{n+1}^{n+1} = c_n^n$ . De (2.3) sabemos que:

$$\begin{aligned} \binom{n+1}{k+1} &= \frac{(n+1)!}{(k+1)!(n-k)!} = \frac{n+1-k}{k+1} \times \frac{(n+1)!}{k!(n+1-k)!} = \frac{n+1-k}{k+1} \binom{n+1}{k} \\ &\Rightarrow \frac{k+1}{n+1} \binom{n+1}{k+1} = \frac{n+1-k}{n+1} \binom{n+1}{k} \\ &\Rightarrow \binom{n}{k} = \frac{k+1}{n+1} \binom{n+1}{k+1}. \end{aligned}$$

Luego:

$$\begin{aligned} B_k^n(x) &= \frac{k+1}{n+1} \binom{n+1}{k+1} (1-x)^{n-k} x^k [(1-x) + x] \\ &= \frac{k+1}{n+1} \binom{n+1}{k+1} (1-x)^{n+1-k} x^k + \frac{k+1}{n+1} \binom{n+1}{k+1} (1-x)^{n-k} x^{k+1}. \end{aligned}$$



De la definición de los polinomios de Bernstein:

$$\begin{aligned} B_k^n(x) &= \frac{n+1-k}{n+1} \binom{n+1}{k} (1-x)^{n+1-k} x^k + \frac{k+1}{n+1} \binom{n+1}{k+1} (1-x)^{n+1-(k+1)} x^{k+1} \\ &= \left(1 - \frac{k}{n+1}\right) B_k^{n+1}(x) + \frac{k+1}{n+1} B_{k+1}^{n+1}(x). \end{aligned} \quad (2.11)$$

Para determinar los coeficientes  $\{c_k^{n+r}\}$  de  $P(x)$  después de aplicar (2.10)  $r$  veces, primero notemos que (2.11) puede ser generalizado para expresar la función base  $\{B_k^n(x)\}$  en términos de cualquier base de grado superior  $\{B_k^{n+r}\}$  como:

$$\begin{aligned} B_k^n(x) &= \binom{n}{k} (1-x)^{n-k} x^k [(1-x) + x]^r = \binom{n}{k} (1-x)^{n-k} x^k \sum_{j=0}^r \binom{r}{j} (1-x)^{r-j} x^j \\ &= \sum_{j=0}^r \binom{n}{k} \binom{r}{j} (1-x)^{n+r-(j+k)} x^{j+k}. \end{aligned}$$

Haciendo  $J = j + k$ , ( $j = 0 \implies J = k$ , y  $j = r \implies J = k + r$ ) resulta:

$$\begin{aligned} B_k^n(x) &= \sum_{J=k}^{k+r} \binom{n}{k} \binom{r}{J-k} (1-x)^{n+r-J} x^J \\ &= \sum_{J=k}^{k+r} \frac{\binom{n}{k} \binom{r}{J-k}}{\binom{n+r}{J}} B_J^{n+r}(x), \quad k = 0, 1, \dots, n \end{aligned} \quad (2.12)$$

Los coeficientes  $\{c_k^{n+r}\}$  de elevación de grado son:

$$c_k^{n+r} = \sum_{j=\max(0, k-r)}^{\min(n, k)} \frac{\binom{r}{k-j} \binom{n}{j}}{\binom{n+r}{k}} c_j^n \quad (2.13)$$

para  $k = 0, 1, \dots, n+r$ . ■

El procedimiento generalizado de elevación de grado (2.13) puede ser requerido en la performance de operaciones aritméticas en polinomios de la forma de Bernstein.

Elevando sucesivamente el grado, el polígono de control converge hacia la curva

(ver Farin [3]).

Distinto a la forma potencial, el actual o mínimo grado de un polinomio en la forma de Bernstein no es inmediatamente aparente por inspección de sus coeficientes. Por eso requerimos de un método para determinar si un ajuste, dado de los puntos de control  $\{c_k^n\}$ , realmente representa la forma de elevación de grado de un polinomio de grado  $< n$ , y si, calculamos sus coeficientes en la base de menor grado posible. Llamemos al último proceso reducción de grado.

El criterio de reducción de grado de un polinomio  $P(x)$  con coeficientes  $\{c_k^n\}$  en el  $n$ -ésimo orden de la base de Bernstein para ser de grado actual  $n - r$  (donde  $r \geq 1$ ).

**Teorema 2.3 (Reducción de Grado).** *Dado un polinomio  $P(x)$  de grado  $n$ , con  $c_n = c_{n-1} = \dots = c_{n-r+1} = 0$ , pero  $c_{n-r} \neq 0$ , puesto que los coeficientes  $c_k^n$  presentan un  $r$ -ésimo grado de reducción de un polinomio de grado actual  $n - r$ , se tiene los coeficientes  $c_k^{n-r}$ , dado por*

$$c_k^{n-r} = \sum_{j=0}^k (-1)^{k-j} \frac{\binom{k-j+r-1}{r-1} \binom{n}{j}}{\binom{n-r}{k}} c_j^n \quad (2.14)$$

para  $k = 0, 1, \dots, n - r$ .

**Prueba:** Consideremos primero el caso  $r = 1$ , supongamos que  $c_n = 0$  para  $c_{n-1} \neq 0$ , y deseamos para derivar los coeficientes  $\{c_k^{n-1}\}$  en la base de grado  $n - 1$  de los coeficientes  $\{c_k^n\}$  dado. Revisando el procedimiento de elevación de grado (2.10) obtenemos el (consistente) sistema de  $n + 1$  ecuaciones lineales:

$$\begin{aligned} \frac{k}{n} c_{k-1}^{n-1} + \left(1 - \frac{k}{n}\right) c_k^{n-1} &= c_k^n \text{ para } k = 1, 2, \dots, n - 1, \\ \text{y } c_0^{n-1} &= c_0^n, \quad c_{n-1}^{n-1} = c_n^n, \end{aligned} \quad (2.15)$$

en los  $n$  desconocidos  $\{c_k^{n-1}\}$ . Por inducción para  $k$ , la solución para (2.15) es:

$$c_1^n = \frac{1}{n}c_0^{n-1} + \left(1 - \frac{1}{n}\right)c_1^{n-1} = \frac{1}{n}c_0^n + \left(1 - \frac{1}{n}\right)c_1^{n-1},$$

$$c_1^{n-1} = \sum_{j=0}^1 (-1)^{1-j} \frac{\binom{n}{j}}{\binom{n-1}{1}} c_j^n.$$

Por la hipótesis inductiva se cumple:

$$c_{n-2}^{n-1} = \sum_{j=0}^{n-2} (-1)^{n-2-j} \frac{\binom{n}{j}}{\binom{n-1}{n-2}} c_j^n.$$

Veamos para  $k = n - 1$ :

$$c_{n-1}^n = \frac{n-1}{n}c_{n-2}^{n-1} + \left(1 - \frac{n-1}{n}\right)c_{n-1}^{n-1}$$

$$= \frac{n-1}{n} + \sum_{j=0}^{n-2} (-1)^{n-2-j} \frac{\binom{n}{j}}{\binom{n-1}{n-2}} c_j^n \left(1 - \frac{n-1}{n}\right) c_{n-1}^{n-1}$$

$$\Rightarrow c_{n-1}^{n-1} = \sum_{j=0}^{n-1} (-1)^{n-1-j} \frac{\binom{n}{j}}{\binom{n-1}{n-1}} c_j^n$$

Entonces se ha probado:

$$c_k^{n-1} = \sum_{j=0}^k (-1)^{k-j} \frac{\binom{n}{j}}{\binom{n-1}{k}} c_j^n, \quad k = 0, 1, \dots, n-1 \quad (2.16)$$

Para el caso general, si  $c_n = c_{n-1} = \dots = c_{n-r+1} = 0$  pero  $c_{n-r} \neq 0$ , puesto que los coeficientes  $\{c_k^n\}$  representan una  $r$ -ésima elevación de grado del polinomio de grado actual  $n - r$ , por inducción en (2.16) obtenemos los coeficientes  $\{c_k^{n-r}\}$  en la base de grado  $n - r$  en términos de los coeficientes dados  $\{c_k^n\}$  como:

$$c_k^{n-r} = \sum_{j=0}^k (-1)^{k-j} \frac{\binom{k-j+r-1}{r-1} \binom{n}{j}}{\binom{n-r}{k}} c_j^n \quad (2.17)$$

para  $k = 0, 1, \dots, n - r$ . ■

Destacamos el procedimiento de reducción de grado (2.17) no puede ser aplicado a polinomios arbitrariamente, solo con aquellos que satisfacen la condición  $c_n = c_{n-1} = \dots = c_{n-r+1} = 0$ , ver Farouki y Rajan '88, (p. 9).

**Teorema 2.4 (Subdivisión).** *Se particiona el polinomio  $P(x)$  en el punto  $x_s$  dando sus coeficientes de Bernstein para los intervalos  $[0, x_s]$  y  $[x_s, 1]$  a la izquierda y derecha de  $x_s$ , entonces:*

$$P_0^{(0)}, P_1^{(1)}, \dots, P_n^{(n)} \quad \text{y} \quad P_n^{(n)}, P_n^{(n-1)}, \dots, P_n^{(0)}, \quad (2.18)$$

son los puntos de Bézier de las curvas a la izquierda y derecha respectivamente, cuando los intervalos son mapeados en  $[0, 1]$  por las transformaciones  $\frac{x}{x_s} \equiv u \in [0, 1]$  y  $\frac{x-x_s}{1-x_s} \equiv v \in [0, 1]$ .

**Prueba :** Notemos primero que las cantidades  $P_k^{(r)}$ , generadas por la recursividad (2.1), pueden ser expresadas en los terminos de los coeficientes de Bernstein  $\{c_k\}$  en la forma:

$$\begin{aligned} P_k^{(r)} &= (1-x_s)P_{k-1}^{(r-1)} + x_s P_k^{(r-1)} \\ &= (1-x_s)[(1-x_s)P_{k-2}^{(r-2)} + x_s P_{k-1}^{(r-2)}] + x_s[(1-x_s)P_{k-1}^{(r-2)} + x_s P_k^{(r-2)}] \\ &\quad \vdots \\ &= \binom{r}{0}(1-x_s)^r P_{k-r}^{(0)} + \binom{r}{1}(1-x_s)^{r-1} x_s P_{k-r+1}^{(0)} + \dots + \binom{r}{r} x_s^r P_k^{(0)} \\ &= \binom{r}{0}(1-x_s)^r c_{k-r} + \binom{r}{1}(1-x_s)^{r-1} x_s c_{k-r+1} + \dots + \binom{r}{r} x_s^r c_k \\ &= \sum_{j=0}^r \binom{r}{j} (1-x_s)^{r-j} x_s^j c_{k-r+j} \quad \text{para } k = r, r+1, \dots, n \\ &= \sum_{j=0}^r B_j^r(x_s) c_{k-r+j} \quad \text{para } k = r, r+1, \dots, n \end{aligned} \quad (2.19)$$

lo cual se verifica fácilmente por inducción. Los coeficientes de (2.7) pueden entonces ser escritos como:

$$P_k^k = \sum_{j=0}^k B_j^k(x_s) c_j \quad \text{y} \quad P_n^{(n-k)} = \sum_{j=k}^n B_{j-k}^{n-k}(x_s) c_j, \quad (2.20)$$

con  $k = 0, 1, \dots, n$ .

Luego, si sustituimos  $x = x_s u$ ,  $1 - x = (1 - u) + (1 - x_s)u$  y  $x = x_s(1 - v) + v$ ,  $1 - x = (1 - x_s)(1 - v)$  en la definición de las funciones de la base de Bernstein  $B_j^n(x)$  en términos de las bases  $\{B_k^n(u)\}$  y  $\{B_k^n(v)\}$  sobre  $u \in [0, 1]$  y  $v \in [0, 1]$ , respectivamente, resulta:

$$\begin{aligned} B_j^n(x) &= \binom{n}{j} (1 - x_s)^{n-j} x_s^j = \binom{n}{j} [(1 - u) + (1 - x_s)u]^{n-j} [x_s u]^j \\ &= \binom{n}{j} \sum_{k=0}^{n-j} \binom{n-j}{n-j-k} (1 - u)^{n-(j+k)} u^{k+j} (1 - x_s)^k x_s^j \\ &= \binom{n}{j} \sum_{K=j}^n \binom{n-j}{n-K} (1 - u)^{n-K} u^K (1 - x_s)^{K-j} x_s^j, \quad (\text{haciendo } K = k + j) \\ &= \sum_{K=j}^n \left\{ \binom{n}{K} (1 - u)^{n-K} u^K \right\} \left\{ \binom{K}{j} (1 - x_s)^{K-j} x_s^j \right\} \\ &= \sum_{K=j}^n B_K^n(u) B_j^K(x_s) \end{aligned} \quad (2.21)$$

En forma análoga se realiza los cambios  $x = x_s(1 - v) + v$  y  $1 - x = (1 - x_s)(1 - v)$  se obtiene:

$$B_j^n(x) = \sum_{K=0}^j B_{n-K}^{n-K}(x_s) B_K^n(v) \quad (2.22)$$

Si ahora introducimos las transformaciones (2.21) y (2.22) entre las bases de Bernstein sobre  $x \in [0, 1]$  y aquellos sobre  $u \in [0, 1]$  y  $v \in [0, 1]$  en la forma

original de  $P(x)$  obtenemos:

$$\begin{aligned}
 P(x) &= \sum_{j=0}^n c_j B_j^n(x) = \sum_{j=0}^n c_j \sum_{k=j}^n B_j^k(x_s) B_k^n(u) \\
 &= \left\{ \sum_{j=0}^0 c_j B_j^0(x_s) \right\} B_0^n(u) + \left\{ \sum_{j=0}^1 c_j B_j^1(x_s) \right\} B_1^n(u) + \cdots + \left\{ \sum_{j=0}^n c_j B_j^n(x_s) \right\} B_n^n(u) \\
 &= \sum_{k=0}^n \left\{ \sum_{j=0}^k c_j B_j^k(x_s) \right\} B_k^n(u) \tag{2.23}
 \end{aligned}$$

En forma análoga se tiene:

$$P(x) = \sum_{k=0}^n \left\{ \sum_{j=k}^n c_j B_{n-j}^{n-k}(x_s) \right\} B_k^n(v) \tag{2.24}$$

Los coeficientes de  $P(x)$  en las bases  $\{B_k^n(u)\}$  y  $\{B_k^n(v)\}$  en los intervalos a la derecha y a la izquierda de  $x_s$  son simplemente las cantidades de (2.23) y (2.24). ■

Note que cada división del polinomio duplica aproximadamente el número de coeficientes de Bernstein distintos requeridos, para representarlo sobre el intervalo original. Con iteración de partición uniforme, los polinomios de control en cada sub-intervalo, convergen al polinomio sobre el intervalo (ver Forauki y Rajan '87, p. 197).

## 2.2 Superficies

Se puede utilizar dos conjuntos de curvas de Bézier para representar superficies de objetos especificados por puntos de control de entrada. La función vectorial paramétrica de la superficie de Bézier se forma como el producto cartesiano de las funciones de combinación de Bézier:

$$P(u, v) = \sum_{j=0}^m \sum_{k=0}^n p_{jk} B_j^m(u) B_k^n(v), \tag{2.25}$$

con  $p_{jk} \in \mathbb{R}^3$ , que especifica las localidades de los  $(m + 1)$  por  $(n + 1)$  puntos de control. Las superficies de Bézier tienen las mismas propiedades que las curvas de Bézier y ofrecen un método adecuado para aplicaciones de diseño interactivas.

### 2.2.1 Algoritmo de De Casteljaou para Superficies

La superficie de Bézier de producto tensorial puede ser obtenida aplicando el algoritmo de De Casteljaou para superficies. Dado los puntos  $p_{jk}$  con  $0 \leq j, k \leq n$  y valores de parámetro  $u, v$  ( $0 \leq u, v \leq 1$ ) el siguiente algoritmo genera un punto sobre la superficie determinado por la red de control  $\{B_{jk}\}$ , considerando para el caso  $m = n$ .

- a) Dados  $\{B_{jk}\}_{j,k}^n$  y  $(u, v) \in [0, 1] \times [0, 1]$ .
- b) Determinar:

$$B_{jk}^{r,r} = [1 - u \ u] \begin{bmatrix} B_{j,k}^{r-1,r-1} & B_{j,k+1}^{r-1,r-1} \\ B_{j+1,k}^{r-1,r-1} & B_{j+1,k+1}^{r-1,r-1} \end{bmatrix} \begin{bmatrix} 1 - v \\ v \end{bmatrix} \quad (2.26)$$

para  $r = 1, 2, \dots, n$  y  $j, k = 0, 1, \dots, n - r$ . Empezando con  $B_{jk}^{0,0} = B_{jk}$ .

- c) El punto  $B_{jk}^{n,n}$  es el punto con valores de parámetro  $(u, v)$  sobre la superficie de Bézier  $P(u, v)$ .

Los algoritmos de subdivisión y elevación de grado se transforman fácilmente en caso de superficies.

## 2.3 Tipos de Evaluación

Los tipos de evaluaciones para los polinomios de Bézier son:

- a) Evaluar  $P(t) = \sum_{k=0}^n c_k B_k^n(t)$  calculando los polinomios de Bernstein,

- b) evaluación por el algoritmo de Horner aplicado a la base de Bernstein,
- c) el algoritmo de De Casteljaou,
- d) transformar la base de Bernstein a la base monomial, y evaluar esta,
- e) por el algoritmo de elevación de grado (el polígono de control converge hacia la curva),
- f) aplicando el algoritmo de reducción de grado, y evaluar el nuevo polinomio,
- g) por el algoritmo de subdivisión, obteniendo así una convergencia rápida del polígono del control a la curva.

## 2.4 Operaciones Aritméticas

Dados dos polinomios de Bézier  $f(x)$  y  $g(x)$  de grado  $m$  y  $n$  con puntos de control  $\{c_k\}, \{d_k\} \subset \mathbb{R}^d$  sobre  $[0,1]$ , donde asumimos  $m \geq n$  sin pérdida de generalidad. Determinaremos los puntos de control por la suma, diferencia, y producto de estos polinomios, y el cociente del polinomio restantes cuando  $f(x)$  es dividido por  $g(x)$ . También comparamos la estructura de la curva de Bézier con estos algoritmos, cuando  $f(x)$  y  $g(x)$  son especificados por sus coeficientes de potencia conocidos,  $\{a_k\}, \{b_k\} \in \mathbb{R}$ , ver en Farouki y Rajan '88 (p. 13).

### Adición y Subtracción

La suma o diferencia  $f(x) \pm g(x)$  es un polinomio de grado  $m$  a lo más, cuyos puntos de control  $\{c'_k\}$  pueden ser obtenidos de la siguiente manera. Si  $n = m$ , nosotros simplemente tomamos la suma o diferencia  $c'_k = c_k \pm d_k, k = 0, 1, \dots, m$ , de sus correspondientes coeficientes de  $f(x)$  y  $g(x)$ . Sin embargo, si  $n < m$ , debemos



de elevar el grado de  $g(x)$  a un total de  $m - n$  veces de (2.14), dando:

$$c'_k = c_k \pm \sum_{j=\max\{0, k-m+n\}}^{\min(n, k)} \frac{\binom{m-n}{k-j} \binom{n}{j}}{\binom{m}{k}} d_j, \text{ para } k = 0, 1, \dots, m. \quad (2.27)$$

Por comparación, cuando  $f(x)$  y  $g(x)$  estan dados en forma potencia, los coeficientes de potencia  $\{a'_k\}$  de los polinomio  $f(x) \pm g(x)$  son simplemente

$$a'_k = a_k \pm b_k \text{ para } k = 0, 1, \dots, n \quad (2.28)$$

y  $a'_k \equiv a_k$  para  $n + 1 \leq k \leq m$ .

## Multiplicación

El producto  $f(x)g(x)$  puede ser expresado como:

$$f(x)g(x) = \sum_{i=0}^m \sum_{j=0}^n c_i d_j \binom{m}{i} \binom{n}{j} (1-x)^{m+n-i-j} x^{i+j}, \quad (2.29)$$

que es un polinomio de grado  $m + n$ . Para cambiar (2.29) en labase de Bernstein  $\{B_k^{m+n}(x)\}$  de grado  $m + n$ , lo reescribimos como una suma simple sobre su indice  $k = i + j$ . Esto da entonces los puntos de control  $\{c'_k\}$  de grado  $m + n$ , de  $f(x)g(x)$  como la suma ponderada de productos de los coeficientes  $\{c_k\}$  y  $\{d_k\}$ . Desarrollando tenemos:

$$\begin{aligned} f(x)g(x) &= \sum_{j=0}^n c_0 d_j \binom{n}{j} (1-x)^{m+n-j} x^j + \sum_{j=0}^n c_1 d_j \binom{m}{1} \binom{n}{j} (1-x)^{m+n-1-j} x^{1+j} \\ &+ \dots + \sum_{j=0}^n c_m d_j \binom{m}{m} \binom{n}{j} (1-x)^{m+n-m-j} x^{m+j} \end{aligned}$$

$$\begin{aligned}
f(x)g(x) &= \binom{m}{0} \binom{n}{0} c_0 d_0 (1-x)^{m+n} + \sum_{j=0}^1 \binom{m}{j} \binom{n}{1-j} c_j d_{1-j} (1-x)^{m+n-1} x^1 \\
&\quad + \cdots + \sum_{j=m}^m \binom{m}{j} \binom{n}{n-j+m} c_j d_{m+n-j} (1-x)^0 x^{m+n} \\
&= \sum_{k=0}^{m+n} \sum_{j=\max(0, k-n)}^{\min(m, k)} \binom{m}{j} \binom{n}{k-j} c_j d_{k-j} (1-x)^{m+n-k} x^k \\
&= \sum_{k=0}^{m+n} \sum_{j=\max(0, k-n)}^{\min(m, k)} \frac{\binom{m}{j} \binom{n}{k-j}}{\binom{m+n}{k}} c_j d_{k-j} \binom{m+n}{k} (1-x)^{m+n-k} x^k \\
\Rightarrow c'_k &= \sum_{j=\max(0, k-n)}^{\min(m, k)} \frac{\binom{m}{j} \binom{n}{k-j}}{\binom{m+n}{k}} c_j d_{k-j} \text{ para } k = 0, 1, \dots, m+n. \tag{2.30}
\end{aligned}$$

La expresión correspondiente a los coeficientes de potencia es:

$$a'_k = \sum_{j=\max(0, k-n)}^{\min(m, k)} a_j b_{k-j} \text{ para } k = 0, 1, \dots, m+n. \tag{2.31}$$

## División

Sean los polinomios  $f(x)$  y  $g(x)$  dados en la forma de Bernstein, donde  $f(x)$  es dividido por  $g(x)$ . Consideremos ahora la determinación de puntos de control para los polinomios cociente y residuo,  $q(x)$  y  $r(x)$  de grados  $m-n$  y  $n-1$ , respectivamente, definidos por:

$$f(x) = q(x)g(x) + r(x), \tag{2.32}$$

los polinomios  $q(x)$  y  $r(x)$  están definidos por  $m-n+1$  y  $n$  puntos de control,  $\{q_k\}$  y  $\{r_k\}$ , respectivamente. Esto es,  $m+1$  coeficientes desconocidos deben ser determinados conjuntamente.

Si aplicamos el procedimiento de multiplicación al producto de los términos  $q(x)g(x)$

en el miembro derecho de (2.32) y elevando el grado del término restante  $r(x)$  un total  $m-n+1$  veces. Expresando ambos miembros de (2.32) en la base de Bernstein de grado  $m$  e igualando los coeficientes de cada base de función  $B_k^m(x)$  en ambos lados de (2.32) se obtiene un sistema lineal de  $m+1$  ecuaciones:

$$c'_k = \sum_{j=\max(0,k-n)}^{\min(m-n,k)} \frac{\binom{m-n}{j} \binom{n}{k-j}}{\binom{m}{k}} q_j d_{k-j} + \sum_{j=\max(0,k-m+n-1)}^{\min(n-1,k)} \frac{\binom{m-n+1}{k-j} \binom{n-1}{j}}{\binom{m}{k}} r_j, \quad (2.33)$$

$k = 0, 1, \dots, m$ , en los  $m+1$  incógnitas  $\{q_k\}$  y  $\{r_k\}$ .

Los coeficientes de potencia  $c'(x)$  y  $d(x)$  deben ser obtenidos resolviendo el sistema lineal (2.33), a diferencia de la adición (2.27) y multiplicación (2.30), no es posible escribir expresiones compactas cerradas para estos coeficientes.

Para el caso especial  $n = m$ , el cociente  $q(x)$  es justamente una constante  $q$  y la ecuación (2.33) se reduce a la forma simple:

$$\begin{aligned} c'_0 &= qd_0 + r_0 \\ &\vdots \\ &= qd_j + \frac{j}{m}r_{j-1} + \left(1 - \frac{j}{m}\right)r_j, \quad j = 1, 2, \dots, m-1 \\ c'_m &= qd_m + r_{m-1} \end{aligned} \quad (2.34)$$

Multiplicando la  $j$ -ésima ecuación líneas arriba por  $(-1)^{m-j} \binom{m}{j}$ , adicionandolas a todas juntas, y usando las transformaciones de bases que se indicará más adelante se tiene  $q = \frac{a_m}{b_m}$  (i.e., la razón de los coeficientes de potencia de  $f(x)$  y  $g(x)$ ) por el valor de los cocientes, y substituyendo este valor en la ecuación (2.34) permiten a los coeficientes  $r_0, r_1, \dots, r_{m-1}$  del residuo a ser calculados directamente.

La división de (2.32) por polinomios  $f(x)$  y  $g(x)$  en forma de potencia se procede en la siguiente forma. Los coeficientes  $q_0, q_1, \dots, q_{m-n}$  y  $r_0, r_1, \dots, r_{n-1}$  del cociente  $q(x)$  y residuo  $r(x)$  en forma potencia son las soluciones del sistema de  $m+1$  ecuaciones

lineales:

$$a_k = \sum_{j=\max(0,k-n)}^{\min(m-n,k)} q_j b_{k-j} + r_k \quad k = 0, 1, \dots, m, \quad (2.35)$$

en las incognitas  $\{q_k\}$  y  $\{r_k\}$  (nosotros hacemos  $r_k \equiv 0$  en (2.35) para  $k \leq n$ ). Para el caso especial  $m = n$ , tenemos  $q(x) = q = \frac{am}{b_m}$  y los coeficientes de residuo pueden ser entonces determinados directamente en la forma  $r_k = a_k - qb_k$  para  $k = 0, 1, \dots, m - 1$ .

## 2.5 Uso de los Coeficientes de Bernstein Escalados

Los algoritmos de (2.27), (2.30), y (2.33) para la adición, multiplicación, y división de los polinomios en forma de Bernstein se asemejan a sus analogos (2.28), (2.31), y (2.35) de la acostumbrada forma de potencia. Esto es mucho mas aparente si usamos los puntos de control escalados.

$$\tilde{c}_k = \binom{n}{k} c_k, \quad k = 0, 1, \dots, n. \quad (2.36)$$

Con coeficientes escalados, el procedimiento para la adición y multiplicación es:

$$\begin{aligned} c'_k &= c_k \pm \sum_{j=\max(0,k-m+n)}^{\min(n,k)} \frac{\binom{m-n}{k-j} \binom{n}{j}}{\binom{m}{k}} d_j, \\ \binom{m}{k} c'_k &= \binom{m}{k} c_k \pm \sum_{j=\max(0,k-m+n)}^{\min(n,k)} \binom{m-n}{k-j} \binom{n}{j} d_j, \\ \tilde{c}'_k &= \tilde{c}_k \pm \sum_{j=\max(0,k-m+n)}^{\min(n,k)} \binom{m-n}{k-j} \tilde{d}_k, \quad k = 0, 1, \dots, m, \\ c_k^{m+n} &= \sum_{j=\max(0,k-n)}^{\min(m,k)} \frac{\binom{m}{j} \binom{n}{k-j}}{\binom{m+n}{k}} c_j d_{k-j}, \end{aligned}$$

$$\binom{m+n}{k} c'_k = \sum_{j=\max(0,k-n)}^{\min(m,k)} \binom{m}{j} c_j \binom{n}{k-j} d_{k-j},$$

$$\tilde{c}'_k = \sum_{j=\max(0,k-n)}^{\min(m,k)} \tilde{c}_j \tilde{d}_{k-j}, \quad k = 0, 1, \dots, m+n, \quad (2.37)$$

mientras que para división, el sistema de  $m+1$  ecuaciones lineales:

$$\begin{aligned} c'_k &= \sum_{j=\max(0,k-n)}^{\min(m-n,k)} \frac{\binom{m-n}{j} \binom{n}{k-j}}{\binom{m}{k}} c_j d_{k-j} + \sum_{j=\max(0,k-m+n-1)}^{\min(n-1,k)} \frac{\binom{m-n+1}{k-j} \binom{n-1}{j}}{\binom{m}{k}} r_j \\ &= \sum_{j=\max(0,k-n)}^{\min(m-n,k)} \tilde{c}_j \tilde{d}_{k-j} + \sum_{j=\max(0,k-m+n-1)}^{\min(n-1,k)} \binom{m-n+1}{k-j} \tilde{r}_j \end{aligned} \quad (2.38)$$

$k = 0, 1, \dots, m$ , debe ser resuelta para los coeficientes escalados de  $q(x)$  y  $r(x)$ . Una comparación de (2.27), (2.37), (2.38) con (2.28), (2.31), y (2.35) revela que las operaciones aritméticas en la forma de Bernstein son ahora casi idénticas a la formulación de potencias, la principal diferencia sobresale de la necesidad de la elevación de grado en la adición de dos polinomios de diferentes grados, y en la determinación del residuo cuando se dividen dos polinomios.

Otros procedimientos son también simplificados por el uso de los puntos de control escalados, incluyendo la fórmula de elevación y reducción de grado, entre las potencias y las formas de Bernstein. Como los coeficientes escalados reducen los cálculos requeridos en las operaciones aritméticas polinómicas, recomendamos su uso en algoritmos en los cuales demandan muchas operaciones de ese tipo. Sin embargo, muchas propiedades del polinomio de Bézier, tales como la propiedad de la cápsula convexa, no se aplican a coeficientes escalados. Cuando es preferible trabajar con coeficientes originales  $\{c'_k\}$ , recomendamos el uso de los coeficientes binomiales factorados, esto ayuda a minimizar explícitamente los costos adicionales y el error incurrido por el cálculo racional de combinaciones de los coeficientes binomiales.

### 3 ERRORES EN LA EVALUACIÓN DE UN POLINOMIO POR EL ALGORITMO DE DE CASTELJAU, ELEVACION DE GRADO Y SUBDIVISIÓN

En este capítulo se muestra el funcionamiento del algoritmo de De Casteljaou, y sus respectivos errores que se generan debido a los cálculos que se realizan por las computadoras.

Las características geométricas atractivas de las curvas de Bézier, nacen de ciertas características elementales de las funciones básicas de Bernstein.

Con el Teorema 2.1 se evalúa un polinomio de Bernstein de la forma  $P(x) = \sum_{k=1}^n c_k \binom{n}{k} (1-x)^{n-k} x^k$  por una secuencia finita de interpolaciones lineales entre los puntos de control. Dados  $c_0, c_1, \dots, c_n \in \mathbb{R}^d$ ,  $d = 1, 2, 3$  y  $x_s \in [0, 1]$ . Sea:

$$\begin{aligned} P_k^{(0)} &= c_k \text{ para } k = 0, 1, \dots, n \\ P_k^{(r)} &= (1 - x_s)P_{k-1}^{(r-1)} + x_s P_k^{(r-1)} \text{ para } k = r, r+1, \dots, n, \\ & r = 1, 2, \dots, n \end{aligned} \quad (3.1)$$

Este algoritmo incurre en  $n(n+1)$  multiplicaciones y en  $\frac{1}{2}n(n+1)$  adiciones, como se muestra en (2.8) produce un único  $P(x_s) - P_n^{(n)}$ .

Otra vez hacemos un análisis del error como en la sección (1.2). Consideremos otra vez  $|\pi_r| = |\sigma_r| = \eta$ , para  $r = 1, 2, \dots, n$ , con el cual se obtendrá el error máximo. Además se considera  $\eta^2 \rightarrow 0$ , porque es prácticamente despreciable para los cálculos, donde  $\widehat{e}_k^{(r)} := fl[P_k^{(r)}] - P_k^{(r)}$ , también se considera que  $\widehat{e}_{k-1}^{(r-1)}\eta = 0$  porque  $\widehat{e}_{k-1}^{(r-1)}\eta = \{fl[P_{k-1}^{(r-1)}] - P_{k-1}^{(r-1)}\}\eta = P_{k-1}^{(r-1)}\eta^2$ .

Los errores pueden ser escritos en la forma siguiente:

$$\begin{aligned}
\widehat{e}_k^{(r)} &= fl[P_k^{(r)}] - P_k^{(r)} \\
&= fl[(1 - x_s)P_{k-1}^{(r-1)} + x_s P_k^{(r-1)}] - P_k^{(r)} \\
&= \{(1 - x_s)[\widehat{e}_{k-1}^{(r-1)} + P_{k-1}^{(r-1)}](1 + \eta) + x_s[\widehat{e}_k^{(r-1)} + P_k^{(r-1)}](1 + \eta)\}(1 + \eta) - P_k^{(r)} \\
&= \{(1 - x_s)\widehat{e}_{k-1}^{(r-1)} + (1 - x_s)P_{k-1}^{(r-1)} + (1 - x_s)\widehat{e}_{k-1}^{(r-1)}\eta + (1 - x_s)P_{k-1}^{(r-1)}\eta + \\
&\quad x_s\widehat{e}_k^{(r-1)} + x_s P_k^{(r-1)} + x_s\widehat{e}_k^{(r-1)}\eta + x_s P_k^{(r-1)}\eta\}(1 + \eta) - P_k^{(r)} \\
&= (1 - x_s)\widehat{e}_{k-1}^{(r-1)} + P_k^{(r)} + (1 - x_s)P_{k-1}^{(r-1)}\eta + x_s\widehat{e}_k^{(r-1)} + x_s P_k^{(r-1)}\eta \\
&\quad (1 - x_s)\widehat{e}_{k-1}^{(r-1)}\eta + P_k^{(r)}\eta + (1 - x_s)P_{k-1}^{(r-1)}\eta^2 + x_s\widehat{e}_k^{(r-1)}\eta + x_s P_k^{(r-1)}\eta^2 - P_k^{(r)} \\
&= (1 - x_s)\widehat{e}_{k-1}^{(r-1)} + (1 - x_s)P_{k-1}^{(r-1)}\eta + x_s\widehat{e}_k^{(r-1)} + x_s P_k^{(r-1)}\eta + P_k^{(r)}\eta
\end{aligned}$$

Para obtener el error máximo, en las cantidades calculadas  $P_k^{(r)}$ , se considera el error absoluto  $e_k^{(r)} = |\widehat{e}_k^{(r)}|$  a cada término, resultando:

$$e_k^{(r)} \leq |1 - x_s|e_{k-1}^{(r-1)} + |x_s|e_k^{(r-1)} + \{|1 - x_s||P_{k-1}^{(r-1)}| + |x_s||P_k^{(r-1)}| + |P_k^{(r)}|\}\eta \quad (3.2)$$

para  $k = r, r + 1, \dots, n$ . La relación (3.2) nos permite estimar las cotas superiores en la magnitud de los errores de redondeo  $|\delta P|$  en el valor calculado  $P(x_s)$  de los polinomios para  $x_s \in [0, 1]$ . Inicializamos con  $e_k^{(0)} = 0$  para  $k = 0, 1, 2, \dots, n$ , y notando que la combinación convexa dado por (3.1) produce cantidades intermedias los cuales estan acotadas por el tamaño de los puntos de control iniciales más grandes, ver en Farouki y Rajan '87, (p. 196).

A partir de la inecuación (3.2), se obtienen:

Para  $n = 1$ , se tiene:

$$\begin{aligned}
e_k^{(1)} &\leq (1 - x_s)e_{k-1}^{(0)} + x_s e_k^{(0)} + \{(1 - x_s)P_{k-1}^{(0)} + x_s P_k^{(0)} + P_k^{(1)}\}\eta \\
&\leq 2\{(1 - x_s)P_{k-1}^{(0)} + x_s P_k^{(0)}\}\eta.
\end{aligned}$$

Para  $n = 2$ , se tiene:

$$\begin{aligned}
e_k^{(2)} &\leq (1 - x_s)e_{k-1}^{(1)} + x_s e_k^{(1)} + \{(1 - x_s)P_{k-1}^{(1)} + x_s P_k^{(1)} + P_k^{(2)}\}\eta \\
&\leq 2\{(1 - x_s)^2 P_{k-2}^{(0)} + x_s(1 - x_s)P_{k-1}^{(0)}\}\eta + 2\{x_s(1 - x_s)P_{k-1}^{(0)} + x_s P_k^{(0)}\}\eta + \\
&\quad [(1 - x_s)\{(1 - x_s)P_{k-2}^{(0)} + x_s P_{k-1}^{(0)}\} + x_s\{(1 - x_s)P_{k-1}^{(0)} + x_s P_k^{(0)}\} + \\
&\quad \{(1 - x_s)P_{k-1}^{(1)} + x_s P_k^{(1)}\}]\eta \\
&\leq 2^2\{(1 - x_s)^2 P_{k-2}^{(0)} + 2x_s(1 - x_s)P_{k-1}^{(0)} + x_s^2 P_k^{(0)}\}\eta.
\end{aligned}$$

Para  $n = r$ , se cumple:

$$e_k^{(r)} \leq 2 \times r\{(1 - x_s)^r P_{k-r}^{(0)} + \binom{r}{1}(1 - x_s)^{r-1} x_s P_{k-r+1}^{(0)} + \cdots + \binom{r}{r} x_s^r P_k^{(0)}\}\eta$$

Para  $n = r + 1$ , se tiene:

$$\begin{aligned}
e_k^{(r+1)} &\leq (1 - x_s)e_{k-1}^{(r)} + x_s e_k^{(r)} + \{(1 - x_s)P_{k-1}^{(r)} + x_s P_k^{(r)} + P_k^{(r+1)}\}\eta \\
&\leq (1 - x_s)2 \times r\{(1 - x_s)^r P_{k-r-1}^{(0)} + \cdots + \binom{r}{r} x_s^r P_{k-1}^{(0)}\}\eta + \\
&\quad x_s 2 \times r\{(1 - x_s)^r P_{k-r}^{(0)} + \binom{r}{1}(1 - x_s)^{r-1} x_s P_{k-r+1}^{(0)} + \cdots + \binom{r}{r} x_s^r P_k^{(0)}\}\eta + \\
&\quad [(1 - x_s)\{(1 - x_s)^r P_{k-r-1}^{(0)} + \cdots + \binom{r}{r} x_s^r P_{k-1}^{(0)}\} + \\
&\quad x_s\{(1 - x_s)^r P_{k-r}^{(0)} + \binom{r}{1}(1 - x_s)^{r-1} x_s P_{k-r+1}^{(0)} + \cdots + \binom{r}{r} x_s^r P_k^{(0)}\} + \\
&\quad \{(1 - x_s)^{r+1} P_{k-r-1}^{(0)} + \cdots + \binom{r+1}{r+1} x_s^{r+1} P_k^{(0)}\}]\eta \\
&\leq 2 \times (r + 1)\{(1 - x_s)^{r+1} P_{k-r-1}^{(0)} + \cdots + \binom{r+1}{r+1} x_s^{r+1} P_k^{(0)}\}\eta.
\end{aligned}$$

Por consiguiente se ha obtenido:

$$e_k^{(r)} \leq 2 \times r\{(1 - x_s)^r P_{k-r}^{(0)} + \binom{r}{1}(1 - x_s)^{r-1} x_s P_{k-r+1}^{(0)} + \cdots + \binom{r}{r} x_s^r P_k^{(0)}\}\eta, \quad (3.3)$$



Definimos:

$$C_M = \max\{2 \times n(1 - x_s)^n c_{k-n}, n \binom{n}{1} (1 - x_s)^{n-1} x_s c_{k-n-1}, \dots, 2 \times n x_s^n c_k\}$$

La cota de error crece linealmente con el grado  $n$  del polinomio, a pesar que el número de operaciones aritméticas crece cuadráticamente con  $n$  para el Algoritmo de De Casteljaou, de Farouki y Rajan [4] se tiene, entonces:

$$|\delta P| \leq 2n C_M \eta \quad (3.4)$$

Podemos definir la condición para la evaluación polinomial en la base de Bernstein de (1.9) se cumple:

$$S_b(x_s) = \sum_{k=0}^n |c_k \binom{n}{k} (1 - x_s)^{n-k} x_s^k| \quad y \quad C_b(x_s) = |P'(x_s)|^{-1} S_b(x_s). \quad (3.5)$$

La suma  $S_b(x_s)\epsilon$  es el número de condición para la evaluación de los polinomios  $P(x)$  en  $x_s$  en la base de Bernstein, ver Farouki y Rajan '87, (p. 196).

**Ejemplo 3.1.** Usando el Ejemplo (1.2), determinamos los números de condición de las raíces en la base de Bernstein, ver Farouki y Rajan '87.

k	$C_b(x_s)$	k	$C_b(x_s)$
1	3.413x10 <sup>0</sup>	11	3.321x10 <sup>6</sup>
2	1.453x10 <sup>2</sup>	12	2.215x10 <sup>6</sup>
3	2.335x10 <sup>3</sup>	13	1.115x10 <sup>6</sup>
4	2.030x10 <sup>4</sup>	14	4.153x10 <sup>5</sup>
5	1.111x10 <sup>5</sup>	15	1.111x10 <sup>5</sup>
6	4.153x10 <sup>5</sup>	16	2.030x10 <sup>4</sup>
7	1.115x10 <sup>6</sup>	17	2.335x10 <sup>3</sup>
8	2.215x10 <sup>6</sup>	18	1.453x10 <sup>2</sup>
9	3.321x10 <sup>6</sup>	19	3.413x10 <sup>0</sup>
10	3.797x10 <sup>9</sup>	20	0.000

Los resultados han sido calculados con 4 dígitos significativos, y las evaluaciones han sido realizados en una vecindad de  $x_s = \frac{1}{2}$ . Comparando con el ejemplo 1.3 se observa que los números de condición de la base de Bernstein son notablemente más pequeños.  $\square$

Por consiguiente es recomendable trabajar con las bases de Bernstein en lugar de la base monomial.

### 3.1 Determinación del Error del Polinomio

En este caso también se debe considerar en primer lugar si los coeficientes del polinomio son o no son números máquina. Entonces se presentan dos casos:

**Primer Caso :** Sean  $c_k$ ,  $k = 0, 1, \dots, n$  y  $x_s$  números de máquina. Para determinar el polinomio de grado  $n$  con su respectivo error, se usará la fórmula recurrente de DE CASTELJAU. Denotamos con  $\pi_i$  y  $\sigma_i$  a los errores de la multiplicación y adición, respectivamente, obtenemos:

$$\begin{aligned}
 P_k^{(1)} &= (1 - x_s)P_{k-1}^{(0)} + x_s P_k^{(0)}, \quad k = 1, 2, \dots, n \\
 fl[P_k^{(1)}] &= fl[(1 - x_s)P_{k-1}^{(0)} + x_s P_k^{(0)}] \\
 &= \{(1 - x_s)c_{k-1}(1 + \pi_1) + x_s c_k(1 + \pi_2)\}(1 + \sigma_1) \\
 &= [(1 - x_s)c_{k-1} + x_s c_k] + [(1 - x_s)c_{k-1}\{\pi_1 + \sigma_1\} + x_s c_k\{\pi_2 + \sigma_1\}] \\
 \\
 P_k^{(2)} &= (1 - x_s)P_{k-1}^{(1)} + x_s P_k^{(1)} \\
 &= (1 - x_s)[(1 - x_s)P_{k-2}^{(0)} + x_s P_{k-1}^{(0)}] + x_s [(1 - x_s)P_{k-1}^{(0)} + x_s P_k^{(0)}]
 \end{aligned}$$

$$\begin{aligned}
fl[P_k^{(2)}] &= fl[(1-x_s)[(1-x_s)c_{k-2} + x_sc_{k-1}] + x_s[(1-x_s)c_{k-1} + x_sc_k] \\
&= \{(1-x_s)[(1-x_s)c_{k-2}(1+\pi_3) + x_sc_{k-1}(1+\pi_4)](1+\sigma_2)(1+\pi_1) + \\
&= x_s[(1-x_s)c_{k-1}(1+\pi_5) + x_sc_k(1+\pi_6)](1+\sigma_3)(1+\pi_2)\}(1+\sigma_1)
\end{aligned}$$

$$\begin{aligned}
fl[P_k^{(2)}] &= [(1-x_s)^2c_{k-2} + 2x_s(1-x_s)c_{k-1} + x_s^2c_k] + \\
&= \{(1-x_s)^2c_{k-2}(\sigma_2 + \sigma_1 + \pi_3 + \pi_1) + \\
&= x_s(1-x_s)c_{k-1}(\sigma_3 + \sigma_2 + 2\sigma_1 + \pi_5 + \pi_4 + \pi_2 + \pi_1) + \\
&= x_s^2c_k(\sigma_3 + \sigma_1 + \pi_6 + \pi_2)\}
\end{aligned}$$

Según se observa no se puede obtener una fórmula recursiva debido a la complejidad del resultado.

**Segundo Caso :** Si  $c_k$ ,  $k = 0, 1, \dots, n$  y  $x_s$  no son números de máquina. Denotamos con  $\delta_i$ ,  $\pi_i$  y  $\sigma_i$  los errores de los números, multiplicación y adición, respectivamente, obtenemos:

$$\begin{aligned}
P_k^{(1)} &= (1-x_s)P_{k-1}^{(0)} + x_sP_k^{(0)}, \quad k = 1, 2, \dots, n \\
fl[P_k^{(1)}] &= fl[(1-x_s)P_{k-1}^{(0)} + x_sP_k^{(0)}] \\
&= \{(1-x_s)(1+\delta_x)c_{k-1}(1+\delta_{k-1})(1+\pi_1) + \\
&= x_s(1+\delta_x)c_k(1+\delta_k)(1+\pi_2)\}(1+\sigma_1) \\
&= [(1-x_s)c_{k-1} + x_sc_k] + [(1-x_s)c_{k-1}\{\delta_x + \delta_{k-1} + \pi_1 + \sigma_1\} + \\
&= x_sc_k\{\delta_x + \delta_k + \pi_2 + \sigma_1\}]
\end{aligned}$$

En este caso solo se ha desarrollado para  $r = 1$  debido a la complejidad de los cálculos, esperando que la idea intuitiva ha sido captado de la manera como se determina la representación en punto flotante del polinomio de mayor grado.

## 3.2 Condición de Bernstein en Elevación de Grado y Subdivisión

Mostraremos más adelante que el número de condición de una raíz real simple  $x_0 \in (0, 1)$ , de un polinomio  $P(x)$  de grado  $n$  es siempre menor en la elevación de grado en la base de Bernstein  $\{B_k^{n+1}(x)\}$  que en la base original  $\{B_k^n(x)\}$ . Sin embargo, como en el caso de la conversión de potencia-a-Bernstein, ningún mejoramiento de la condición de la raíz ha resultado de la elevación de grado. Supongamos que cada coeficiente nominal  $\{c_k^n\}$  sufre un error aleatorio de magnitud relativa máxima  $\varepsilon$ . De la ecuación (2.10), vemos que:

$$\begin{aligned} |c_k^{n+1}| &= \left| \frac{k}{n+1} c_{k-1}^n + \left(1 - \frac{k}{n+1}\right) c_k^n \right| \\ &\leq \frac{k}{n+1} |c_{k-1}^n| + \left(\frac{n+1-k}{n+1}\right) |c_k^n|. \end{aligned}$$

Se define ahora  $w_k$ ,  $k = 1, 2, \dots, n$ , como

$$w_k := \frac{k|c_{k-1}^n| + (n+1-k)|c_k^n|}{(n+1)|c_k^{n+1}|} \geq 1 \quad (3.6)$$

y  $w_0 = w_{n+1} = 1$ . Sea  $\Omega := \max w_k$  una cota superior sobre el factor por el cual el número de condición de la raíz  $c_{b,n+1}(x_0)$  en la base de Bernstein  $\{B_k^{n+1}(x)\}$  es reducido relativamente a  $c_{b,n}(x_0)$  en la base  $\{B_k^n(x)\}$ . Esto puede significar, que en el peor de los casos, el error de amplificación en los coeficientes (3.6) en el desarrollo de elevación de grado, anula los mejoramientos del condicionamiento de la raíz de la elevación de grado de las bases.

El mismo razonamiento se aplica a la subdivisión polinomial usando el algoritmo de De Casteljaou. Si  $P(x)$  tiene puntos de control  $\{c_k^n\}$  sobre  $[0, 1]$ , este puede ser subdividido en el punto  $x_s \in (0, 1)$ , para obtener sus puntos de control  $\{l_k^n\}$  y  $\{r_k^n\}$ ,

en los intervalos  $[0, x_s]$  y  $[x_s, 1]$  a la izquierda y a la derecha del punto de corte  $x_s$ .

$$l_k^n = \sum_{j=0}^k B_j^k(x_s) c_j^n \quad y \quad r_k^n = \sum_{j=k}^n B_{j-k}^{n-k}(x_s) c_j^n, \quad (3.7)$$

para  $k = 0, 1, \dots, n$ . Sabemos del teorema 2.4, que las raíces de  $P(x)$  en  $[0, x_s]$  y  $[x_s, 1]$  son menos sensitivos a perturbaciones de los coeficientes  $\{l_k^n\}$  y  $\{r_k^n\}$ , respectivamente, que áquellas perturbaciones en los coeficientes originales  $\{c_k^n\}$ .

Si los coeficientes  $\{c_k^n\}$  están sujetos a errores aleatorios de magnitud máxima relativa  $\varepsilon$ , sin embargo, los errores relativos máximos en los coeficientes (3.7) excederán al error  $\varepsilon$  por los factores:

$$w_k = \frac{1}{l_k^n} \sum_{j=0}^k |B_j^k(x_s) c_j^n| \quad y \quad \frac{1}{r_k^n} \left| \sum_{j=k}^n B_{j-k}^{n-k}(x_s) c_j^n \right|, \quad (3.8)$$

para  $k = 0, 1, \dots, n$ , respectivamente. Las cantidades  $w_k$  en (3.8) son números de condición para los puntos de control subdivididos  $\{l_k^n\}$  y  $\{r_k^n\}$ , con respecto a los coeficientes originales  $\{c_k^n\}$ , y por la desigualdad triangular, estos satisfacen la cota inferior  $w_k \geq 1$ . Si caracterizamos la subdivisión por el número de condición simple  $\Omega = \max\{w_k\}$ , por la conversión de potencia-a-Bernstein se prueba que  $\Omega$  es una cota superior en el factor, por lo cual los números de condición de las raíces, son reducidos a través de la subdivisión. Esto significa, en el peor de los casos, que ninguna ventaja en la estabilidad de la raíz resulta del uso explícito de la subdivisión (ver Farouki y Rajan [5]).

Consideremos luego el comportamiento de los números de condición de la raíz en las bases de Bernstein bajo el procedimiento de elevación de grado.

**Teorema 3.1.** (vea Farouki y Rajan '87, p. 203) *Para un polinomio arbitrario  $P(x)$  de grado  $n$  expresado en la base Bernstein sobre  $[0, 1]$ , sea  $C_{b,v}(x_s)$  el número de condición de cualquier raíz  $x_s \in [0, 1]$ , cuando  $P(x)$  es elevado de grado  $v \geq n$ . Entonces  $C_{b,v+1}(0) = C_{b,v}(0) = 0$  y  $C_{b,v+1}(1) = C_{b,v}(1) = 0$ , mientras que para cualquier raíz  $x_s$  sobre un intervalo abierto  $\langle 0, 1 \rangle$  tenemos  $C_{b,v+1}(x_s) < C_{b,v}(x_s)$ ,*

es decir, el número de condición decrece monotonamente bajo la elevación de grado.

**Prueba :** De (2.10), consideremos el comportamiento de la suma:

$$S_{b,v}(x_s) = \sum_{k=0}^v |c_k^v| \binom{v}{k} (1-x_s)^{n-k} x_s^k \quad (3.9)$$

$v \geq n$ , bajo la transformación (2.10). Primero calculamos el caso de los puntos límites del intervalo unitario.

$$S_{b,v}(x_s) = |c_0^v| \binom{v}{0} (1-x_s)^n + |c_1^v| \binom{v}{1} (1-x_s)^{n-1} x_s + \cdots + |c_v^v| \binom{v}{v} (1-x_s)^{n-v} x_s^v,$$

$$x_s = 0 \implies S_{b,v}(0) = |c_0^v| + 0 + \cdots + 0 = |c_0^v| = |c_0^{v+1}| = S_{b,v+1}(0),$$

$$x_s = 1 \implies S_{b,v}(1) = 0 + 0 + \cdots + |c_v^v| = |c_v^v| = |c_{v+1}^{v+1}| = S_{b,v+1}(1).$$

Tenemos entonces:

$$\begin{cases} S_{b,v}(0) = |c_0^v| = |c_0^{v+1}| = S_{b,v+1}(0) \\ S_{b,v}(1) = |c_v^v| = |c_{v+1}^{v+1}| = S_{b,v+1}(1) \end{cases}, \quad (3.10)$$

y en realidad estas cantidades se anulan cuando  $x_s = 0$  ó  $x_s = 1$ , respectivamente. El número de condición (1.6) ó (1.8) de una raíz para los puntos extremos de  $[0, 1]$  es invariante bajo el grado de elevación. Entonces consideramos la raíz en el intervalo abierto  $\langle 0, 1 \rangle$ , usando (2.10), (3.9) y aplicando la inecuación triangular para cada termino  $|kc_{k-1}^v + (v+1-k)c_k^v|$  en la suma aproximada. Obtenemos usando la regla de Descartes:

$$\begin{aligned} S_{b,v+1}(x_s) &= \sum_{k=0}^{v+1} |c_k^{v+1}| \binom{v+1}{k} (1-x_s)^{v+1-k} x_s^k \\ &= |c_0^v| (1-x_s)^{v+1} + \\ &= \sum_{k=1}^v \frac{k}{v+1} c_{k-1}^v + \left(1 - \frac{k}{v+1}\right) c_k^v \binom{v+1}{k} (1-x_s)^{v+1-k} x_s^k + |c_v^v| x_s^{v+1} \end{aligned}$$

$$S_{b,v+1}(x_s) < |c_0^v|(1-x_s)^{v+1} + \sum_{k=1}^v \frac{k|c_{k-1}^v|}{v+1} \binom{v+1}{k} (1-x_s)^{v+1-k} x_s^k + \\ |c_v^v| x_s^{v+1} + \sum_{k=1}^v \frac{(v+1-k)|c_k^v|}{v+1} \binom{v+1}{k} (1-x_s)^{v+1-k} x_s^k.$$

Haciendo:

$$\binom{v+1}{k} = \frac{(v+1)!}{k!(v+1-k)!} = \frac{(v+1)v!}{k(k-1)!(v-(k-1))!} = \frac{v+1}{k} \binom{v}{k-1},$$

$$\binom{v+1}{k} = \frac{(v+1)v!}{k!(v+1-k)(v-k)!} = \frac{v+1}{v+1-k} \binom{v}{k},$$

tenemos reemplazando:

$$S_{b,v+1}(x_s) < |c_v^v| x_s^{v+1} + \sum_{k=1}^v |c_{k-1}^v| \binom{v}{k-1} (1-x_s)^{v+1-k} x_s^k + \\ |c_0^v|(1-x_s)^{v+1} + \sum_{k=1}^v |c_k^v| \binom{v}{k} (1-x_s)^{v+1-k} x_s^k \\ = x_s \sum_{k=0}^v |c_k^v| \binom{v}{k} (1-x_s)^{v-k} x_s^k + (1-x_s) \sum_{k=0}^v |c_k^v| \binom{v}{k} (1-x_s)^{v-k} x_s^k \\ = \sum_{k=0}^v |c_k^v| \binom{v}{k} (1-x_s)^{v-k} x_s^k = S_{b,v}(x_s). \quad (3.11)$$

De esto y de (3.5) se tiene:

$$\Rightarrow |P'(x_s)|^{-1} S_{b,v+1}(x_s) < |P'(x_s)|^{-1} S_{b,v}(x_s) \\ \Rightarrow C_{b,v+1}(x_s) < C_{b,v}(x_s). \quad \blacksquare$$

**Ejemplo 3.2.** Dado el polinomio  $P(x) = x^2 - x + 0.25$ , y el polinomio perturbado  $\tilde{P}(x) = x^2 - 1.00001x + 0.25$  y  $\tilde{P}'(x) = 2x - 1.00001$ . Resolviendo el polinomio perturbado se tiene  $x_1 = 0.5022$  y  $x_2 = 0.4978$ . Consideramos  $x_s = x_1$ , entonces los coeficientes son  $c_0^{(2)} = 0.25$ ,  $c_1^{(2)} = -0.250005$  y  $c_2^{(2)} = 0.24999$ , donde el número de condición es  $C_{b,2}(0.5022) = 56.9$ , usando  $c_k^{v+1} = \frac{k}{v+1} c_{k-1}^v + (1 - \frac{k}{v+1}) c_k^v$  se determina

para  $n = 2$  y  $v = 2$ :

$$\begin{aligned} k = 1 \quad c_1^{(3)} &= \frac{1}{3}0.25 + (1 - \frac{1}{3})(-0.250005) = -0.08334 \\ k = 2 \quad c_2^{(3)} &= \frac{2}{3}(-0.250005) + (1 - \frac{2}{3})(0.24999) = -0.08334, \end{aligned}$$

también se cumple  $c_0^{(3)} = c_0^{(2)} = 0.25$  y  $c_3^{(3)} = c_3^{(2)} = 0.24999$ .

Luego  $S_{b,3}(0.5022) = 0.12501$  y el número de condición es  $C_{b,3}(0.5022) = 28.4752$ .

Entonces se tiene que  $C_{b,3}(0.5022) < C_{b,2}(0.5022)$ .

Las soluciones reales son  $x_1 = x_2 = 0.5$ , donde los errores relativos son 0.0044 en ambos casos.

El polinomio de Bernstein es  $P(x) = 0.25 \binom{2}{0} (1-x)^2 + -0.25 \binom{2}{1} (1-x)x + 0.25 \binom{2}{2} x^2$ , y el polinomio perturbado  $\tilde{P} = 0.25 \binom{2}{0} (1-x)^2 + -0.25001 \binom{2}{1} (1-x)x + 0.25 \binom{2}{2} x^2$ , donde los puntos de control son:  $c_0^{(2)} = 0.25$ ,  $c_1^{(2)} = -0.25001$ , y  $c_2^{(2)} = 0.25$ , y los nuevos  $c_{b,2}$  puntos de control son:  $c_0^{(3)} = 0.25$ ,  $c_1^{(3)} = -0.08334$ ,  $c_2^{(3)} = -0.08334$ ,  $c_3^{(3)} = 0.25$ , donde  $S_{b,3}(0.50223) = 0.12501$  y  $P'(0.50223) = 0.00446$  entonces  $C_{b,3}(0.50223) = 28.0291$ , el cual también satisface que  $C_{b,3}(x_s) < C_{b,2}(x_s)$ .  $\square$

Otro procedimiento el cual produce aproximaciones poligonales convergentes al polinomio de Bernstein es la tecnica de subdivisión asociada con el algoritmo de Casteljau. Esto también induce a un decrecimiento monótono de los números de condición de la raíz.

**Teorema 3.2.** (vea Farouki y Rajan '87, p. 204) Para un polinomio arbitrario  $P(x)$  expresado en la base de Bernstein sobre  $[0, 1]$ , sea  $C_b(x_0)$  el número de condición de cualquier raíz  $x_0 \in [0, 1]$ . Ahora dividimos  $P(x)$  en algún punto  $x_s \in (0, 1)$ . Por el algoritmo de De Casteljau, obtenemos los coeficientes de Bernstein en las bases transformadas  $\{B_k^n(u)\}$  y  $\{B_k^n(v)\}$  definido sobre  $\frac{x}{x_s} \equiv u \in [0, 1]$  y  $\frac{(x-x_s)}{(1-x_s)} \equiv v \in [0, 1]$ , respectivamente. Sea  $C'_b(x_0)$  el número de condición en las bases  $\{B_k^n(u)\}$  y  $\{B_k^n(v)\}$  despues de la partición, de acuerdo a que si  $x_0$  está a la izquierda o a la derecha de  $x_s$ . Entonces en los puntos extremos de  $[0, 1]$  tenemos  $C'_b(0) = C_b(0) = 0$  y  $C'_b(1) = C_b(1) = 0$ , y



los puntos de división  $C'_b(x_s) = 0 < C_b(x_s)$ . Finalmente, si  $x_0$  cae sobre cualquiera de los intervalos abiertos  $(0, x_s)$  ó  $(x_s, 1)$ , entonces la desigualdad estricta  $C'_b(x_0) < C_b(x_0)$  se mantiene, es decir, el número de condición decrece monótonamente bajo la subdivisión polinomial.

**Prueba :** Examinemos el caso  $x_0 \in [0, x_s]$  solamente, donde los argumentos de  $x_0 \in [x_s, 1]$  son exactamente analogos. Consideremos la suma:

$$S'_b(x_0) = \sum_{k=0}^n |P_k^{(k)}| B_k^n(u_0) \quad (3.12)$$

donde  $u_0 = \frac{x_0}{x_s}$ , caracterizando el número de condición  $C'_b(x_s)$  en la base  $\{B_k^n(u)\}$ , definido en  $u \in [0, 1]$  despues de la partición. Comparamos (3.12) con la suma  $S_b(x_0)$ , que caracteriza el número de condición  $C_b(x_0)$  antes de la partición.

Para la izquierda en el punto final, desarrollando (3.12) y evaluando  $x_0 = 0 (\implies u_0 = 0)$ , obtenemos:

$$S'_b(0) = |P_0^{(0)}| B_0^n(0) + |P_1^{(1)}| B_1^n(0) + \dots + |P_n^{(n)}| B_n^n(0), \quad (3.13)$$

donde

$$B_k^n(x) = \binom{n}{k} (1-x)^{n-k} x^k$$

$$\begin{array}{ll} B_0^n(x) = (1-x)^n & B_0^n(0) = 1 \\ B_1^n(x) = \binom{n}{1} (1-x)^{n-1} x & B_1^n(0) = 0 \\ & \vdots \\ B_n^n(x) = \binom{n}{n} x^n & B_n^n(0) = 0 \end{array}$$

Reemplazando en (3.12) se tiene  $S'_b(0) = |P_0^{(0)}| = |c_0| = S_b(0)$  ( $= 0$ , cuando  $x_0 = 0$  es una raíz). También, si  $x_0 = x_s$ , entonces  $u_0 = 1$  tenemos:

$$S'_b(x_0) = \sum_{k=0}^n |P_k^{(k)}| B_k^n(u_0)$$

$$S'_b(x_s) = \sum_{k=0}^n |P_k^{(k)}| B_k^n(1)$$

$$= |P_0^{(0)}| B_0^n(1) + |P_1^{(1)}| B_1^n(1) + \dots + |P_n^{(n)}| B_n^n(1)$$

$$B_k^n(u) = \binom{n}{k} \frac{(x_s - u)^{n-k} u^k}{(x_s)^n}$$

Luego:

$$B_0^n(u) = (1 - u)^n \quad B_0^n(1) = 0$$

$$B_1^n(u) = \binom{n}{1} (1 - u)^{n-1} u \quad B_1^n(1) = 0$$

⋮

$$B_n^n(u) = \binom{n}{n} u^n \quad B_n^n(1) = 1.$$

Reemplazando se tiene:

$$S'_b(x_s) = |P_0^{(0)}| \times 0 + |P_1^{(1)}| \times 0 + \dots + |P_n^{(n)}| \times 1,$$

$$S'_b(x_s) = |P_n^{(n)}| = 0,$$

ya que el valor  $P_n^{(n)}$  determinado por el algoritmo de De Casteljau debe de ser cero para una raíz  $x_s$  de  $P(x)$  (i.e. si el polinomio esta dividido por una raíz, esta raíz puede tener condiciones perfectas). Por contraste  $S_b(x_s) > 0$ , desde que toda la función base sobre  $[0, 1]$  es positivo para  $x_s \in ]0, 1[$ .

Consideremos el caso  $x_0 \in ]0, x_s[$ . Usando  $P_k^k = \sum_{j=0}^n c_j B_j^k(x_s)$  podemos escribir la

suma (3.12) como:

$$\begin{aligned}
S'_b(x_0) &= \sum_{k=0}^n \left| \sum_{j=0}^k B_j^k(x_s) c_j B_k^n(u_0) \right| < \sum_{k=0}^n \sum_{j=0}^k |c_j| B_j^k(x_s) B_k^n(u_0) \\
&= \sum_{j=0}^0 |c_j| B_j^0(x_s) B_0^n(u_0) + \cdots + \sum_{j=0}^n |c_j| B_j^n(x_s) B_n^n(u_0) \\
&= \sum_{k=0}^n |c_0| B_0^k(x_s) B_k^n(u_0) + \sum_{k=1}^n |c_1| B_1^k(x_s) B_k^n(u_0) + \cdots + \\
&\quad + \sum_{k=n}^n |c_n| B_n^k(x_s) B_k^n(u_0) \\
&< \sum_{j=0}^n |c_j| \sum_{k=j}^n B_j^k(x_s) B_k^n(u_0) \\
&= \sum_{j=0}^n |c_j| B_j^n(x_s)
\end{aligned}$$

$$S'_b(x_0) \leq \sum_{j=0}^n |c_j| B_j^n(x_0) = S_b(x_0) \quad (3.14)$$

$$|P'(x_0)|^{-1} S'_b(x_0) \leq |P'(x_0)|^{-1} S_b(x_0)$$

$$C'_b(x_0) \leq C_b(x_0) \quad (\text{por (3.5)})$$

Si el número de condición  $C'_b(x_0)$  es definido usando las derivadas con respecto a la transformación de la variable dependiente  $u = \frac{x}{x_s}$ , este deberá crecer en un factor  $\frac{1}{x_s}$ . Sin embargo, argumentamos que las derivadas con respecto a la variable original  $x$  son las apropiadas para  $C'_b(x_0)$ , desde que estamos interesados en la magnitud de las perturbaciones relativas de la raíz en el intervalo unitario original que a los subintervalos generados por el corte. Ignoraremos entonces este factor. ■

**Ejemplo 3.3.** Dado el polinomio  $\tilde{P}(x) = (1-x)^2 - 2\binom{2}{1}(1-x)x + 1.00001x^2$  y  $\tilde{P}'(x) = 12.00002x - 6$ , resolviendo el polinomio perturbado se tiene  $x_1 = 0.7887$  y  $x_2 = 0.2113$ . Consideramos  $x_s = x_1$  entonces los coeficientes  $c_0 = 1$ ,  $c_1 = -2$  y  $c_2 = 1.00001$ , donde el número de condición es  $C_b(0.7887) = 038487$ , considerando

$x_s = 0.5 \in \langle 0, 1 \rangle$  se determina  $v = 0.5774$  y usando (3.14) se tiene  $S'_b(0.7887) = 1.2441$  y el número de condición es  $C'_b(0.7887) = 0.3591$ . Entonces se tiene que  $C'_b(0.7887) < C_b(0.7887)$ .

Las soluciones reales son  $x_1 = 0.7887$  y  $x_2 = 0.2113$ , donde los errores relativos son cero en ambos casos.  $\square$

## 4 TRANSFORMACIONES ENTRE LA BASE MONOMIAL A LA BASE DE BERNSTEIN

El objetivo de este capítulo es comparar la estabilidad numérica de las dos bases que se encuentran frecuentemente en los modelos de aplicaciones geométricas. Trabajaremos en el dominio unitario  $[0, 1]$  y luego generalizaremos para un intervalo arbitrario  $[\alpha, \beta]$ .

### 4.1 Conversión Básica

Dado el polinomio de grado  $n$ , expresados en las bases monomial y de Bernstein sobre el intervalo unitario  $[0, 1]$ .

$$P(x) = \sum_{k=0}^n a_k x^k = \sum_{k=0}^n c_k \binom{n}{k} (1-x)^{n-k} x^k, \quad k = 0, 1, \dots, n. \quad (4.1)$$

Cada función de base de potencia puede ser expresado como una combinación lineal de  $n + 1$  funciones en la base de Bernstein, y vice-versa.

$$x^j = \sum_{k=0}^n \Lambda_{jk} \binom{n}{k} (1-x)^{n-k} x^k, \quad \binom{n}{j} (1-x)^{n-j} x^j = \sum_{k=0}^n \Lambda_{jk}^{-1} x^k \quad (4.2)$$

para  $j = 0, 1, \dots, n$ . Determinamos los elementos de la transformación matricial  $\Lambda$ ,  $x^j = x^j \{(1-x) + x\}^{n-j}$  donde:

$$x^j = x^j \sum_{k=0}^{n-j} \binom{n-j}{k} (1-x)^{n-j-k} x^k = \sum_{k=0}^{n-j} \binom{n-j}{n-(k+j)} (1-x)^{n-j-k} x^{k+j}.$$

Poniendo  $k + j = K$ , obtenemos:

$$x^j = \sum_{K=j}^n \binom{n-j}{n-K} (1-x)^{n-K} x^K. \quad (4.3)$$

Sabemos:

$$\binom{n-j}{n-K} = \frac{(n-j)!}{(n-K)!(K-j)!} = \frac{j! \binom{K}{j} \times \frac{n!}{K!(n-K)!}}{j!(n-j)!} = \frac{\binom{K}{j} \binom{n}{K}}{\binom{n}{j}} \quad (4.4)$$

Reemplazando (4.4) en (4.3) tenemos:

$$x^j = \sum_{K=j}^n \frac{\binom{K}{j} \binom{n}{K}}{\binom{n}{j}} (1-x)^{n-K} x^K. \quad (4.5)$$

Los elementos de la transformación inversa  $\Lambda^{-1}$  son obtenidos de la expansión:

$$\begin{aligned} \binom{n}{j} (1-x)^{n-j} x^j &= \binom{n}{j} x^j \sum_{k=0}^{n-j} \binom{n-j}{k} (-x)^k, \text{ (por expansión binomial)} \\ &= \binom{n}{j} \sum_{k=0}^{n-j} (-1)^k \binom{n-j}{n-j-k} x^{k+j}. \end{aligned}$$

Poniendo  $k+j=K$ , y por (4.4) obtenemos:

$$\binom{n}{j} (1-x)^{n-j} x^j = \sum_{K=j}^n (-1)^{K-j} \binom{K}{j} \binom{n}{K} x^K \quad (4.6)$$

De (4.2), (4.5), y (4.6) se tiene:

$$\Lambda_{jk} = \binom{k}{j} \binom{n}{j}^{-1}, \quad \Lambda_{jk}^{-1} = (-1)^{k-j} \binom{n}{k} \binom{k}{j} \quad (4.7)$$

para  $j \leq k$ , mientras  $\Lambda_{jk} = \Lambda_{jk}^{-1} = 0$  para  $j > k$ . Vemos que las matrices  $\Lambda$  y  $\Lambda^{-1}$  son triangulares superiores, y que los elementos de  $\Lambda$  son todos no-negativos.

Las siguientes transformaciones entre los coeficientes potenciales  $\{a_k\}$  y los coeficientes de Bernstein  $\{c_k\}$  son obtenidos:

$$P(x) = \sum_{j=0}^n a_j \left( \sum_{k=j}^n \underbrace{\frac{\binom{k}{j} \binom{n}{K}}{\binom{n}{j}}}_{\Lambda_{jk}} \binom{n}{K} (1-x)^{n-K} x^K \right) = \sum_{k=0}^n c_k \binom{n}{k} (1-x)^{n-k} x^k.$$

Desarrollando se tiene:

$$a_0 \sum_{k=0}^n \wedge_{0k} \binom{n}{k} (1-x)^{n-k} x^k + \dots + a_l \sum_{k=l}^n \wedge_{lk} \binom{n}{k} (1-x)^{n-k} x^k + \dots = \\ c_0 \binom{n}{0} (1-x)^{n-0} x^0 + \dots + c_l \binom{n}{l} (1-x)^{n-l} x^l + \dots,$$

donde  $0 \leq l \leq n$ . Igualando el factor común  $\binom{n}{l} (1-x)^{n-l} x^l$  se obtiene:

$$a_0 \wedge_{0l} + \dots + a_l \wedge_{ll} + \dots + a_n \wedge_{nl} = c_l, \quad \text{donde } 0 \leq l \leq n.$$

Sabemos que  $\wedge_{jl} = 0$  cuando  $j > l$ , y haciendo  $l = k$  se tiene:

$$c_k = \sum_{j=0}^n a_j \wedge_{jk} = \sum_{j=0}^k \binom{k}{j} \binom{n}{j}^{-1} a_j. \quad (4.8)$$

Luego:

$$P(x) = \sum_{k=0}^n a_k x^k = \sum_{j=0}^n c_j \sum_{k=j}^n (-1)^{k-j} \binom{n}{k} \binom{k}{j} x^k = \sum_{j=0}^n c_j \sum_{k=j}^n \wedge_{jk}^{-1} x^k$$

En forma analoga se desarrolla para el otro caso, se tiene:

$$a_0 x^0 + \dots + a_l x^l + \dots = c_0 \sum_{k=0}^n \wedge_{0k}^{-1} x^k + \dots + c_l \sum_{k=l}^n \wedge_{lk}^{-1} x^k + \dots + c_n \sum_{k=n}^n \wedge_{nk}^{-1} x^k,$$

donde  $0 \leq l \leq n$ . Igualando el factor común  $x^l$  se tiene:

$$a_l = c_0 \wedge_{0l}^{-1} + \dots + c_l \wedge_{ll}^{-1} + \dots, \quad c_n \wedge_{nn}^{-1} \quad 0 \leq l \leq n. \\ a_k = \sum_{j=0}^n c_j \wedge_{jl}^{-1} = \sum_{j=0}^l c_j \wedge_{jl}^{-1} = \sum_{j=0}^l (-1)^{l-j} \binom{n}{l} \binom{l}{j} c_j.$$

Haciendo  $l = k$  tenemos:

$$a_k = \sum_{j=0}^n c_j \wedge_{jk}^{-1} = \sum_{j=0}^k (-1)^{k-j} \binom{n}{k} \binom{k}{j} c_j. \quad (4.9)$$

Los elementos  $\Lambda_{jk}$  y  $\Lambda_{jk}^{-1}$  de las matrices de conversión pueden ser calculados usando un algoritmo de aritmética racional exacta, y luego convertidos a punto flotante por una división simple (convertirlos en números de máquina). Los coeficientes potencial y de Bernstein pueden ser dados como punto flotante, entonces en cualquiera de dos casos, calculamos cada  $a_k$  o  $c_k$  usando (4.8) y (4.9). Entonces se requiere a lo más  $k + 1$  multiplicaciones y  $k$  adiciones en punto flotante, ver Farouki y Rajan '87, (p. 195).

## 4.2 Conversión Básica Matricial

Dados dos vectores de coeficientes  $a$  y  $c$  de  $P(t)$ , de potencia y de Bernstein, respectivamente. Las matrices de transformación  $M := (m_{jk})$  y  $M^{-1} := (m_{jk}^{-1})$  son ambas triangular superior y están dada por:

$$m_{jk} = \begin{cases} (-1)^{j-k} \binom{n}{j} \binom{j}{k} & \text{si } j \geq k \\ 0 & \text{si } j < k \end{cases} \quad \text{y} \quad m_{jk}^{-1} = \begin{cases} \frac{\binom{j}{k}}{\binom{n}{k}} & \text{si } j \geq k \\ 0 & \text{si } j < k. \end{cases} \quad (4.10)$$

**Lema 4.1.** *Las matrices  $M$  y  $M^{-1}$  son matrices simétricas en la diagonal secundaria  $j = n - k$ , es decir:*

$$m_{n-k, n-j} = m_{jk} \quad \text{y} \quad m_{n-k, n-j}^{-1} = m_{jk}^{-1} \quad \text{para } 0 \leq j, k \leq n.$$

**Prueba:** De (4.10) para  $j \geq k$  se cumple

$$\begin{aligned} m_{n-k, n-j} &= (-1)^{(n-k)-(n-j)} \binom{n}{n-k} \binom{n-k}{n-j} = (-1)^{j-k} \binom{n}{j} \binom{j}{k} = m_{jk} \\ &\implies m_{n-k, n-j} = m_{jk} \quad \text{para } 0 \leq j, k \leq n. \end{aligned} \quad (4.11)$$



De manera similar se procede para el otro caso, para  $j \geq k$  de (4.10):

$$\begin{aligned}
 m_{n-k,n-j}^{-1} &= \frac{\binom{n-k}{n-j}}{\binom{n}{n-j}} = \frac{\binom{j}{k}}{\binom{n}{k}} = m_{jk}^{-1} \\
 \implies m_{n-k,n-j}^{-1} &= m_{jk}^{-1} \text{ para } 0 \leq j, k \leq n. \quad \blacksquare
 \end{aligned}
 \tag{4.12}$$

### 4.2.1 Los Números de Condición

**Ejemplo 4.1.** Asumamos los calculos obtenidos con 32-bit y 64-bit en una precisión simple y doble, respectivamente. Sean  $s = 24$  y  $s = 56$  la cantidad de bits de la mantisa en la representación del punto flotante. Sean las matrices  $M$  y  $M^{-1}$  dadas por (4.10):

n	$\ M\ _\infty$	$\ M^{-1}\ _\infty$	$Cond_\infty(M)$	$Cond_2(M)$
2	4	3	12	8.8
3	12	4	48	26.7
4	32	5	160	80.8
5	80	6	480	244.7
6	240	7	1680	741
7	672	8	5376	2242
8	1792	9	16128	6781
9	5376	10	53760	20494
10	15360	11	$1.69 \times E^{+5}$	61904
11	42240	12	$5.07 \times E^{+5}$	$1.86 \times E^{+5}$
12	$1.27 \times E^{+5}$	13	$1.65 \times E^{+6}$	$5.64 \times E^{+5}$
13	$3.66 \times E^{+5}$	14	$5.12 \times E^{+6}$	$1.70 \times E^{+6}$
14	$1.02 \times E^{+6}$	15	$1.54 \times E^{+7}$	$5.13 \times E^{+6}$
15	$3.07 \times E^{+6}$	16	$4.92 \times E^{+7}$	$1.55 \times E^{+7}$
16	$8.95 \times E^{+6}$	17	$1.52 \times E^{+8}$	$4.66 \times E^{+7}$
17	$2.53 \times E^{+7}$	18	$4.56 \times E^{+8}$	$1.40 \times E^{+8}$
18	$7.60 \times E^{+7}$	19	$1.44 \times E^{+9}$	$4.23 \times E^{+8}$
19	$2.22 \times E^{+8}$	20	$4.44 \times E^{+9}$	$1.27 \times E^{+9}$
20	$6.35 \times E^{+8}$	21	$1.33 \times E^{+10}$	$3.83 \times E^{+9}$

Esta tabla nos muestra los valores establecidos que tienen las  $\|\cdot\|_\infty$  y  $Cond_\infty(M)$  para un valor de  $n$  que se encuentra entre 2 y 20, ver Daniel y Doubisse, (p. 125).

Suponemos que los componentes de  $p$  son precisos y que la perturbación asociada solo afecta el último bit de la mantisa. En este caso,  $c_p$  equivale a  $c_{max}$ , tenemos la mayor perturbación en la computadora.

$$10^{-c_p} = 10^{-c_{max}} \quad 2^{-s}$$

$$\implies \log_{10} 10^{-c_p} = \log_{10} 2^{-s} \implies c_p = s \log_{10} 2.$$

Para  $s = 24$  resulta  $c_p = 24(0.301) = 7.22$ .

Para  $s = 56$  obtenemos  $c_p = 56(0.301) = 16.86$ .

Se obtiene la siguiente tabla, ver Daniel y Doubisse '89, (p. 126).

n	$c_p \simeq 7.22$	$c_p \simeq 16.86$	$c_p \simeq 3$
2	6	16	2
3	6	15	1
4	5	15	1
5	5	14	0
6	4	14	0
7	3	13	0
8	3	13	0
9	2	12	0
10	2	12	0
11	2	11	0
12	1	11	0
13	1	10	0
14	0	10	0
15	0	9	0
16	0	9	0
17	0	8	0
18	0	8	0
19	0	7	0
20	0	7	0

Esta tabla nos indica los números de dígitos significativos cuando  $n$  toma un valor entre 2 y 20.

Denotamos con  $R_j$  y  $C_k$  la suma de los valores absolutos de los elementos matriciales  $m_{jk}$  a través de filas  $j$  y columnas  $k$  de la conversión matricial básica  $M$ , respectivamente. En forma analoga, sean  $R_j^{-1}$  y  $C_k^{-1}$  las correspondientes sumas de la matriz inversa  $M^{-1}$ .

**Lema 4.2.** *Las sumas de filas y columnas de la matriz  $M$  son dados por:*

$$R_j = C_{n-j} = \binom{n}{j} 2^j \text{ para } j = 0, 1, \dots, n$$

**Prueba:** *Consideramos primero la suma de las filas  $R_j$  donde tenemos:*

$$R_j = \sum_{k=0}^n |m_{jk}|$$

Sabemos que  $m_{jk}$  y  $m_{jk}^{-1}$  son obtenidos de (4.10), resulta:

$$R_j = \sum_{k=0}^n |m_{jk}| = \sum_{k=0}^j |m_{jk}| + \sum_{k=j+1}^n |m_{jk}| = \sum_{k=0}^j \binom{n}{j} \binom{j}{k} = \binom{n}{j} 2^j, \quad (4.13)$$

para  $j = 0, 1, \dots, n$ . Para la suma de columnas  $C_k$  podemos recurrir al Lema 4.1 y escribimos:

$$C_k = \sum_{j=0}^n |m_{jk}| = \sum_{j=0}^n |m_{n-k, n-j}| = R_{n-k} = \sum_{j=0}^{n-k} \binom{n}{n-k} \binom{n-k}{j} = \binom{n}{n-k} 2^{n-k}.$$

Haciendo  $k = n - j$ ,  $j = n - k$ , obtenemos:

$$C_{n-j} = \binom{n}{j} 2^j \text{ para } j = 0, 1, \dots, n. \quad \blacksquare \quad (4.14)$$

**Lema 4.3.** Las normas  $\|\cdot\|_1$  y  $\|\cdot\|_\infty$  de la transformación matricial  $M$  son dadas por:

$$\|M\|_1 = \|M\|_\infty = \binom{n}{v} 2^v$$

donde  $v$  denota el mayor entero que no excede a  $\frac{2(n+1)}{3}$ .

**Prueba:** Como:

$$\|M\|_1 = \max_{0 \leq k \leq n} \sum_{j=0}^n |m_{jk}|, \quad \|M\|_\infty = \max_{0 \leq k \leq n} \sum_{k=0}^n |m_{jk}|,$$

tenemos por el Lema (4.2):

$$\|M\|_1 = \max_{0 \leq k \leq n} C_k, \quad \|M\|_\infty = \max_{0 \leq k \leq n} R_j,$$

donde  $\{C_k\}$  es la suma de la columna  $k$  y en la suma de la fila  $j$ . Se cumple:

$$\|M\|_1 = \max_{0 \leq k \leq n} C_k = \max_{0 \leq k \leq n} R_{n-k} = \max_{0 \leq n-j \leq n} R_j = \max_{0 \leq j \leq n} R_j = \|M\|_\infty$$

$$\implies \|M\|_1 = \|M\|_\infty. \quad (4.15)$$

Definimos las diferencias:

$$\begin{aligned} \Delta R_j &= R_j - R_{j-1} \\ &= \binom{n}{j} 2^j - \binom{n}{j-1} 2^{j-1}, \quad (\text{por Lema (4.2)}) \\ &= \frac{n! 2^{j-1}}{(n-j)!(j-1)!} \left[ \frac{2(n-j+1) - j}{j(n-j+1)} \right], \quad j = 1, 2, \dots, n, \end{aligned}$$

y si  $R_j$  es no decreciente, (i.e.  $\Delta R_j \geq 0$ ), cuando:

$$\begin{aligned} 2(n-j+1) - j &\geq 0, \\ j &\leq \frac{2(n+1)}{3}. \end{aligned} \quad (4.16)$$

y si  $R_j$  es decreciente ( $\Delta R_j < 0$ ) de otro modo. Se dan dos casos según el grado de  $n$ :

i) Si  $n$  no es de la forma  $3r + 2$  para  $r = 0, 1, \dots$ , la ecuación en (4.16) es realizada por el entero  $j$ , y tenemos:

$$R_0 < \dots < R_{v-1} < R_v > R_{v+1} > \dots > R_n,$$

donde  $v$  denota el entero par de  $\frac{2(n+1)}{3}$ , el cual es par.

ii) Si  $n$  es de la forma  $3r + 2$  para  $r = 0, 1, \dots$ , donde  $\frac{2(n+1)}{3}$  es un entero y de (4.16) tenemos la ecuación de  $j = \frac{2(n+1)}{3}$ :

$$R_0 < \dots < R_{v-1} = R_v > \dots > R_n,$$

donde  $v = \frac{2(n+1)}{3}$ . ■

**Lema 4.4.** Las normas de la transformación matricial  $M^{-1}$  estan dadas por:

$$\|M^{-1}\|_1 = \|M^{-1}\|_\infty = n + 1$$

**Prueba:** Es suficiente consideramos las sumas de las filas ó columnas solamente.

De la suma de la  $j$  - esima fila y por (4.10) tenemos:

$$R_j^{-1} = \sum_{k=0}^n |m_{jk}^{-1}| = \sum_{k=0}^j |m_{jk}^{-1}| + \sum_{k=j+1}^n |m_{jk}^{-1}| = \sum_{k=0}^j \frac{\binom{j}{k}}{\binom{n}{k}} = \binom{n}{j}^{-1} \sum_{k=0}^j \binom{n-k}{n-j}.$$

Haciendo  $l = j - k$ , resulta:

$$R_j^{-1} = \binom{n}{j}^{-1} \sum_{l=0}^j \binom{n-j+l}{n-j}. \quad (4.17)$$

Sabemos que:

$$\sum_{i=0}^r \binom{m+i}{m} = \binom{m+r+1}{m+1}.$$

Reemplazando se tiene:

$$R_j^{-1} = \binom{n}{j}^{-1} \binom{n-j+j+1}{n-j+1} = \binom{n}{j}^{-1} \binom{n+1}{n-j+1},$$

por la simetría de filas y columnas:

$$R_j^{-1} = C_{n-j}^{-1} = \frac{n+1}{n-j+1} \text{ para } j = 0, 1, \dots, n.$$

Luego, como sabemos:

$$\|M^{-1}\|_{\infty} = \max_{0 \leq j \leq n} R_j^{-1} = \max_{0 \leq j \leq n} \frac{n+1}{n-j+1} = n+1.$$

Por el Lema 4.3 se tiene:

$$\|M^{-1}\|_{\infty} = \|M^{-1}\|_1 = n+1. \quad \blacksquare \quad (4.18)$$

**Proposición 4.1.** (ver Farouki '91) El número de condición de la transformación de los polinomios  $P(t)$  de grado  $n$  en la base de Bernstein sobre  $t \in [0, 1]$  a la base de potencia, cuando  $t = 0$  (o vice-versa) en las normas  $\|\cdot\|_1$  y  $\|\cdot\|_{\infty}$ , está dada por:

$$\text{Cond}_1(M) = \text{Cond}_{\infty}(M) = (n+1) \binom{n}{v} 2^v,$$

donde  $v$  es el mayor entero que no excede  $\frac{2(n+1)}{3}$ .

**Prueba:** De (4.18) y por el Lema 4.3, obtenemos:

$$\|M\|_1 \|M^{-1}\|_1 = \|M\|_{\infty} \|M^{-1}\|_{\infty} = (n+1) \binom{n}{v} 2^v$$

donde  $v$  denota el mayor entero que no excede  $\frac{2(n+1)}{3}$ .

Entonces:

$$\text{Cond}_1(M) = \text{Cond}_{\infty}(M) = (n+1) \binom{n}{v} 2^v, \quad (4.19)$$

donde  $v$  denota el mayor entero que no excede  $\frac{2(n+1)}{3}$ . ■

## 4.2.2 Una Expresión Aproximada de $\text{Cond}(M)$

El número de condición está dado por (4.19). Recordando la aproximación de Stirling para la función factorial.

$$m! \simeq \sqrt{2\pi m} e^{-m} m^m, \quad (4.20)$$

tenemos:

$$\binom{n}{v} = \frac{n!}{(n-v)!v!} \simeq \frac{n^{n+1/2}}{(n-v)^{(n-v)+1/2}\sqrt{2\pi v}v} = \frac{1}{\sqrt{2\pi v}} \left[\frac{n}{n-v}\right]^{n+1/2} \left[\frac{n-v}{v}\right]^v.$$

Sustituyendo en (4.19) tenemos:

$$\text{Cond}(M) \simeq \frac{n+1}{\sqrt{2\pi v}} \left[\frac{n}{n-v}\right]^{n+1/2} \left[\frac{2(n-v)}{v}\right]^v.$$

Para simplificar notemos que  $\frac{n}{n-v} \approx 3$ , donde  $n$  es de la forma  $3r+2$ , y de otro modo es ligeramente mayor que 3, el residuo  $\frac{n}{n-v}-3$  disminuye cuando  $n$  se incrementa. Análogamente,  $\frac{2(n-v)}{v} \approx 1$ , si  $n=3r+2$ , de otro modo es ligeramente menor que 1 con un residuo  $1-\frac{2(n-v)}{v}$  que disminuyen con  $n$ .

Si consideramos la aproximación  $\frac{n}{n-v} \simeq 3$  y  $\frac{2(n-v)}{v} \simeq 1$  se obtiene:

$$\text{Cond}(M) \simeq \frac{n+1}{\sqrt{2\pi v}} \times 3^{n+1/2} \times 1^v \simeq \frac{3^{n+1/2} \times (n+1)}{\sqrt{2\pi v}}.$$

Finalmente, asumiendo que  $v \simeq \frac{2n}{3}$ , y  $\sqrt{\frac{n+1}{n}} \simeq 1$  reemplazando:

$$\text{Cond}(M) \simeq \frac{3^{n+1/2} \times (n+1)}{\sqrt{2\pi(\frac{2n}{3})}} \simeq \frac{3^{n+1}}{2\sqrt{\pi}} \sqrt{n+1}. \quad (4.21)$$

Si el valor preciso de la  $Cond(M)$ , (4.21) puede ser simplificado por el factor  $\frac{3}{2\sqrt{\pi}} \sim 0.85$ , escribiendo.

$$Cond(M) \simeq 0.85 \times 3^n \times \sqrt{n} + 1.$$

La aproximación de (4.21) es más fácilmente calculado y ilustrado que (4.19) al extremo de la sensibilidad del proceso de conversión básico del polinomio de grado  $n$ .

Por ejemplo, es evidente que si  $n$  es grande, el valor de la  $Cond(M)$  es incrementado por un factor aproximadamente 3, para un incremento en la unidad del grado. Ilustramos el aumento en forma exacta de (4.19) para  $Cond(M)$  de grado  $n$ , ver Farouki '91.

n	$Cond_1 = Cond_\infty$	$Cond_2$
2	$1.20 \times 10^1$	$7.71 \times 10^0$
3	$4.80 \times 10^1$	$2.31 \times 10^1$
4	$1.60 \times 10^2$	$6.95 \times 10^1$
5	$4.80 \times 10^2$	$2.09 \times 10^2$
6	$1.68 \times 10^3$	$6.30 \times 10^2$
7	$5.38 \times 10^3$	$1.90 \times 10^3$
8	$1.61 \times 10^4$	$5.71 \times 10^3$
9	$5.38 \times 10^4$	$1.72 \times 10^4$
10	$1.69 \times 10^5$	$5.17 \times 10^4$
11	$5.07 \times 10^5$	$1.56 \times 10^5$
12	$1.65 \times 10^6$	$4.68 \times 10^5$

### 4.2.3 Las Condiciones de Conversión entre la Base Monomial y de Bernstein

Las operaciones del punto flotante nos obligan a utilizar adecuadamente las propiedades asociativas.

Supongamos que se nos da coeficientes de potencia  $\{a_k\}$ , los cuales sufren errores aleatorios intrínsecos de una magnitud relativa máxima  $\varepsilon$ , y calculamos los puntos



de Bézier  $\{c_j\}$  usando (4.8). El error relativo máximo  $\delta_j$  en  $c_j$  será:

$$\delta_j := w_j \varepsilon, \quad \text{donde} \quad w_j := \frac{1}{|c_j|} \sum_{k=0}^j \frac{\binom{j}{k}}{\binom{n}{k}} |a_k| \geq 1 \quad (4.22)$$

para  $j = 0, 1, \dots, n$ . Llamamos  $w_j$  al número de condición de la conversión de los puntos de Bézier  $c_j$ , con respecto a los coeficientes de potencia  $\{a_k\}$ , y el menor valor  $w_j \geq 1$  sale de la aplicación de la desigualdad triangular por (4.8) (note que  $w_j = 1$  solamente cuando  $a_0, a_1, \dots, a_j$  son todos del mismo signo).

Esto es, los errores relativos en los coeficientes de potencia son siempre ampliados en los puntos de Bézier cuando los coeficientes finales son obtenidos por la conversión (4.8). Veremos en la siguiente sección que este efecto cancela el mejoramiento de la raíz por la condición de la forma de Bernstein. Si fuéramos a caracterizar la conversión de potencia-a-Bernstein en (4.8) por un número de condición simple  $\Omega$ , se considera:

$$\Omega = \max_{0 \leq j \leq n} \{w_j\}, \quad (4.23)$$

puesto que los números de condición siempre miden la perturbación peor posible de las salidas.

Debera notarse que los errores de amplificación de los coeficientes de la conversión de potencia-a-Bernstein demostradas líneas arriba realmente ocurren en la conversión entre cualquier dos bases (incluyendo la conversión de Bernstein al caso de potencia). (ver Farouki y Rajan '88).

### 4.3 Número de Condición de las Raíces

Los números de condición  $C_p(x_s)$  y  $C_b(x_s)$  de una raíz real simple  $x_s \in [0, 1]$  del polinomio  $P(x)$  en las formas de Potencia y de Bernstein,

respectivamente, son dados por:

$$C_p(x_s) = |P'(x_s)|^{-1}S_p(x_s) \quad , \quad C_b(x_s) = |P'(x_s)|^{-1}S_b(x_s). \quad (4.24)$$

donde  $S_p(x_s) = \sum_{j=0}^n |a_j|x_s^j$ , y  $S_b(x_s) = \sum_{j=0}^n |c_j|\binom{n}{j}(1-x_s)^{n-j}x_s^j$  según (1.6). Veremos que  $S_b(x_s) \leq S_p(x_s)$ ,  $\forall x_s \in [0, 1]$  (la desigualdad se cumple solamente en el caso  $x_s = 0$ ). Bajando el factor  $|P'(x_s)|^{-1}$  en (4.24), el cual es independiente de las bases, comparamos las sumas  $S_p(x_s)$  y  $S_b(x_s)$  los cuales caracterizan el número de condición en las bases de potencia y de Bernstein. Por (3.5) y (4.22), tendremos para  $x_s \in [0, 1]$ :

$$S_b(x_s) = \sum_{j=0}^n |c_j|\binom{n}{j}(1-x_s)^{n-j}x_s^j = \sum_{j=0}^n \frac{1}{w_j} \sum_{k=0}^j \frac{\binom{j}{k}}{\binom{n}{k}} |a_k| \binom{n}{j} (1-x_s)^{n-j}x_s^j, \quad (4.25)$$

y introduciendo  $\Omega$  por (4.23), reagrupando y reordenando la sumatoria, se obtiene:

$$\begin{aligned} S_b(x_s) &= \sum_{j=0}^n \min \left\{ \frac{1}{w_j} \right\} \sum_{k=0}^j \frac{\binom{j}{k}}{\binom{n}{k}} |a_k| \binom{n}{j} (1-x_s)^{n-j}x_s^j \\ &\geq \frac{1}{\Omega} \sum_{j=0}^n \sum_{k=0}^j \frac{\binom{j}{k}}{\binom{n}{k}} |a_k| \binom{n}{j} (1-x_s)^{n-j}x_s^j \\ &= \frac{1}{\Omega} \sum_{k=0}^n |a_k| \sum_{j=k}^n \frac{\binom{j}{k}}{\binom{n}{k}} \binom{n}{j} (1-x_s)^{n-j}x_s^j \\ &= \frac{1}{\Omega} \sum_{k=0}^n |a_k|x_s^k = \frac{1}{\Omega} S_p(x_s) \quad (\text{por (4.5)}). \end{aligned} \quad (4.26)$$

Cuando se cumple esta desigualdad, esto significa que la cantidad  $\Omega$ , dada por (4.22) y (4.23), representa una cota superior en el factor  $\frac{1}{\Omega}$ , por el cual también el número de condición de la raíz en la base de Bernstein es relativamente menor que en la base monomial, tanto por el cual los errores relativos de los coeficientes son amplificados en la conversión potencia-a-Bernstein (4.8). En el peor caso, la pobre condición de las bases de conversión esencialmente anula el mejoramiento en la condición de la raíz obtenida en la base de Bernstein.

Concluimos que las conversiones explícitas de la base potencia-a-Bernstein a través de (4.8) no son una vía significativa para el mejoramiento de la estabilidad numérica. En su lugar, se recomienda que los problemas deben ser expresados en la forma de Bernstein inicialmente, y deberían formularse algoritmos para aceptar las formas de Bernstein en las entradas y genera salidas sobre ellos. (ver Farouki y Rajan '88).

## 4.4 Propiedades de Transformación de los Números de Condición

Mostraremos dos importantes propiedades de la transformación de los números de condición de (1.6) y (1.8).

- 1.- Tenemos la base  $\{\phi_k(x)\}$  la cual es linealmente independiente. Entonces generamos una nueva base dado por:

$$\tilde{\phi}_k(x) = s_k \phi_k(x), \quad k = 0, 1, \dots, n, \quad (4.27)$$

donde los  $\{s_k\}$  son factores escalares arbitrarios diferentes de cero. Los coeficientes de (1.2) cambiarían simplemente por:

$$\tilde{\lambda}_k = \frac{1}{s_k} \lambda_k, \quad k = 0, 1, \dots, n. \quad (4.28)$$

Entonces la suma que caracteriza los números de condición de (1.6) y (1.8) es la misma en la nueva base que en la base original. Equivalentemente, podemos decir que el número de condición de una cierta raíz de un polinomio dado, depende solamente de la configuración de la raíz de las funciones bases en las cuales está expresado.

- 2.- Supongamos que tenemos que ver con raíces en un intervalo arbitrario  $[\alpha, \beta]$  de un polinomio  $P(x)$  dado por coeficientes  $\{\lambda_k\}$  en la base  $\{\phi_k\}$

en el intervalo. Si mapeamos el intervalo  $x \in [\alpha, \beta]$  en  $t \in [0, 1]$  a través de la transformación

$$t = \frac{x - \alpha}{\beta - \alpha}, \quad (4.29)$$

y definimos un nuevo conjunto de funciones bases  $\{\Theta_k(t)\}$  en  $[0, 1]$  desde el  $\{\phi_k(x)\}$  por:

$$\Theta_k(t) = \phi_k(\alpha + t(\beta - \alpha)), \quad k = 0, 1, \dots, n. \quad (4.30)$$

Entonces:  $x = \alpha \implies t = 0$ ,  $x = \beta \implies t = 1$ , luego

$$\implies C_{[\alpha, \beta]} = (\beta - \alpha)C_{[0, 1]}, \quad (4.31)$$

donde  $C_{[\alpha, \beta]}$  es el número de condición de una raíz  $x_s$  definido en  $P(x)$  con coeficientes  $\{\lambda_k\}$  en la base  $\{\phi_k(x)\}$  sobre  $[\alpha, \beta]$ , y  $C_{[0, 1]}$  es el número de condición de la raíz  $\varepsilon_0 = \frac{(x_s - \alpha)}{(\beta - \alpha)}$  del polinomio con coeficientes idénticos  $\{\lambda_k\}$  en la base  $\{\Theta_k(t)\}$  sobre  $[0, 1]$ .

Claramente, las sumas en los números de condición (1.6) ó (1.8) son las mismas cuando son calculadas en la base original sobre  $[\alpha, \beta]$  en las bases transformadas sobre  $[0, 1]$ ; el factor  $(\beta - \alpha)$  es debido a las expresiones derivadas multiplicando estas sumas. (ver Farouki y Rajan '87)

## 4.5 Condiciones de las Raíces en las Bases de Potencia y de Bernstein

Comparamos el número de condición (1.6) y (1. ) de las raíces reales simples y múltiples de los polinomios expresados en las bases de potencia y de Bernstein en un intervalo unitario  $[0, 1]$ . Para polinomios arbitrarios, el número de

condición de una raíz  $x_s \in [0, 1]$  en la base de Bernstein es siempre menor o igual que en la base de potencia (será obtenido la igualdad solamente con las raíces en el origen). Eso lo probaremos en los Teoremas 4.1 y 4.2. También consideraremos la generalización de los argumentos para la condición de las raíces reales sobre un intervalo arbitrario  $[\alpha, \beta]$ .

**Teorema 4.1.** (ver Farouki y Rajan '87, pag. 201) Para un polinomio arbitrario  $P(x)$  con una raíz simple  $x_s \in [0, 1]$ , sean  $C_p(x_s)$  y  $C_b(x_s)$  los números de condición de (1.6) la raíz en las bases de potencia y de Bernstein sobre  $[0, 1]$ , respectivamente. Entonces  $C_b(x_s) \leq C_p(x_s) \forall x_s \in [0, 1]$ . En particular  $C_b(0) = C_p(0) = 0$ . Mientras que  $x_s \in ]0, 1[$  tenemos la desigualdad estricta  $C_b(x_s) < C_p(x_s)$ .

**Prueba :** Primero examinemos la condición de una raíz simple  $x_s = 0$  para el origen.

$$S_p(x_s) = |a_0| + |a_1 x_s| + \dots + |a_n x_s^n|$$

$$S_b(x_s) = |c_0(1 - x_s)^n| + |c_1 \binom{n}{1} (1 - x_s)^{n-1} x_s| + \dots + |c_n x_s^n|$$

Resulta:  $S_p(0) = |a_0|$ ,  $S_b(0) = |c_0|$ . Sin embargo, de la transformación de los coeficientes (4.7) y (4.8) se tiene, haciendo  $k = 0$ :  $c_0 = a_0 = 0$  (desde  $x_s = 0$  es una raíz), entonces  $S_p(0) = S_b(0) = 0$ . Considerando nuevas raíces  $x_s$  sobre el intervalo  $]0, 1[$ , obtenemos de la desigualdad triangular para la sumatoria las siguientes relaciones sobre  $x_s \in ]0, 1[$ . Note que los elementos  $\wedge_{jk}$  son definidos no-negativos:

$$S_b(x_s) = \sum_{k=0}^n \left| \sum_{j=0}^k a_j \wedge_{jk} \binom{n}{k} (1 - x_s)^{n-k} x_s^k \right| \leq \sum_{k=0}^n \sum_{j=0}^k |a_j| \wedge_{jk} \binom{n}{k} (1 - x_s)^{n-k} x_s^k.$$

Por la regla de signos de Descartes se tiene que los coeficientes  $a_j$  no pueden ser todos del mismo signo. Arreglando la orden de la sumatoria resulta:

$$S_b(x_s) < \sum_{j=0}^0 |a_j| \wedge_{j0} \binom{n}{0} (1 - x_s)^n + \dots + \sum_{j=0}^n |a_j| \wedge_{jn} \binom{n}{n} x_s^n$$

$$S_b(x_s) = \sum_{j=0}^n |a_j| \sum_{k=j}^n \wedge_{jk} \binom{n}{k} (1-x_s)^{n-k} x_s^k. \quad (4.32)$$

Recordemos que la segunda suma en (4.32) es simplemente la  $j$ -ésima función de la base potencial evaluado en  $x_s$ , y de (4.5) y (4.7) tenemos:

$$S_b(x_s) < \sum_{j=0}^n |a_j| x_s^j = S_p(x_s) \quad \text{para } x_s \in < 0, 1] \quad (4.33)$$

$$|P'(x_s)|^{-1} S_b(x_s) < |P'(x_s)|^{-1} S_p(x_s)$$

$$\implies C_b(x_s) < C_p(x_s). \quad \blacksquare$$

**Ejemplo 4.2.** Dado el polinomio  $P(x) = x^2 - x + 0.25 = (x - \frac{1}{2})^2$  y el polinomio perturbado  $\tilde{P}(x) = x^2 - 1.00001x + 0.25$  y  $\tilde{P}'(x) = 2x - 1.00001$ , donde las raíces son  $\tilde{x}_s = 0.5022$  y  $0.4978$ . Determinemos  $S_p(x_s)$  para  $x_s = 0.5022$ ,  $P'(0.5022) = -0.00439$ , donde los coeficientes no perturbados y perturbados de potencia son  $a_0 = 0.25$ ,  $a_1 = -1.00001$  y  $a_2 = 1$ , y  $\tilde{a}_0 = 0.25$ ,  $\tilde{a}_1 = -1.00001$  y  $\tilde{a}_2 = 1$ . Se tiene que:

$$S_p(0.5022) = |0.25| + |-1.00001(0.5022)| + |1(0.5022)^2| = 1.004$$

Luego el número de condición potencial es  $C_p(0.5022) = \frac{1.004}{0.00439} = 228.8$ . Procedemos a determinar los coeficientes de Bernstein por (4.7), donde:

$$\tilde{c}_0 = \tilde{a}_0 = 0.25$$

$$\tilde{c}_1 = \frac{\binom{1}{0}}{\binom{2}{0}} \tilde{a}_0 + \frac{\binom{1}{1}}{\binom{2}{1}} \tilde{a}_1 = 0.25 + 0.5(-1.00001) = -0.250005$$

$$\tilde{c}_2 = \frac{\binom{2}{0}}{\binom{2}{0}} \tilde{a}_0 + \frac{\binom{2}{1}}{\binom{2}{1}} \tilde{a}_1 + \frac{\binom{2}{2}}{\binom{2}{2}} \tilde{a}_2 = 0.25 - 1.00001 + 1 = 0.24999$$

En forma analoga se calculan los  $c_0 = 0.25$ ,  $c_1 = -0.25$  y  $c_2 = 0.25$ . Luego:

$$\begin{aligned} S_b(0.5022) &= |0.25| \binom{2}{0} (1 - 0.5022)^2 + |-0.25| \binom{2}{1} (1 - 0.5022)0.5022 + \\ &\quad |0.25| \binom{2}{2} (0.5022)^2 \\ &= 0.06195 + 0.1250 + 0.06305 = 0.25 \end{aligned}$$

El número de condición de Bernstein es  $C_b(0.5022) = \frac{0.2500}{0.00439} = 56.9$ . Entonces se tiene que  $C_b(0.5022) < C_p(0.5022)$ .

Las soluciones reales son  $x_1 = x_2 = 0.5$ , los errores relativos son 0.0044 en ambos casos.

El polinomio de Bernstein es  $P(x) = 0.25 \binom{2}{0} (1-x)^2 - 0.25 \binom{2}{1} (1-x)x + 0.25 \binom{2}{2} x^2$ , el perturbado  $\tilde{P}(x) = 0.25 \binom{2}{0} (1-x)^2 - 0.25001 \binom{2}{1} (1-x)x + 0.25 \binom{2}{2} x^2$ ,  $\tilde{P}'(x) = 2.00004x - 1.00002$ . Las raíces son  $\tilde{x}_1 = 0.4978$  y  $\tilde{x}_2 = 0.5022$ ,  $\tilde{P}'(0.5022) = 0.0044$  y  $S_b(0.5022) = 0.25$  entonces  $C_b(0.5022) = 56.8$ , el cual también satisface  $C_b(x_s) < C_p(x_s)$ .

Las soluciones reales son  $x_1 = x_2 = 0.5$ , los errores relativos son 0.0044 en ambos casos. Los errores relativos son practicamente los mismos.  $\square$

A pesar de un cambio de base no se puede remediar el crecimiento no lineal (1.8) del desplazamiento de la raíz con el parámetro perturbación que caracteriza las raíces multiples. Mostramos que el factor constante en (1.8) es sin embargo más pequeño en la base de Bernstein que en la base Monomial.

**Teorema 4.2.** (vea Farouki y Rajan '87, pag. 202) Para un polinomio  $P(x)$  arbitrario con una raíz  $x_s \in [0, 1]$  de multiplicidad  $m \geq 2$ , sean  $C_p^{(m)}(x_s)$  y  $C_b^{(m)}(x_s)$  los números de condición (1.8) de la raíz multiple en las bases de potencia y de Bernstein sobre  $[0, 1]$ , respectivamente. Entonces  $C_b^{(m)}(0) = C_p^{(m)}(0) = 0$ , y para  $x_s \in ]0, 1]$  tenemos la desigualdad estricta  $C_b^{(m)}(x_s) < C_p^{(m)}(x_s)$ .

**Prueba :** El resultado es consecuencia de la inecuación (1.8) y de la prueba del teorema (4.1). Podemos ignorar el factor  $\left\{ \frac{m!}{|P^{(m)}(x_s)|} \right\}^{1/m}$ , que es independiente

de la base, y comparamos las cantidades:

$$\sqrt[m]{S_p(x_s)} \quad \text{y} \quad \sqrt[m]{S_b(x_s)}. \quad (4.34)$$

Hemos probado que  $S_b(0) = S_p(0) = 0$  y donde  $C_b^{(m)}(0) = C_p^{(m)}(0) = 0$ . Para una raíz  $x_s \in ]0, 1[$  tenemos  $S_b(x_s) < S_p(x_s)$  por (4.33). Esta inecuación es preservado en la  $m$ -ésima raíz en cada lado ( $m \geq 2$ ), y por eso  $C_b^{(m)}(x_s) < C_p^{(m)}(x_s)$  para  $x_s \in ]0, 1[$ . ■

**Ejemplo 4.3.** Sea el polinomio  $P(x) = x^2 - 2x + 1$ ,  $P'(x) = 2x - 2$  y  $P''(x) = 2$ , donde las raíces son  $x_s = 1$  de multiplicidad  $m = 2$ , determinemos el  $S_p(x_s)$  para  $x_s = 1$ ,  $P'(1) = 0$ ,  $P''(1) = 2$ , se obtiene  $\frac{2!}{2} = 1$  y donde los coeficientes de potencia son  $a_0 = 1$ ,  $a_1 = -2$  y  $a_2 = 1$ , se tiene que:

$$\sqrt{S_p(1)} = \sqrt{|1| + |-2(1)| + |1(1)^2|} = \sqrt{1 + 2 + 1} = 2.$$

Luego el número de condición para la base de potencia es  $C_p^{(2)}(1) = \frac{2}{1} = 2$ .

Procedemos a determinar los coeficientes de Bernstein por (4.7), donde:

$$\begin{aligned} c_0 &= a_0 = 1, \\ c_1 &= \frac{\binom{1}{0}}{\binom{2}{0}} a_0 + \frac{\binom{1}{1}}{\binom{2}{1}} a_1 = 1 + 0.5(-2) = 0, \\ c_2 &= \frac{\binom{2}{0}}{\binom{2}{0}} a_0 + \frac{\binom{2}{1}}{\binom{2}{1}} a_1 + \frac{\binom{2}{2}}{\binom{2}{2}} a_2 = 1 - 2 + 1 = 0. \end{aligned}$$

Luego:

$$\sqrt{S_b(1)} = \sqrt{|1| \binom{2}{0} (1)^2 + |0| \binom{2}{1} (1)^2 + |0| \binom{2}{2} (1)^2} = \sqrt{1 + 0 + 0} = 1.$$

El número de condición de Bernstein es  $C_b^{(2)}(1) = \frac{1}{1} = 1$ . Entonces se tiene que  $C_b^{(2)}(1) < C_p^{(2)}(1)$ .

Si perturbamos el coeficiente  $a_1$  por 0.0001 del polinomio, se obtiene las nuevas raíces



$x_1 = 0.9968$  y  $x_2 = 1.0032$ , de los cuales el error relativo es 0.0032.

El polinomio de Bernstein es  $P(x) = (1 - x)^2$ , donde para cualquier perturbación que se realizara, se obtendria las mismas raices ya dadas anteriormente.  $\square$

El resultado de los teoremas 4.1 y 4.2 no depende de la elección del intervalo  $[0, 1]$  de ninguna forma esencial, por lo tanto fácilmente podemos probar que los números de condición de la raíz son mejores en la base de Bernstein sobre un intervalo  $[\alpha, \beta]$  que en la generalización de las bases de potencia en ese intervalo. Es decir la expansión de las series de Taylor sobre  $x = \alpha$ , y su número de condición esta dado por  $C_t(x_s) = |P'(x_s)|^{-1} \sum_{j=0}^n |f_j \phi_j(x)|$ , siendo  $f_j = \frac{P^{(j)}(\alpha)}{j!}$  y las funciones  $\phi_j(x) = (x - \alpha)^j$ .

**Teorema 4.3.** (vea Farouki y Rajan '87, pag. 203) Para un polinomio  $P(x)$  arbitrario con una raíz  $x_s$  sobre un intervalo general  $[\alpha, \beta]$ , sean  $C_t(x_s)$  y  $C_b(x_s)$  los números de condición de la raíz en una expansión de la serie de Taylor alrededor de  $x = \alpha$ , y en la base de Bernstein sobre  $[\alpha, \beta]$ . Entonces  $C_b(\alpha) = C_t(\alpha) = 0$ , mientras que para  $x_s \in ]\alpha, \beta]$ , se cumple la desigualdad estricta  $C_b(x_s) < C_t(x_s)$ .

**Prueba :** El polinomio  $P(x)$  esta dado por:

$$P(x) = \sum_{j=0}^n f_j (x - \alpha)^j = \sum_{k=0}^n c_k \binom{n}{k} \frac{(\beta - x)^{n-k} (x - \alpha)^k}{(\beta - \alpha)^n} \quad (4.35)$$

donde  $f_j = \frac{P^{(j)}(\alpha)}{j!}$ . Aplicamos la transformación (4.29) con  $P(x)$  sobre  $[\alpha, \beta]$  dentro el polinomio  $P(t)$  sobre  $[0, 1]$ . De (4.30) la base de Bernstein transforma dentro de la base de Bernstein sobre  $[0, 1]$ , la serie de Taylor de la base de funciones  $\phi_j(x) = (x - \alpha)^j$  se convierte en  $\Theta_j(t) = (\beta - \alpha)^j t^j$ . Ignorando el factor  $(\beta - \alpha)$ , introducido en ambas bases (4.31) por la transformación, la magnitud comparativa del número de condición de la raíz  $x_s$  en la serie de Taylor, cuando se aproxima a  $x = \alpha$  en la base de Bernstein sobre  $[\alpha, \beta]$ , puede ser determinada por la suma.

$$t_0 = \frac{x_s - \alpha}{\beta - \alpha} \implies x_s - \alpha = (\beta - \alpha)t_0$$

$$P(x_s) = \sum_{j=0}^n f_j [(\beta - \alpha)t_0]^j \implies |P(x_s)| \leq \sum_{j=0}^n |f_j| (\beta - \alpha)^j t_0^j = S_t(t_0)$$

Sabemos que  $\beta - x_s = (\beta - \alpha)(1 - t_0)$ , entonces  $\beta - \alpha = \frac{x_s - \alpha}{t_0}$ . Reemplazando obtenemos:

$$P(x_s) = \sum_{k=0}^n c_k \binom{n}{k} \frac{(\beta - x_s)^{n-k} (x_s - \alpha)^k}{(\beta - \alpha)^n} = \sum_{k=0}^n c_k \binom{n}{k} (1 - t_0)^{n-k} t_0^k$$

$$|P(x_s)| = \left| \sum_{k=0}^n c_k \binom{n}{k} (1 - t_0)^{n-k} t_0^k \right| \leq \sum_{k=0}^n |c_k| \binom{n}{k} (1 - t_0)^{n-k} t_0^k = S_b(t_0)$$

El número de condición es independiente de los escalares de la base funcional. Podemos reescribir  $S_t(t_0)$  como una suma sobre términos de la forma  $|a_j|t_0^j$ , donde  $a_j = (\beta - \alpha)^j f_j$  son los coeficientes de la serie de potencia de  $P(t)$ , y son relacionados con los coeficientes de Bernstein  $c_k$  por (4.7) y (4.8). Las inecuaciones derivadas en la prueba del teorema 4.1 pueden ser aplicadas en la suma (4.35) para establecer la prueba del teorema 4.3.

Para  $t_0 = 0$  obtenemos  $x_s = \alpha$ , luego  $S_t(0) = |a_0|$  donde  $a_j = (\beta - \alpha)^j f_j$  donde  $S_b(0) = |c_0|$  entonces  $S_t(0) = S_b(0) = 0$ . Luego:

$$S_b(t_0) = \sum_{k=0}^n |c_k| \binom{n}{k} (1 - t_0)^{n-k} t_0^k$$

$$S_b(t_0) < S_t(t_0) \text{ para } t_0 \in (0, 1]. \blacksquare$$

**Ejemplo 4.4.** Dados el polinomio  $P(x) = (x - 2)^2 + 5(x - 2) + 2$ , y el polinomio perturbado  $\tilde{P}(x) = 1.00001(x - 2)^2 + 5(x - 2) + 2$ , donde  $\tilde{P}'(x) = 2.00002(x - 2) + 5$ . Haciendo  $t = x - 2$  y resolviendo el polinomio perturbado, se tiene  $t_1 = -0.43845$  y  $t_2 = -0.45615$ , por lo tanto  $x_1 = 1.5616 \in [1, 2]$  y  $x_2 = -2.5615 \in [-3, -2]$ . Consideramos  $x_s = x_2$ ,  $\alpha = -3$  y  $\beta = -2$ , además  $f_0 = 2$ ,  $f_1 = 5$  y  $f_2 = 1.00001$  por la transformación

$a_j = (\beta - \alpha)^j f_j$ ,  $j = 0, 1, 2$ , por lo tanto  $a_0 = 2$ ,  $a_1 = 5$  y  $a_2 = 1.00001$ . Resulta

$$S_t(-2.5615) = |1.00001(-2.5615)^2| + |5(-2.5615)| + |2| = 21.3688$$

Luego el número de condición es  $C_t(-2.5615) = 5.1828$ .

Procedemos a determinar los coeficientes de Bernstein por (4.7), donde:

$$\begin{aligned}\tilde{c}_0 &= \tilde{a}_0 = 2 \\ \tilde{c}_1 &= \frac{\binom{1}{0}}{\binom{2}{0}}\tilde{a}_0 + \frac{\binom{1}{1}}{\binom{2}{1}}\tilde{a}_1 = 2 + 0.5(5) = 4.5 \\ \tilde{c}_2 &= \frac{\binom{2}{0}}{\binom{2}{0}}\tilde{a}_0 + \frac{\binom{2}{1}}{\binom{2}{1}}\tilde{a}_1 + \frac{\binom{2}{2}}{\binom{2}{2}}\tilde{a}_2 = 8.00001\end{aligned}$$

En forma analoga se obtiene  $c_0 = 2$ ,  $c_1 = 4.5$  y  $c_2 = 8$ . Luego  $S_b(-2.5615) = 4.3848$  y el número de condición de Bernstein es  $C_b(-2.5615) = 1.0635$ . Entonces se tiene que  $C_b(-2.5615) < C_t(-2.5615)$ .

Las soluciones reales son  $x_1 = 1.5616$  y  $x_2 = -2.5616$ , donde los errores relativos son 0 y 0.00004.

El polinomio de Bernstein es  $P(t) = 2\binom{2}{0}(1-t)^2 + 4.5\binom{2}{1}(1-t)t + 8t^2$ , y el polinomio perturbado es  $\tilde{P}(t) = 2\binom{2}{0}(1-t)^2 + 4.50001\binom{2}{1}(1-t)t + 8t^2$ , con raíces  $x_1 = 1.56156$  y  $x_2 = -2.56167$ ,  $\tilde{P}'(t) = 1.99996t + 5.00002$  con  $\tilde{P}'(-0.43844) = 4.12316$  y  $S_b(1.56156) = 11.3521$ , entonces  $C_b(2.43844) = 2.75325$ . También resulta que  $C_b(x_s) < C_t(x_s)$ .

Las soluciones reales son  $x_1 = 1.5615$  y  $x_2 = -2.5616$ , donde los errores relativos son 0.00004 en ambos casos.  $\square$

Afortunadamente, podemos probar en forma general que las bases de Bernstein exhiben un óptimo condicionamiento de la raíz entre una familia grande (sino de todos) de las bases polinomiales sobre  $[0, 1]$  las cuales estan caracterizadas por simples propiedades.

**Teorema 4.4.** (Farouki y Rajan '87, p. 206) Una base polinomial  $\{\phi_j^n(x)\}$  de grado  $n$  sobre  $[0, 1]$  será llamada una combinación no-negativa de las bases de Bernstein  $\{B_k^n(x)\}$ , si cada  $\phi_j^n(x)$  es una suma ponderada de las  $n + 1$  funciones base de Bernstein con coeficientes no-negativos. Sean  $C_\phi(x_s)$  y  $C_b(x_s)$  los números de condición de cualquier raíz  $x_s \in [0, 1]$  de un polinomio arbitrario  $P(x)$  expresado en la base arbitraria y de Bernstein, respectivamente, entonces  $C_b(x_s) \leq C_\phi(x_s)$ .

**Prueba :** Consideramos una base polinomial  $\{\phi_j^n(x)\}$  definida sobre  $[0, 1]$ , la cual puede ser expresada en la forma:

$$\phi_j^n(x) = \sum_{k=0}^n m_{jk} B_k^n(x) \text{ para } j = 0, 1, \dots, n \quad (4.36)$$

donde  $m_{jk} \geq 0$  para  $j, k = 0, 1, \dots, n$ . Las funciones base de Bernstein son no-negativas sobre  $[0, 1]$ , los  $\{\phi_j^n(x)\}$  son también no-negativas en el intervalo. Un polinomio  $P(x)$  con coeficientes  $\{\lambda_j\}$  en la base  $\{\phi_j^n(x)\}$  puede ser expresado en forma de Bernstein como:

$$P(x) = \sum_{k=0}^n c_k B_k^n(x) \text{ donde } c_k = \sum_{j=0}^n \lambda_j m_{jk}, \quad k = 0, 1, \dots, n \quad (4.37)$$

El número de condición de una raíz  $x_s \in [0, 1]$  en la base  $\{\phi_j^n(x)\}$  es dado por:

$$C_\phi = |P'(x_s)|^{-1} S_\phi(x_s) \text{ donde } S_\phi(x_s) = \sum_{j=0}^n |\lambda_j \phi_j^n(x_s)|. \quad (4.38)$$

Los elementos de la matriz  $\{m_{jk}\}$  son todos no-negativos.

$$\begin{aligned} S_b(x_s) &= \sum_{k=0}^n |c_k B_k^n(x_s)| = \sum_{k=0}^n |c_k| B_k^n(x_s) \\ &\leq \sum_{k=0}^n \sum_{j=0}^n |\lambda_j m_{jk}| B_k^n(x_s) \text{ por (4.37)} \\ &= \sum_{j=0}^n |\lambda_j| \sum_{k=0}^n |m_{jk}| B_k^n(x_s) \end{aligned}$$

$$S_b(x_s) = \sum_{j=0}^n |\lambda_j| \phi_j^n(x_s) = S_\phi(x_s) \text{ (por (4.38))} \quad (4.39)$$

$$|P'(x_s)|^{-1} S_b(x_s) \leq |P'(x_s)|^{-1} S_\phi(x_s)$$

$$C_b(x_s) \leq C_\phi(x_s)$$

La igualdad se cumple cuando la raíz es para  $x_s = 0$  ó  $1$ , donde solamente uno de las cantidades  $\{m_{j0}\}$  y  $\{m_{jn}\}$ , respectivamente, es diferente de cero. ■

**Ejemplo 4.5.** Usando el polinomio del ejemplo (4.5), donde la matriz  $M$  está dado por

$$M = \begin{pmatrix} 1.8 & -2.3 & 2 \\ 4.0 & 3.2 & 10 \\ 3.0 & -1.0 & 5 \end{pmatrix},$$

se obtiene de (4.37)  $\lambda_0 = 0.1518$ ,  $\lambda_1 = 2.3947$  y  $\lambda_2 = 3.1173$  y usando (4.38) se tiene  $S_\phi(0.7887) = 26.993$  con  $\tilde{P}'(0.7887) = 3.4644$ , donde los números de condición son  $C_\phi(0.7887) = 7.7912$  y  $C_b(0.7887) = 0.38487$ . Entonces se tiene que  $C_b(0.7887) < C_\phi(0.7887)$ . □

El número de condición de una raíz en las bases de Bernstein sobre  $[0, 1]$  es más pequeño o igual que en cualquier otra base sobre  $[0, 1]$ . Las funciones bases pueden ser expresadas como una combinación lineal no negativa de las funciones bases de Bernstein. Consideramos ahora la naturaleza de las bases las cuales satisfacen está condición.

**Teorema 4.5.** (Farouki y Rajan '87, p. 207) Una condición suficiente (pero no necesaria) para que las bases  $\{\phi_j^n(x)\}$  sean una combinación no-negativa de las bases de Bernstein sobre  $[0, 1]$  es que cada cada raíz de cada función base  $\phi_j^n(x)$  debe caer fuera del disco abierto  $D$  en el plano complejo el cual tiene al intervalo  $(0, 1)$  como diámetro.

**Prueba :** Desde que  $B_k^n(x) > 0$  para  $k = 0, 1, \dots, n$ ,  $x \in (0, 1)$ , las funciones

dadas en (4.39), donde  $m_{jk} \geq 0$  para  $j, k = 0, 1, \dots, n$ , claramente no tienen raíces sobre  $(0, 1)$  tampoco. Un polinomio  $\phi_j^n(x)$  sin raíces reales sobre  $(0, 1)$  puede ser descompuesto en factores de tres tipos básicos, los cuales están expresados en forma de Bernstein como sigue:

$$x + \zeta = \zeta B_0^1(x) + (1 - \zeta) B_1^1(x), \text{ donde } \zeta \geq 0,$$

$$\eta - x = \eta B_0^1(x) + (\eta - 1) B_1^1(x), \text{ donde } \eta \geq 1,$$

$$\begin{aligned} \{x - (\rho + i\sigma)\} \{x - (\rho - i\sigma)\} &= [\rho^2 + \sigma^2] B_0^2(x) + \left\{ \left(\rho - \frac{1}{2}\right) + \sigma^2 - \left(\frac{1}{2}\right)^2 \right\} B_1^2(x) + \\ & \quad [(\rho - 1)^2 + \sigma^2] B_2^2(x). \end{aligned}$$

Estos son polinomios de Bernstein con coeficientes no-negativos, dado que las raíces conjugadas complejas  $\rho \pm i\sigma$  de cada  $\phi_j^n(x)$ ,  $j = 0, 1, \dots, n$ , fuera del disco abierto  $D$  en el plano complejo sobre el intervalo  $(0, 1)$  como diametro.

Probando lo anterior:  $x + \zeta = x + x\zeta - x\zeta + \zeta = \zeta(1 - x) + (1 + \zeta)x$ .

Como se cumple  $B_0^1(x) = \binom{1}{0}(1-x)^{1-0}x^0 = (1-x)$ ,  $B_1^1(x) = \binom{1}{1}(1-x)^{1-1}x^1$  tenemos  $x + \zeta = \zeta B_0^1(x) + (1 + \zeta) B_1^1(x)$ , donde  $\zeta \geq 0$ .

Luego es  $\eta - x = \eta - x\eta + x\eta - x = \eta(1 - x) + (\eta - 1)x$ , entonces  $\eta - x = \eta B_0^1(x) + (\eta - 1) B_1^1(x)$ , donde  $\eta \geq 1$ .

Finalmente:

$$\begin{aligned} \{x - (\rho + i\sigma)\} \{x - (\rho - i\sigma)\} &= \{(\rho + i\sigma) - x\} \{(\rho - i\sigma) - x\} \\ &= \{(\rho + i\sigma) B_0^1(x) + [(\rho + i\sigma) - 1] B_1^1(x)\} \times \\ & \quad \{(\rho - i\sigma) B_0^1(x) + [(\rho - i\sigma) - 1] B_1^1(x)\} \end{aligned}$$

$$\begin{aligned} \{x - (\rho + i\sigma)\} \{x - (\rho - i\sigma)\} &= (\rho + \sigma) B_0^2(x) + \left\{ \left(\rho - \frac{1}{2}\right) + \sigma^2 - \left(\frac{1}{2}\right)^2 \right\} B_1^2(x) + \\ & \quad \{(\rho - 1)^2 + \sigma^2\} B_2^2(x) \end{aligned}$$

donde:

$$\begin{aligned}
 B_0^1(x)B_0^1(x) &= (1-x)(1-x) = \binom{2}{0}(1-x)^{2-0}x^0 = B_0^2(x) \\
 2B_1^1(x)B_0^1(x) &= 2x(1-x) = \binom{2}{1}(1-x)^{2-1}x^1 = B_1^2(x) \\
 B_1^1(x)B_1^1(x) &= x^2 = \binom{2}{2}(1-x)^{2-2}x^2 = B_2^2(x)
 \end{aligned}$$

Si trabajamos con la conjugada de  $\rho \pm i\sigma$  se obtendra:

$$\begin{aligned}
 \{x - \overline{(\rho + i\sigma)}\}\{x - \overline{(\rho - i\sigma)}\} &= \{x - (\rho - i\sigma)\}\{x - (\rho + i\sigma)\} \\
 &= [\rho^2 + \sigma^2]B_0^2(x) + \{(\rho - \frac{1}{2})^2 + \sigma^2 - (\frac{1}{2})^2\}B_1^2(x) + \\
 &\quad \{(\rho - 1)^2 + \sigma^2\}B_2^2(x)
 \end{aligned}$$

El producto de dos polinomios de Bernstein con coeficientes no-negativos es otro polinomio de Bernstein con coeficientes no-negativo, y por eso vemos que una condición suficiente para los  $\{\phi_j^n(x)\}$ ,  $j=0,1, \dots, n$  puede estar exterior a  $D$ . ■

La condición de arriba no es necesaria como se puede ver en un contraejemplo. Las raíces cúbicas  $x_s = -1$  y  $\frac{1}{2} \pm \frac{4}{9}i$  tienen coeficientes de Bernstein no-negativos, y justamente con otros tres polinomios linealmente independientes con coeficientes de Bernstein no-negativos  $(1, x, x^2)$  forman una base de la forma (4.37), aún cuando las raíces  $\frac{1}{2} \pm \frac{4}{9}i$  están en el interior de  $D$ .

## CONCLUSIONES

En el análisis del punto flotante es recomendable trabajar con números de máquina (en base dos, octal o hexadecimal), debido a que error relativo resulta menor en ese caso.

Debe quedar claro que cuando realizamos procedimientos con polinomios, ya sea para determinar su raíz, o determinar un polinomio aproximante, el error relativo total que se debe considerar, es el error propio más el error del cálculo, lo cual nos indica si el algoritmo utilizado para el polinomio es estable o no.

Se han realizado comparaciones entre polinomios de base potencial y de Bernstein, llegando a la conclusión que es recomendable trabajar en la base de Bernstein directamente antes de usar la transformación de la base de potencia a la base de Bernstein, ya que trabajando directamente en la base de Bernstein se tiene una variedad de algoritmos en los cuales decrece monotomamente el número de condición. A pesar de ello no podemos afirmar que dichos algoritmos son más eficientes, porque tenemos que analizar su aritmética de punto flotante en cada operación que se realiza, la cual depende también del grado del polinomio.

Es preferible trabajar en la base de Bernstein porque las raíces que se obtiene pueden ser transformadas para trabajar sobre el intervalo unitario  $[0, 1]$  lo que facilita el trabajo para los algoritmos. Además, si no son considerados los errores de calculo, la interpretación apropiada de nuestros resultados puede ser expresada de la siguiente manera:

- a) Si es posible derivar ambos, tanto los coeficientes de potencia como de Bernstein para un polinomio de la misma exactitud relativa, la forma de Bernstein es preferible.



- b) Si los coeficientes de Bernstein pueden ser determinados en una exactitud relativa dada en intervalos diferentes, el menor intervalo conocido que contiene todas las raíces de interés es preferible.
- c) Si los coeficientes de Bernstein en un intervalo fijo pueden ser determinados por una exactitud relativa dada en las bases de Bernstein de diferente grado, los coeficientes en las bases de mayor grado son preferibles (asumiendo que el mayor grado del polinomio puede ser procesado con igual exactitud).

Es además conveniente, volver a recalcar que las conversiones explícitas de potencia-a-Bernstein, la subdivisión, y la elevación o reducción de grado todos degradaran la aproximación relativa si se desarrollan en punto flotante. Es entonces deseable adoptar formulaciones algorítmicas las cuales satisfacen los criterios mencionados anteriormente sin considerar en tales procedimientos.

Comparando los algoritmos presentados se tiene que el método usual requiere de  $\frac{n(n+1)}{2}$  multiplicaciones y  $n$  adiciones, el método de Horner con  $n$  multiplicaciones y  $n$  adiciones, la evaluación en la base de Bernstein requiere de  $n(n+1)$  multiplicaciones y  $\frac{n(n+1)}{2}$  adiciones, el método de elevación ó reducción de grado tiene  $O(n)$  operaciones, el método de la subdivisión tiene  $O(n^2)$  operaciones. Todos ellos tienen un grado de dificultad, el cual depende del grado del polinomio y de la evaluación de los cálculos. Si no se tiene en cuenta este último se puede garantizar que el método de subdivisión es el más conveniente de utilizar, en otro caso no se puede garantizar que algoritmo es eficiente.

## BIBLIOGRAFIA

- [1] Burden R., Faires J.: Análisis Numérico. Grupo Editorial Iberoamérica, México, 1985.
- [2] Daniel M., Daubisse J.: The numerical Problem of Using Bézier Curves and Surfaces in the Power Basis, Computer Aided Geometric Design, Vol. 6, pp 121 - 128, 1989.
- [3] Farin G.: Curves and Surfaces for Computer Aided Geometric Design, Third Edition. Academic Press Inc, San Diego, 1992.
- [4] Farouki R., Rajan V.: On the Numerical Condition of Polynomials in Bernstein form, Computer Aided Geometric Design, Vol. 4, pp 191 - 216, 1987.
- [5] Farouki R., Rajan V.: Algorithms for Polynomials in Bernstein form, Computer Aided Geometric Design, Vol. 5, p. 1 - 26, 1988.
- [6] Farouki R.: On the Atability of Transformations Between Power and Bernstein Polynomial forms, Computer Aided Geometric Design, Vol. 8, pp 29 - 36, 1991.
- [7] Kincaid D., Cheney W.: Análisis Numérico, Addison-Wesley Iberoamericana, Estados Unidos, 1994.
- [8] Melvin J., Robert J.: Análisis Numérico, Compañía Editorial Continental, Vol. 1, México, 1995.
- [9] Nieves A., Domínguez F.: Métodos Numéricos, Compañía Editorial Continental, Vol. 1, México, 1995.
- [10] Stoer J., Bulirsch R.: Introduction to Numerical Analysis, Springer-Verlag, Vol. 2, New York, 1993.