Spigot Algorithm

Juan Antonio Rodríguez[1[0009−0001−6727−2321]], Sofia Salinas Rico[1[0009−0008−7938−9747]], and Pawell Steven Torres[1[2222−−3333−4444−5555]]

Universidad Nacional de Colombia http://www.unal.edu.co

Resumen En este informe se abordará el Algoritmo de Spigot, inicialmente se explicarán algunos conceptos matemáticos clave que se emplean en la ejecución del Algoritmo como las Series de Leibniz, un vistazo rápido a la transformación de Euler para acelerar la convergencia de series y las bases de raíz mixta. Luego tratará el algoritmo generalizado y posteriormente su uso para hallar los primeros n términos de la representación decimal de π.

Keywords: Pi · Algoritmo de Spigot · Serie de Leibniz · Bases de Raíz Mixta.

1. Conocimientos Matemáticos Previos

1.1. Series de Aproximación

Las series de aproximación son series numéricas convergentes que, cuando $n \to \infty$, aproximan uniformemente un número real. Presentan aplicaciones notables cuando se emplean para aproximar números trascendentales. Para su determinación, es posible recurrir al uso de funciones y sus respectivas series de Taylor, evaluadas en un valor particular de x. A continuación deducimos las series de aproximación de e y de π. La serie de aproximación de π es un resultado de importancia, tanto es así que es conocida como "Serie de Leibniz".

Serie de Leibniz para $\boldsymbol{\pi}$ A partir de la serie de Taylor para calcular la tangente inversa:

$$tan^{-1}(x) = \sum_{n \in \mathbf{N}} \frac{x^{2n+1}}{2n+1} \tag{1}$$

Se obtiene la serie de Leibniz cuando $x = 1$:

$$tan^{-1}(1) = \frac{\pi}{4} = \sum_{n \in \mathbf{N}} \frac{(-1)^n}{2n+1} \tag{2}$$

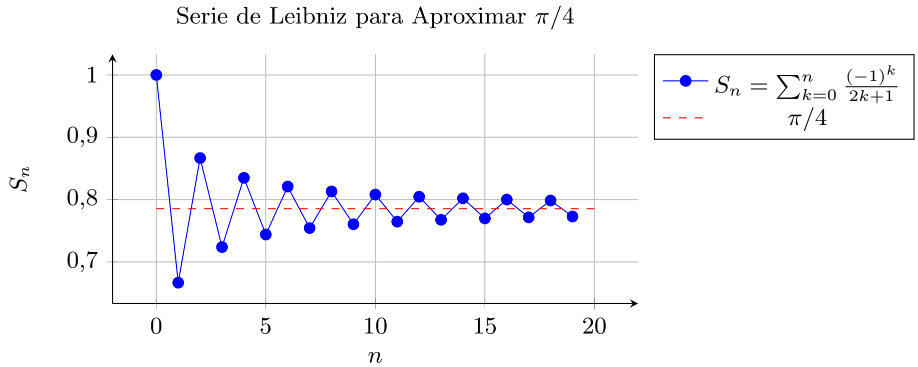

Figura 1. Convergencia de la Serie de Leibniz hacia $\pi/4$

Serie de Aproximación para *e* Podemos aproximar e usando la serie de Taylor para calcular e^x:

$$e^x = \sum_{n \in \mathbb{N}} \frac{x^n}{n!} \tag{3}$$

Se obtiene la serie de aproximación cuando $x = 1$:

$$e = \sum_{n \in \mathbb{N}} \frac{1}{n!} \tag{4}$$

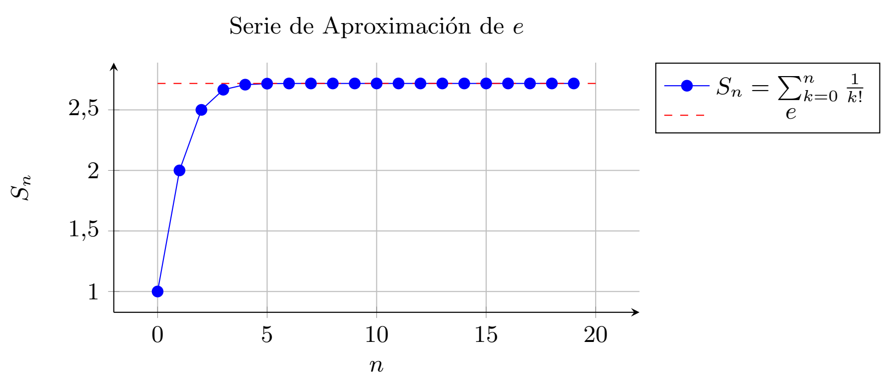

Figura 2. Convergencia de la Serie de Aproximación hacia e

1.2. Bases de Raíz Mixta

Los números reales se expresan comúnmente en base 10 (decimal). Es decir, un número en la forma abc,def, donde cada $a, b, c, d, e, f \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, es equivalente a:

$$a \times 10^2 + b \times 10^1 + c \times 10^0 + d \times 10^{-1} + e \times 10^{-2} + f \times 10^{-3}$$

Esta generalización nos permite expresar cualquier número real en una base distinta de 10. A modo de ejemplo, podemos representar cualquier número en base binaria, es decir, base 2.

Para continuar, considere el número irracional $\sqrt{2} = 1{,}41421356...$ escrito en base decimal. Este número se puede expresar como:

$$\sqrt{2} = 1 + \frac{1}{10}\left(4 + \frac{1}{10}\left(1 + \frac{1}{10}\left(4 + \frac{1}{10}\left(2 + \frac{1}{10}\left(1 + \cdots\right)\right)\right)\right)\right) \tag{5}$$

Podemos definir la base usada como la secuencia $(\frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \dots)$ y los coeficientes entonces corresponderían a la secuencia $(1; 4, 1, 4, 2, 1, \dots)$. Note que el primer elemento de la secuencia de los coeficientes en este caso coincide con la parte entera de $\sqrt{2}$.

Las bases de raíz mixta son entonces aquellas que los términos de la secuencia base son diferentes. En particular, tome la secuencia $b = (\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \dots)$ y consideremos la secuencia de coeficientes que se obtiene para un número a: $(a_0; a_1, a_2, a_3, \dots)$, con cada a_i un número entero no negativo, esto es:

$$a = a_0 + \frac{1}{2}\left(a_1 + \frac{1}{3}\left(a_2 + \frac{1}{4}\left(a_3 + \frac{1}{5}(a_4 + \cdots)\right)\right)\right) \tag{6}$$

Si se tiene que $0 \leq a_i \leq i$ para todo $i \geq 1$ la representación se denomina *regular*. Se denotarán las representaciones de bases de raíz mixta como $(a_0; a_1, a_2, a_3, \dots)_b$, donde b es la secuencia de la base. En particular, para la base $b = (\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \dots)$ todo número real positivo tiene una representación regular y representaciones son únicas si eliminamos aquellas representaciones que terminan con dígitos maximales (por ejemplo, tome $\frac{1}{2}$, se tiene que $\frac{1}{2} = (0; 1, 0, 0, \dots)_b = (0; 0, 2, 3, 4, 5, 6, \dots)_b$). Vamos a excluir estas representaciones de ésta y todas las bases de raíz mixta.

Base de raíz mixta para e Teniendo en cuenta la serie de aproximación de e que se encuentra en la subsección anterior de este documento, la cual es:

$$e = \sum_{n \in \mathbb{N}} \frac{1}{n!} = 2 + \frac{1}{2} + \frac{1}{2 \cdot 3} + \frac{1}{2 \cdot 3 \cdot 4} + \cdots \tag{7}$$

Se puede re-escribir esta expresión usando la forma vista para las bases de raíz mixta como:

$$e = 2 + \frac{1}{2}\left(1 + \frac{1}{2}\left(1 + \frac{1}{3}\left(1 + \cdots\right)\right)\right) \tag{8}$$

Con lo cual se obtiene que $e = (2; 1, 1, 1, \ldots)$. Note que para ésta base $b = (\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \ldots)$, e tiene una representación periódica.

1.3. Aceleración de la convergencia de series, Transformación de Euler

Encontrar una representación en base b para números racionales que no terminan o números irracionales no es tarea sencilla, pues ya se vió con los casos de e que las representaciones son muy diferentes de como lucen en base decimal. Por lo tanto, para π debemos encontrar una representación en bases de raíz mixta para alguna base que no necesariamente sea b que de ser posible, sea periódica (así como e lo fue con la base b). Para ello, vamos a emplear la serie de Leibniz descrita en el inicio de este documento, la cual es:

$$\frac{\pi}{4} = \sum_{n \in \mathbf{N}} \frac{(-1)^n}{2n+1} \tag{9}$$

Y se emplea la Transformación de Euler que se puede apreciar en mayor detalle en [3].

Se define $\Delta x_k = x_k - x_{k+1}$ y $\Delta^n x_k = \Delta^{n-1} x_k - \Delta^{n-1} x_{k+1}$.

Se tiene que la serie alternada convergente

$$\sum_{k=0}^{\infty} (-1)^k a_k \tag{10}$$

Converge al mismo número y tiene la misma suma que la serie

$$\sum_{n=0}^{\infty} \frac{\Delta^n a_o}{2^{n+1}} \tag{11}$$

Representación en base de raíz mixta para π Usando la Transformación de Euler y la Serie de Leibniz para π, se deduce la siguiente expansión (más a detalle en la página 256 de [3]):

$$\frac{\pi}{4} = \frac{1}{2}\left(1 + \frac{1}{3} + \frac{1 \cdot 2}{3 \cdot 5} + \frac{1 \cdot 2 \cdot 3}{3 \cdot 5 \cdot 7} + \cdots\right) \tag{12}$$

De la que podemos obtener la representación en base de raíz mixta para π, con base $c = (\frac{1}{3}, \frac{2}{5}, \frac{3}{7}, \frac{4}{9}, \cdots)$, entonces $\pi = (2; 2, 2, 2, \cdots)_c$.

$$\pi = 2 + \frac{1}{3}\left(2 + \frac{2}{3 \cdot 5}\left(2 + \frac{3}{3 \cdot 5 \cdot 7}\left(2 + \cdots\right)\right)\right)$$

Sin embargo, algunas de las propiedades que se tienen para la base b no se tienen para la base c. En particular, se tiene que para un número $a = (a_0; a_1, a_2, a_3, \cdots)_c$ la parte entera puede ser a_0 o $a_0 + 1$. Esto genera que la precisión de la representación requiera de más términos, en particular, para poder calcular los primeros n términos de π usando su representación en base c se requieren $\lfloor 10n/3 \rfloor$ dígitos de raíz mixta (coeficientes de la secuencia). (Una explicación más detallada de esto se encuentra en [1] en la sección 6, "Spigot Algorithms" y en el apéndice de la sección 13, "A Spigot Algorithm For The Digits of π" que se encuentra en [2]).

2. Explicación General Del Algoritmo

En general, los algoritmos de Spigot son algoritmos que aproximan los dígitos decimales de un número trascendental (como e o como π). El algoritmo produce los dígitos uno a uno de izquierda a derecha. El algoritmo optimiza el uso de memoria a expensas del número de operaciones que se emplean.

El algoritmo utiliza la representación de un número en una base de raíz mixta (en lo posible que sea la base b) y la división entera (algoritmo de Euclides para $\mathbb{Z}$ de igual manera usa una serie de arreglos para almacenar variables que se van reutilizando a medida que el algoritmo se ejecuta.

A continuación una breve ilustración de cómo funciona el algoritmo:

$$Base = \boxed{\tfrac{c_1}{d_1}, \tfrac{c_2}{d_2}, \tfrac{c_3}{d_3}, \tfrac{c_4}{d_4}, \tfrac{c_5}{d_5}, \tfrac{c_6}{d_6}, \tfrac{c_7}{d_7}, \tfrac{c_8}{d_8}, \tfrac{c_9}{d_9}, \tfrac{c_{10}}{d_{10}}, \tfrac{c_{11}}{d_{11}}}$$

$$\text{Rerpresentación} = \boxed{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}}$$

$$\downarrow \times 10$$

$$10a_1, 10a_2, 10a_3, 10a_4, 10a_5, 10a_6, 10a_7, 10a_8, 10a_9, 10a_{10}, 10a_{11}$$

$$q_{10} \times c_{10} \leftarrow q_{11} \times c_{11} \leftarrow 0$$

$$10a_{10} + q_{11} \times c_{11} = d_{10} \times q_{10} + r_{10} \qquad 10a_{11} + 0 = d_{11} \times q_{11} + r_{11}$$

$$\downarrow \qquad \downarrow$$

$$r_{10} \qquad r_{11}$$

$$\downarrow \times 10$$

$$10r_1, 10r_2, 10r_3, 10r_4, 10r_5, 10r_6, 10r_7, 10r_8, 10r_9, 10r_{10}, 10r_{11}$$

Figura 3. Se muestra meidante el uso de flechas el flujo y las operaciones y procesos que se llevan a cabo durante la ejecución del algoritmo usando una base arbitraria y una representación arbitraria para un número arbitrario.

Vemos que el algoritmo usa la representación en la base de raíz mixta cortada a n elementos (en particular en el diagrama anterior, está cortada a 11 elementos) y va haciendo los cálculos de derecha a izquierda, donde el proceso es el mismo:

1. Los elementos del arreglo a se multiplican por 10.
2. De derecha a izquierda: Se suma el elemento i-ésimo del arreglo a con el elemento i-ésimo del arreglo b (para el primer cálculo de cada iteración, este elemento es 0). Este valor se puede guardar en una variable temporal pues solo se va a usar una vez.
3. Se almacena en la posición i-ésima del arreglo a el resultado de la suma del paso anterior reducido módulo el denominador del elemento i-ésimo de la base de raíz mixta que se esté empleando. (O en otras palabras, el residuo de la división entera de estos valores)
4. Se almacena en la posición i-1-ésima del arreglo b el cociente de la división entera del paso anterior multiplicado por el numerador del elemento i-ésimo de la base de raíz mixta.
5. Cuando se llega al final (o inicio) de los arreglos, el resultado que se obtenga del paso anterior es precisamente el dígito siguiente del número trascendental que se está aproximando (usando la notación del diagrama éste digito sería precisamente $q_1 \times c_1$).

Teniendo en cuenta los pasos anteriores, se puede concluir que la precisión (o número de dígitos decimales) del número trascendental que se está aproximando depende del número de elementos de la representación en la base de raíz mixta y dependiendo la base, las condiciones adicionales que se tengan de acuerdo a ésta.

3. Algoritmo de Spigot para e

El algoritmo de Spigot para e es muy sencillo pues la base que se emplea es la base b y la representación $(2; 1, 1, 1, 1, 1, \dots)_b$. Como se usa la base b la parte entera de e no es calculada pues es brindada inmediatamente por la representación $(2; 1, 1, 1, 1, 1, \dots)_b$, es decir que el algoritmo empieza calculando el primer dígito a la derecha de la coma decimal en adelante.

Aquí podemos ver un poco como funcionaría el algoritmo calculando los primeros digitos decimales de e:

Cuadro 1. Los primeros 5 dígitos de e calculados usando el Algoritmo de Spigot. Adaptado de [2]

Paso	Dígitos	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	$\frac{1}{8}$	$\frac{1}{9}$	$\frac{1}{10}$	$\frac{1}{11}$
1	**2**	1	1	1	1	1	1	1	1	1	1
2		10	10	10	10	10	10	10	10	10	10

Paso	Dígitos	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	$\frac{1}{8}$	$\frac{1}{9}$	$\frac{1}{10}$	$\frac{1}{11}$
3	**7**	+4	+3	+2	+1	+1	+1	+1	+1	+0	–
4		14	13	12	11	11	11	11	11	10	10
5		0	1	0	1	5	4	3	2	0	10
6		0	10	0	10	50	40	30	20	0	100
7	**1**	+3	+0	+3	+9	+6	+4	+2	+0	+9	–
8		3	10	3	19	56	44	32	20	9	100
9		1	1	3	4	2	2	0	2	9	1
10		10	10	30	40	20	20	0	20	90	10
11	**8**	+6	+9	+8	+3	+2	+0	+3	+9	+0	–
12		16	19	38	43	22	20	3	29	90	10
13		0	1	2	3	4	6	3	2	0	10
14		0	10	20	30	40	60	30	20	0	100
15	**2**	+5	+6	+7	+8	+9	+4	+2	+0	+9	–
16		5	16	27	38	49	64	32	20	9	100
17		1	1	3	3	1	1	0	2	9	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

En la tabla anterior, podemos ver el proceso paso a paso que usa el Algoritmo de Spigot para el cálculo de los dígitos de e. En el paso 1, el Algoritmo "carga"la representación del número a aproximar (e) y "carga"la base a usar. En este caso, carga la base b y la representación de $e = (2; 1, 1, 1, 1, 1, \dots)_b$. Como se está usando la base b, la parte entera es determinada por el primer término de la representación (en este caso, 2). Los pasos $2, 6, 10, 14, \dots$ hacen referencia al paso en donde se multiplica la fila anterior por 10. Los pasos $3, 7, 11, 15, \dots$ y $5, 9, 13, 17, \dots$ hacen referencia al cálculo del residuo y el cociente respectivamente (éste último multiplicado por el numerador del respectivo término de la base) del respectivo término de los pasos $4, 8, 12, 14, \dots$ entre el denominador del respectivo término de la base. Los pasos $4, 8, 12, 16, \dots$ hacen referencia al resultado de la suma de las dos filas anteriores.

4. Algoritmo de Spigot para π

El algoritmo de Spigot para π es más complejo que para e, pues la pues la base que se emplea es la base c y la representación $(2; 2, 2, 2, \dots)_c$. Como se usa la base c la parte entera de π no es dada por la representación $(2; 2, 2, 2, \cdots)_c$, es decir que el algoritmo empieza calculando desde la parte entera y luego la parte decimal de izquierda a derecha. De igual manera, como se está usando la base c, la precisión de la aproximación está dada por el número de términos que se toman de la representación. Si queremos aproximar los primeros n dígitos de π debemos tomar la representación con $\lfloor 10n/3 \rfloor$ términos.

Aquí podemos ver un poco como funcionaría el algoritmo calculando los primeros dígitos decimales de π:

Cuadro 2. Los primeros 4 dígitos de π calculados usando el Algoritmo de Spigot. Adaptado de [2]

Paso	Dígitos		$\frac{1}{3}$	$\frac{2}{5}$	$\frac{3}{7}$	$\frac{4}{9}$	$\frac{5}{11}$	$\frac{6}{13}$	$\frac{7}{15}$	$\frac{8}{17}$	$\frac{9}{19}$	$\frac{10}{21}$	$\frac{11}{23}$	$\frac{12}{25}$
1		2	2	2	2	2	2	2	2	2	2	2	2	2
2		20	20	20	20	20	20	20	20	20	20	20	20	20
3	**3**	10	+12	+12	+12	+10	+12	+7	+8	+9	+0	+0	+0	–
4		30	32	32	32	30	32	27	28	29	20	20	20	20
5		0	2	2	4	3	10	1	13	12	1	20	20	20
6		0	20	20	40	30	100	10	130	120	10	200	200	200
7	**1**	13	+20	+33	+40	+65	+48	+98	+88	+72	+150	+132	+96	–
8		13	40	53	80	95	148	108	218	192	160	332	296	200
9		3	1	3	3	5	5	4	8	5	8	17	20	0
10		30	10	30	30	50	50	40	80	50	80	170	200	0
11	**4**	11	+24	+30	+40	+40	+42	+63	+64	+90	+120	+88	+0	–
12		41	34	60	70	90	92	103	144	140	200	258	200	0
13		1	1	0	0	0	4	12	9	4	10	6	16	0
14		10	10	0	0	0	40	120	90	40	100	60	160	0
15	**1**	4	+2	+9	+24	+55	+84	+63	+48	+72	+60	+66	+0	–
16		14	12	9	24	55	124	183	138	112	160	126	160	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

En la tabla anterior, podemos ver el proceso paso a paso que usa el Algoritmo de Spigot para el cálculo de los dígitos de π. En el paso 1, el Algoritmo "carga"la representación del número a aproximar (π) y "carga"la base a usar. En este caso, "carga"la base c y la representación de $\pi = (2; 2, 2, 2, \ldots)_c$. Como se está usando la base c, la parte entera es calculada en la primera iteración. De igual manera, note que el primer término (de izquierda a derecha) de la base se omite, pero es $\frac{1}{10}$. Los pasos $2, 6, 10, 14, \ldots$ hacen referencia al paso en donde se multiplica la fila anterior por 10. Los pasos $3, 7, 11, 15, \ldots$ y $5, 9, 13, \ldots$ hacen referencia al cálculo del residuo y el cociente respectivamente (éste último multiplicado por el numerador del respectivo término de la base) del respectivo término de los pasos $4, 8, 12, 14, \ldots$ entre el denominador del respectivo término de la base. Los pasos $4, 8, 12, 16, \ldots$ hacen referencia al resultado de la suma de las dos filas anteriores. De igual manera, se puede apreciar que como en el caso de π se utiliza la base c y no la base b, los enteros que se utilizan a lo largo del algoritmo son comparativamente mucho mayores que los que se usan para e y la base b.

Referencias

1. Arndt, J., Haenel, C.: Pi - Unleashed. Springer (2001). https://doi.org/10.1007/978-3-642-56735-3
2. Bailey, D.H., Borwein, J.M.: Pi: The Next Generation A Sourcebook on the Recent History of Pi and Its Computation. Springer (2016). https://doi.org/10.1007/978-3-319-32377-0
3. Knopp, K.: Infinite sequences and series. Dover Books on Mathematics, Dover Puublications, first edn. (Jun 1956)