Miscelánea Matemática **56** (2013) 27–39 SMM

Números computables y números normales

Guillermo Morales-Luna
Departmento de Computación
CINVESTAV-IPN, México, D.F.
gmorales@cs.cinvestav.mx

Resumen

Hacemos un recuento de las bases formales del análisis computable y bosquejamos la construcción debida a Turing de conjuntos consistentes de números reales absolutamente normales. A grandes rasgos: un número real es computable si lo es la sucesión de sus dígitos respecto a cualquier base de representación. Un número es normal, respecto a una base, si todos los dígitos aparecen con una misma probabilidad en su expansión en esa base, y es absolutamente normal si es normal respecto a cualquier base. Turing bosquejó una construcción, sin demostrarla por completo, de conjuntos, de medida casi 1, de números absolutamente normales, en el intervalo real unitario, junto con un algoritmo de complejidad doblemente exponencial, para producir números absolutamente normales. Su demostración fue completada en la década pasada. Presentamos aquí las ideas básicas en esa construcción.

1. Introducción

Este artículo pretende recordar a Alan Turing como matemático, en ocasión del centenario de su nacimiento. Turing es bien conocido como lógico, como creador de la computación y como criptólogo. Un célebre texto en el que varios lógicos de primera línea rememoran varias anécdotas de Turing es [**5**]. También, en [**6**], puede verse las facetas de Turing como lógico y como inventor.

Aquí haremos una somera revisión de sus principales resultados en la teoría, prácticamente debida a él, de los números normales.

En la sección 2 presentamos las nociones básicas de los números computables. Estos son aquellos calculables procedimentalmente en su total expansión decimal. Precisamente, al formalizar aquí la noción de procedimiento efectivo es que Turing introduce sus famosas máquinas [**8**]. Luego, en la sección 3 presentamos las funciones computables y las nociones básicas del análisis computable. En la sección 4 recordamos el famoso número Ω de Chaitin [**3**] para ilustrar la noción de incomputabilidad. Finalmente, en la sección 5 presentamos los resultados de Turing respecto a los números normales, de acuerdo a la presentación de Becher *et al.* [**1**] quienes completaron el manuscrito inédito de Turing sobre este tema.

2. Números computables

Sea $b \in \mathbb{N}$ un número natural mayor que 1, entendido como una *base de representación*. El conjunto $[\![0, b-1]\!] = \{0, \ldots, b-2, b-1\}$ es el de los *b-dígitos*. Si $b = 2$ se tiene el sistema de representación *binario*, si $b = 10$ se tiene el *decimal* usual y si $b = 20$ se tiene el de base 20 que se utilizó en Mesoamérica y en la antigua Galia.

Todo real no-nulo $a \in \mathbb{R} - \{0\}$ se expresa mediante una tripleta $(\varepsilon, m, \mathbf{a}) \in \{-1, +1\} \times \mathbb{Z} \times [\![0, b-1]\!]^{\mathbb{N}}$ (aquí $[\![0, b-1]\!]^{\mathbb{N}}$ denota a todas las sucesiones de b-dígitos): Si $\mathbf{a} = (a_n)_{n\in\mathbb{N}}$ entonces

$$a = \varepsilon \sum_{n=0}^{+\infty} a_n b^{m-n}. \tag{1}$$

En esta expresión, ε es el *signo* y el valor m indica prácticamente la posición del *punto decimal* al escribir el número real a en base b.

Convenimos en representar al número $0 \in \mathbb{R}$ mediante la sucesión constante $\mathbf{a} = \mathbf{0}$, la posición $m = 0$ del punto decimal y el signo $\varepsilon = +1$.

El número $a \in \mathbb{R}$ es *computable* si la sucesión de dígitos $\mathbf{a}$ lo es, es decir si la función $n \mapsto a_n$ lo es. Esto significa que hay un algoritmo que dado n como entrada, produce el n-ésimo dígito a_n.

Por ejemplo, Arquímedes, desde el siglo III A.C., describió la manera de aproximar circunferencias por perímetros de polígonos regulares, lo que muestra en la práctica que π es computable. La base e de los logaritmos naturales es también computable viéndola ya sea como límite de una sucesión de números racionales o como una serie rápidamente convergente, ambas maneras siempre figurando en cualquier texto de cálculo superior.

Todo algoritmo queda codificado mediante un *programa* el cual es una cadena de símbolos, bien formada de acuerdo con alguna sintaxis, y dotada de una connotación procedimental. Un programa que calcule a la función $n \mapsto a_n$ correspondiente a un número computable $a \in \mathbb{R}$ es una *descripción finita* de la sucesión (potencialmente) infinita de dígitos de a. De esta forma, un número computable es un objeto con una descripción potencialmente infinita (su sucesión de dígitos) pero también con una descripción finita (el programa que lo calcula).

La proposición siguiente da varias caracterizaciones de los números computables [**9**]:

Proposición 2.1. *Para todo $a \in \mathbb{R}$ las siguientes aseveraciones son equivalentes a pares:*

- *a es computable.*
- *Existe una sucesión computable de racionales $(q_n)_{n\geq 0}$ que converge rápidamente a a, es decir,*

$$\forall n \in \mathbb{N}: \quad |a - q_n| < 2^{-n}.$$

- *Existen dos sucesiones computables de racionales, $(x_n)_{n\geq 0}$, $(y_n)_{n\geq 0}$, la primera creciente y la segunda decreciente, tales que*

$$\lim_{n\to+\infty} x_n = a = \lim_{n\to+\infty} y_n.$$

- *El conjunto $\mathbb{Q}\cap]-\infty, a]$ es decidible.*
- *Existe un entero $a_0 \in \mathbb{Z}$ y un conjunto decidible $A \subset \mathbb{N}$ tal que $a = a_0 + \sum_{j\in A} 2^{j+1}$.*
- *a posee una expansión como fracción continuada que es computable, es decir, existe un entero $a_0 \in \mathbb{Z}$ y una función computable $f : \mathbb{N} \to \mathbb{Z}^+$ tal que*

$$a = a_0 + \cfrac{1}{f(0) + \cfrac{1}{f(1) + \cfrac{1}{f(0)+\cdots}}}$$

Sea NUMCOMP $\subset \mathbb{R}$ la colección de números computables. Esta colección contiene a todos los racionales y es un campo con las operaciones usuales. En la práctica, contiene a todos "los números útiles en la ingeniería":

Proposición 2.2 (Propiedades de numerabilidad)**.**

- *Todo número racional es computable:* $\mathbb{Q} \subset \text{NumComp}$.
- *Todo número algebraico es computable.*
- *Toda aritmética de punto flotante está contenida en* NumComp.
- *Solo hay una cantidad numerable de números computables:*

$$\text{card}\,(\text{NumComp}) = \aleph_0.$$

- *La gran mayoría de los números irracionales no son computables.*

A grandes rasgos, esto se debe a que la colección de programas es numerable en tanto que la recta real no lo es.

Desde el punto de vista algebraico, NumComp es un campo que es una extensión de $\mathbb{Q}$ de grado numerable, y es también un subcampo de $\mathbb{R}$.

Proposición 2.3 (Propiedades algebraicas)**.**

- NumComp *es un campo. Es una extensión de* $\mathbb{Q}$ *y es un subcampo de* $\mathbb{R}$.
- NumComp *es un campo real cerrado, es decir, si* $p(X)$ *es un polinomio con coeficientes en* NumComp*, toda raíz de él en* $\mathbb{R}$ *ha de estar también en* NumComp*, o sea rige la implicación siguiente:*

$$p(X) \in \text{NumComp}[X]\,,\, x \in \mathbb{R}\,,\, p(x) = 0 \implies x \in \text{NumComp}.$$

- *Existe un conjunto numerable* $R \subset \mathbb{R}$ *tal que* $\text{NumComp} = \mathbb{Q}[R]$ *(aquí,* $\mathbb{Q}[R]$ *denota la extensión de campo de* $\mathbb{Q}$ *al adjuntarle los elementos de* R*).*

En efecto, tomando una enumeración de $\mathbb{Q}$, seleccionando algorítmicamente números distintos en el n-ésimo dígito del n-ésimo número racional se puede tener una sucesión de números computables fuera de $\mathbb{Q}$ para formar al conjunto R.

Esta argumentación se puede utilizar también para mostrar que existen números reales no-computables.

Desde el punto de vista topológico, NumComp satisface la siguiente:

Proposición 2.4 (Propiedades topológicas)**.**

- NumComp *es un espacio topológico metrizable con la topología usual inducida por* $\mathbb{R}$*.*
- NumComp *no es completo.*
- *Sea* $U_C = [0,1] \cap$ NumComp*. Entonces:* U_C *posee la topología de Cantor, es decir, la topología producto en* $\{0,1\}^{\mathbb{N}}$*.*

Para examinar nociones fuertes de continuidad, nos permitimos recordar que $\rho(X,Y) = \frac{1}{2}(X+Y)(X+Y+1)$ es un polinomio de grado 2 que define de hecho una biyección $\mathbb{N}^2 \to \mathbb{N}$, llamada *enumeración de Cantor*.

Una sucesión real $(a_n)_{n\geq 0}$ se dice ser *computable* si existe una sucesión computable de racionales $(q_k)_{k\geq 0}$ tal que

$$\forall n, m \in \mathbb{N} : \ |a_n - q_{\rho(n,m)}| < 2^{-m}.$$

De acuerdo con la proposición 2.2, NumComp es numerable. Independientemente de cualquier numeración que se defina en NumComp, vale la siguiente:

Proposición 2.5. NumComp *no puede ser una subsucesión de ninguna sucesión computable.*

Esto se demuestra mediante un argumento *de tipo diagonal* y lo que indica es que NumComp es un supremo en la clase de conjuntos computables.

Si $(a_n)_{n\geq 0}$ es una sucesión real computable y convergente a un número real $a_\infty \in \mathbb{R}$ se dice que una función $\nu : \mathbb{N} \to \mathbb{N}$ es un *módulo de convergencia* si vale la implicación:

$$\forall m \in \mathbb{N}\ \forall n \in \mathbb{N}\ \left[n > \nu(m) \implies |a_n - a_\infty| < 2^{-m}\right].$$

Se dice que la sucesión $(a_n)_{n\geq 0}$ *converge computablemente* si converge y posee un módulo de convergencia que es computable, es decir, hay una máquina de Turing que dado un número natural, produce el valor del módulo de convergencia en ese número.

La siguiente propiedad es similar a la que se obtiene cuando se considera convergencia uniforme de sucesiones de funciones:

Proposición 2.6. *El límite de una sucesión computable que converge computablemente es computable.*

3. Funciones computables

Una función de los enteros $f : \mathbb{N}^n \to \mathbb{N}$ es *computable* si lo es por una máquina de Turing. Obviamente en esto hay una función de identificación del conjunto de los números naturales con las sucesiones finitas de dígitos, $\mathbb{N} \approx [\![0, b-1]\!]^*$.

Pasando a funciones definidas en el campo de los números reales, se dice que una función real $f : \mathbb{R} \to \mathbb{R}$ es *computable* si para cada $x \in$ NUMCOMP, $k, i \in \mathbb{N}$ es posible producir procedimentalmente sendos $p, q \in \mathbb{Q}$ tales que $|x - p| < 2^{-i}$ y $|f(x) - q| < 2^{-k}$.

Así, por ejemplo, si $f($NUMCOMP$) \subseteq$ NUMCOMP entonces f es computable.

Proposición 3.1. *Las funciones aritméticas más usuales en ingeniería son computables.*

Proposición 3.2. *Toda función computable es continua.*

De hecho esta aseveración puede reforzarse.

Para cada $n \in \mathbb{N}$, sea $(B_n(j))_{j\geq 0}$ una enumeración de las bolas abiertas en $\mathbb{R}^n$ con centro en puntos de coordenadas racionales y con radio racional.

Para una función $f : \mathbb{R}^n \to \mathbb{R}$ sea

$$R_f = \{k \in \mathbb{N} |\ \exists i, j \in \mathbb{N} :\ [k = \rho(i, j)\ \&\ f(B_n(j)) \subset B_1(i)]\}\,.$$

Es decir, a grandes rasgos, R_f codifica a "parejas" de esferas: En cada una, la primera componente está en el dominio de f, la segunda en su contradominio y la imagen bajo f de la primera viene a ser un subconjunto de la segunda. La función f es *efectivamente continua* si R_f es semidecidible (recursivamente enumerable) y

$$\forall x \in \mathbb{R}^n \forall \varepsilon > 0\ \exists k = \rho(i, j) \in R_f :\ x \in B_n(j)\ \&\ \text{radio}(B_1(i)) < \varepsilon. \quad (2)$$

Observamos aquí que la pareja de esferas $(B_1(i), B_n(j))$ depende del punto x: Dados $x \in \mathbb{R}^n$ y $\varepsilon > 0$ es posible encontrar sendas esferas $B_n(j) \subset \mathbb{R}^n$ (que será una vecindad de x) y $B_1(i) \subset \mathbb{R}$ de manera que $\rho(i, j) \in R_f$ (por lo que $B_1(i)$ ha de ser una vecindad de $f(x)$) y el radio de $B_1(i)$ no exceda ε. Es por esto que está apareciendo la conjunción lógica "&" en (2) en vez de la implicación "$\Rightarrow$" como en la definición de continuidad en el sentido usual (en la que el cuantificador $\forall x$ cambia de lugar respecto al cuantificador existencial).

Proposición 3.3. *Una función es computable si y solo si es efectivamente continua.*

Un curso muy recomendable sobre análisis computable es [**9**]. En línea está, del mismo autor, [**10**] donde el lector puede ver las técnicas de análisis en el contexto de computabilidad.

4. Incomputabilidad

Sea $\Pi = (\pi_n)_{n\in\mathbb{N}}$ una enumeración de los programas. La función $U_\Pi : (n,m) \mapsto U_\Pi(n,m) = \pi_n(m)$ que en cada pareja (n,m) de enteros evalúa al n-ésimo programa en m, se dice ser *universal* para el esquema Π. Como un resultado de computabilidad, se tiene que en cualquier esquema de cómputo efectivo, la función universal es también computable. Así, en el marco de las máquinas de Turing, la *máquina universal de Turing* evalúa a cualquier máquina de Turing en cualquier entrada. En una pareja (n,m), se ve a n como un *programa* y a m como su *entrada*, de donde se obtiene la clásica *arquitectura de von Neumann*.

Para dos índices $m, n \in \mathbb{N}$, se escribe $\pi_n(m)\downarrow$ para denotar que el n-ésimo programa *se para* (o *converge*) cuando actúa sobre la entrada m y, en sentido contrario, $\pi_n(m)\uparrow$ para denotar que *no se para* (o *diverge*). La llamada *paradoja del mentiroso* ("¿Te miento cuando te digo que te miento?") se codifica en el ámbito de la computabilidad con el *problema de la palabra*: ¿Es computable la función característica del conjunto $\{(n,m)|\ \pi_n(m)\downarrow\}$ (dominio de la función universal)? Turing mismo demostró, en [**8**], que no lo es. En otras palabras, Turing mostró que el problema de la parada no es efectivamente resoluble.

Sea $\Pr : \mathcal{P}(\mathbb{N}) \to [0,1]$, $A \mapsto \Pr(A) = \sum_{a\in A} 2^{-(a+1)}$. Esta es una medida de probabilidad.

Proposición 4.1 (Chaitin)**.** *Sea* $\Omega = \{n \in \mathbb{N}|\ \pi_n(n)\downarrow\}$ *el conjunto de índices de programas que se paran con sus propios índices, y sea* $\omega = \Pr(\Omega)$*. Entonces* $\omega \notin$ NumComp.

Demostración. En efecto, esto resulta de la irresolubilidad del problema de la parada. □

Desde los años 60, *Chaitin* (1947–) ha estudiado ω [**4**, **3**].

Un objeto será tanto más *determinado* cuanto sea más corta la ley que lo describe. Será tanto más *azaroso* cuanto sea más larga la ley que lo describe.

El número real ω es pues lo más azaroso posible porque una descripción de ese número solo puede consistir de una transcripción de él mismo: Aunque se haya calculado los primeros n bits en ω, no se

podría tener certeza alguna sobre cuál ha de ser el siguiente bit. ω es absolutamente irreducible.

Para mostrar la noción de problema incomputable, veamos, de manera alternativa al siempre citado *problema de la parada*, un problema de suma importancia en computación, el de *optimización de programas*.

Digamos que un programa es *elegante* si es uno de la menor longitud entre todos los que son equivalentes a él.

Entonces, por el *principio del buen orden*, toda función computable posee un programa elegante que la calcula.

Proposición 4.2. *No se puede determinar cuándo se tiene un programa elegante.*

Demostración. En efecto, supongamos por un momento que sea computable la función $E : \mathbb{N} \to \{0, 1\}$ tal que $[E(n) = 1 \Leftrightarrow \pi_n \text{ es elegante}]$. Sea $F : \mathbb{N} \to \mathbb{N}$,

$$n \mapsto F(n) = \text{ Arg Min}\{m > n | \text{ long}(\pi_m) > \text{long}(\pi_n) \ \& \ E(m) = 1\}$$

y sea $G : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, $(n, m) \mapsto G(n, m) = U(F(n), m)$. Entonces G ha de ser también computable.

Sea $n_G \in \mathbb{N}$ un índice de G: π_{n_G} calcula G.

Sea $n_0 = F(n_G + k)$. Entonces $\pi(n_0)$ debe ser elegante, equivalente a $m \mapsto G(n_G + k, m)$, pero éste último es más corto que $\pi(n_0)$ para una elección adecuada de k.

Esta contradicción muestra que E no puede ser computable. □

Consecuencia técnica No existe un procedimiento general para optimizar programas. La optimización de programas ha de ser un trabajo artesanal.

5. Números normales

Un número normal es tal que toda secuencia de dígitos tiene la misma probabilidad de aparecer en la expansión del número. Si se piensa a los dígitos como "resultados posibles de un juego" (por ejemplo soles y águilas que caen al echar volados, o ganancias en tiradas consecutivas de un juego de azar) entonces un número normal sería el historial de un juego perfectamente legal, en donde ninguna secuencia predeterminada de salidas tiene prioridad, es decir, todas las posibilidades de salidas son equiprobables.

La noción de números normales apareció al final de la primera década del siglo XX [**2**] y ocupó la atención de matemáticos como E. Borel o W. Sierpiński [**7**]. Para precisar esta noción es necesario recurrir a varios tecnicismos.

Sea $b \in \mathbb{Z}^+$, $b > 1$, una base de representación. Expresemos a cada real $a \in \mathbb{R}$ de la forma (1), $a = \varepsilon \sum_{\nu=0}^{+\infty} a_\nu b^{m-\nu}$. Si $a \in]0,1[$, entonces $\varepsilon = 1$, $m = 0$, $a_0 = 0$, por lo que a se identifica propiamente con la sucesión $\mathbf{a} = (a_\nu)_{\nu\geq 1} \in [\![0, b-1]\!]^{\mathbb{Z}^+}$. Para cada $n \in \mathbb{Z}^+$, sea $\mathbf{a}|_n = (a_\nu)_{\nu=1}^n$ el prefijo de longitud n de la sucesión $\mathbf{a}$.

Una palabra $\sigma \in [\![0, b-1]\!]^r$ *aparece en la posición i de a* si vale la igualdad

$$\sigma = a_i a_{i+1} \dots a_{i+r-1},$$

es decir, si $\mathbf{a}|_{i+r-1} = \mathbf{a}|_{i-1} \star \sigma$.

Para cada $n \in \mathbb{Z}^+$, sea $N(n; a, b, \sigma)$ el número de veces que aparece σ en el prefijo $\mathbf{a}|_n$, aún cuando haya superposiciones de esas apariciones, y $\Pr(n; a, b, \sigma) = \frac{1}{n} N(n; a, b, \sigma)$ la *frecuencia*, o *probabilidad*, con la que aparece.

Un número real $a \in \mathbb{R}$ es *b-normal* si

$$\forall \sigma \in [\![0, b-1]\!]^* : \quad \lim_{n\to+\infty} \Pr(n; a, b, \sigma) = b^{-r} \quad \text{con} \quad r = \log(\sigma)\,. \tag{3}$$

a es *absolutamente normal* si es b-normal para cada $b \geq 2$.

a es *simplemente b-normal* si vale la relación (3) considerando solamente palabras de longitud $r = 1$, es decir, considerando solo a los b-dígitos.

Por ejemplo, sea σ_b una palabra formada como una permutación de los b-dígitos. Entonces

$$s_b = \sum_{k\geq 1} \sigma_b b^{-kb} = \sigma_b \sum_{k\geq 1} (b^b)^{-k}$$

es simplemente b-normal, pero no necesariamente simplemente b'-normal para otra base $b' \neq b$.

Proposición 5.1 (Borel)**.** *Un real $a \in]0,1[$ es absolutamente normal si es simplemente b-normal para cualquier base b de representación.*

Turing plantea la construcción de conjuntos en el intervalo $[0,1]$, de números absolutamente normales, cuyas medidas de Lebesgue son cercanas a 1.

Esto lo hace en un manuscrito inédito e incompleto, publicado faccimilarmente en 1992 y resanado, desde el punto de vista lógico y matemático, por Becher en 2007 [**1**]. Vamos a bosquejar aquí esta construcción.

Se define a los operadores siguientes:

- Para dos palabras $\tau, \sigma \in [\![0, b-1]\!]^*$ sea $g(\tau, \sigma)$ el número de veces que σ aparece en τ.
- Para una palabra $\sigma \in [\![0, b-1]\!]^*$, un entero $k \in \mathbb{N}$, y una longitud $n \in \mathbb{N}$, consideremos el conjunto de palabras de longitud n en las que σ aparece k veces:

$$H(\sigma, k, n) = \{\tau \in [\![0, b-1]\!]^n |\ g(\tau, \sigma) = k\}$$
$$\text{y} \quad h(\sigma, k, n) = \operatorname{card}\left(H(\sigma, k, n)\right).$$

Resulta claro que si $\sigma = j \in [\![0, b-1]\!]$ es un solo dígito, entonces

$$h(j, k, n) = \binom{r}{k}(b-1)^{n-k},$$

y en consecuencia

$$\sum_{k=0}^{n} h(j, k, n) = b^n.$$

Lema 5.1 (Indemostrado de Turing)**.** *Sea $\sigma \in [\![0, b-1]\!]^*$ una palabra de dígitos y sea $r = \log(\sigma)$. Sea $t \in \mathbb{R}$ tal que $t\frac{b^r}{n} < \frac{3}{10}$. Entonces:*

$$\sum \left\{ h(\sigma, k, n) |\ \left| k - \frac{n}{b^r} \right| > t \right\} < 2b^n \exp\left(-\frac{t^2 b^r}{4n} \right). \tag{4}$$

Como se ve, este lema indemostrado es una forma de la Ley de los Grandes Números: el lado derecho de (4) tiende a ser cero rápidamente cuando t se incrementa, y del lado izquierdo tenemos que, cuando k difiere más que en t del "promedio de subpalabras" de longitud r en una de longitud n, entonces la probabilidad de que σ aparezca k veces es aún menor. Solo cuando k esté cerca del promedio $\frac{n}{b^r}$, la probabilidad de que σ ocurra k veces es significativa.

Una versión del lema indemostrado de Turing, con cotas menos justas es la siguiente [**1**]:

Lema 5.2 (Becher)**.** *Sea $\varepsilon \in [6/\lfloor n/r \rfloor, b^{-r}]$. Entonces:*

$$\sum \left\{ h(\sigma, k, n) |\ \left| k - \frac{n}{b^r} \right| \geq \varepsilon n \right\} < 2b^{n+2r-2}\ r \exp\left(-\frac{\varepsilon^2 n\, b^r}{6\ r} \right).$$

Ahora, consideremos los conjuntos de la forma

$$G(\sigma, \varepsilon, n, b) = \left\{ a \in [0, 1] |\ \left| N(n; a, b, \sigma) - \frac{n}{b^r} \right| < \varepsilon n \right\},$$

donde $N(n; a, b, \sigma)$, definido al inicio de esta sección, es el número de veces que σ aparece en el prefijo de longitud n de a.

Proposición 5.2. *Si* $\varepsilon \in [6/\lfloor n/r \rfloor, b^{-r}]$, *entonces:*

$$\mu(G(\sigma,\varepsilon,n,b)) > 1 - 2b^{2r-2}\ r\ \exp\left(-\frac{\varepsilon^2 n\, b^r}{6\ r}\right),$$

donde μ *es la medida de Lebesgue en* $\mathbb{R}$.

Puede verse que $G(\sigma,\varepsilon,n,b)$ es una unión finita de intervalos con extremos racionales. Para $b_0, r_0 \in \mathbb{Z}^+$ sea

$$G^*(\varepsilon,n,r_0,b_0) = \bigcap_{b=2}^{b_0} \bigcap_{r=1}^{r_0} \bigcap_{\sigma \in [\![0,b-1]\!]^r} G(\sigma,\varepsilon,n,b).$$

Se tiene que $G^*(\varepsilon,n,r_0,b_0)$ es también una unión finita de intervalos.

Proposición 5.3. *Si* $\varepsilon \in \left[6/\lfloor n/r_0 \rfloor, b_0^{-r_0}\right]$, *entonces:*

$$\mu(G^*(\varepsilon,n,r_0,b_0)) > 1 - 2b_0^{3r_0-1}\ r_0\ \exp\left(-\frac{\varepsilon^2 n}{3\ r_0}\right).$$

Para cada $\kappa \in \mathbb{Z}^+$, tómese $n = \kappa$, $r_0 = \lfloor \frac{1}{4}\sqrt{\ln \kappa} \rfloor$, $b_0 = \lfloor e^{r_0} \rfloor$ y $\varepsilon = b_0^{-r_0}$. Sea

$$G_\kappa = G^*\left((\lfloor \exp(\lfloor \frac{1}{4}\sqrt{\ln\kappa}\rfloor)\rfloor)^{-\lfloor \frac{1}{4}\sqrt{\ln\kappa}\rfloor}\ ,\ \kappa\ ,\ \lfloor \frac{1}{4}\sqrt{\ln\kappa}\rfloor\ ,\ \lfloor \exp(\lfloor \frac{1}{4}\sqrt{\ln\kappa}\rfloor)\rfloor\right).$$

Proposición 5.4. *Existe* $\kappa_0 \in \mathbb{Z}^+$ *tal que* $\forall \kappa \geq \kappa_0$*:* $\mu(G_\kappa) \geq 1 - \frac{1}{\kappa(\kappa-1)}$.

A partir de aquí Turing define una doble sucesión de conjuntos:

$n = 0$. Para $k \in \mathbb{N}$, $A_{nk} =]0,1[$.

$n > 0$. Para $k \in \mathbb{N}$, $A_{nk} = G_{\kappa_0+k+n} \cap A_{n-1,k} \cap [u_{nk}, 1]$, donde $u_{nk} \in [0,1]$ es tal que $\mu(A_{nk}) = 1 - \frac{1}{k} + \frac{1}{n+k}$.

Cada conjunto A_{nk} es la unión finita de intervalos con extremos racionales en $[0,1]$ y por tanto los puntos u_{nk} pueden ser racionales también.

Teorema 5.1 (Turing)**.** *Existe una sucesión computable doble* $(A_{nk})_{n,k\in\mathbb{N}}$ *de conjuntos en* $[0,1]$ *tal que:*

1. *Para cada* $n,k \in \mathbb{N}$, A_{nk} *es la unión finita de intervalos con extremos racionales en* $[0,1]$ *y* $\mu(A_{nk}) > 1 - \frac{1}{k}$.

2. *Para cada* $k \in \mathbb{N}$*, la sucesión* $(A_{nk})_{n\in\mathbb{N}}$ *es decreciente.*
3. *Para cada* $k \in \mathbb{N}$*,* $E_k = \bigcap_{n\in\mathbb{N}} A_{nk}$ *es de medida* $1 - \frac{1}{k}$ *y consta de números absolutamente normales.*

De hecho, éste puede volverse más constructivo:

Teorema 5.2 (Turing)**.** *Existe un procedimiento efectivo tal que dada una sucesión de bits* $\alpha \in [\![0,1]\!]^{\mathbb{Z}^+}$ *y dos enteros* $k, n \in \mathbb{Z}^+$ *produce un número real* $a \in [0,1]$*, absolutamente normal tal que a queda dentro de un conjunto determinado de medida de Lebesgue al menos* $1 - \frac{2}{k}$ *y los primeros n dígitos binarios de a coinciden con los primeros n bits en* α*.*

Este algoritmo es de complejidad doblemente exponencial, como lo es la construcción de Sierpiński [**7**], por lo que es de muy poca utilidad práctica. Si se requiriese el dígito en la posición n habría que realizar un número de operaciones que es proporcional a b^{b^n} y este valor crece muy rápido para $b \geq 2$. Por ejemplo, para $b = 2$ se tendría que del orden de $2^{2^n} = 2^{1024}$ operaciones serían necesarias para calcular el dígito en la posición $n = 10$. Si cada operación se realizara en 2^{-9} segundos, se requeriría del orden de 2^{1015} segundos. Un año consta de aproximadamente 2^{62} segundos, por lo que el cómputo requerido llevaría 2^{953} años: una cantidad inmensa de tiempo, y ¡esto solo para calcular el décimo dígito!

Bibliografía

1. V. Becher, S. Figueira, y R. Picchi, Turing's unpublished algorithm for normal numbers, *Theor. Comput. Sci.* **377** (2007) 126–138.
2. E. Borel, Les probabilités denombrables et leurs applications arithmétiques., *Palermo Rend.* **27** (1909) 247–271.
3. G. Chaitin, *Meta Math! : The Quest for Omega*, Pantheon, octubre 2005.
4. G. J. Chaitin, *The Limits of Mathematics : A Course on Information Theory and the Limits of Formal Reasoning (Discrete Mathematics and Theoretical Computer Science)*, Springer, octubre 2002.
5. J. N. Crossley, Reminiscences of logicians, en *Algebra and Logic: Papers from the 1974 Summer Research Institute of the Australian Mathematical Society*, Springer, 1975, 1–62. Lecture Notes in Math., Vol. 450.
6. A. Hodges, *Alan Turing, Logical and Physical*, no. 2, 3–15, Springer, 2008.
7. M. Sierpiński, Démonstration élémentaire du théorème de M. Borel sur les nombres absolument normaux et détermination d'un tel nombre, *Bulletin de la Société Mathtématiques de France* **45** (1917) 127–132.

8. A. M. Turing, On computable numbers, with an application to the *entscheidungsproblem*, *Proceedings of the London Mathematical Society* **42** (1936) 230–265.
9. K. Weihrauch, *Computable analysis: an introduction*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2000.
10. K. Weihrauch y N. Zhong, Computable analysis of the abstract Cauchy problem in a Banach space and its applications (I), *Electr. Notes Theor. Comput. Sci.* **167** (2007) 33–59.