SOBRE LA CONJETURA DE COLLATZ

Universidad Nacional de Córdoba, Argentina
Lucas Nicolas Depetris, 2022

Introducción.

En el campo de las matemáticas se podría decir que los problemas se dividen en cuatro categorías: aquellos fáciles de enunciar y resolver, difíciles de enunciar pero fáciles de resolver, difíciles de enunciar y resolver, y fáciles de enunciar pero difíciles de resolver. "La matemática es la reina de las ciencias, y la teoría de números es la reina de la matemática" (Gauss). Esta (teoría de números) es la madre creadora de problemas del último tipo, área en la que yace nuestro tema de interés (incluso no resuelto).

El problema $3x+1$, la conjetura de Collatz, el problema de Syracusa o la conjetura de Ulam concierne al comportamiento de las iteraciones de la función $\mathrm{col} : \mathbb{N} \longrightarrow \mathbb{N}$ que asigna a cada número par n el valor $n/2$ y cada impar $3n+1$. Veamos que ocurre si probamos esto con algunos números, empezando digamos por 5, 6, y 11.

$$\begin{tikzcd}
5 \arrow[r] & 16 \arrow[r] & 8 \arrow[r] & 4 \arrow[r] & 2 \arrow[r] & 1 \arrow[ll, bend left] & & \\
6 \arrow[r] & 3 \arrow[r] & 10 \arrow[ull] & & & & & \\
11 \arrow[r] & 34 \arrow[r] & 17 \arrow[r] & 52 \arrow[r] & 26 \arrow[r] & 13 \arrow[r] & 40 \arrow[r] & 20 \arrow[ulllll, bend right]
\end{tikzcd}$$

Como se ve, los tres y todos otros los números involucrados llegan a 1. Más aún, se ha verificado computacionalmente que todo $n \leq 2{,}95 \times 10^{20}$ converge eventualmente al ciclo (1, 4, 2) [3]. Para una mejor visualización podemos construir el siguiente diagrama: Partimos con 1 en la base y vamos ascendiendo. Dado un n en el dibujo, si $n = \mathrm{col}_1(m)$ (m es un predecesor impar de n) o $n = \mathrm{col}_2(\tilde{m})$ ($\tilde{m}$ es un predecesor par de n), entonces escribimos m o $\tilde{m}$ arriba. Notar que estas dos condiciones pueden darse a la vez, por ejemplo $\mathrm{col}_1(3) = 10$ y $\mathrm{col}_2(20) = 10$. En tal caso situamos m a la derecha y $\tilde{m}$ a la izquierda. Luego se une el diagrama con flechas de arriba hacia abajo y da como resultado

$$\begin{tikzcd}
\ddots \arrow[dr] & \ddots \arrow[dr] & \ddots \arrow[dr] & \vdots \arrow[d] & \vdots \arrow[d] & \vdots \arrow[d] & \vdots \arrow[d] & \vdots \arrow[d] & \iddots \arrow[dl] & \iddots \arrow[dl] \\
& 2048 \arrow[dr] & 341 \arrow[d] & 340 \arrow[d] & 336 \arrow[d] & 320 \arrow[d] & 53 \arrow[dl] & 52 \arrow[dl] & 48 \arrow[dl] & \\
& & 1024 \arrow[d] & 170 \arrow[d] & 168 \arrow[d] & 160 \arrow[d] & 26 \arrow[d] & 24 \arrow[d] & & \\
& & 512 \arrow[dr] & 85 \arrow[d] & 84 \arrow[d] & 80 \arrow[d] & 13 \arrow[dl] & 12 \arrow[dl] & & \\
& & & 256 \arrow[d] & 42 \arrow[d] & 40 \arrow[d] & 6 \arrow[d] & & & \\
& & & 128 \arrow[dr] & 21 \arrow[d] & 20 \arrow[d] & 3 \arrow[dl] & & & \\
& & & & 64 \arrow[d] & 10 \arrow[d] & & & & \\
& & & & 32 \arrow[dr] & 5 \arrow[d] & & & & \\
& & & & & 16 \arrow[d] & & & & \\
& & & & & 8 \arrow[d] & & & & \\
& & & & & 4 \arrow[d] & & & & \\
& & & & & 2 \arrow[d] & & & & \\
& & & & & 1 \arrow[uu, bend right] & & & &
\end{tikzcd}$$

Por esto se tiene:

Conjetura de Collatz: Todo natural bajo la iteración de la función col termina en el ciclo (1, 4, 2).

Como el nombre lo indica (y mencionado anteriormente), este es un problema sin resolver. No se sabe a ciencia cierta cuándo fue descubierto por primera vez, aunque el planteo formal del mismo se hizo en 1937 por Lothar Collatz [11]. Extensa investigación se ha realizado al respecto (en [7] se encuentra una recopilación de la bibliografía de esto último) y toda forma sensata de abordaje a este problema hasta la fecha no ha dado resultados con lo que respecta a su solución. En lo computacional se tiene [2], que estudia ciertas cotas en el comportamiento de la función col, y a [6, 8], en donde se desarrollan métodos de "verificación exhaustiva" para la búsqueda de un posible contraejemplo. En lo teorico [1] aborda a la conjetura desde el álgebra lineal, [10] desde el álgebra, [14] desde la topología y [15, 9] lo aproximan como un sistema dinámico. En este documento se mirará al problema desde la aritmética modular y desde la topología, además de mostrar un resultado considerado como el más significativo hasta el momento logrado por Terence Tao.

"Las matemáticas aún no están preparadas para tales problemas" (Paul Erdős), "Es un problema extremadamente difícil, completamente fuera del alcance de las matemáticas actuales" (Jeffrey Lagarias). Estas son algunas citas de matemáticos conocedores del tema. Incluso versiones más débiles de la conjetura están aún sin resolver:

Conjetura de los ciclos finitos: Hay una cantidad finita de números cíclicos. Esto es, la cantidad de naturales x tal que $\mathrm{col}^n(x) = x$ para cierto $n \in \mathbb{N}$ es finita.

Conjetura de la no existencia de una trayectoria divergente: No existe $x \in \mathbb{N}$ tal que $\lim_{n\to\infty} \mathrm{col}^n(x) = \infty$.

Por otra parte cabe mencionar que se tiene la siguiente conjetura sobre una de las generalizaciones de Collatz:

Conjetura de extensión a los enteros: Si se considera a la función de Collatz como $\mathrm{col} : \mathbb{Z} \longrightarrow \mathbb{Z}$, entonces para todo $x \in \mathbb{Z}$ existe $k \in \mathbb{N}$ tal que $\mathrm{col}^k(x)$ pertenece a uno de los ciclos

i. (1, 4, 2), ii. (−1), iii. (0), iv. (−5, −14, −7, −20, −10),
v. (−17, −50, −25, −74, −37, −110, −55, −164, −82, −41, −122, −61, −182, −91, −272, −136, −68, −34).

1. Abordaje con el uso de Aritmética Modular.

Denotamos $\mathrm{col}_1(q) = 3q + 1$ y $\mathrm{col}_2(q) = q/2$. Notar que cuando $n \in \mathbb{N}$ es impar, $\mathrm{col}(n) = 3n + 1$ es par. Esto motiva definir

$$\begin{aligned}
&G_r : \mathbb{Q} \longrightarrow \mathbb{Q}, && G_r(q) = (\mathrm{col}_2^r \circ \mathrm{col}_1)(q) = \frac{3}{2^r}q + \frac{1}{2^r},\\
&L : \bigcup_{k\in\mathbb{N}} \mathbb{N}^k \longrightarrow \mathbb{Q}, && L(n_1, \cdots, n_k) = (G_{n_k}^{-1} \circ G_{n_{k-1}}^{-1} \circ \cdots \circ G_{n_1}^{-1})(1),\\
&\tilde{L} : \bigcup_{k\in\mathbb{N}} \mathbb{N}^k \longrightarrow \mathbb{N}, && \tilde{L}(n_1, \cdots, n_k) = 3^k L(n_1, \cdots, n_k).
\end{aligned}$$

El hecho de que $\tilde{L}$ llega a $\mathbb{N}$ lo vamos a mostrar a continuación. Ahora veremos la importancia de estas funciones. Vamos a usar la notación $N_l = (n_1, \cdots, n_l)$ para $l \in [\![1, k]\!]$ y $N = N_k$. Empecemos calculando L y $\tilde{L}$. Planteamos

$$q = \frac{3}{2^r}G_r^{-1}(q) + \frac{1}{2^r} \implies G_r^{-1}(q) = \frac{q \cdot 2^r - 1}{3},$$

y por lo tanto

$$\begin{aligned}
&G_{n_1}^{-1}(1) = \frac{2^{n_1} - 1}{3}, \qquad (G_{n_2}^{-1} \circ G_{n_1}^{-1})(1) = \frac{\frac{2^{n_1}-1}{3} \cdot 2^{n_2} - 1}{3} = \frac{(2^{n_1} - 1)2^{n_2} - 3}{9},\\
&(G_{n_3}^{-1} \circ G_{n_2}^{-1} \circ G_{n_1}^{-1})(1) = \frac{((2^{n_1} - 1)2^{n_2} - 3)2^{n_3} - 9}{27}, \qquad \cdots\\
&\qquad \cdots, \qquad L(n_1, \cdots, n_k) = \frac{(\cdots((2^{n_1} - 1)2^{n_2} - 3)\cdots)2^{n_k} - 3^{k-1}}{3^k},
\end{aligned}$$

de donde también obtenemos $\tilde{L}$. Por comodidad de escritura, para $k = 0$ tomamos $\tilde{L}(N_0) = 1$ (extensión natural al 0, como en $n!$) y por lo tanto $L(N_0) = 1$. Luego L y $\tilde{L}$ cumplen la siguiente recursividad:

$$\left.\begin{aligned}
&L(N_0) = 1, \quad L(N_l) = \frac{L(N_{l-1}) \cdot 2^{n_l} - 1}{3}\\
&\tilde{L}(N_0) = 1, \quad \tilde{L}(N_l) = \tilde{L}(N_{l-1}) \cdot 2^{n_l} - 3^{l-1}
\end{aligned}\right. \qquad \forall\, l \in [\![1, k]\!].$$

Con esto se tiene la siguiente proposición.

Lema 1.1: Las siguientes afirmaciones son equivalentes:

i. Se cumple Collatz, ii. $\forall\, n \in \mathbb{N}\ \exists\ \{n_i\}_{i=1}^k \subseteq \mathbb{N} : 1 = (G_{n_1} \circ \cdots \circ G_{n_k})(n)$,

iii. $\forall\, n \in \mathbb{N}\ \exists\ \{n_i\}_{i=1}^k \subseteq \mathbb{N} : n = L(n_1, \cdots, n_k)$.

Demostración. (i.) $\Rightarrow$ (ii.) $\mathrm{col}(n) = \mathrm{col}_1(n)$ que es par. Luego

$$\mathrm{col}^{\ell_2(\mathrm{col}_1(n))+1}(n) = (\mathrm{col}_2^{\ell_2(\mathrm{col}_1(n))} \circ \mathrm{col}_1)(n),$$

que es un número impar. Se repite el proceso y como vale Collatz, eventualmente se llega a 1. Además (ii.) $\Rightarrow$ (i.) es claro por la definición de G_r y (ii.) $\Leftrightarrow$ (iii.) también por la definición de L.

$\blacksquare$

Notar que no cualquier elección de $N \in \mathbb{N}^k$ cumple $L(N) \in \mathbb{N}$. Por esto definimos el conjunto $S_k = \{N \in \mathbb{N}^k \mid \tilde{L}(N) \equiv 0\,(3^k)\}$, lo que da lugar a la siguiente equivalencia de la conjetura de Collatz (reformulación del inciso iii. del lema anterior):

Lema 1.2: Collatz $\iff$ $2\mathbb{N} - 1 = \bigcup_{k\in\mathbb{N}} \{L(N)\}_{N\in S_k}$.

Observar que también se podría escribir $\mathbb{N} = \bigcup_{k\in\mathbb{N}} \{L(N) \cdot 2^j\}_{\substack{N\in S_k \\ j\in\mathbb{N}_0}}$ y es equivalente.

Contrario a la intuición, es posible hallar S_k en su totalidad para un k arbitrario. Más aún, cada uno de estos queda completamente descripto por un subconjunto de cardinalidad finita, algo esperable en ecuaciones modulares. Ahora procederemos a dar unos preliminares matemáticos que nos harán posible estudiar S_k.

Prerrequisitos Matemáticos.

Nuestro objetivo en este apartado es ahondar más profundo en el tema de la aritmética módular, desarrollar herramientas que permitan resolver "ecuaciónes exponenciales en congruencia". Sean a, b, $n \in \mathbb{N}$. Denotamos la congruencia: si a es congruente a b módulo n, $a \equiv b\,(n)$. Más aún, si $b \in [\![0,\, n-1]\!]$, $\mathrm{mod}(a,\, n) = b$. Se denota a la clase de equivalencia de a módulo n por $[a]_n = \{b \in \mathbb{Z} \mid b \equiv a\,(n)\}$. El máximo común divisor de a y b: $(a,\, b)$; y el mínimo común múltiplo de a y b: $[a,\, b]$. Dos números se dicen coprimos si su máximo común divisor es 1.

Lema 1.3: Dados a, $n \in \mathbb{N}$, $\exists\ m \in \mathbb{N}$ tal que $a^m \equiv 1(n)$ $\iff$ a y n son coprimos.

Definición 1.1: Sean a, $n \in \mathbb{N}$ coprimos. Se define el "orden de a módulo n" como

$$\mathrm{ord}_n(a) = \mathrm{mín}\{m \in \mathbb{N} \mid a^m \equiv 1(n)\}.$$

Observar que por el lema anterior $\mathrm{ord}_n(a)$ está bien definido.

Lema 1.4: Sean a, n coprimos. $a^k \equiv 1(n)$ $\iff$ $\mathrm{ord}_n(a) \mid k$.

Definición 1.2: Dado $n \in \mathbb{N}$ se define

$$\mathcal{U}_n := \{m \in [\![1,\, n]\!] \mid (a,\, n) = 1\}, \qquad \varphi(n) = |\mathcal{U}_n|,$$

esta última llamada "Función φ de Euler".

Lema 1.5: *Propiedades de la función φ de Euler.* Sean $m,\, n,\, k \in \mathbb{N}$, p primo. Se cumplen

$$\text{i.} \quad \varphi(p^k) = p^k(1 - 1/p), \qquad \text{ii.} \quad \varphi(mn) = \varphi(m)\varphi(n)\frac{(m,\, n)}{\varphi((m,\, n))},$$

$$\text{iii.} \quad \varphi(n^m) = n^{m-1}\varphi(n), \qquad \text{iv.} \quad \varphi(n) = n \prod_{\substack{p|n \\ p \text{ primo}}} \left(1 - \frac{1}{p}\right),$$

$$\text{v.} \quad \sum_{d|n} \varphi(d) = n, \qquad \text{vi.} \quad \varphi(n) \text{ es par } \forall\ n \geq 3.$$

Teorema 1.1: *(Euler, generalización del Pequeño Teorema de Fermat).* Sean $a,\, n \in \mathbb{N}$ coprimos. Se cumple $a^{\varphi(n)} \equiv 1(n)$.

Corolario 1.1: Sean $a,\, n \in \mathbb{N}$ coprimos. Entonces $\mathrm{ord}_n(a) \mid \varphi(n)$.

Definición 1.3: i. Dados $r,\, n \in \mathbb{N}$ coprimos, se dice que r es una "raíz primitiva módulo n" si $\mathrm{ord}_n(r) = \varphi(n)$.

ii. Denotamos al conjunto de las raíces primitivas módulo n como $\mathrm{PR}(n)$.

Observación 1.1: i. Si $r \in \mathrm{PR}(n)$, entonces $\{1,\, r,\, \cdots,\, r^{\varphi(n)-1}\} \equiv \mathcal{U}_n\,(n)$. Esto es, cada elemento del conjunto de la izquierda es congruente módulo n a algún elemento del conjunto de la derecha.

En efecto, como r y n son coprimos se tiene que $(r^k,\, n) = 1$ para $k \in [\![0,\, \varphi(n)-1]\!]$, $|\{r^k\}_{k=0}^{\varphi(n)-1}| = \varphi(n)$ y para $i,\, j \in [\![0,\, \varphi(n) - 1]\!]$, $i \neq j$ se tiene $r^i \not\equiv r^j(n)$. De esta forma $\{\mathrm{mod}(r^k,\, n)\}_{k=0}^{\varphi(n)-1} \subseteq [\![1,\, n]\!]$ es un conjunto de cardinal $\varphi(n)$ y sus elementos son coprimos con n, luego es igual a $\mathcal{U}_n$.

ii. $r \equiv \tilde{r}(n) \implies \mathrm{ord}_n(r) = \mathrm{ord}_n(\tilde{r})$.

Ejemplo 1.1: i. $n = 13$, $\varphi(13) = 12$. Por el corolario anterior los posibles órdenes módulo 13 son 1, 2, 3, 4, 6, y 12. 1 no puede ser raíz primitiva pues $\mathrm{ord}_{13}(1) = 1$. Veamos si 2 lo es.

$$2^1 \equiv 2\,(13), \quad 2^2 \equiv 4\,(13), \quad 2^3 \equiv 8\,(13), \quad 2^4 \equiv 3\,(13),$$
$$2^6 = 2^4 \cdot 2^2 \equiv 12\,(13), \quad \text{y} \quad 2^{12} = 2^{\varphi(13)} \equiv 1\,(13),$$

con lo que $\mathrm{ord}_{13}(2) = 12 = \varphi(13)$. Se sigue que 2 es una raíz primitiva módulo 13.

ii. $n = 8$, $\varphi(8) = 4$. Los posibles órdenes módulo 8 son 1, 2, 4 y los candidatos a raíces primitivas son 1, 3, 5, y 7 (ya que tienen que ser coprimos con 8). Claramente 1 no es, además,

$$3^2 \equiv 1\,(8), \qquad 5^2 \equiv 1\,(8), \qquad \text{y} \qquad 7^2 \equiv 1\,(8),$$

con lo cual $\mathrm{ord}_8(3) = \mathrm{ord}_8(5) = \mathrm{ord}_8(7) = 2 \neq 4$. Se concluye que no hay raíces primitivas módulo 8.

Lema 1.6: Sean $a,\, n \in \mathbb{N}$ coprimos. $a^k \equiv a^l\,(n) \iff k \equiv l\,(\mathrm{ord}_n(a))$.

Lema 1.7: Sean $a,\, n \in \mathbb{N}$ coprimos. Entonces $\mathrm{ord}_n(a^k) = \dfrac{\mathrm{ord}_n(a)}{(k,\, \mathrm{ord}_n(a))}$.

Corolario 1.2: Si $\mathrm{PR}(n) \neq \emptyset$ entonces $|\mathrm{PR}(n)| = \varphi(\varphi(n))$.

Teorema 1.2: $\mathrm{PR}(n) \neq \emptyset \iff n \in \{1,\, 2,\, 4\} \cup \{p^m\}_{\substack{m \in \mathbb{N} \\ p \text{ primo impar}}} \cup \{2p^m\}_{\substack{m \in \mathbb{N} \\ p \text{ primo impar}}}$.

Lema 1.8: Si $a \in \mathrm{PR}(p)$, entonces $a \in \mathrm{PR}(p^k)$ $\forall\, p \geq 3$ primo, $k \in \mathbb{N}$.

Definición 1.4: Sea $r \in \mathrm{PR}(n)$ y $a \in \mathbb{N}$ coprimo con n. Se define al "logaritmo discreto de a respecto a r (o de base r) módulo n" como $I_r^n(a) = \min\{m \in \mathbb{N} \mid r^m \equiv a\,(n)\}$. Se escribirá $I_r(a)$ cuando el módulo se entienda del contexto.

Observación 1.2: Por el inciso i. de la observación anterior, el logaritmo discreto, bajo las hipótesis de la definición, siempre está definido.

Ejemplo 1.2: 3 es una raíz primitiva módulo 7 y

$$3^1 \equiv 3\,(7), \quad 3^2 \equiv 2\,(7), \quad 3^3 \equiv 6\,(7), \quad 3^4 \equiv 4\,(7), \quad 3^5 \equiv 5\,(7), \quad \text{y} \quad 3^6 \equiv 1\,(7).$$

Así,

$$I_3(3) = 1, \quad I_3(2) = 2, \quad I_3(6) = 3, \quad I_3(4) = 4, \quad I_3(5) = 5, \quad \text{y} \quad I_3(3) = 1.$$

Lema 1.9: Sea p primo, a coprimo con p y $r \in \mathrm{PR}(p)$. Se cumple $I_r^p(a) \equiv I_r^{p^k}(a)\,(p-1)$ $\forall\, k \in \mathbb{N}$.

Con esto damos por terminado los preliminares. Continuamos con el estudio de S_k. Recordar que si $N = (n_1,\, \cdots,\, n_k)$, denotamos $N_l = (n_1,\, \cdots,\, n_l)$ para $l \in [\![1,\, k]\!]$.

Teorema 1.3: Para todo $k \in \mathbb{N}$ se cumple

$$S_k = \left\{ N \in \mathbb{N}^k \;\middle|\; \begin{array}{l} n_i \equiv -I_2^9(L(N_{i-1})) + 2r_i \,(6) \text{ con } r_i \in \{1,\, 2\} \text{ para } i \in [\![1,\, k-1]\!] \\ n_k \equiv -I_2^3(L(N_{k-1}))\,(2) \end{array} \right\}.$$

Demostración. Usaremos inducción. Para $k = 1$, notar que 2 es una raíz primitiva módulo 3 y $\varphi(3) = 2$. Se sigue que

$$\tilde{L}(n_1) \equiv 0\,(3) \iff 2^{n_1} \equiv 1 \equiv 2^0\,(3) \iff n_1 \equiv 0 \equiv -I_2^3(L(N_0))\,(2).$$

Ahora supongamos que la afirmación vale para un $l \in \mathbb{N}$. Veamos que también lo hace para $l + 1$. Por hipótesis sabemos que las soluciones de $\tilde{L}(N_l) \equiv 0\,(3^l)$ están dadas por

$$\begin{array}{l} n_i \equiv -I_2^9(L(N_{i-1})) + 2r_i\,(6) \text{ con } r_i \in \{1,\, 2\} \text{ para } i \in [\![1,\, l-1]\!] \\ n_l \equiv -I_2^3(L(N_{l-1}))\,(2) \end{array}.$$

Así,

$$\tilde{L}(N_{l+1}) \equiv 0\,(3^{l+1}) \iff \underbrace{\tilde{L}(N_l) \cdot 2^{n_{l+1}} \equiv 3^l\,(3^{l+1})}_{\sharp} \implies \begin{cases} \text{i.} & \tilde{L}(N_i) \not\equiv 0\,(3^{i+1}) \\ \text{ii.} & \tilde{L}(N_i) \equiv 0\,(3^i) \end{cases}$$

para todo $i \in [\![1,\, l]\!]$. Esto último se deduce de: Es inmediato de ($\sharp$) que $\tilde{L}(N_l) \cdot 2^{n_{l+1}} \equiv 0\,(3^l)$ y como $(2^{n_{l+1}},\, 3^l) = 1$, se obtiene $\tilde{L}(N_l) \equiv 0\,(3^l)$. Repitiendo este razonamiento se obtiene ii. Además,

$$\tilde{L}(N_{i+1}) \equiv 0\,(3^{i+1}) \implies \tilde{L}(N_i) \cdot 2^{n_{i+1}} \equiv 3^i\,(3^{i+1}) \implies \tilde{L}(N_i) \not\equiv 0\,(3^{i+1})$$

para $i \in [\![1,\, l]\!]$ y se tiene i. Se sigue que

$$(\text{i.}, i = l) + (\text{ii.}, i = l) \implies \left(\frac{\tilde{L}(N_l)}{3^l},\, 3 \right) = 1$$

y por lo tanto

$$(\sharp) \iff \underbrace{\frac{\tilde{L}(N_l)}{3^l}}_{=L(N_l)} \cdot 2^{n_{l+1}} \equiv 1\,(3) \iff n_{l+1} \equiv -I_2^3(L(N_l))\,(2).$$

Más aún,

$$(\text{i.}, i = l-1) + (\text{ii.}, i = l-1) \implies (L(N_{l-1}),\, 9) = 1$$

y entonces

$$\begin{aligned} \tilde{L}(N_l) \not\equiv 0\,(3^{l+1}) \iff \tilde{L}(N_{l-1}) \cdot 2^{n_l} \not\equiv 3^{l-1}\,(3^{l+1}) &\iff L(N_{l-1}) \cdot 2^{n_l} \not\equiv 1\,(9) \\ &\iff n_l \not\equiv -I_2^9(L(N_{l-1}))\,(6) \end{aligned}$$

($\varphi(9) = 6$ y 2 es una raíz primitiva módulo 9). De esta forma se tiene

$$\begin{cases} n_l \equiv -I_2^3(L(N_{l-1}))\,(2) \\ n_l \not\equiv -I_2^9(L(N_{l-1}))\,(6) \end{cases} \overset{\natural}{\iff} n_l \equiv -I_2^9(L(N_{l-1})) + 2r_l\,(6)$$

($\natural$) Lema anterior.

en donde $r_l \in \{1, 2\}$. Finalmente

$$\begin{aligned} n_i &\equiv -I_2^9(L(N_{i-1})) + 2r_i \,(6) \text{ con } r_i \in \{1, 2\} \text{ para } i \in [\![1, l]\!] \\ n_{l+1} &\equiv -I_2^3(L(N_l))\,(2) \end{aligned}$$

y concluímos que vale para $l + 1$.

$\blacksquare$

Observación 1.3: De la última demostración podemos resaltar lo siguiente: Dado $N \in S_k$, entonces $N_i \in S_i$ y $(L(N_i), 3) = 1$ para $i \in [\![1, k-1]\!]$.

A continuación mostramos un algoritmo en Python para calcular los elementos de S_k (lo llamaremos **sol**). Este tiene tres entradas: k: natural que indica la dimensión, A: tupla de longitud $k-1$ con cada coordenada teniendo valor 1 o 2, pues esta indica la elección de los r_i's, y por último B: tupla de longitud k cuyas coordenadas indican la traslación de los valores (de 6 en 6 para los primeros $k-1$ y de 2 en 2 para el último) involucrados en el cálculo del elemento de S_k. Tiene como salida un $N \in S_k$. La función $\tilde{L}$ la implementaremos con el nombre **Ltil** y a L con **L**. La función **modstar** es simplemente auxiliar.

```
import numpy as np
from sympy import discrete_log


def Ltil(h):
    a = 1
    for k in range(len(h)):
        a = a * 2 ** h[k] - 3 ** k
    return a


def L(h):
    return Ltil(h) // 3 ** len(h)


def modstar(a, n):
    b = np.mod(a, n)
    if b != 0:
        return b
    else:
        return n


def sol(k, A, B):
    if k == 1:
        return [2 + B[0] * 2]
    N = [2 * A[0] + 6 * B[0]]
    for i in range(1, k - 1):
        c = -discrete_log(9, L(N), 2) + 2 * A[i]
        c = modstar(c, 6) + 6 * B[i]
        N.append(c)
        N = [int(float(a)) for a in N]
    c = -discrete_log(3, L(N), 2)
```

```python
    c = modstar(c, 2) + 2 * B[k - 1]
    N.append(c)
    return [int(float(a)) for a in N]
```

Click aquí para acceder al archivo con los algoritmos. Veamos algunos ejemplos:

```python
>>> sol(1, [], [3])
[8]
>>> sol(3, [1, 2], [0, 2, 5])
[2, 16, 11]
>>> sol(4, [2, 2, 1], [7, 4, 0, 1])
[46, 25, 5, 4]
>>> sol(7, [1, 2, 1, 2, 2, 1], [32, 14, 6, 0, 23, 7, 5])
[194, 90, 39, 2, 143, 43, 12]
```

y podemos verificar que efectivamente están en S_1, S_3, S_4, y S_7 respectivamente:

```python
>>> Ltil([8]) % 3
0
>>> Ltil([2, 16, 11]) % 3 ** 3
0
>>> Ltil([46, 25, 5, 4]) % 3 ** 4
0
>>> Ltil([194, 90, 39, 2, 143, 43, 12]) % 3 ** 7
0
```

Así, ahora conocemos qué forma tienen las soluciones de $\tilde{L}(N) \equiv 0\,(3^k)$. Sin embargo, dado un elemento de S_k, no está claro que modificaciones le podemos hacer a esa tupla (además de trasladar la última coordenada en múltiplos de 2) para conseguir otro elemento también en S_k, pues al realizar un cambio todos los valores siguientes deben ser recalculados. Por esto, daremos una herramienta para, dado $N \in S_k$, hallar toda una familia de elementos también en S_k en base a este último.

Lema 1.10: Dado $N \in S_k$, todo $\tilde{N} \in \mathbb{N}^k$ con $\tilde{n}_i \equiv n_i\,(\varphi(3^{k-i+1}))$ para $i \in [\![1, k]\!]$ cumple $\tilde{N} \in S_k$.

Demostración. En primer lugar, sea $\tilde{N} \in \mathbb{N}^k$ con

$$\begin{aligned} \tilde{n}_i &= n_i \text{ para } i \in [\![1, k]\!] \setminus \{m\} \\ \tilde{n}_m &\equiv n_m\,(\varphi(3^{k-m+1})) \end{aligned},$$

esto es, $\tilde{N} = (n_1, \cdots, n_{m-1}, n_m + \tilde{k} \cdot \varphi(3^{k-m+1}), n_{m+1}, \cdots, n_k)$. Luego

$$
\begin{aligned}
&\tilde{L}(N) \equiv \tilde{L}(\tilde{N})\,(3^k) \iff \frac{\tilde{L}(N_{k-1}) \cdot 2^{n_k} - 1}{3} \equiv \frac{\tilde{L}(\tilde{N}_{k-1}) \cdot 2^{n_k} - 1}{3}\,(3^k) \iff \\
&\iff \tilde{L}(N_{k-1}) \equiv \tilde{L}(\tilde{N}_{k-1})\,(3^k) \iff \cdots \iff \tilde{L}(N_m) \equiv \tilde{L}(\tilde{N}_m)\,(3^k) \iff \\
&\iff \frac{\tilde{L}(N_{m-1}) \cdot 2^{n_m} - 1}{3} \equiv \frac{\tilde{L}(\tilde{N}_{m-1}) \cdot 2^{n_m + \tilde{k}\cdot\varphi(3^{k-m+1})} - 1}{3}\,(3^k) \iff \\
&\iff \tilde{L}(N_{m-1}) \cdot 2^{n_m} \equiv \tilde{L}(\underbrace{\tilde{N}_{m-1}}_{=N_{m-1}})2^{n_m + \tilde{k}\cdot\varphi(3^{k-m+1})}\,(3^k) \iff \\
&\iff \tilde{L}(N_{m-1}) \equiv \tilde{L}(N_{m-1}) \cdot 2^{\tilde{k}\cdot\varphi(3^{k-m+1})}\,(3^k) \iff \\
&\qquad\qquad\iff L(N_{m-1}) \equiv L(N_{m-1}) \cdot \underbrace{2^{\tilde{k}\cdot\varphi(3^{k-m+1})}}_{\equiv 1\,(3^{k-m+1})}\,(3^{k-m+1}),
\end{aligned}
$$

y como lo último se cumple trivialmente, vale que $\tilde{L}(\tilde{N}) \equiv 0\,(3^k)$, o sea $\tilde{N} \in S_k$. Notar que $\tilde{L}(N) \equiv \tilde{L}(\tilde{N})\,(3^k)$ no es válido en general, pues en el último paso ocupamos que $N \in S_k$ para poder dividir a $\tilde{L}(N_{m-1})$ por 3^{m-1} (observación anterior). Aplicando lo recién mostrado para cada coordenada se tiene el lema.

■

Con esto último surge una pregunta natural: ¿Es posible quedarnos con un subconjunto finito $S_k^* \subseteq S_k$, de forma tal que variando las coordenadas de sus elementos según el lema anterior, genere a todo S_k? Esto es, en concreto,

$$¿\exists\ S_k^* \subseteq S_k \text{ con } |S_k^*| < \infty : \{N \in \mathbb{N}^k \mid n_i \equiv n_i^*\,(\varphi(3^{k-i+1}))\ \forall\ i \in [\![1,\, k]\!]\}_{N^* \in S_k^*} = S_k?$$

La respuesta es sí y lo veremos a continuación. Notar que

$$\{N \in \mathbb{N}^k \mid n_i \equiv n_i^*\,(\varphi(3^{k-i+1}))\ \forall\ i \in [\![1,\, k]\!]\}_{N^* \in S_k^*} = \bigcup_{N^* \in S_k^*} \prod_{i=1}^{k} \left([n_i^*]_{\varphi(3^{k-i+1})} \cap \mathbb{N}\right).$$

Denotamos a tal conjunto como $\langle S_k^* \rangle$ y lo llamamos "el generado de S_k^*". Si $n \in \mathbb{N}$ definimos

$$
\begin{aligned}
&\bmod^* : \mathbb{Z} \times \{n\} \longrightarrow [\![1,\, n]\!] \qquad \text{dada por} \\
&\bmod^*(a,\, n) = \begin{cases} \bmod(a,\, n) & \text{si} \quad \bmod(a,\, n) \neq 0 \\ n & \text{si} \quad \bmod(a,\, n) = 0 \end{cases}.
\end{aligned}
$$

Así, el conjunto que va a cumplir lo que buscamos es

$$
\begin{aligned}
S_k^* = \Bigg\{ N^* \in \mathbb{N}^k \ \Bigg|\ &\begin{matrix} n_i^* = \bmod^*(-I_2^9(L(N_{i-1}^*)) + 2r_i,\, 6) + m_i \\ n_k^* = \bmod^*(-I_2^3(L(N_{k-1}^*)),\, 2) \end{matrix}, \text{ con} \\
&r_i \in \{1,\, 2\}, \begin{matrix} m_i \in [\![0,\, \varphi(3^{k-i+1}) - 1]\!] \\ m_i \equiv 0\,(6) \end{matrix}\ \forall\ i \in [\![1,\, k-1]\!] \Bigg\}.
\end{aligned}
$$

Empecemos calculando su cardinal. Veamos primero la siguiente proposición correspondiente a conteo.

Lema 1.11: Dados $m,\, n \in \mathbb{N}$, se cumple $\big|[\![1,\, m]\!] \cap n\mathbb{N}\big| = \dfrac{m - \operatorname{mod}(m,\, n)}{n}$.

Demostración. Recordar que la función "piso " $\lfloor \cdot \rfloor : \mathbb{R} \longrightarrow \mathbb{Z}$ asigna a cada número real r el único entero a tal que $a \leq r < a+1$. Ahora, $n\mathbb{N} = \{n,\, 2n,\, 3n,\, \cdots\}$ y

$$kn \leq m \iff k \leq \frac{m}{n} \iff k \leq \left\lfloor \frac{m}{n} \right\rfloor.$$

Luego es claro que $\big|[\![1,\, m]\!] \cap n\mathbb{N}\big| = \left\lfloor \dfrac{m}{n} \right\rfloor$. Ahora,

$$\left\lfloor \frac{m}{n} \right\rfloor \leq \frac{m}{n} < \left\lfloor \frac{m}{n} \right\rfloor + 1 \iff n\left\lfloor \frac{m}{n} \right\rfloor \leq m < n\left\lfloor \frac{m}{n} \right\rfloor + n \iff 0 \leq m - n\left\lfloor \frac{m}{n} \right\rfloor < n$$

y $m - \left(m - n\left\lfloor \dfrac{m}{n} \right\rfloor\right) = n\left\lfloor \dfrac{m}{n} \right\rfloor$ por lo que $m \equiv \underbrace{m - n\left\lfloor \dfrac{m}{n} \right\rfloor}_{\in [\![0,\, n-1]\!]} (n)$. Se sigue que $\operatorname{mod}(m,\, n) = m - n\left\lfloor \dfrac{m}{n} \right\rfloor$ y entonces $\left\lfloor \dfrac{m}{n} \right\rfloor = \dfrac{m - \operatorname{mod}(m,\, n)}{n}$.

■

Observar que de esto último es inmediato que $\big|[\![0,\, m]\!] \cap n\mathbb{N}_0\big| = \dfrac{m - \operatorname{mod}(m,\, n)}{n} + 1$. Calculemos $|S_k^*|$.

Notar que en cada coordenada $i \in [\![1,\, k-1]\!]$ se tienen 2 opciones para r_i y

$$\begin{aligned}
\big|[\![0,\, \varphi(3^{k-i+1}) - 1]\!] \cap 6\mathbb{N}_0\big| &= \frac{\varphi(3^{k-i+1}) - 1 - \operatorname{mod}(\varphi(3^{k-i+1}) - 1,\, 6)}{6} + 1 \\
&= \frac{6 \cdot 3^{k-i-1} - 1 - \operatorname{mod}(6 \cdot 3^{k-i-1} - 1,\, 6)}{6} + 1 \\
&= \frac{6 \cdot 3^{k-i-1} - 1 - \operatorname{mod}(-1,\, 6)}{6} + 1 \\
&= \frac{6 \cdot 3^{k-i-1} - 1 - 5}{6} + 1 \\
&= \frac{6 \cdot 3^{k-i-1} - 6}{6} + 1 \\
&= 3^{k-i-1} - 1 + 1 \\
&= 3^{k-i-1}
\end{aligned}$$

para m_i, por lo que en cada una de estas se tienen $2 \cdot 3^{k-i-1}$ opciones. Así,

$$|S_k^*| = \prod_{i=1}^{k-1} 2 \cdot 3^{k-i-1} = 2^{k-1} \cdot 3^{(k-2)+(k-3)+\cdots+1} = 2^{k-1} \cdot 3^{\frac{(k-2)(k-1)}{2}}.$$

Ahora,

Teorema 1.4: $\langle S_k^* \rangle = S_k$.

Demostración. ($\subseteq$) Dado $N^* \in S_k^*$, cumple $\begin{cases} n_i^* & \equiv -I_2^9(L(N_{i-1}^*)) + 2r_i \, (6) \\ n_k^* & \equiv -I_2^3(L(N_{k-1}^*)) \, (2) \end{cases}$ con $r_i \in \{1,\, 2\}$ para $i \in [\![1,\, k-1]\!]$, por lo que claramente $N^* \in S_k$ y entonces $S_k^* \subseteq S_k$. Más aún, por el lema 1.10 se tiene $\langle S_k^* \rangle \subseteq S_k$.

($\supseteq$) Sea $N \in S_k$, cumple $\begin{cases} n_i & \equiv -I_2^9(L(N_{i-1})) + 2r_i \, (6) \\ n_k & \equiv -I_2^3(L(N_{k-1})) \, (2) \end{cases}$ con $r_i \in \{1,\, 2\}$ para $i \in [\![1,\, k-1]\!]$. Veamos que $N \in \langle S_k^* \rangle$.

Si $i = 1$, $n_1 = \underbrace{-I_2^9(\underbrace{L(N_0)}_{=1})}_{=0} + 2r_1 + 6h_1$ con $h_1 \in \mathbb{N}_0$. Usando el algoritmo de la división con $6h_1$ siendo el dividendo y $\varphi(3^{k-i+1}) = \varphi(3^k)$ el divisor, se obtiene $6h_1 = m_1 + q_1 \cdot \varphi(3^k)$ con $m_1 \in [\![0,\, \varphi(3^k) - 1]\!]$. Además, como $\varphi(3^k) = 6 \cdot 3^{k-2}$ se tiene $m_1 \equiv 0 \, (6)$. Se sigue que $n_1 = 2r_1 + m_1 + q_1 \cdot \varphi(3^k) = \text{mod}^*(-I_2^9(L(N_0^*)) + 2r_1,\, 6) + m_1 + q_1 \cdot \varphi(3^k)$ y por lo tanto $n_1 = n_1^* + q_1 \cdot \varphi(3^k)$ con $n_1^* := \text{mod}^*(-I_2^9(L(N_0^*)) + 2r_1,\, 6) + m_1$. Notar que

$$L(N_1) \equiv L(N_1^*) \, (3^{k-1}) \iff \frac{2^{n_1} - 1}{3} \equiv \frac{2^{n_1^*} - 1}{3} \, (3^{k-1}) \iff 2^{n_1} \equiv 2^{n_1^*} \, (3^k) \iff$$
$$\iff n_1 \equiv n_1^* \, (\varphi(3^k)),$$

y por lo recién visto lo último es cierto.

Ahora supongamos que para un $i \in [\![1,\, k-2]\!]$ se tiene mostrado que

$$\begin{array}{lcl} n_i = n_i^* + q_i \cdot \varphi(3^{k-i+1}) & \text{con} & n_i^* = \text{mod}^*(-I_2^9(L(N_{i-1}^*)) + 2r_i,\, 6) + m_i \\ L(N_i) \equiv L(N_i^*) \, (3^{k-i}) & & m_i \in [\![0,\, \varphi(3^{k-i+1}) - 1]\!],\, m_i \equiv 0 \, (6) \end{array}.$$

Veamos que también se cumple para $i+1$. Sabemos que $n_{i+1} \equiv -I_2^9(L(N_i)) + 2r_{i+1} \, (6)$ y

$$I_2^9(L(N_i)) \equiv I_2^9(L(N_i^*)) \, (6) \iff L(N_i) \equiv L(N_i^*) \, (9).$$

Por hipótesis se tiene $L(N_i) \equiv L(N_i^*) \, (3^{k-i})$, e $i \leq k-2$ implica $k - i \geq 2$. Por esto se cumple $L(N_i) \equiv L(N_i^*) \, (9)$ y luego $n_{i+1} \equiv -I_2^9(L(N_i^*)) + 2r_{i+1} \, (6)$. Se sigue que $n_{i+1} = \text{mod}^*(-I_2^9(L(N_i^*)) + 2r_{i+1},\, 6) + 6h_{i+1}$ con $h_{i+1} \in \mathbb{N}_0$. Ahora aplicamos el algoritmo de la división siendo $6h_{i+1}$ el dividendo y $\varphi(3^{k-(i+1)+1}) = \varphi(3^{k-i})$ el divisor y se obtiene $6h_{i+1} = m_{i+1} + q_{i+1} \cdot \varphi(3^{k-i})$ con $m_{i+1} \in [\![0,\, \varphi(3^{k-i}) - 1]\!]$. Además, $\varphi(3^{k-i}) = 6 \cdot 3^{k-i-2}$ y $k - i - 2 \geq 0$, por lo que resulta $m_{i+1} \equiv 0 \, (6)$. Se sigue que $n_{i+1} = n_{i+1}^* + q_{i+1} \cdot \varphi(3^{k-(i+1)+1})$ con $n_{i+1}^* := \text{mod}^*(-I_2^9(L(N_i^*)) + 2r_{i+1},\, 6) + m_{i+1}$. Por último,

$$L(N_{i+1}) \equiv L(N_{i+1}^*) \, (3^{k-(i+1)}) \iff \frac{L(N_i) \cdot 2^{n_{i+1}} - 1}{3} \equiv \frac{L(N_i^*) \cdot 2^{n_{i+1}^*} - 1}{3} \, (3^{k-i-1}) \iff$$
$$\iff L(N_i) \cdot 2^{n_{i+1}} \equiv \underbrace{L(N_i^*)}_{\equiv L(N_i) \, (3^{k-i})} \cdot 2^{n_{i+1}^*} \, (3^{k-i}) \overset{\sharp}{\iff} 2^{n_{i+1}} \equiv 2^{n_{i+1}^*} \, (3^{k-i}) \iff$$
$$\iff n_{i+1} \equiv n_{i+1}^* \, (\varphi(3^{k-i})),$$

($\sharp$) $(L(N_i),\, 3) = 1$ por la observación 1.3.

y por lo recién visto, lo último es cierto.

Si $i = k$ se tiene $n_k = -I_2^3(L(N_{k-1}))\,(2)$ y

$$I_2^3(L(N_{k-1})) \equiv I_2^3(L(N_{k-1}^*))\,(2) \iff L(N_{k-1}) \equiv L(N_{k-1}^*)\,(3),$$

y esto se cumple pues ya vimos que $L(N_{k-1}) \equiv L(N_{k-1}^*)\,(3^{k-(k-1)})$. Se sigue que $n_k \equiv -I_2^3(L(N_{k-1}^*))\,(2)$ y por lo tanto $n_k = \mathrm{mod}^*(-I_2^3(L(N_{k-1}^*)), 2) + q_k \cdot \varphi(3) = n_k^* + q_k \cdot \varphi(3)$ con $n_k^* := \mathrm{mod}^*(-I_2^3(L(N_{k-1}^*)), 2)$.

Finalmente, es claro que $N^* \in S_k^*$ y $N \in \prod_{i=1}^{k} \left([n_i^*]_{\varphi(3^{k-i+1})} \cap \mathbb{N}\right) \subseteq \langle S_k^* \rangle$.

$\blacksquare$

A continuación damos un algoritmo que como entrada tiene un k natural que indica la dimensión y como salida una lista con todos los elementos de S_k^*. A este lo llamaremos **solstar**. Es necesario que **discrete_log**, las funciones definidas anteriormente (**Ltil**, **L**, **modstar** y **sol**), y el módulo **numpy** estén importados. Las funciones **prodcart** y **cart** son simplemente auxiliares.

```
def prodcart(A, B):
    result = []
    for i in range(0, len(A)):
        for j in range(0, len(B)):
            if type(A[i]) != list:
                A[i] = [A[i]]
            temp = [num for num in A[i]]
            temp.append(B[j])
            result.append(temp)
    return result

def cart(A):
    temp = A[0]
    for i in range(1, len(A)):
        temp = prodcart(temp, A[i])
    return temp

def solstar(k):
    if k == 1:
        return [[2]]
    if k == 2:
        return [[2, 2], [4, 1]]
    A = cart([[1, 2] for i in range(k-1)])
    B = [[i for i in range(0, 3 ** (k-j-1))] for j in range(1, k)]
    C = cart(B)
    for c in C:
        c.append(0)
    h = [sol(k, a, c) for a in A for c in C]
    return h


def param(A):
    k = len(A)
```

```
    if k == 1:
        return [], []
    R = []
    M = []
    if A[0] % 6 == 2:
        R.append(1)
        M.append(A[0] - 2)
    else:
        R.append(2)
        M.append(A[0] - 4)
    for i in range(1, k-1):
        x = -dilog9(L(A[:i])) + 2
        if A[i] % 6 == x % 6:
            R.append(1)
            M.append(A[i] - modstar(x, 6))
        else:
            R.append(2)
            M.append(A[i] - modstar(x + 2, 6))
    del M[k - 2]
    X = []
    X.append(R)
    X.append(M)
    return X


def suma(A, B):
    k = len(A)
    if k == 1:
        return [2]
    ra, ma = param(A)
    rb, mb = param(B)
    ma.append(0)
    mb.append(0)
    rsum = [modstar(ra[i] + rb[i], 2) for i in range(k-1)]
    msum = [np.mod(ma[i] + mb[i], phi(3 ** (k-(i+1)+1))) for i in range(k-1)]
    N = [int(float(2 * rsum[0] + msum[0]))]
    for i in range(1, k-1):
        c = modstar(-dilog9(L(N)) + 2 * rsum[i], 6) + msum[i]
        N.append(c)
        N = [int(float(a)) for a in N]
    c = modstar(-dilog3(L(N)), 2)
    N.append(c)
    N = [int(float(a)) for a in N]
    return N


def gen(A):
    k = len(A)
    e = id(k)
    G = []
    G.append(e)
    if A == e:
        return G
    G.append(A)
    B = suma(A, A)
    while B != e:
        G.append(B)
```

```
        B = suma(B, A)
    return G

def ord(A):
    return len(gen(A))

def inver(A):
    H = gen(A)
    k = len(H)
    return H[k-1]

def id(k):
    return sol(k, [2 for i in range(k-1)], [0 for i in range(k)])

def elemord(k, m): #da una lista con los elementos de orden m en S*_k
    A = solstar(k)
    B = []
    for i in range(len(A)):
        x = A[i]
        if ord(x) == m:
            B.append(x)
    return B

def contord(k, m): #cuenta la cantidad de elementos de orden m en S*_k
    A = [ord(a) for a in solstar(k)]
    j = 0
    for i in range(len(A)):
        if A[i] == m:
            j += 1
    return j

def contord2(A, m): #cuenta la cantidad de elementos de orden m en Z_A[0]x...xZ_A[k-1], k = len(A)
    k = len(A)
    if k == 1:
        G = Group("add", A[0])
    else:
        G = Group(["add" for i in range(k)], A)
    B = G.go()
    j = 0
    for i in range(len(B)):
        if B[i] == m:
            j += 1
    return j

def xi(A, B):
    if not len(A) == len(B) + 1:
        print("dimension error")
        return
    B.append(0)
    B.append(0)
    return sol(len(A) + 1, A, [b // 6 for b in B])

def retrac(N): #tiene como entrada un N en S*_k y como salida la retracci n de N, en S*_(k-1)
    k = len(N)
    if k == 2:
        return [2]
```

```
    if k == 1:
        return []
    x = param(N)
    rho = x[0]
    mu = x[1]
    del rho[k - 2]
    del mu[k - 3]
    mu = [np.mod(mu[i], phi(3 ** (k - (i + 1)))) for i in range(len(mu))]
    rho = [int(float(a)) for a in rho]
    mu = [int(float(a)) for a in mu]
    return xi(rho, mu)

def atrac(N, t): #tiene como entrada un N en S*_k y como salida la atracci n de N (can nica, de orden o natural seg n el tipo)
    k = len(N)
    if t == 'can':
        return [2] + N
    x = param(N)
    rho = x[0]
    mu = x[1]
    if t == 'ord':
        return xi([2] + rho, [0] + mu)
    if t == 'nat':
        return xi(rho + [2], mu + [0])

def setisize(A):
    a = []
    for x in A:
        if not x in a:
            a.append(x)
    return a

def atrac2(A, j): #tiene como entrada un N en S*_k y como salida N^(j), j >= k
    while len(A) != j:
        A = atrac(A, 'nat')
    return A

def atrac3(A, j): #tiene como entrada un N en S*_k y como salida N^(j), j >= k
    while len(A) != j:
        A = atrac(A, 'ord')
    return A


def retrac2(A, j): #tiene como entrada un N en S*_k y como salida N^(j), j <= k
    while len(A) != j:
        A = retrac(A)
    return A

def suma1(A, B, t = '+'): #tiene como entradas dos elementos en S*_inf y como salida su suma (mayor o menor dependiendo del tipo t)
    k = len(A)
    l = len(B)
    if t == '+':
        m = max(k, l)
```

```
        A = atrac2(A, m)
        B = atrac2(B, m)
        return suma(A, B)
    if t == '-':
        n = min(k, l)
        A = retrac2(A, n)
        B = retrac2(B, n)
        return suma(A, B)

def suma2(A, B, t = '+'): #tiene como entradas dos elementos en S*_inf
    y como salida su suma (mayor o menor dependiendo del tipo t)
    k = len(A)
    l = len(B)
    if t == '+':
        m = max(k, l)
        A = atrac3(A, m)
        B = atrac3(B, m)
        return suma(A, B)
    if t == '-':
        n = min(k, l)
        A = retrac2(A, n)
        B = retrac2(B, n)
        return suma(A, B)


def mover(A, j): #tiene como entrada un N en S*_k y como salida N^(j)
    k = len(A)
    if j >= k:
        return atrac2(A, j)
    else:
        return retrac2(A, j)

def retracalt(N):
    k = len(N)
    if k == 2:
        return [2]
    if k == 1:
        return []
    x = param(N)
    rho = x[0]
    mu = x[1]
    rho = rho[1:]
    mu = mu[1:]
    rho = [int(float(a)) for a in rho]
    mu = [int(float(a)) for a in mu]
    return xi(rho, mu)

def retracalt2(N, j):
    while len(N) != j:
        N = retracalt(N)
    return N

def mover2(A, j):
    k = len(A)
    if j >= k:
        return atrac3(A, j)
    else:
        return retracalt2(A, j)
```

```
def inf(A):
    return [L(A), fact(L(A)), aux(L(A))]

def parte_basica(N): #tiene como entrada N en S_k y como salida la
    parte basica de N
    k = len(N)
    A = [modstar(N[i], phi(3 ** (k - (i + 1) + 1))) for i in range(k
    -1)] + [modstar(N[k-1], 2)]
    return [int(float(A[i])) for i in range(k)]

def traslacion(N): #tiene como entrada N en S_k y como salida la
    traslacion de N
    k = len(N)
    A = parte_basica(N)
    B = [N[i] - parte_basica(N)[i] for i in range(k)]
    return[B[i] // phi(3 ** (k - (i + 1) + 1)) for i in range(k)]

def invW(l):
    k = (l + 1) // 2
    factor = fact(3 * k - 1)
    if 2 in list(factor):
        n = factor[2] + 1
    else:
        n = 1
    A = list(filter(lambda x: x != 2, factor))
    B = [i ** factor[i] for i in A]
    return [int(np.prod(A)), n]
```

Click aquí para acceder al archivo con los algoritmos. Veamos algunos ejemplos:

```
>>> solstar(1)
[[2]]
>>> solstar(2)
[[2, 2], [4, 1]]
>>> solstar(3)
[[2, 2, 2], [8, 6, 2], [14, 4, 2], [2, 4, 1], [8, 2, 1], [14, 6, 1],
    [4, 3, 2], [10, 5, 2], [16, 1, 2], [4, 5, 1], [10, 1, 1], [16, 3,
    1]]
```

Cabe aclarar que a fines prácticos se puede usar **solstar** hasta $k = 5$, pues para $k = 6$ se tiene $|S_6^*| = 2^{6-1} \cdot 3^{(6-2)(6-1)/2} = 32 \cdot 3^{10} > 1,000,000$.

Ahora, recordar que por la propiedad recursiva de L se tiene que los elementos que formarían parte de los conjuntos involucrados en la descomposición de $2\mathbb{N} - 1$ (lema 1.2) son de la forma

$$\frac{1 \cdot 2^{n_1} - 1}{3}, \qquad \frac{\frac{2^{n_1} - 1}{3} \cdot 2^{n_2} - 1}{3}, \qquad \frac{\frac{\frac{2^{n_1} - 1}{3} \cdot 2^{n_2} - 1}{3} \cdot 2^{n_3} - 1}{3}, \qquad \ldots$$

Por esto, se podría decir que a medida en que aumentamos k en S_k, estamos "refinando" el conjunto en donde buscamos naturales impares. Más aún, se puede ver que los números generados en S_k no se pierden en S_{k+1}. En concreto,

Lema 1.12: Para todo $k \in \mathbb{N}$ se cumple $\{L(N)\}_{N \in S_k} \subseteq \{L(N)\}_{N \in S_{k+1}}$.

Demostración. Sea $N = (n_1, \cdots, n_k)$.

$$L(N) = L(2, N) \iff (G_{n_k}^{-1} \circ \cdots \circ G_{n_1}^{-1})(1) = (G_{n_k}^{-1} \circ \cdots \circ G_{n_1}^{-1} \circ G_2^{-1})(1) \iff$$
$$\iff 1 = G_2^{-1}(1)$$

y $G_2^{-1}(1) = \dfrac{3}{2^2} \cdot 1 + \dfrac{1}{2^2} = 1$.

■

Observación 1.4: Cabe resaltar de la demostración anterior que si $N \in S_k$, entonces $(2,\, N) \in S_{k+1}$ y $L(2,\, N) = L(N)$.

Veamos ahora dos propiedades muy importantes de L.

Lema 1.13: i. $L(N)$ es impar para $N \in S_k$ y ii. $L_{|_{\mathbb{N}^k}}$ es inyectiva. Más aún, dados $N \in S_k$, $M \in S_l$ con $l > k$, entonces $L(N) = L(M)$ si y solo si $M = (\underbrace{2, \cdots, 2}_{l-k}, N)$.

Demostración. i. Como $(2,\, 3) = 1$ se tiene

$$\mathrm{mod}(L(N),\, 2) = \mathrm{mod}\left(\frac{L(N_{k-1}) \cdot 2^{n_k} - 1}{3},\, 2\right) = \mathrm{mod}(L(N_{k-1}) \cdot 2^{n_k} - 1,\, 2) =$$
$$= \mathrm{mod}(-1,\, 2) = 1.$$

ii. Primero veamos que $L_{|_{\mathbb{N}^k}}$ es inyectiva. Por comodidad de escritura vamos a usar L, pero recordar que está restringida a $\mathbb{N}^k$. Usaremos inducción. Si $k = 1$,

$$L(n_1) = L(\tilde{n}_1) \iff \frac{2^{n_1} - 1}{3} = \frac{2^{\tilde{n}_1} - 1}{3} \iff 2^{n_1} = 2^{\tilde{n}_1} \iff n_1 = \tilde{n}_1.$$

Ahora supongamos que L es inyectiva en $\mathbb{N}^l$. Vamos a ver que también lo es en $\mathbb{N}^{l+1}$.

$$L(N_{l+1}) = L(\tilde{N}_{l+1}) \iff \frac{L(N_l) \cdot 2^{n_{l+1}} - 1}{3} = \frac{L(\tilde{N}_l) \cdot 2^{\tilde{n}_{l+1}} - 1}{3} \iff$$
$$\iff \underbrace{L(N_l)}_{\equiv 1\,(2)} \cdot 2^{n_{l+1}} = \underbrace{L(\tilde{N}_l)}_{\equiv 1\,(2)} \cdot 2^{\tilde{n}_{l+1}} \iff L(N_l) = L(\tilde{N}_l) \quad \text{y} \quad 2^{n_{l+1}} = 2^{\tilde{n}_{l+1}}.$$

Luego es claro que $N_{l+1} = \tilde{N}_{l+1}$.

Ahora sea $N \in S_k$ y $M = (2, \cdots, 2,\, N) \in \mathbb{N}^l$. Por la observación anterior es claro que $M \in S_l$ y $L(N) = L(M)$. Por otra parte, si $N \in S_k$, $M \in S_l$ con $l > k$ y $L(N) = L(M)$, se tiene que $L(\underbrace{2, \cdots, 2,\, N}_{\in S_l}) = L(M)$, y ya mostramos que $L_{|_{\mathbb{N}^l}}$ es inyectiva. Luego $M = (2, \cdots, 2,\, N)$.

■

Corolario 1.3: Los incisos i. y ii. del lema anterior también se cumplen para $\tilde{L}$.

Después de todo lo visto, naturalmente surgen las siguientes preguntas: ¿Cuál es la distribución de los naturales impares sobre L bajo los S_k's? (o la de $\mathbb{N}$ sobre $2^j L$ bajo los S_k's y j's, que es equivalente a lo anterior como ya vimos) y por otra parte, Dado un $n_0 \in \mathbb{N}$, ¿Qué tanto hay que refinar en el conjunto de búsqueda (aumentar k) para encontrar a todos los impares menores que n_0 en tal conjunto? (o para encontrar a todo natural menor que n_0 en el conjunto anterior mencionado). Todo esto, en concreto, es estudiar el comportamiento de

$$f(n) = \text{mín}\left\{\tilde{k} \mid L^{-1}(n) \cap \mathbb{N}^{\tilde{k}} \neq \emptyset\right\} \qquad \text{y}$$
$$g(n_0) = \text{mín}\left\{\tilde{k} \mid 2\mathbb{N} - 1 \cap [\![1,\, n_0]\!] \subseteq \{L(N)\}_{N \in S_{\tilde{k}}}\right\}.$$

Notar que $g(n_0) = \max\limits_{n' \in [\![1,\, n_0]\!]} f(n')$. Si bien no vamos a ahondar de forma profunda en estas cuestiones, podemos hacer un gráfico para tener una idea visual de como son f y g. Vamos a implementar a la función **graf**, que tiene como entradas n_0, que indica la tolerancia en el eje x para los números impares graficados, y m, que indica la tolerancia en el refinamiento k. Como salida tiene un esbozo del conjunto $\{(n,\, f(n)) \in 2\mathbb{N} - 1 \times \mathbb{N} \mid n \leq n_0 \text{ y } f(n) \leq m\}$. Las funciones **collatz** y **cont** son auxiliares.

```python
import numpy as np
import matplotlib.pyplot as plt


def collatz(n):
    q = np.array([])
    while n != 1:
        if n % 2 == 1:
            n = (3 * n + 1) // 2
            q = np.append(q, 1)
        else:
            n = n / 2
            q = np.append(q, 2)
    a = len(q)
    q0 = np.zeros(a)
    for i in range(a):
        q0[i] = q[a - i - 1]
    return q0.astype('int')


def cont(n):
    j = 0
    a = collatz(n)
    for i in range(len(a)):
        if a[i] == 1:
            j = j + 1
    return j


```

```
def graf(n_0, m):
    A = [cont(n) for n in range(1, n_0 + 1)]
    X = []
    for n in range(1, n_0 + 1):
        if A[n - 1] <= m and n % 2 == 1:
            X.append(n)
    B = [A[n - 1] for n in X]
    C = []
    for i in range(len(B)):
        l = 0
        q = B[i]
        for j in range(len(C)):
            if C[j] == q:
                l = 1
        if l == 0:
            C.append(q)
    C.sort()
    Y = []
    for i in range(len(X)):
        Y.append(A[X[i] - 1])
    plt.scatter(X, Y, s=5, c='r')
    x = np.linspace(0, n_0, 2)
    for i in range(len(C)):
        h = C[i]
        y = [h, h]
        if i % 2 == 0:
            plt.plot(x, y, 'k--', linewidth=0.5)
            plt.text(n_0 + 0.5, h, f"${h}$", fontsize='x-small', color='k')
        else:
            plt.plot(x, y, 'b--', linewidth=0.5)
            plt.text(n_0 + 0.5, h, f"${h}$", fontsize='x-small', color='b')
    plt.yticks([])
    plt.xticks([0, n_0])
    plt.show()

def graff(n_0, m):
    A = [cont(n) for n in range(1, n_0 + 1)]
    X = []
    for n in range(1, n_0 + 1):
        if A[n - 1] <= m:
            X.append(n)
    B = [A[n - 1] for n in X]
    C = []
    for i in range(len(B)):
        l = 0
        q = B[i]
        for j in range(len(C)):
            if C[j] == q:
                l = 1
        if l == 0:
            C.append(q)
    C.sort()
    Y = []
    for i in range(len(X)):
        Y.append(A[X[i] - 1])
    plt.scatter(X, Y, s=5, c='r')
```

```
    plt.yticks([])
    plt.xticks([0, n_0])
    plt.show()


def xi(k):
    return (2**totient(3**k)-1)//3**k


def numberToBase(n, b):
    if n == 0:
        return [0]
    digits = []
    while n:
        digits.append(int(n % b))
        n //= b
    return digits[::-1]
```

Click aquí para acceder al archivo con los algoritmos. Veamos un ejemplo:

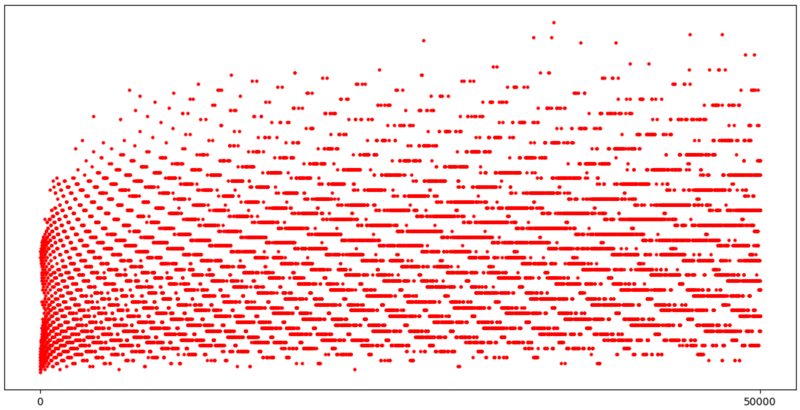

Figura 1: Gráfico de $f(n)$ sobre los primeros impares hasta 50,000

Notar que en cada corte vertical, el punto de más arriba representa un valor de g. Vale la pena destacar algunas regiones interesantes de la imágen anterior:

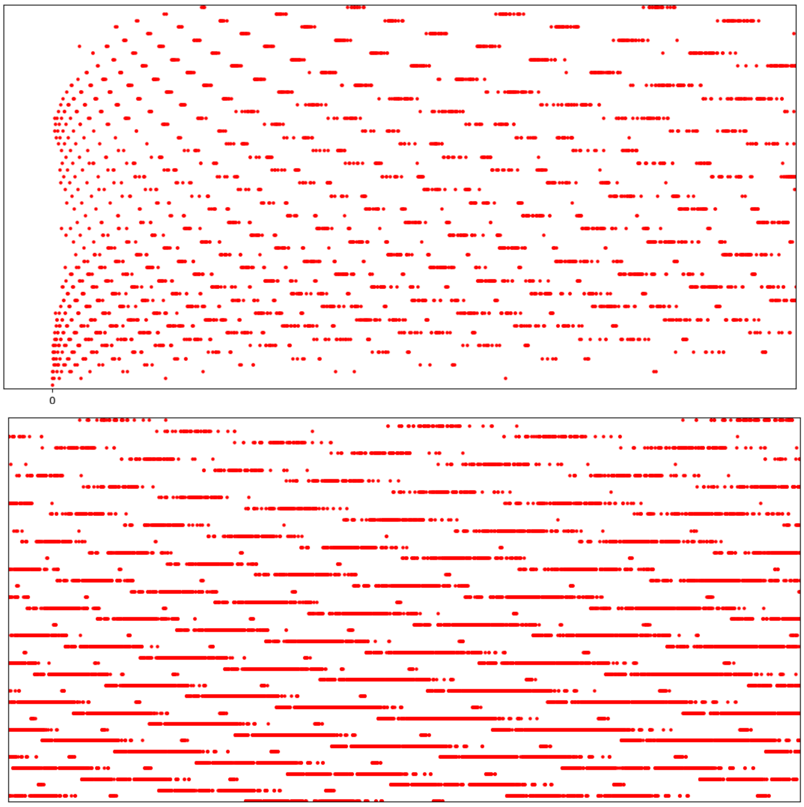

Figura 2: Secciones notables de la Figura 1

2. Abordaje Topológico.

Para esta sección nos basamos en [14, 12].

Definición 2.1: Un espacio topológico se dice "de Alexandroff" si es cerrado bajo intersección arbitraria de abiertos. Esto es, dados $\{U_i\}_{i\in I}$ abiertos, entonces $\bigcap_{i\in I} U_i$ es abierto.

Observación 2.1: Dado X un espacio topológico de Alexandroff y $a \in X$, $V(a) := \bigcap_{\substack{U \text{ abierto} \\ a \in U}} U$ es el entorno abierto de a más chico. Es decir, si V es entorno abierto de a, entonces $V(a) \subseteq V$.

Lema 2.1: Un espacio topológico X es de Alexandroff si y solo si para todo $a \in X$ existe un entorno más chico de a.

Demostración. ($\Rightarrow$) es la observación anterior. ($\Leftarrow$) Supongo que $\forall\, a \in X$ $\exists\, V(a)$ entorno abierto de a con la propiedad de que si U es un entorno abierto de a, entonces $V(a) \subseteq U$. Luego $V(a) \subseteq \bigcap_{\substack{U \text{ abierto} \\ a \in U}} U$, pero $\bigcap_{\substack{U \text{ abierto} \\ a \in U}} U \subseteq V(a)$ trivialmente y por lo tanto son iguales. Así, sean $\{U_i\}_{i\in I}$ abiertos y $V = \bigcap_{i\in I} U_i$. Si $V = \emptyset$ listo, de lo contrario para $x \in V$ se tiene que para cada $i \in I$, $x \in U_i$ y entonces $V(x) \subseteq U_i$. Se sigue que $V(x) \subseteq V$ para todo $x \in V$ y en consecuencia $\bigcup_{x\in V} V(x) \subseteq V$, pero claramente $V \subseteq \bigcup_{x\in V} V(x)$ y entonces son iguales. Luego V es abierto.

■

Definición 2.2: Sea $X \neq \emptyset$, $f : X \longrightarrow X$ función. Se define la "topología primal generada por f" a $\tau_f = \{A \subseteq X \mid f^{-1}(A) \subseteq A\}$. En tal caso a X lo llamamos espacio topológico primal.

Observación 2.2: τ_f es una topología de Alexandroff, pues es claro que $\emptyset$, $X \in \tau_f$, y si $\{A_i\}_{i\in I} \subseteq \tau_f$, $f^{-1}\left(\bigcup_{i\in I} A_i\right) = \bigcup_{i\in I} f^{-1}(A_i) \subseteq \bigcup_{i\in I} A_i$ y $f^{-1}\left(\bigcap_{i\in I} A_i\right) = \bigcap_{i\in I} f^{-1}(A_i) \subseteq \bigcap_{i\in I} A_i$.

Recordar que si $Y \subseteq X$, la clausura de Y, $\overline{Y}$ (o $\mathrm{cl}(Y)$), es el cerrado más chico que contiene a Y. Esto es, si $A \subseteq X$ es cerrado y cumple $Y \subseteq A$, entonces $\overline{Y} \subseteq A$.

Lema 2.2: Sea X un espacio topológico primal y $Y \subseteq X$. Se cumplen

i. $\displaystyle\bigcup_{n\in\mathbb{N}_0} f^n(Y)$ es el subconjunto de X f-invariante que contiene a Y más chico,

ii. Y es cerrado si y solo si es f-invariante, iii. $\displaystyle\overline{Y} = \bigcup_{n\in\mathbb{N}_0} f^n(Y)$.

Demostración. Denotamos $f^{\mathbb{N}_0}(Y) = \bigcup_{n\in\mathbb{N}_0} f^n(Y)$. i. Es claro que $Y \subseteq f^{\mathbb{N}_0}(Y)$ y que es f-invariante. Veamos que es el más chico. Sea $A \subseteq X$ f-invariante con $Y \subseteq A$. Veamos que $f^{\mathbb{N}_0}(Y) \subseteq A$. En primer lugar, $f(A) \subseteq A$, con lo que $f^n(A) \subseteq A$ para todo $n \geq 0$. Además, $Y \subseteq A$ implica $f^n(Y) \subseteq f^n(A) \subseteq A$. Se sigue que $f^{\mathbb{N}_0}(Y) \subseteq A$.

ii. $(\Rightarrow)$ Y^c es abierto y entonces $f^{-1}(Y^c) \subseteq Y^c \iff Y \subseteq (f^{-1}(Y^c))^c = f^{-1}(Y) \iff f(Y) \subseteq f(f^{-1}(Y)) \subseteq Y$ y por lo tanto es f-invariante.

$(\Leftarrow)$ $f(Y) \subseteq Y \iff Y \subseteq f^{-1}(f(Y)) \subseteq f^{-1}(Y) \implies (f^{-1}(Y))^c = f^{-1}(Y^c) \subseteq Y^c$. Se sigue que Y es cerrado.

iii. En primer lugar $\overline{Y}$ es f-invariante pues es cerrado (inciso anterior) y $Y \subseteq \overline{Y}$, con lo que $f^{\mathbb{N}_0}(Y) \subseteq \overline{Y}$ (inciso i.). Más aún, $f^{\mathbb{N}_0}(Y)$ es cerrado pues claramente es f-invariante, pero $\overline{Y}$ es el cerrado más chico que contiene a Y. Se sigue que $\overline{Y} \subseteq f^{\mathbb{N}_0}(Y)$ y entonces son iguales.

■

Definición 2.3: La órbita de $x \in X$ se define como $O_f(x) := \{f^n(x)\}_{n\in\mathbb{N}_0}$ y su núcleo $\ker_f(x) = \bigcup_{n\in\mathbb{N}_0} f^{-n}(x)$ (para buena definición notar que $(f^n)^{-1} = (f^{-1})^n$). Cuando la función f se entienda del contexto se escribirá $O(x)$ y $\ker(x)$.

Recordar que dado un conjunto no vacío, una relación en este se dice cuasi-orden si es reflexiva y transitiva. El conjunto se dice parcialmente ordenado si la relación es reflexiva, antisimétrica y transitiva. Dado un orden parcial $(A, \leq)$, $x \in A$ se dice elemento minimal si $y \leq x$ implica $y = x$.

Definición 2.4: i. Dado (X, τ_f), se define el cuasi-orden $\leq_f$ en X dado por $x \leq_f y \iff y \in O_f(x)$ (fácilmente se verifica que efectivamente es un cuasi-orden).

ii. Se definen los conjuntos $\uparrow x := \{y \in X \mid x \leq_f y\}$ y $\downarrow x := \{y \in X \mid y \leq_f x\}$.

Observación 2.3: i. $\uparrow x = \mathrm{cl}(x)$ y $\downarrow x = \ker(x)$.

ii. Dada una familia de conjuntos, la relación definida por $A \leq B \iff A \subseteq B$ define un orden parcial en tal familia.

Definición 2.5: Dado (X, τ_f), $W \subseteq X$ se dice un conjunto minimal de f si es un elemento minimal del orden parcial $(\{A \subseteq X \mid A \neq \emptyset \text{ cerrado}\}, \subseteq)$

Lema 2.3: Los conjuntos minimales de un espacio topológico primal (X, τ_f) son exactamente la órbita de los puntos periódicos.

Demostración. Sea $A \subseteq X$ un conjunto minimal. Es claro que para todo $x \in X$, $O(x)$ es cerrado (pues es la clausura de x). Así, si $x \in A$, como A es cerrado se tiene $f^n(A) \subseteq A$ (inciso ii. del lema anterior) y entonces $f^n(x) \in A$ para $n \geq 0$. Luego $O(x) \subseteq A$ y como A es minimal se sigue que $O(x) = A$, además, con el mismo argumento se obtiene $O(f(x)) = A$, de donde se obtiene $x \in O(f(x))$ por lo que x es periódico.

Por otra parte, Sea x periódico y $A \neq \emptyset$ cerrado (por lo tanto f-invariante) con $A \subseteq O(x) = \{x, f(x), \cdots, f^k(x)\}$ para algún $k \geq 0$. Si $a \in A$ entonces $f^n(a) \in A$ para todo $n \geq 0$. Luego es claro que $A = O(x)$.

■

Definición 2.6: Un espacio topológico X se dice w-R_0 si $\bigcap_{x \in X} \mathrm{cl}(x) = \emptyset$

Teorema 2.1: Un espacio topológico primal (X, τ_f) no es w-R_0 si y solo si hay una única órbita periódica O y $\forall\, x \in X \setminus O\ \exists\, n \in \mathbb{N}$ tal que $f^n(x) \in O$.

Demostración. ($\Leftarrow$) Para todo $x \in X \setminus O$ se tiene $O \subseteq \mathrm{cl}(x)$ (lema anterior). Luego

$$\bigcap_{x\in X} \mathrm{cl}(x) = \left(\bigcap_{x\in X\setminus O} \mathrm{cl}(x)\right) \cap \left(\bigcap_{x\in O} \mathrm{cl}(x)\right) \overset{\natural}{=} \left(\bigcap_{x\in X\setminus O} \mathrm{cl}(x)\right) \cap O = O \neq \emptyset$$

y se sigue que X no es w-R_0.

($\Rightarrow$) Definimos $O := \bigcap_{x\in O} \mathrm{cl}(x)$. Claramente es cerrado por X ser de Alexandroff, veamos que es un minimal de f. Si $A \neq \emptyset$ es cerrado, se tiene $\mathrm{cl}(A) = \bigcup_{a\in A} \mathrm{cl}(A) = A$ (recordar que la clausura de la unión es la unión de las clausuras). Así, dado $o \in O$, $o \in \mathrm{cl}(x)$ para todo $x \in X$ y por lo tanto $o \in A$. Por el lema anterior O es una órbita periódica. Si $O' \neq O$ es otra órbita periódica, $O' \cap O = \emptyset$ y X sería w-R_0, contradiciendo la hipótesis.

Si $x \in X \setminus O$, $O \subseteq \mathrm{cl}(x)$ (obvio de la definición de O) y luego para todo $a \in O$ se tiene $a \in \mathrm{cl}(x)$, con lo que existe $n \in \mathbb{N}$ tal que $a = f^n(x) \in O$.

■

Corolario 2.1: (X, τ_f) es w-R_0 si y solo si se cumple alguna de las siguientes:

i. X no tiene órbitas periódicas, ii. X tiene 2 o más órbitas periódicas,

iii. X tiene solo una órbita periódica y $\exists\, a \in X$ tal que $O \cap \mathrm{cl}(a) = \emptyset$.

Definición 2.7: $\mathrm{Per}(f) := \{\text{puntos periódicos de } f\}$.

Lema 2.4: i. $\downarrow x \subseteq \downarrow y$, $\downarrow y \subseteq \downarrow x$ o $\downarrow x \cap \downarrow y = \emptyset$ y ii. $\downarrow x$ es compacto para todo $x \in X$.

Demostración. Si $\downarrow x \cap \downarrow y \neq \emptyset$ existe $w \in X$ tal que $x \in O(w)$ y $y \in O(w)$, esto es, $x = f^k(w)$ y $y = f^{\tilde{k}}(w)$ para ciertos $k, \tilde{k} \geq 0$. Si $k = \tilde{k}$ listo, de lo contrario, sin pérdida de generalidad asumimos $k > \tilde{k}$. Luego es claro que $x \in O(y)$ y entonces $y \in \downarrow x$. Ahora sea $v \in \downarrow y$, se tiene $v \leq y$, $y \leq x$ y por transitividad se sigue que $v \leq x$, con lo cual $v \in \downarrow x$, de donde se obtiene $\downarrow y \subseteq \downarrow x$.

ii. Sea $\{U_i\}_{i\in I}$ un cubrimiento por abiertos de $\downarrow x$, es decir, $\downarrow x \subseteq \bigcup_{i\in I} U_i$. Existe $i_0 \in I$ tal que $x \in U_{i_0}$, y como $f^{-1}(U_{i_0}) \subseteq U_{i_0}$ se tiene que $f^{-1}(x) \subseteq U_{i_0}$. Luego $f^{-n}(x) \subseteq U_{i_0}$ para todo $n \geq 0$ y se concluye $\downarrow x = \bigcup_{n\in\mathbb{N}_0} f^{-n}(x) \subseteq U_{i_0}$. Finalmente $\downarrow x$ es compacto.

■

Teorema 2.2: Sea (X, τ_f) un espacio primal. Las siguientes afirmaciones son equivalentes:

i. X es compacto, ii. $|\mathrm{Per}(f)| < \infty$ y $\forall\, x \in X\ \exists\, n \in \mathbb{N} : f^n(x) \in \mathrm{Per}(f)$.

($\natural$) $x \in O \implies \mathrm{cl}(x) = O$.

Demostración. (i.) $\Rightarrow$ (ii.) Claramente $\{\downarrow x\}_{x\in X}$ es un cubrimiento por abiertos de X, que como es compacto $\exists\, a_1, \cdots, a_n \in X$ tal que $X = \bigcup_{i=1}^{n} \downarrow a_i$. Por el inciso i. del lema anterior, sin pérdida de generalidad asumimos que $\downarrow a_i \cap \downarrow a_j = \emptyset$ para $i \neq j$. Ahora veamos que los a_i's son puntos periódicos y $\mathrm{Per}(f) = \bigcup_{i=1}^{n}\{f^k(a_i)\}_{k\geq 0}$. Para cada i, trivialmente $f(a_i) \in \bigcup_{k=1}^{n} \downarrow a_k$, con lo que $f(a_i) \in \downarrow a_j$ para algún $j \in [\![1, n]\!]$ y entonces $f(a_i) \leq a_j$. Más aún, es claro que $a_i \leq f(a_i)$ por lo tanto $a_i \leq a_j$ y luego $i = j$. Así, $f(a_i) \leq a_i$, de donde se obtiene $a_i \in \mathrm{Per}(f)$.

Ahora sea $\alpha \in \mathrm{Per}(f)$. Trivialmente $\alpha \in \bigcup_{k=1}^{n} \downarrow a_k$ por lo que $\alpha \leq a_i$ para algún $i \in [\![1, n]\!]$. Si $\pi = \min\limits_{\substack{r\in\mathbb{N}\\ f^r(\alpha)=\alpha}} r$ es el período de α, $a_i = f^k(\alpha)$ para cierto $k \in [\![0, \pi - 1]\!]$ y luego $f^{\pi-k}(a_i) = \alpha$. Se sigue que $\mathrm{Per}(f) = \bigcup_{i=1}^{n}\{f^k(a_i)\}_{k\geq 0}$, que claramente tiene cardinal finito.

Por último, si $x \in X$ $\exists\, i \in [\![1, n]\!]$ tal que $x \leq a_i$, con lo que $a_i = f^m(x) \in \mathrm{Per}(f)$ para cierto $m \in \mathbb{N}$.

(ii.) $\Rightarrow$ (i.) Es fácil ver que $X = \bigcup_{a\in\mathrm{Per}(f)} \downarrow a$, y es compacto pues es unión finita de compactos.

$\blacksquare$

Teorema 2.3: Un espacio primal no es w-R_0 si y solo si es compacto y tiene una única órbita periódica.

Demostración. Se deduce de los últimos dos teoremas.

$\blacksquare$

Observación 2.4: Todo espacio primal no compacto es w-R_0.

En [5] se muestra:

Teorema 2.4: Sean X, Y dos espacios topológicos y $f : X \longrightarrow Y$ cerrada e inyectiva. Si X es w-R_0 entonces Y lo es también.

Definición 2.8: Un espacio topológico X se dice "supercompacto" si todo cubrimiento por abiertos contiene a X. Esto es, si $\{U_i\}_{i\in I}$ son abiertos que cumplen $X = \bigcup_{i\in I} U_i$, entonces $X = U_i$ para algún $i \in I$.

Lema 2.5: Sea (X, τ_f) un espacio primal y $x \in X$. Si $\downarrow x = X$ entonces x es un punto periódico.

Demostración. $X = \downarrow x$ si y solo si para todo $y \in X$ se tiene $x \in \mathrm{cl}(y)$, por lo que $\mathrm{cl}(x) \subseteq \mathrm{cl}(y)$ para todo $y \in X$ (pues $\mathrm{cl}(x)$ es el cerrado más chico que contiene a x). Ahora, si $A \neq \emptyset$ es cerrado, $A = \mathrm{cl}(A) = \bigcup_{a\in A} \mathrm{cl}(a)$ y queda claro que $\mathrm{cl}(x) \subseteq A$. Luego $\mathrm{cl}(x)$ es un conjunto minimal de f y por lema 2.3 se sigue que x es un punto periódico.

■

Teorema 2.5: Un espacio primal es supercompacto si y solo si es compacto y tiene una única órbita periódica.

Demostración. ($\Rightarrow$) Si (X, τ_f) es supercompacto, trivialmente es compacto, más aún, como $\{\downarrow x\}_{x\in X}$ es un cubrimiento por abiertos de X se tiene que existe $q \in X$ tal que $\downarrow q = X$. Por el lema anterior q es un punto periódico. Ahora, si $O = O(o)$ es una órbita periódica, $x \in \downarrow q$ implica $q \in O(x)$ para todo $x \in X$ y por lo tanto $q \in O(o)$, de donde se obtiene $O = O(q)$ y luego solo hay una única órbita periódica.

($\Leftarrow$) Sea $O = O(p)$ la única órbita periódica de X. Como es compacto, para todo $x \in X$ existe $n \in \mathbb{N}$ tal que $f^n(x) \in \mathrm{Per}(f)$ por el teorema 2.2, en cuya demostración vimos que $\mathrm{Per}(f)$ es la unión de todas las órbitas periódicas, en este caso, $\mathrm{Per}(f) = O$, con lo cual existe $m \in \mathbb{N}$ tal que $f^m(x) = p$ y entonces $x \in f^{-m}(p) \subseteq \downarrow p$. Se sigue que $X = \downarrow p$. Así, sea $\{U_i\}_{i\in I}$ un cubrimiento por abiertos de X. Existe $i_0 \in I$ tal que $p \in U_{i_0}$ y como $f^{-1}(U_{i_0}) \subseteq U_{i_0}$, $f^{-1}(p) \subseteq U_{i_0}$. Se sigue que $\downarrow p \subseteq U_{i_0}$ con lo cual $X = U_{i_0}$ concluyendo que es supercompacto.

■

Definición 2.9: Dado un espacio topológico X, $x \in X$ se dice un "punto aislado" si el singulete $\{x\}$ es abierto.

Recordar que en un espacio topológico X un subconjunto A se dice denso si $\overline{A} = X$.

Definición 2.10: Un espacio topológico se dice "resoluble" si es la unión de dos subconjuntos densos disjuntos.

La demostración del siguiente teorema se puede encontrar en [4].

Teorema 2.6: Un espacio topológico de Alexandroff es resoluble si y solo si no tiene puntos aislados.

Teorema 2.7: Sea el espacio primal (X, τ_f) con f suryectiva. Si $\{x\}$ es abierto, entonces es cerrado.

Demostración: Como $\{x\}$ es abierto, $X \setminus \{x\}$ es cerrado y por el inciso ii. del lema 2.2 se tiene $f(X \setminus \{x\}) \subseteq X \setminus \{x\}$. Como f es suryectiva se sigue que $f(x) = x$ y por lo tanto $\{x\}$ es cerrado.

■

Corolario 2.2: Sea (X, τ_f) con f suryectiva. Si no tiene puntos cerrados entonces es resoluble.

Demostración. Si ningún singulete es cerrado, ninguno es abierto por el teorema anterior y entonces no hay ningún punto aislado. Se sigue que X es resoluble.

Observación 2.5: Sea col la función de Collatz (que claramente es suryectiva). El espacio primal $(\mathbb{N}, \tau_{\text{col}})$ es resoluble.

En efecto, $\nexists\, n \in \mathbb{N}$ tal que $\text{col}(n) = n$, con lo que ningún punto es cerrado.

En [14] se demuestra el siguiente teorema:

Teorema 2.8: Sea $(\mathbb{N}, \tau_{col})$ el espacio primal generado por col. La conjetura de Collatz es cierta si y solo si $\mathbb{N} = \Omega_1 \cap \Omega_2$ con $\Omega_1 \subseteq \downarrow 1$ y Ω_1, Ω_2 son subconjuntos densos disjuntos.

Finalmente, los siguientes teoremas muestran equivalencias de la conjetura de Collatz desde la topología.

Teorema 2.9: Las siguientes afirmaciones son equivalentes:

i. Se cumple la conjetura de Collatz, ii. $(\mathbb{N}, \tau_{col})$ no es w-R_0,

iii. $(\mathbb{N}, \tau_{col})$ es compacto y tiene una única órbita periódica,

iv. $(\mathbb{N}, \tau_{col})$ es supercompacto,

v. Cualquier cerrado F en τ_{col} cumple $1 \in F$.

Demostración. (i.) $\Rightarrow$ (ii.) Para todo $n \in \mathbb{N}$ existe $k \in \mathbb{N}$ tal que $\text{col}^k(n) = 1$ y hay una única órbita periódica $O(1)$. Luego por el teorema 2.1 $(\mathbb{N}, \tau_{col})$ no es w-R_0.

(ii.) $\Leftrightarrow$ (iii.) Directo del teorema 2.3.

(iii.) $\Leftrightarrow$ (iv.) Directo del teorema 2.5.

(iv.) $\Rightarrow$ (v.) Si vale iv. entonces se cumple iii. y por lo tanto hay una única órbita periódica, que es $O(1)$. Más aún, como es compacto, por teorema 2.2 se tiene que para todo $n \in \mathbb{N}$ existe $k \in \mathbb{N}$ tal que $f^k(n) = 1$. Ahora, si F es cerrado, por iii. del lema 2.2 se tiene $F = \overline{F} = \bigcup_{n \geq 0}\{f^n(x) \mid x \in F\}$ y claramente $1 \in F$.

(v.) $\Rightarrow$ (i.) $1 \in \uparrow n$ para todo $n \in \mathbb{N}$ (pues $\uparrow x = \text{cl}(x)$ que es cerrado), con lo que $1 \in O(n)$ y luego se cumple la conjetura de Collatz.

■

Recordar que dado $X \neq \emptyset$ y $p \in X$, se define la topología de "punto excluido" como $\tau_{E(p)} = \{S \subseteq X \mid p \notin S\} \cup X$.

Teorema 2.10: La conjetura de Collatz es cierta si y solo si existe $h : (\mathbb{N}, \tau_{col}) \longrightarrow (\mathbb{N}, \tau_{E(1)})$ cerrada e inyectiva.

Demostración. ($\Rightarrow$) Por el inciso v. del teorema anterior, para todo conjunto cerrado F, $1 \in F$. Luego definiendo $h(x) = x$ se tiene $1 \in h(F)$ y por lo tanto h es cerrada e inyectiva.

($\Leftarrow$) Sea $h : (\mathbb{N}, \tau_{col}) \longrightarrow (\mathbb{N}, \tau_{E(1)})$ cerrada e inyectiva. Notar que $(\mathbb{N}, \tau_{E(1)})$ no es w-R_0 (pues $1 \in \mathrm{cl}(x)$ para todo $x \in \mathbb{N}$ y por lo tanto la intersección es no vacía). Por el teorema 2.4 se sigue que $(\mathbb{N}, \tau_{col})$ no es w-R_0 y entonces vale la conjetura de Collatz.

$\blacksquare$

Con esto se da por terminada la sección del abordaje a Collatz usando topología.

3. Resultado de Terence Tao.

Definición 3.1: Sea $\mathrm{col} : \mathbb{N} \longrightarrow \mathbb{N}$ la función de Collatz. Dado $n \in \mathbb{N}$ se define $\mathrm{col}_{\text{mín}}(n) = \inf_{k \in \mathbb{N}} \mathrm{col}^k(n)$, que es el mínimo elemento de la órbita de n.

Observar que si la conjetura de Collatz es cierta clarmaente $\mathrm{col}_{\text{mín}}(n) = 1$ para todo $n \in \mathbb{N}$.

Teorema 3.1: *Casi todas las órbitas de Collatz alcanzan valores casi acotados.* Sea $f : \mathbb{N} \longrightarrow \mathbb{R}$ cualquier función con $\lim_{n\to\infty} f(n) = \infty$. Luego $\mathrm{col}_{\text{mín}}(n) < f(n)$ para casi todo $n \in \mathbb{N}$ (en el sentido de densidad logarítmica).

La demostración de este último se puede encontrar en [13]. "Esto es lo máximo posible que se puede avanzar en el problema de Collatz sin realmente resolverlo" (Terence Tao).

Referencias

[1] JF Alves, MM Graca, ME Sousa Dias, and J Sousa Ramos. A linear algebra approach to the conjecture of collatz. *Linear algebra and its applications*, 394:277–289, 2005.

[2] David Applegate and Jeffrey C Lagarias. Density bounds for the 3x+ 1 problem. ii. krasikov inequalities. *Mathematics of computation*, 64(209):427–438, 1995.

[3] David Barina. Convergence verification of the collatz problem. *The Journal of Supercomputing*, 77(3):2681–2688, 2021.

[4] Intissar Dahane, Sami Lazaar, Tom Richmond, and Tarek Turki. On resolvable primal spaces. *Quaestiones Mathematicae*, 42(1):15–35, 2019.

[5] Giuseppe DIMAIO. A separation axiom weaker than r0. *INDIAN JOURNAL OF PURE & APPLIED MATHEMATICS*, 16(4):373–375, 1985.

[6] Yasuaki Ito and Koji Nakano. Efficient exhaustive verification of the collatz conjecture using dsp blocks of xilinx fpgas. *International Journal of Networking and Computing*, 1(1):49–62, 2011.

[7] Jeffrey C Lagarias. The 3x+ 1 problem: An annotated bibliography. *preprint*, 2004.

[8] Gary T Leavens and Mike Vermeulen. 3x+ 1 search programs. *Computers & Mathematics with Applications*, 24(11):79–99, 1992.

[9] Simon Letherman, Dierk Schleicher, and Reg Wood. The 3n+ l-problem and holomorphic dynamics. *Experimental Mathematics*, 8(3):241–251, 1999.

[10] Daniel Nichols. Analogues of the 3 x+ 1 problem in polynomial rings of characteristic 2. *Experimental Mathematics*, 27(1):100–110, 2018.

[11] Robertson E.F O'Connor, J.J. Lothar collatz. *St Andrews University School of Mathematics and Statistics, Scotland*, 2006.

[12] Fatemah Ayatollah Zadeh Shirazi and Nasser Golestani. Functional alexandroff spaces. *Hacettepe Journal of Mathematics and Statistics*, 40(4):515–522, 2011.

[13] Terence Tao. Almost all orbits of the collatz map attain almost bounded values. In *Forum of Mathematics, Pi*, volume 10. Cambridge University Press, 2022.

[14] Jorge Vielma and Angel Guale. A topological approach to the ulam–kakutani–collatz conjecture. *Topology and its Applications*, 256:1–6, 2019.

[15] Günther J Wirsching. *The dynamical system generated by the 3n+ 1 function*. Springer, 2006.