Criba de Primalidad Basada en Frecuencias Cíclicas

1. Introducción

El estudio de los números primos ha sido una constante en la historia de las matemáticas, no solo por su importancia teórica, sino también por su aplicación en áreas como la criptografía, la teoría de la información y la computación. A lo largo del tiempo, diversos métodos han sido propuestos para la generación o verificación de primos, destacando entre ellos la criba de Eratóstenes, los test de primalidad probabilísticos y los algoritmos de factorización.

Este trabajo presenta un enfoque alternativo, basado en una observación empírica de patrones numéricos, que permite detectar números primos mayores que 3 a partir de propiedades cíclicas asociadas a los números de la forma $6n \pm 1$. La clave del método radica en la aparición de **frecuencias regulares** dentro de una secuencia generada exclusivamente por operaciones aritméticas simples.

2. Fundamento Teórico

Todo número natural mayor que 1 puede clasificarse según su residuo módulo 6. De estas, solo los números de las formas $6n \pm 1$ pueden ser primos cuando $n > 0$, ya que el resto corresponden a múltiplos de 2 o de 3.

3. Generación de la Secuencia Base

Este patrón alternante de sumas $+2, +4$ produce la sucesión exacta de los números $6n \pm 1$. Se asocian dos frecuencias a cada número base:

- $f_2 = 4n \pm 1$
- $f_1 = f_2 + 4n$

4. La Criba por Frecuencias

Cada número base genera un patrón de cancelación alternando f_1 y f_2, eliminando compuestos a partir de su cuadrado. Esta cancelación respeta el principio modular y se aplica solo dentro del dominio $6n \pm 1$.

5. Evaluación Directa de Primalidad

Un número N se evalúa verificando si existe alguna combinación de frecuencias f_1, f_2 tal que:

$$\frac{N - \text{offset}}{f_1 + f_2} \in \mathbb{Z}$$

Ejemplo: 77 con base 5 da colisión inmediata:

$$\frac{77 - 7}{10} = 7 \Rightarrow \text{Compuesto}$$

6. Optimización Observada y Reutilización de Patrones

Los patrones de cancelación tienden a repetirse en bloques estructurados, especialmente al alcanzar productos de primos. Esta propiedad permite reutilización de patrones y sugiere eficiencia computacional.

7. Validación Computacional (Nota metodológica)

Se han realizado pruebas computacionales en Python para confirmar la precisión del modelo. Versiones anteriores permitieron verificar números como $2^{127} - 1$ de forma muy eficiente. El modelo actual da prioridad a la fidelidad estructural más que al rendimiento.

8. Discusión

Ventajas:

- Independiente de la factorización.
- Cubre todos los primos mayores que 3.
- Modularidad reutilizable.

Limitaciones:

- No analiza los primos 2 y 3.
- Requiere formalización matemática completa.

Potencial: Prometedor como herramienta educativa y de investigación.

9. Conclusiones

Este modelo ofrece un enfoque estructurado, replicable y alternativo al análisis de primalidad. Su lógica modular permite tanto evaluaciones teóricas como futuras optimizaciones en implementaciones reales.

10. Análisis Predictivo de Incidencia de Frecuencias

Además del recorrido secuencial, el modelo permite una forma predictiva para determinar si un número será alcanzado por una frecuencia específica. Se aplica la expresión general:

$$\frac{N - \text{offset}}{f} \in \mathbb{Z}$$

donde:

- $\text{offset}_1 = f_1 + i$
- $\text{offset}_2 = f_1 + f_2 + i$

Las fórmulas utilizadas anteriormente:

$$\frac{N - f_1 - i}{f}, \quad \frac{N - (f_1 + f_2) - i}{f}$$

son casos particulares de este análisis predictivo.

Apéndice: Ejemplo de Código en Python

Análisis con Predicción Directa

```
def verificar_colision(N, f1, f2, index, max_pasos=10):
f = f1 + f2
offset = f1 + index
for paso in range(max_pasos):
if (N - offset) % f == 0:
return True  # Colision detectada
offset += f1 if paso % 2 == 0 else f2
return False
```

Este código aplica el enfoque predictivo descrito en la sección 9.1, verificando si un número N es alcanzado por un patrón alternante de frecuencias.