# Algoritmos simbólico-numéricos para el aislamiento de raíces reales.

Marta González Marquina.

Máster en Ingeniería Matemática.
Facultad de Ciencias Matemáticas. Universidad Complutense de Madrid.

Noviembre, 2013

- Aislaremos las raíces de cualquier polinomio

$$f(x) = \sum_{i=0}^{n} a_i x^i.$$

$\tau$ será el número de bits del mayor $|a_i|$.

## 1637. Regla de los signos de Descartes.

Sea $f(x) = \sum_{i=0}^{n} a_i x^i$ un polinomio de grado $n$ con coeficientes reales. Entonces,

$N_f(0,\infty) = V(a_0, a_1, \ldots, a_n) - 2\kappa$, para algún $\kappa \in \mathbb{N}_0$.

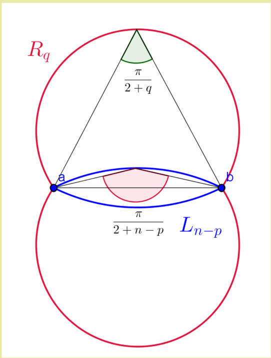
- **Cambio de Jacobi.**

  $I = (c, d)$. $V_f(I) = V(b_0, b_1, \ldots, b_n)$, siendo

$$g(x) = \sum_{i=0}^{n} b_i x^i = (1+x)^n f\left(\frac{cx+d}{x+1}\right).$$

  Así podemos aplicar la Regla de Descartes a cualquier intervalo $I$.

## Regiones de Obreshkoff.

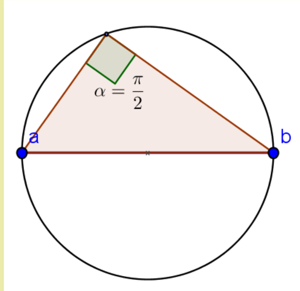


n° raíces en $L_n \leq V_f(I) \leq$ n° raíces en $R_n$

- **Akritas** desarrolla el método más utilizado para aislamiento de raíces reales.
  El criterio de parada del método de Akritas se basa en los dos siguientes resultados.

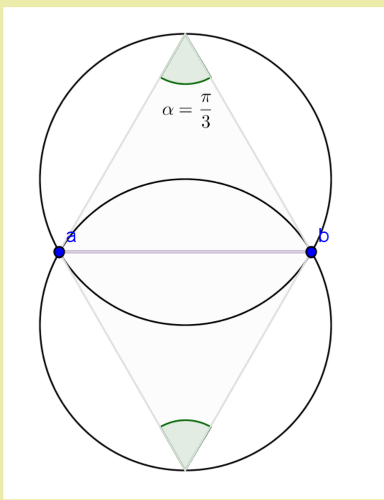
## Teorema del círculo.

No hay ninguna raíz de $f$ en el círculo $\rightarrow V_f(I) = 0$



## Teorema de los dos círculos.

Hay una única raíz de $f$ en los círculos $\rightarrow V_f(I) = 1$



## Método de Akritas. Descartes con bisección.

- **Entrada:** $f(x) \in \mathbb{R}[x]$. $I_0 = (a, b)$.
- **Criterio de parada.** Para cada intervalo $I$ comprobamos:
  - $V_f(I) = 0 \rightarrow I$ no contiene raíces. Descartamos $I$.
  - $V_f(I) = 1 \rightarrow I$ contiene una única raíz real. $I$ es intervalo aislante.
- Para cada $I$ con $V_f(I) > 1$, realizamos bisección: $I = I_1 \cup I_2$. Calculamos $V_f(I_i)$, $i = 1, 2$ con el **algoritmo de De Casteljau.**
- **Problemas:**
  - Clúster de raíces: tarda mucho en separarlas.
  - Raíces complejas muy juntas: $V_f(I) = 2$ hasta que no salen las raíces de la lente de Obreshkoff.
- **Complejidad:** $O(n^5(\tau + \log n)^2)$.

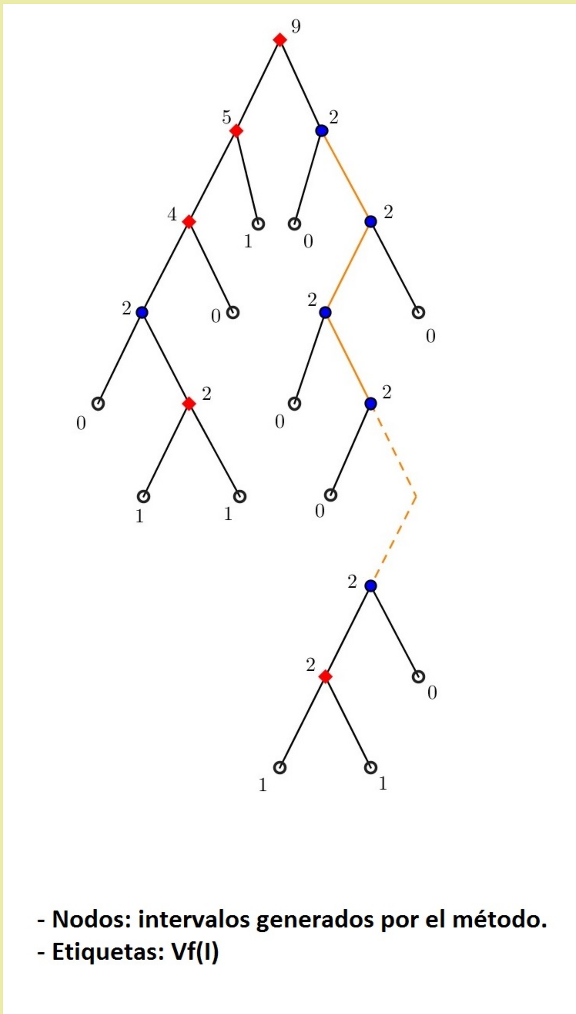
## Método de Sagraloff. NEWDSC.

- **2012.** *Sagraloff* propone un nuevo método más rápido en las situaciones con clústers.
- Este método viene inspirado por la mejora del método de falsa posición propuesto por *Abbot* en el 2012: **QIR.**

EJEMPLO DE QIR. El polinomio $x^5 - 2$ contiene una raíz real en el intervalo $I = (1, 2)$. Con $x_0 = 1$, cinco iteraciones de Newton producen una aproximación con un denominador de más de 500 dígitos, mientras que seis iteraciones de QIR obtenemos una precisión similar y la solución es la siguiente:

$$\left[\frac{4705}{4096}, \frac{2353}{2048}\right]$$

- **Idea:** Los clúster de $m$ raíces se comportan como una raíz $m$-múltiple, por lo que en estos casos usaremos iteraciones de Newton *a la Abbot* en lugar de pasos de bisección.
- Alcanzaremos convergencia cuadrática.
- La complejidad es óptima $\tilde{O}(n^3\tau)$ y se basa en utilizar Newton para reducir"la longitud de las ramas del árbol de recursión del algoritmo anterior en las que no disminuyen las variaciones de signo (equivalente a la existencia de un clúster):




- **Algoritmo.**
  - Sea $v = V_f(I)$, $I = (a, b)$ en una cierta iteración.
  - Puede que exista un clúster de $v$ raíces, así que calculamos

$$\lambda = t - v\frac{f(t)}{f'(t)}$$

  para algún $t \in [a, b]$ (por ejemplo, $t = a$ ó $t = b$).
  - Descomponemos $I$ (a la Abbott) en $N_I \geq 4$ subintervalos.
  - $I'$ es aquel que contiene a $\lambda$: $I' \subset I$, $\lambda \in I'$ y $w(I') << w(I)$.
  - Si $V_f(I') = v$ seguimos con $I'$ y $N_{I'} = N_I^2$.
  - Si $V_f(I') \neq v$, bisecamos $I = I_1 \cup I_2$ y $N_{I_i} = max(4, N_I)$.
  - Si existía el clúster, hemos conseguido una convergencia muy rápida; si no, al menos hemos bisecado.

- A.G. Akritas, G.E. Collins, "Polynomial real root isolation using Descartes' Rule of Signs", ACMSAC76. 1976.
- M. Sagraloff, "When Newton meets Descartes: a simple and fast algorithm to isolate the real roots of a polynomial", ISAAC'12. Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation.
- M.E. Alonso, A. Galligo, "A root isolation algorithm for sparse univariate polynomials", ISAAC'12. Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation.

## Método de GALLIGO.

- Método especializado en polinomios *sparse* (grado muy alto y pocos monomios).
- **Características:**
  - **Raíces virtuales.** Raíces reales de $f$ y algunas raíces de sus $\mathbb{F}$-derivadas, que proporcionan una igualdad en la fórmula de *Budan-Fourier*. Las raíces virtuales dependen de forma continua de los coeficientes de $f$.
  - **Nuevo criterio de parada** basado en que los inversos de las raíces reales de $f$ son raíces de su recíproco ($Rf$); hecho que generalmente no es cierto para las virtuales.

EJEMPLO CON ARITMÉTICA EXACTA.
Aplicamos nuestra implementación en SAGE de una versión simplificada del método de Galligo al polinomio

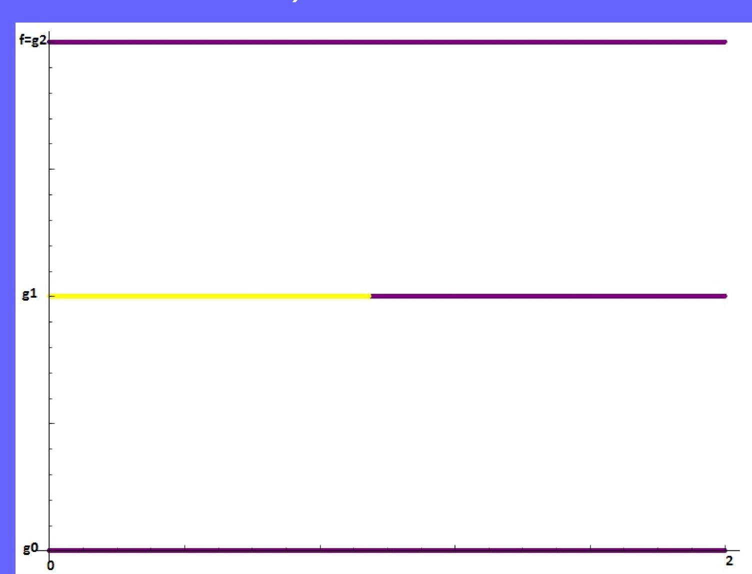
$$f(x) = (x^{150} + x^2 - 10^3x + 1)(x^2 - 10^3x + 1).$$

Obtenemos que el polinomio tiene una raíz real en el intervalo $[1, 2]$, dos raíces reales en el intervalo $[0, 1]$ con más de 100 cifras iguales y otra mayor. Al aplicar la función *f.roots()* de SAGE (que sigue el método de Akritas), obtenemos que en el intervalo $[0, 1]$ hay una única raíz real de multiplicidad 2, lo que es falso.

EJEMPLO CON ARITMÉTICA APROXIMADA.
Aplicando la misma implementación en SAGE adaptada a una aritmética aproximada, tratamos ejemplos en los que es difícil la evaluación como en el siguiente:

$$f(x) = x^{317811} - cx^{196418} + 1.$$

Para $c > 1{,}94$, el polinomio tiene dos raíces reales muy juntas. Así, para $c = 2$ obtenemos que $f$ tiene una raíz real en 1 y otra a su derecha. El método de Akritas (tanto en Maple16 como en SAGE) y nuestra implementación del método de Sagraloff no consiguen aislar las raíces de este polinomio.
Para ver lo próximas que están las raíces, a continuación mostramos la *Tabla de Budan* de $f$ (en amarillo los puntos donde $f < 0$ y en morado $f > 0$).
Representando la Tabla de Budan para el intervalo $[0, 2]$ (no vemos nada):



En la tabla para el intervalo $[1 - 1/1000000, 1 + 1/100000]$ vemos perfectamente las dos raíces reales: