Búsqueda de Valores Extremos en una Gráfica Regular de Funciones Booleanas Balanceadas

Daniel López Fernández
Guillermo Morales Luna

Centro de Investigación y de Estudios Avanzados - IPN
Departamento de Ingeniería Eléctrica, Sección Computación

Definición

Sea $\mathbb{F}_2 = \{0, 1\}$ y $\mathbb{F}_2^n$ el espacio de vectores en $\mathbb{F}_2$.
Una función booleana es de la forma :

$$f : \mathbb{F}_2^n \to \mathbb{F}_2$$

Espacio de Funciones Booleanas

$\forall n \in \mathbb{N}$: se cumple: $car\{\mathbb{F}_2\} = 2^n \Rightarrow car\{\mathbb{F}_2^n\} = 2^{2^n}$

Definición

Sea $f \in \mathbb{F}_2^n$

Soporte : $Spt(f) : \{\delta \in \mathbb{F}_2^n | f(\delta) = 1\}$

Parte Nula : $Null(f) : \{\delta \in \mathbb{F}_2^n | f(\delta) = 0\}$

Se dice que f es **balanceada** si:

$$card\{Spt(f)\} = card\{Null(f)\}[3]$$

Definición

- Sean $f, h : \mathbb{F}_2^n \to \mathbb{F}_2$. f en Forma Normal Algebraica (FNA) es:

$$f(x) = \bigoplus_{a_1,\cdots,a_n \in \mathbb{F}_2^n} h(a_1,\cdots,a_n) x_1^{a_1},\cdots,a_n^{a_n}$$

- El **grado algebraico** $gr(f)$: El número de variables en el término más grande de $x_1^{a_1},\cdots,a_n^{a_n}$ en FNA(f).
- Función **afín**: De grado a lo sumo 1.
- Función **lineal**: Afín con un término constante igual a cero.
- **No Linealidad** de $f : \mathbb{F}_2^n \to \mathbb{F}_2$ [4],[2]

$$min\{DH(f, l) | l \in A\}$$

Gráfica

- (f, g) arista en $\mathcal{G} \Leftrightarrow f \oplus g$ es balanceada.
- Num. de Nodos: $\binom{2^n}{2^{n-1}}$
- Num. de vecinos: $\binom{2^{n-1}}{2^{n-2}}$
- Diámetro igual a 2

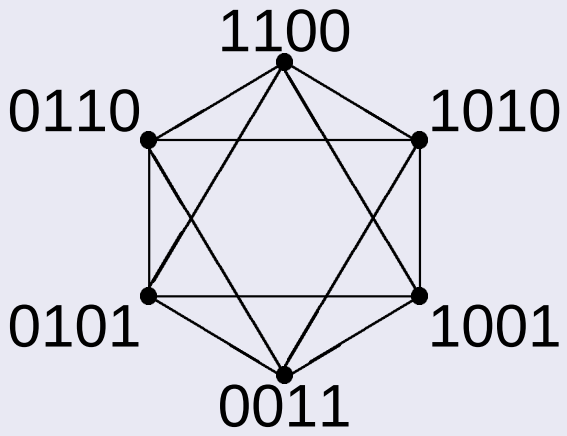


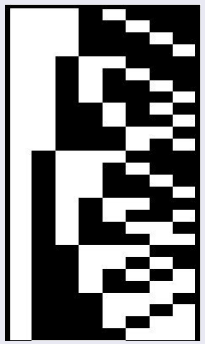
Figura: Gráfica $\mathcal{G}(2)$.

Definición

Sea $f \in \mathbb{F}_2^n$:

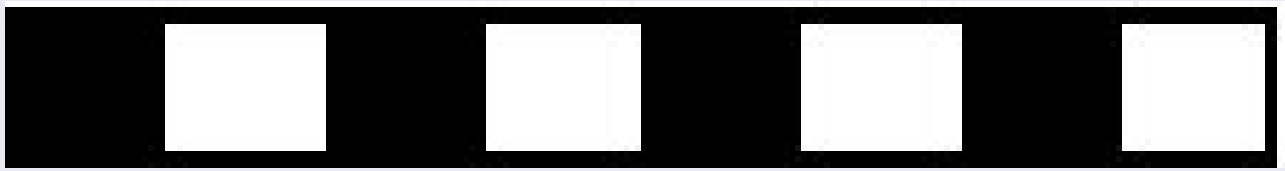
- $f(\delta) = 0$ ■
- $f(\delta) = 1$ □
- $f(\delta) = 0/1$ ■

Funciones de grado 5



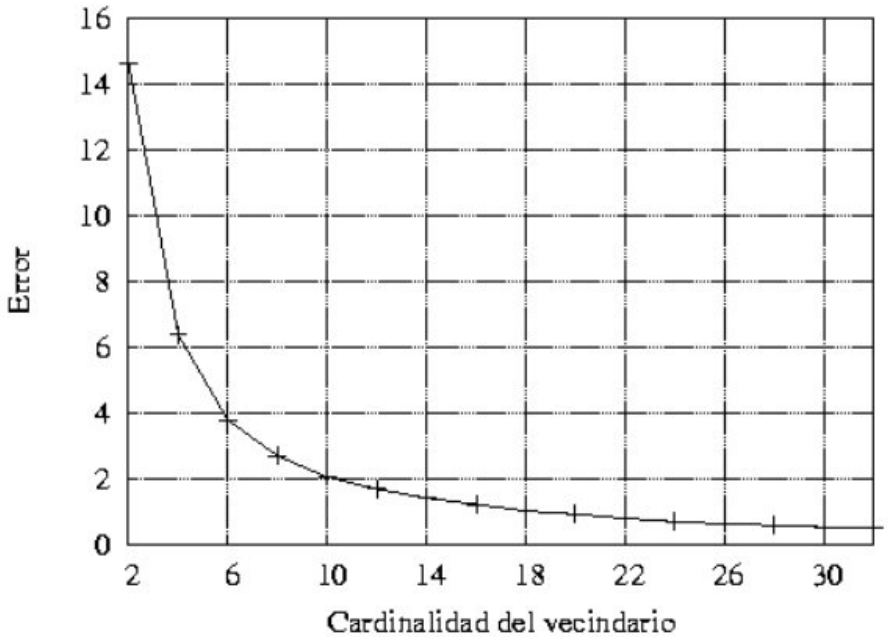
Ejemplo

$$f = \begin{pmatrix} 000 & 001 & 010 & 011 & 100 & 101 & 110 & 111 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}.$$



Búsqueda Local [1]

begin
f_c = función inicial;
repeat
seleccionar $F \subset \mathbb{F}_2^n; card(F) = k$
if $\exists f \in F : nl(f) \geq nl(f_c)$ **then**
$f_c := f$
end if
until ninguna mejora se haya hecho
end

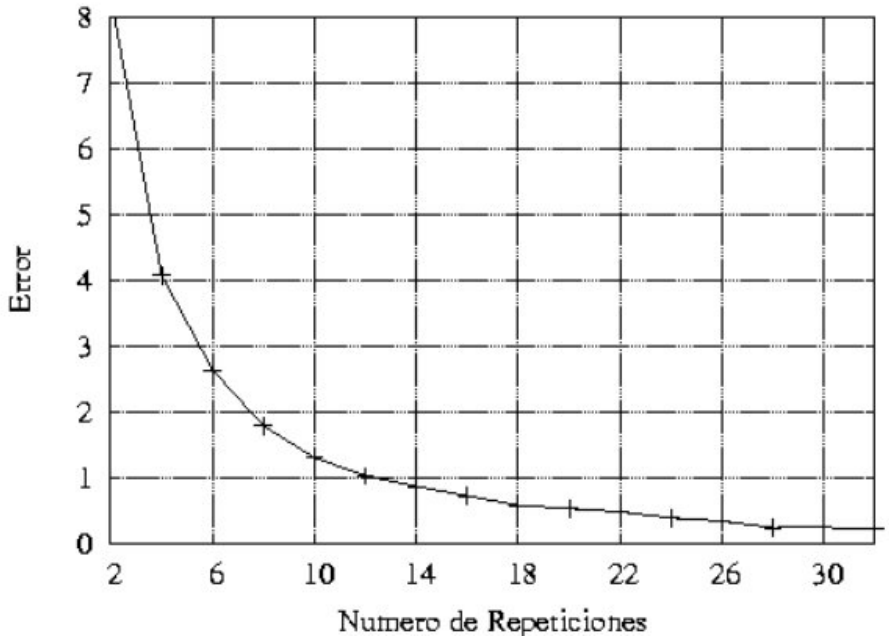


Parámetros de los algoritmos

- $f(n) = 1.30381n^3 - 14.8007n^2 + 50.8046n - 44.2.$

Recocido Simulado[1]

begin
f_i = función inicial;
repeat
  for $l := 1$ to k **do**
    seleccionar $f_j \in \mathbb{F}_2^n$;
    if $nl(f_j) \leq nl(f_i)$ **then**
      $f_i := f_j$
    else
      if
      $exp\left(\frac{1}{c_k}\left(nl(f_i) - nl(f_j)\right)\right) >$
      $random[0, 1)$ **then**
        $f_i := f_j$
      end if
    end if
  end for
  $c_k := c_k \cdot \alpha$
until Criterio de paro
end



Parámetros de los algoritmos

- $f(n) = 1.30381n^3 - 14.8007n^2 + 50.8046n - 44.2.$
- $k = 0.550137\ n^2 - 3.81909\ n + 7.4011.$
- $c_k = 0.0731352\ n^4 - 1.08566\ n^3 + 5.17046\ n^2 - 8.24651\ n + 4.33334.$

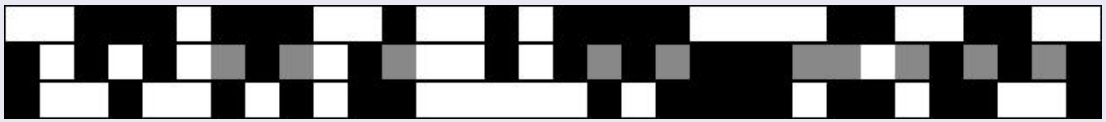
Valores Extremos con B.L y R.S.

Grado n	Búsqueda Local	Recocido Simulado
5	10	12
6	24	24
7	52	52
8	108	110
9	226	230
10	468	472
11	956	960
12	1948	1952
13	3948	3952
14	7974	7976
15	16056	16068
16	32288	32304

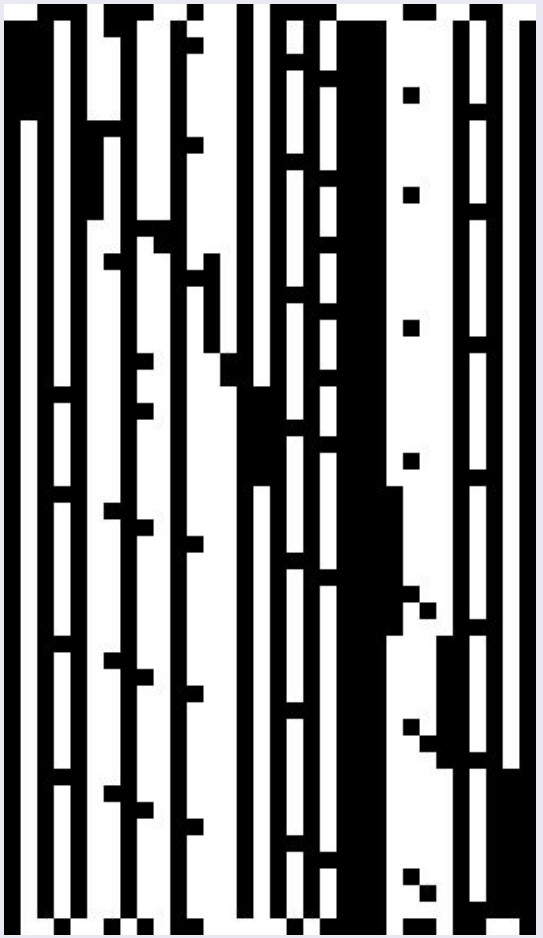
Algoritmo de k-means

begin
Sea (**k**) número de cúmulos;
Seleccionar **k** puntos como *centros*.
repeat
    Asignar cada punto al centro del cúmulo mas cercano.
    Calcular el nuevo centro del cúmulo.
until Centros no se modifiquen.
end

Resultado



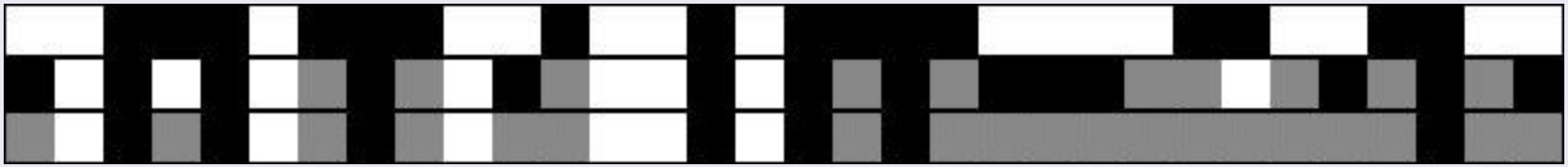
Ejemplo



Algoritmo de Selección

begin
sea $G :=$ patrón producido por k-means (k=3).
sea $g := 0$; \ \ g es el vecino producido
sea $s := 0$; \ \ s el contador de vls. 1
sea $H :=$ lista vacía ; \ \ H es una lista de vls. 0/1
for $\delta \in \mathbb{F}_2^2$ **do**
  if $f(\delta) == 1$ y $G(\delta) ==$ negro **then**
    sea $g(\delta) := 1$; $s++$;
  end if
  if $f(\delta) == 0$ y $G(\delta) ==$ blanco **then**
    sea $g(\delta) := 0$;
  else
    agregar δ a la lista H;
  end if
end for
para $2^{n-1} - s$ elementos $\delta \in H$ sea $g(\delta) = 1$;
función de salida g
end

Resultado



Resultados de la Heurística

Grado	S.A.	Heurística
5	12	12
6	24	26
7	52	52
8	110	110
9	230	230
10	472	470
11	960	960
12	1952	1952
13	3952	3954
14	7974	7974

n	R.S. en $\mathcal{G}(n)$	Heurística
9	220	226
9	222	226
9	216	226
9	224	224
9	216	224
9	224	226
9	224	222
9	224	226
9	224	226
9	218	226

Nuevos parámetros de los algoritmos [1]

- $f(n) = 1.30381\, n^3 - 14.8007\, n^2 + 50.8046\, n - 44.2.$
- $f(n) = 2.87229\, n^2 + 50.7208\, n - 264.03.$

Heurística

$card\{\mathbb{G}\}$	Vecinos de $f \in \mathbb{G}$	$nl(f_0) = nl(f_1) - 2$	Patrón
$\binom{2^n}{2^{n-1}}$	$\binom{2^{n-1}}{2^{n-2}}$		

Figura: Reducción del número de vecinos

Características

- Reducción del número de vecinos.
- Los óptimos globales siempre pueden ser alcanzados.
- Los patrones aumentan probabilidad de localizar óptimos.
- Recocido Simulado escapa de óptimos locales.

Conclusiones

- El algoritmo de búsqueda local tiene un buen desempeño en espacio de búsqueda moderados.
- La cardinalidad del espacio de funciones booleanas aumenta la complejidad de los algoritmos de optimización.
- El análisis de cúmulos mejora el desempeño pero aun no es óptimo.
- La heurística puede ser competitiva para n pequeños.
- Es necesario comprobar que se puede extender a n mayores.


Emile Aarts and Jan Korst.
Simulated annealing and Boltzmann machines: a stochastic approach to combinatorial optimization and neural computing.
John Wiley & Sons, Inc., New York, NY, USA, 1989.


Thomas Johansson and Enes Pasalic.
A construction of resilent functions with high nonlinearity.
IMA International Conference, 1746:35–44, 2000.


Ren Kui, Park Jaemin, and Kim Kwangio.
On the construction of cryptographically strong boolean functions with desirable trade-off.
Journal of Zhejiang University SCIENCE, 6A(5):358–364, 2005.


Piotr Porwik.
The spectral test of the boolean function linearity.
Journal Appl, Math. Comput Sci, 13:567–575, 2003.