Departamento de Arquitectura Instituto de Computación

Universidad de la República

Montevideo - Uruguay

Notas de Teórico

Álgebra de Boole

Arquitectura de Computadoras

(Versión 4.3a - 2016)

4 ALGEBRA DE BOOLE

4.1 Introducción.

El álgebra de Boole es una herramienta de fundamental importancia en el mundo de la computación. Las propiedades que se verifican en ella sirven de base al diseño y la construcción de las computadoras que trabajan con objetos cuyos valores son discretos, es decir las computadoras digitales, en particular las binarias (en las cuales los objetos básicos tienen solo 2 valores posibles) las que son, en definitiva, la totalidad de las computadoras de uso corriente.

Desde ya adelantemos que no se verán aquí detalles formales de la construcción algebraica, ni todas las propiedades que se verifican, así como tampoco todos los métodos de síntesis de funciones booleanas que habitualmente se incluyen en este tema en cursos de lógica y/o diseño lógico.

Como toda álgebra, la de Boole parte de un cuerpo axiomático, el cual puede adquirir diversas formas, variando la cantidad y calidad de los axiomas. Aquí en particular tomaremos uno: el propuesto por Huntington en 1904 que tiene la ventaja de ser consistente e independiente.

4.2 Axiomas.

1. Existe un conjunto G de objetos, sujetos a una relación de equivalencia, denotada por "=" que satisface el principio de sustitución.

   Esto significa que si $a = b$, b puede sustituir a a en cualquier expresión que la contenga, sin alterar la validez de la expresión.

2. (a) Se define una regla de combinación "+" en tal forma que $a + b$ está en G siempre que tanto a como b lo estén.
   (b) Se define una regla de combinación "·" en tal forma que $a \cdot b$ está en G siempre que tanto a como b lo estén.

3. *Neutros*.
   (a) Existe un elemento 0 en G tal que para cada a de G: $a + 0 = a$
   (b) Existe un elemento 1 en G tal que para cada a de G: $a \cdot 1 = a$

4. *Conmutativos*.
   Para todo par de elementos a y b pertenecientes a G se cumple:

   (a) $a + b = b + a$
   (b) $a \cdot b = b \cdot a$

5. *Distributivos*.
   Para toda terna de elementos a, b, c pertenecientes a G se cumple:

   (a) $a + (b \cdot c) = (a + b) \cdot (a + c)$
   (b) $a \cdot (b + c) = a \, . \, b + a \cdot c$

6. *Complemento*.
   Para cada elemento a de G existe un elemento $\bar{a}$ tal que:
   $$a \cdot \bar{a} = 0$$
   $$a + \bar{a} = 1$$

7. Existen por lo menos dos elementos x, y en G tal que $x <> y$

Existe similitud de muchos de estos postulados con los del álgebra común. Sin embargo, la primera de las reglas distributivas (sobre la suma) y la existencia del complemento diferencian en forma fundamental esta álgebra de la común.

4.3 Modelo aritmético.

El ejemplo más simple del álgebra de Boole se compone de un conjunto G de 2 elementos: "0" y "1". Como es natural estos dos elementos deben coincidir con los neutros de las reglas de combinación para satisfacer el axioma 3. Las reglas de combinación debemos definirlas de manera de satisfacer los axiomas.

Así de acuerdo al axioma 3 :

$$0+0=0 \qquad 0\cdot 1=0 \quad (\mathrm{a}=0)$$
$$1+0=1 \qquad 1\cdot 1=1 \quad (\mathrm{a}=1)$$

de acuerdo al axioma 4

$$0+1=1 \qquad 1\cdot 0=0$$

y teniendo presente el axioma 5 :

$$1+(1\cdot 0)=(1+1)\cdot(1+0) \qquad (\text{5a con a}=1, \text{b}=1, \text{c}=0)$$
$$1+0=(1+1)\cdot 1$$
$$1=1+1 \qquad (\text{por axioma 3})$$

$$0\cdot(0+1)=0\cdot 0+0\cdot 1 \qquad (\text{5b con a}=0, \text{b}=0, \text{c}=1)$$
$$0\cdot 1=0\cdot 0+0$$
$$0=0\cdot 0 \qquad (\text{por axioma 3})$$

Por lo tanto las reglas *completas* son:

$$0+0=0 \qquad 0\cdot 0=0$$
$$0+1=1 \qquad 0\cdot 1=0$$
$$1+0=1 \qquad 1\cdot 0=0$$
$$1+1=1 \qquad 1\cdot 1=1$$

Nosotros usaremos esta versión "binaria" del álgebra de Boole.

4.4 Propiedades.

- **Dualidad**

Si analizamos los postulados veremos que los mismos se presentan de a pares y en tal forma que uno de la pareja se obtiene de otro cambiando "0" por "1" junto con "+" por "·" (y viceversa). Esto asegura que cada propiedad que se demuestre en esta Álgebra tiene una "dual" que también es cierta (para demostrar la dual bastaría con repetir la demostración realizada sustituyendo cada postulado o propiedad utilizada por su dual).

- **Asociativa**

a) $a+(b+c)=(a+b)+c$

b) $a\cdot(b\cdot c)=(a\cdot b)\cdot c$

Si bien las leyes asociativas son muchas veces incluidas dentro del cuerpo axiomático, de hecho son demostrables a partir de los axiomas aquí presentados (demostración que no haremos) por lo cual las presentamos como propiedades.

- **Idempotencia**

Para todo elemento en G se cumple:

$$a+a=a$$
$$a\cdot a=a$$

Demostración:

$$a+a=(a+a)\cdot 1 \qquad \text{(3b)}$$
$$a+a=(a+a)\cdot(a+\overline{a}) \qquad \text{(6)}$$
$$a+a=a+(a\cdot\overline{a}) \qquad \text{(5a)}$$
$$a+a=a+0 \qquad \text{(6)}$$
$$\Rightarrow a+a=a$$

$$\Rightarrow a\cdot a=a \qquad \text{(Dualidad)}$$

- **Neutros Cruzados**

Para todo elemento en G se cumple

$$a+1=1$$
$$a\cdot 0=0$$

Demostración:

$$a+1 = a+(a+\overline{a}) \qquad (6)$$
$$a+1 = (a+a)+\overline{a} \qquad \text{(asociativa)}$$
$$a+1 = a+\overline{a} \qquad \text{(idempotencia)}$$
$$\Rightarrow a+1 = 1$$

$$\Rightarrow a \cdot 0 = 0 \qquad \text{(Dualidad)}$$

Entonces los axiomas 1, 2, 3, 4, 5 y 7 se satisfacen por definición y es fácil verificar que el G (complemento) también es cierto.

Construimos por lo tanto un modelo "aritmético" de álgebra de Boole que podemos denominar "binario" y es en definitiva con la que trabajaremos.

Muchas veces las reglas de combinación se presentan como tablas (como las funciones booleanas más generales que veremos más tarde)

a	***b***	***a+b***
0	0	0
0	1	1
1	0	1
1	1	1

a	***b***	***a·b***
0	0	0
0	1	0
1	0	0
1	1	1

En general notaremos a·b como ab, además la operación “·” tendrá mayor precedencia que la operación “+”.

- **Complemento de complemento**

Para cada elemento de G se cumple : $a = \overline{\overline{a}}$

Para todo par de elementos de G se cumple :

$$a + ab = a$$
$$a(a+b) = a$$

Para todo par de elementos de G se cumple :

$$a + \overline{a}b = a + b$$
$$a(\overline{a} + b) = ab$$

- **Ley de *De Morgan***

Para todo par de elementos de G se cumple :

$$\overline{(a+b)} = \overline{a}\,\overline{b}$$
$$\overline{(ab)} = \overline{a} + \overline{b}$$

Estas reglas de De Morgan pueden generarse para cualquier número de variables.

$$\overline{(a_1 + a_2 + ... + a_n)} = \overline{a_1}\,\overline{a_2} ... \overline{a_n}$$
$$\overline{(a_1 a_2 ... a_n)} = \overline{a_1} + \overline{a_2} + ... + \overline{a_n}$$

4.5 Modelo lógico.

Los valores que pueden asignarse a un juicio, desde el punto de vista lógico, son dos: verdadero (V) o falso (F).

Un juicio al cual se le aplica el operador lógico no (negación) forma un nuevo juicio.

Dos juicios pueden combinarse para formar un tercero mediante los operadores lógicos "o" e "y".

Si vinculamos los valores booleanos 0 y 1 con los valores lógicos F y V respectivamente, encontramos que las operaciones del álgebra de Boole "binaria" asigna correctamente los valores lógicos del juicio combinación.

Esto se comprueba observando que:

verdadero o verdadero es verdadero
verdadero o falso es verdadero
falso o verdadero es verdadero
falso o falso es falso

verdadero y verdadero es verdadero
verdadero y falso es falso
falso y verdadero es falso
falso y falso es falso

Por lo cual se puede concluir que el modelo "lógico" es isomorfo con el "aritmético" (binario) realizando la correspondencia.

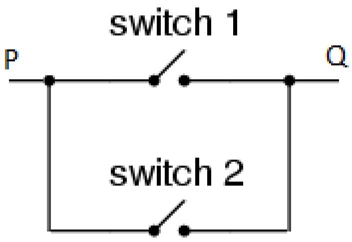
$$F \Leftrightarrow 0$$
$$V \Leftrightarrow 1$$
$$\vee \Leftrightarrow +$$
$$\wedge \Leftrightarrow .$$
$$\neg \Leftrightarrow \overline{}$$

Es posiblemente consecuencia de este isomorfismo que las reglas de combinación "+" y "." del álgebra de Boole reciban los nombres de OR ("o" en inglés) y AND ("y" en inglés) respectivamente.

4.6 Modelo circuital.

Otro modelo posible es el que surge de considerar llaves eléctricas y asociar el valor A a la llave abierta (no pasa corriente, circuito abierto) y el valor C con la llave cerrada (pasa corriente, circuito cerrado).

Es fácil comprobar que la combinación de llaves en paralelo o en serie cumplen las mismas reglas definidas en el modelo aritmético para "+" y "." respectivamente.



paralelo

serie

LL1	LL2	Circuito Paralelo
A	A	A
A	C	C
C	A	C
C	C	C

LL1	LL2	Circuito Serie
A	A	A
A	C	A
C	A	A
C	C	C

Entonces existe también un isomorfismo entre el modelo "circuital" y el "aritmético" si hacemos la asociación

$$A \Leftrightarrow 0$$
$$C \Leftrightarrow 1$$
$$\text{paralelo} \Leftrightarrow +$$
$$\text{serie} \Leftrightarrow \cdot$$

Este isomorfismo es de fundamental importancia para la construcción práctica de las computadoras binarias.

4.7 Expresiones booleanas.

Llamamos constante a todo elemento del conjunto G que define al álgebra. Las variables podrán tomar como valor cualquier elemento de G (0 o 1 en el caso en que trabajamos).

Una expresión la podemos definir recursivamente como

1) las constantes y las variables
2) el complemento de una expresión booleana
3) el OR (+) o el AND (·) de dos expresiones booleanas.

4.8 Funciones booleanas.

Una función "F" de n variables x_1 ... x_n booleanas es una aplicación del espacio G^n sobre el espacio G de tal forma que para cada valor posible de la n-upla x_1 ... x_n, donde cada variable puede tomar cualquier valor del conjunto (en nuestro caso {0, 1}), se asocia un valor del recorrido G.

Una de las formas de expresar F es a través de las denominadas tablas de verdad que indican el resultado de F para cada valor posible de la n-upla; por ejemplo :

a	*b*	*c*	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Otras formas de representar F incluyen el método de indicar sólo los puntos en los cuales F vale 1 o sólo los puntos en los cuales vale 0 (representaciones Σ y Π respectivamente). Para indicar los puntos en que la función vale 1 puede usarse la notación σ en lugar de Σ.

Por ejemplo, la función anterior se puede expresar :

$f(a,b,c) = \Sigma(1,4,5,7)$ $\qquad$ $f(a,b,c) = \Pi(0,2,3,6)$

Otra forma de expresar las funciones es a través de expresiones; ejemplo, la función anterior sería:

$$f = ac + a\overline{b} + \overline{a}\overline{b}c$$

4.8.1 Conectivas binarias.

Un caso interesante de estudiar es el de las funciones booleanas de 2 variables. Por ser dos variables las combinaciones posibles son 4, es decir "F" tiene 4 duplas (4 puntos) por tanto existen 16 funciones booleanas de dos variables posibles. Algunas de ellas no son de interes, veamos las tablas de verdad de las más útiles.

a	b	OR	AND	XOR	NOR	NAND	Equiv.	Idemp	Tautol.
0	0	0	0	0	1	1	1	0	1
0	1	1	0	1	0	1	0	0	1
1	0	1	0	1	0	1	0	0	1
1	1	1	1	0	0	0	1	0	1

Nota :
la NOR es el complemento de la OR
la NAND es el complemento de AND
la XOR ("O" exclusivo) puede definirse como:

$$a \oplus b = \overline{a}b + a\overline{b}$$

4.8.2 OR exclusivo.

El XOR es una función muy importante (es la suma aritmética binaria módulo 2) y cumple las propiedades :

1) Asociativa
2) Conmutativa
3) Distributiva: $a(b \oplus c) = ab \oplus ac$
4) $a \oplus 0 = a$
5) $a \oplus 1 = \overline{a}$

6) $a \oplus a = 0$
7) Cancelativa: $a \oplus b = a \oplus c \rightarrow b = c$

4.8.3 Suma de productos canónicos.

Desarrollaremos a continuación un método sistemático para encontrar una expresión algebraica para una función cualquiera dada.

Definamos producto canónico de *n* variables x_1 ... x_n al producto de todas ellas en el cual cada variable aparece una y sólo una vez, en forma simple o complementada.
Una suma de productos canónicos es una expresión formada sumando productos canónicos.

Existe un teorema (que no demostraremos) que afirma que toda función *f* de *n* variables puede expresarse como:

$$\begin{aligned} f(x_1, x_2, \ldots, x_n) = {} & x_1 \cdot x_2 \cdot x_3 \cdot \ldots \cdot x_n \cdot f(1,1,1,\ldots,1) + \\ & \overline{x_1} \cdot x_2 \cdot x_3 \cdot \ldots \cdot x_n \cdot f(0,1,1,\ldots,1) + \\ & x_1 \cdot \overline{x_2} \cdot x_3 \cdot \ldots \cdot x_n \cdot f(1,0,1,\ldots,1) + \\ & \ldots\ldots\ldots\ldots + \\ & \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \ldots \cdot \overline{x_n} \cdot f(0,0,0,\ldots,0) \end{aligned}$$

Es decir toda función de *n* variables puede expresarse como la suma de todos sus productos canónicos afectado cada producto canónico por un coeficiente. Este coeficiente es el valor de la función evaluado en el punto tal que las variables que en el producto canónico asociado aparecen simples tengan el valor 1 y las que aparecen complementadas el valor 0.

Este teorema, de fundamental importancia nos permite enunciar un método de construcción de una expresión que represente una función dada. El método es el siguiente: se tienen en cuenta sólo los puntos en los que la función vale 1 (por el teorema los productos asociados con los puntos en los que la función vale 0 desaparecen por estar afectados por un coeficiente nulo: el propio valor de la función); en esos puntos se busca el producto canónico asociado que es aquel donde la variable aparece simple si en el punto vale 1 o complementada si vale 0. De esta manera la función puede expresarse como suma de los productos canónicos así elegidos.

Por ejemplo sea la función de tres variables:

a	*b*	*c*	*F*	
0	0	0	0	
0	**0**	**1**	**1**	$\Rightarrow \overline{a}\,\overline{b}c$
0	1	0	0	
0	**1**	**1**	**1**	$\Rightarrow \overline{a}bc$
1	**0**	**0**	**1**	$\Rightarrow a\overline{b}\,\overline{c}$
1	0	1	0	
1	1	0	0	
1	1	1	0	

Entonces f puede expresarse como $f(a,b,c) = \overline{a} \cdot \overline{b} \cdot c + \overline{a} \cdot b \cdot c + a \cdot \overline{b} \cdot \overline{c}$

4.8.4 Productos de sumas canónicas.

Como todo en el álgebra de Boole, existe un método dual del anterior: el producto de sumas canónicas. En este caso deben considerarse los puntos en los que la función vale 0 y buscar las sumas canónicas asociadas que son aquellas en las que la variable aparece simple si tiene valor 0 y complementada si tiene valor 1.

4.9 Operadores lógicamente completos.

Un conjunto de operadores se llama lógicamente completo si cualquier función booleana puede expresarse mediante los mismos.

Del teorema de los productos canónicos, ya enunciado, se extrae una conclusión fundamental tanto del punto de vista lógico como del circuital: el conjunto de operadores +, . y ' es lógicamente completo.

Otra consecuencia es que para probar que un cierto conjunto de operadores es lógicamente completo, alcanza con probar que con ellos se pueden implementar el OR, el AND y el NOT (complemento).

Es fácil probar que el conjunto OR, NOT es lógicamente completo notando que el AND se puede construir como:

$$\overline{(ab)} = \overline{a} + \overline{b} \quad (\text{por De Morgan}) \Rightarrow a \cdot b = \overline{(\overline{a} + \overline{b})}$$

Más interesante aún es mostrar que un solo operador, como el NAND, es lógicamente completo. Debemos ver como implementar el NOT y el AND y el OR (representaremos con # el NAND):

$$a \# a = \overline{(a.a)} = \overline{a} \qquad (\text{complemento})$$

$$(a\#b)\#(a\#b) = \overline{(a\#b)} = \overline{\overline{(a.b)}} = a.b \qquad (\text{AND})$$

$$(a\#a)\#(b\#b) = \overline{(\overline{a}.\overline{b})} = \overline{\overline{(a+b)}} = a + b \qquad (\text{OR}) \text{ por De Morgan}$$

Por lo cual el NAND es lógicamente completo.

4.10 Simplificación.

Hasta ahora hemos visto un método sistemático de expresar las funciones booleanas como expresión de sus variables. Pero este método no asegura que la expresión lograda sea la más simple posible. El hecho que la expresión de una función sea lo más simple posible no es algo trivial o caprichoso, es de fundamental importancia en la construcción práctica de circuitos lógicos, por eso analizaremos algunos métodos para simplificar expresiones booleanas, de manera de aplicarlos a las expresiones obtenidas como sumas de productos canónicos.

4.10.1 Método algebraico.

El método consiste en la aplicación, más o menos ingeniosa, de transformaciones algebraicas de manera de lograr expresiones más sencillas. Por supuesto que este no es un método sistemático, pero es la base, al fin, de los métodos sistemáticos.

Resumamos aquí algunas propiedades vistas del álgebra que serán de utilidad en la tarea de simplificar:

1) $f \cdot \overline{f} = 0$
2) $f + \overline{f} = 1$
3) $g \cdot f + \overline{g} f = f$
4) $g \cdot f + f = f$
5) $f + \overline{f} \cdot g = f + g$

Veamos un par de ejemplos de como se aplican estas propiedades para reducir expresiones:

a) Sea $f_1 = \overline{a}bc + \overline{a}b\overline{c} + ab\overline{c}$

Por la aplicación de la propiedad 3 a los primeros dos términos y a los dos últimos queda

$$f_1 = \overline{a}b + b\overline{c}$$

b) Sea $f_2 = \overline{a}\overline{b}c + a\overline{b}c + a\overline{b}\overline{c} + \overline{a}bc$

Aplicando la propiedad 3 a los dos primeros términos queda

$f_2 = \overline{b}c + a\overline{b}\overline{c} + \overline{a}bc$

$f_2 = \overline{b}(c + a\overline{c}) + \overline{a}bc$ (por la propiedad distributiva)

$f_2 = \overline{b}(c + a) + \overline{a}bc$ (por la propiedad 5 al paréntesis)

$f_2 = c(\overline{b} + \overline{a}b) + a\overline{b}$ (por distributiva aplicada dos veces)

$f_2 = c(\overline{a} + \overline{b}) + a\overline{b}$ (por propiedad 5 al paréntesis)

Entonces la expresión de f2 a la que llegamos es:

$$f_2 = a\overline{b} + \overline{a}c + \overline{b}c$$

Esta sin embargo no es la expresión más reducida de f_2. Vemos como hubiera quedado aplicando la propiedad 3 al primer y cuarto miembro y al segundo y tercero

$$f_2 = \overline{a}c + a\overline{b}$$

siendo esta sí, la expresión más reducida.

Como vemos entonces el procedimiento descrito no asegura reducir la expresión a un mínimo ya que depende de como se elijan las propiedades a aplicar y los términos sobre los que se aplican.

4.10.2 Métodos sistemáticos.

Los modelos sistemáticos se basan en la propiedad 3 $g \cdot f + \overline{g} f = f$ y son básicamente uno "gráfico" (Diagrama de Karnaugh) y otro "algorítmico" o "numérico" (Método de Quine-McKlusky)

A continuación veremos una introducción al método gráfico.

4.10.3 Diagrama de Karnaugh.

Este método consiste en representar en forma gráfica una función como suma de productos canónicos y hacerlo de tal forma que sea sencillo establecer procedimientos sistemáticos para hallar las agrupaciones de términos más convenientes para simplificar la expresión. Esto se logra utilizando una cuadrícula en la cual a cada cuadrado elemental corresponde un producto canónico posible y tal que al pasar de uno a otro cualquiera de sus cuatro adyacentes solo cambie el valor de una de las variables en juego. Por ejemplo para tres variables la cuadrícula es:

c\ab	00	01	11	10
0				
1				

Para cuatro variables:

cd\ab	00	01	11	10
00				
01				
11				
10				

Para cinco variables:

e = 0

cd\ab	00	01	11	10
00				
01				
11				
10				

e = 1

cd\ab	00	01	11	10
00				
01				
11				
10				

En esta cuadrícula se marcan con "1" los lugares para los cuales la combinación de valores de las variables hace que la función valga 1 y el método consiste en buscar agrupar los "unos" formando los rectángulos más grandes posibles (que tengan todos 1 en su interior), repitiendo este proceso hasta que todos los puntos donde la función vale "1" estén comprendidos en algún rectángulo (siendo la cantidad total de rectángulos utilizados la menor posible). Es necesario aclarar que la cantidad de elementos agrupados debe ser una potencia de 2.

Nota: el diagrama es circular (los elementos de cada borde son adyacentes con los del borde simétrico)

Ejemplos:

cd\ab	00	01	11	10
00	1	1		
01	1	1	1	
11				
10				

cd\ab	00	01	11	10
00	1		1	1
01	1	1		
11				
10			1	1

Una vez realizado el proceso anterior la mínima expresión de la función se obtiene sumando los términos asociados a cada rectángulo los cuales son el producto de las variables (simples o complementadas) cuyo valor no cambia en él.

Por ejemplo en los diagramas anteriores sería

1) $f_3 = \bar{a}\bar{c} + b\bar{c}d$

2) $f_4 = a\bar{d} + \bar{a}\bar{b}\bar{c} + \bar{a}\bar{c}d$

Para finalizar veamos como se aplica el método al caso ya visto de la función:

$$f_2 = \bar{a}\bar{b}c + a\bar{b}c + a\bar{b}\bar{c} + \bar{a}bc$$

El diagrama de Karnaugh correspondiente es

c\ab	00	01	11	10
0				1
1	1	1		1

Entonces la función queda $f_2 = \bar{a}c + a\bar{b}$ la cual coincide, como era de esperar, con la expresión más reducida hallada por el método "algebraico".

La primer expresión de f_2 hallada con dicho método tenía además el término $\bar{b}c$ que corresponde al rectángulo formado por los dos extremos inferiores del diagrama que aquí queda evidente que no era necesario puesto que con los otros dos términos cubrimos todos los puntos donde la función vale "1".

4.10.4 Uso de "comodines" ("don't care" ó "X")

Cuando la función booleana que se desea minimizar tiene puntos en los cuáles no está definida, es decir que no importa el valos que toma para ciertas combinaciones específicas de sus variable de entrada, se puede utilizar esta propiedad para eventualmente lograr una minimización más importante de la expresión algebraica.

Para ello se colocan "x" en el diagrama de Karnaugh correspondiente a esos puntos no definidos y las "x" se podrán tomar como 1 en caso que permitan lograr un rectángulo de mayor tamaño y 0 en caso contrario. De esta manera al lograr rectángulos de mayor tamaño minimizo los términos de la expresión y eventualmente también la cantidad de los mismos.

Cabe aclarar que no siempre se logra una mayor minimización porque existen casos donde no es posible asignarle 1 a la “x” para lograr rectángulos más grandes. Notar además que no tiene sentido alguno agrupas solamente “x” (porque así estaríamos agregando términos, cuando la idea de usar las “x” es lograr una mayor minimización).