Complejidad (II)

Problemas NP-completos

Gregorio Hernández

UPM

Matemática Discreta II

Complejidad (o coste) de un algoritmo

Algoritmo: secuencia de instrucciones para resolver un problema

Programa: algoritmo escrito en un lenguaje de programación, junto con las estructuras de datos utilizadas

La complejidad mide la eficiencia del algoritmo en la resolución del problema. Se mide en:

- Tiempo (número de operaciones realizadas)
- Espacio o cantidad de memoria utilizada

    - en el peor caso
    - en media

Complejidad de un problema

MST Algoritmo de Prim

Complejidad
- $O(n^3)$
- $O(n^2)$

Algoritmo polinómico

TSP Algoritmo “fuerzabruta”

Complejidad
- $O(n!)$

Algoritmo no polinómico

Complejidad

	$\log n$	n	n^2	2^n	$n!$
2	1	2	4	4	2
8	3	8	64	256	40320
32	5	32	1024	$4{,}3\times10^{9}$	$2{,}6\times10^{35}$
100	6	100	10^4	$1{,}2\times10^{27}$	$9{,}3\times10^{177}$

Un siglo tiene $3{,}1\times10^{9}$ segundos

Si la edad del Universo es de 13500 millones de años,
el big-bang ocurrió hace $4{,}5\times10^{16}$ segundos

Problema INDECIDIBLE

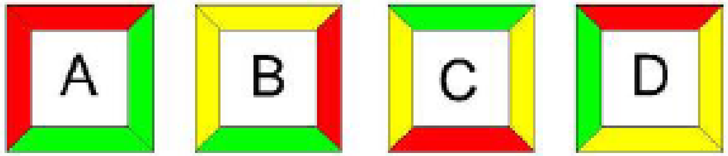
Si no existe ningún algoritmo que lo resuelva.

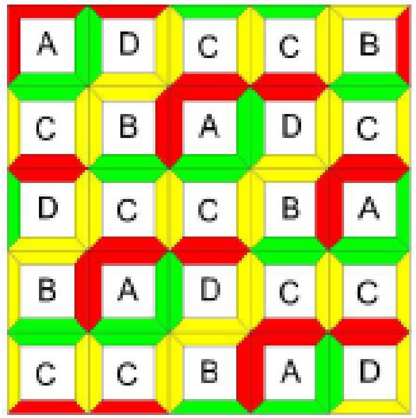
- Entscheidungsproblem (problema de decisión): Dada una frase del cálculo de predicados de primer orden, decidir si es un teorema. (Church y Turing)
- El Problema de la parada: Dado un programa y su entrada, decidir si ese programa terminará para esa entrada o si correrá indefinidamente. (Turing)
- Un número computable es un número real que puede ser aproximado por un algoritmo con un nivel de exactitud arbitrario. Turing demostró que casi todos los números no son computables.
- 10º problema de Hilbert: Dada una ecuación diofántica (con coeficientes enteros) ¿existe solución entera? (Matijasevic 1970)

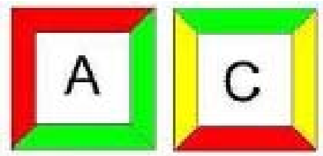
Problema INDECIDIBLE

Si no existe ningún algoritmo que lo resuelva.

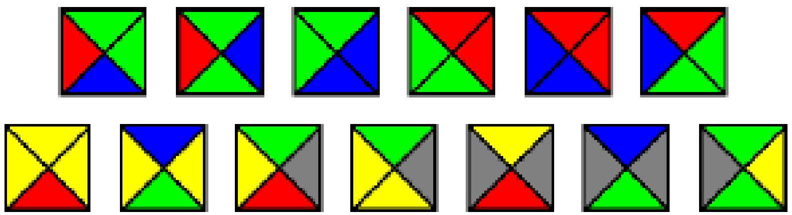
- Embaldosados del plano: Dado un conjunto finito de baldosas cuadradas con un color en cada lado, ¿puede embaldosarse el plano de forma que lados contiguos tengan el mismo color? (Domino problem, Wang, Berger, 1966)







No hay solución



Hay solución no periódica

Problema INDECIDIBLE

- Matrix Mortality
  Dado un conjunto finito de matrices de tamaño n×n, ¿e puede obtener la matriz nula multiplicando algunas de ellas?
  El problema es indecidible para n = 3 y no se sabe para n = 2
- Problema 3n + 1 (Collatz Problem)

  Sea a_0 un entero positivo. Se construye la sucesión (a_n) así

$$a_n = \begin{cases} \frac{1}{2} a_{n-1} & \text{si } a_{n-1} \text{ es par} \\ 3a_{n-1} + 1 & \text{si } a_{n-1} \text{ es impar} \end{cases}$$

¿En esta sucesión se alcanza siempre el 1?

No se sabe si este problema es decidible o indecidible

Problemas INDECIDIBLES

Problemas DECIDIBLES

Si existen algoritmos que los resuelven

- MST de un grafo con pesos en las aristas
- Dado un grafo G, ¿es hamiltoniano?
- Hallar el camino de peso mínimo entre dos vértices de un grafo.
- Hallar el número cromático de un grafo G.

Clase P
Problemas resolubles en tiempo polinómico

Clase NP
Problemas verificables en tiempo polinómico

Clase P
Problemas resolubles en tiempo polinómico

Existe un algoritmo de complejidad polinómica $O(n^k)$ que resuelve el problema (k constante, n tamaño de la entrada del problema)

- Hallar el MST de un grafo con pesos en las aristas.
- Hallar el camino de peso mínimo entre dos vértices de un grafo.
- Decidir si un grafo es euleriano.

$\mathbf{P \subset NP}$

Clase NP (No determinista Polinómico)
Problemas verificables en tiempo polinómico

¿$\mathbf{P \neq NP}$?

Dado un “certificado” (una posible solución), podemos verificar si es correcto en tiempo polinómico en el tamaño del problema.

- Decidir si un grafo es 4-coloreable.
- Decidir si un grafo es hamiltoniano.
- Hallar el camino más largo entre dos vértices de un grafo

Clase NP (No determinista Polinómico)
Problemas verificables en tiempo polinómico

Dado un "certificado" (una posible solución), podemos verificar si es correcto en tiempo polinómico en el tamaño del problema.

- Decidir si un grafo es 5-coloreable

Dada una instancia I del problema, es decir, un grafo G, un certificado C(I) es una asignación de colores a los vértices

$$V(G) \rightarrow \{1,2,3,4,5\}$$

que podemos comprobar si es una 5-coloración válida en tiempo $O(n^2)$ mirando los extremos de cada arista

Clase NP (No determinista Polinómico)
Problemas verificables en tiempo polinómico

Dado un “certificado” (una posible solución), podemos verificar si es correcto en tiempo polinómico en el tamaño del problema.

- Decidir si un grafo es hamiltoniano

Dada una instancia I del problema, es decir, un grafo G, un certificado C(I) es una permutación de los vértices,

$$v_6\, v_3\, v_1\, v_7\, v_5\, v_8 \ldots v_9\, v_2$$

que podemos comprobar si es un ciclo hamiltoniano averiguando si en G existen las aristas v_6v_3, $v_3\, v_1$, v_1v_7, … , v_9v_2, v_2v_6 en tiempo $O(n^2)$

PROBLEMAS DE DECISIÓN

Problemas en que la respuesta es SÍ o NO

Los problemas de optimización se transforman en problemas de decisión

Hallar el camino mínimo entre dos vértices u y v de un grafo G	→	Dados el grafo G, los vértices u,v y un entero k, decidir si entre u y v existe un camino con a lo más k aristas.
Problema del Viajante Hallar el ciclo hamiltoniano de peso mínimo	→	Dados el grafo G y un entero k, decidir si existe un ciclo hamiltoniano de peso menor o igual a k.

TRANSFORMACIÓN DE PROBLEMAS

El problema A es polinómicamente reducible a B si existe T, algoritmo polinómico, que convierte cada instancia I de A en una instancia T(I) para el problema B tal que:

La respuesta a I es SÍ $\Leftrightarrow$ La respuesta a T(I) es SÍ

La notación es A $\propto$ B

Problema A
Dado un grafo G y un entero k, ¿existe un conjunto independiente en G de tamaño k?

Problema B
Dado un grafo G y un entero k, ¿existe una clique en G de tamaño k?

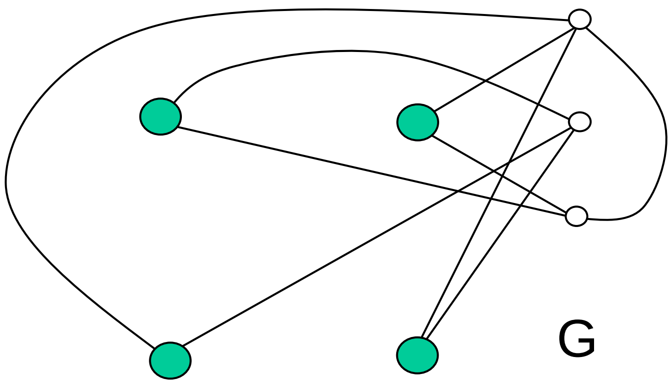
IS $\propto$ CLIQUE

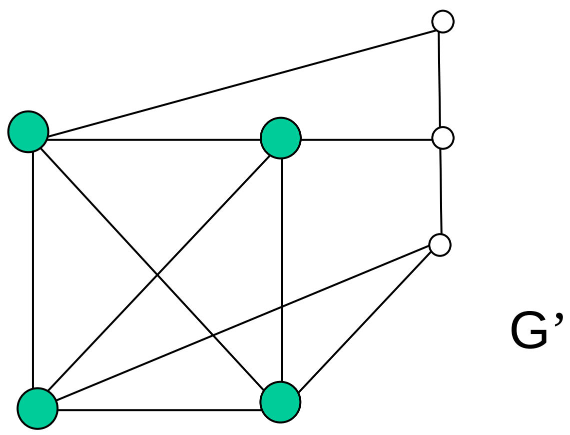
TRANSFORMACIÓN DE PROBLEMAS

IS $\propto$ CLIQUE

Una instancia de IS es un grafo G y un entero k. Debemos transformarla en una entrada de datos para B. La transformación consiste en pasar al grafo complementario G'. El coste de la transformación es $O(n^2)$

G tiene un conjunto independiente de tamaño k (la respuesta para la instancia es SÍ) $\Leftrightarrow$
G' contiene una clique de tamaño k (la respuesta para T(I) es SÍ)





PROBLEMAS NP-duros y NP-completos

Un problema Π es NP-duro si cualquier problema de la clase NP puede transformarse polinómicamente a Π

Es decir, si resolviendo Π se pueden resolver TODOS los problemas de la clase NP.

Un problema Π es NP-completo si

- es NP-duro
- está en la clase NP

¿Existen problemas NP-duros?

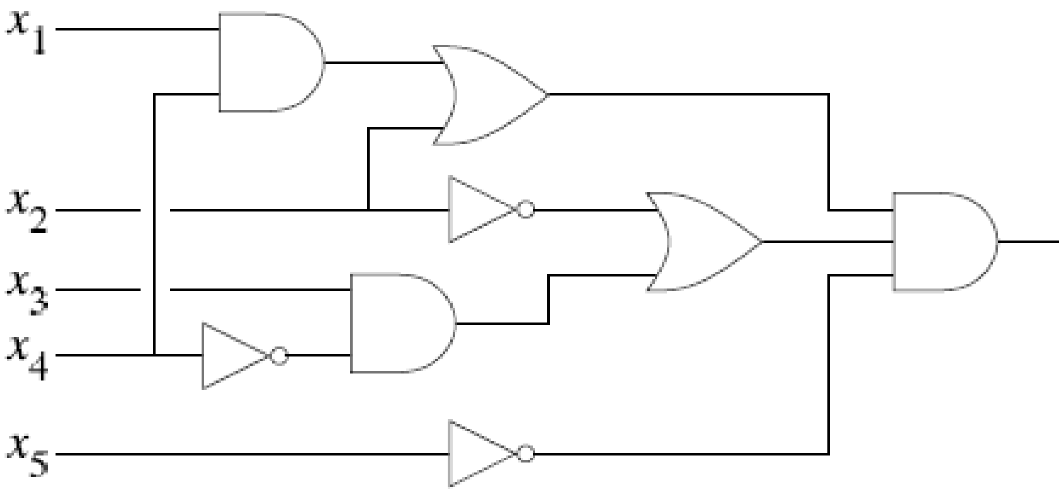
¿Existen problemas NP-completos?

SAT

Satisfactibilidad de circuitos o expresiones booleanas

En 1971 Cook demostró que es un problema NP-completo

Dado un circuito C o expresión booleana, ¿existe una asignación de valores 0, 1 a las variables de modo que la respuesta de C sea 1?



$$C = (x_1x_4 + x_2)\,(x'_2 + x_3x'_4)\,x'_5$$

SAT es un problema de la clase NP

SAT es un problema de decisión

En 1971 Cook demostró que es un problema NP-completo

Dado un circuito C o expresión booleana, ¿existe una asignación de valores 0, 1 a las variables de modo que la respuesta de C sea 1?

Una asignación de valores a las variables es una aplicación

$$f:\{x_1, x_2, \dots , x_n\} \rightarrow \{0,1\}$$

y esto es un certificado para el circuito dado C. Comprobar que la asignación satisface el circuito (responde con 1) es fácil, basta calcular todas las operaciones indicadas en el circuito o expresión booleano. Se puede hacer en tiempo polinómico.

$$C = (x_1x_4 + x_2)\ (x'_2 + x_3x'_4)\ x'_5$$

Si $x_1, x_4 = 1$, $x_2, x_3, x_5 = 0$ entonces $C = 1$

Problemas NP-completos

SAT (Cook, 1971)

3-SAT (Karp, 1972)

Dada una expresión booleana E en forma normal conjuntiva con tres literales (variables o negación de variables) por cláusula (factores), ¿existe una asignación de valores 0, 1 a las variables de modo que la respuesta de E sea 1?

$$E = (x_1 + x'_4 + x_2)\ (x'_2 + x_3 + x'_4)\ (x'_2 + x_3 + x_5)$$

SAT $\propto$ 3-SAT

¿Cómo demostrar que un problema Q es NP-completo

1. Demostrar que **Q** está en la clase NP (es lo más sencillo)
2. Elegir un problema **Q’** que sea NP-completo
3. Demostrar que **Q’** $\propto$ **Q**

En su artículo de 1972, Karp demostró, utilizando esta estrategia, que muchos problemas eran NP-completos: 3-coloración, recubrimiento por vértices, conjunto independiente, ciclo hamiltoniano, etc.

3-coloración es NP-completo

1. Demostrar que **3C** está en la clase NP (ya lo hemos visto)
2. Elegir un problema, **3-SAT** que es NP-completo
3. Demostrar que **3-SAT $\propto$ 3C**

Consideramos una instancia de 3-SAT, una expresión booleana E, y construimos a partir de ella un grafo G tal que:

E se satisface $\Leftrightarrow$ G es 3-coloreable

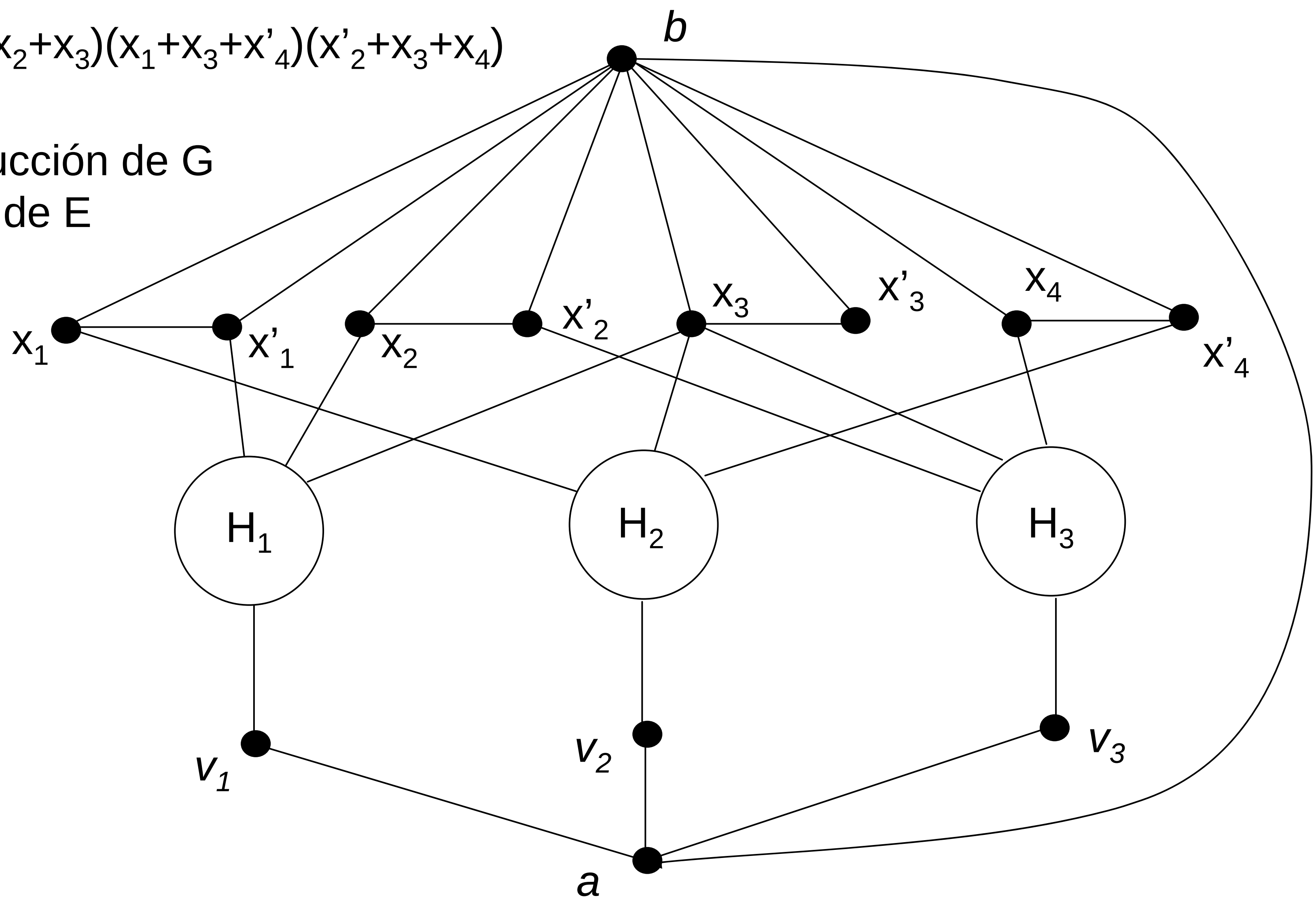
Existe una asignación de valores 0,1 a las variables de forma que E toma el valor 1

G admite una 3-coloración válida

3-coloración es NP-completo

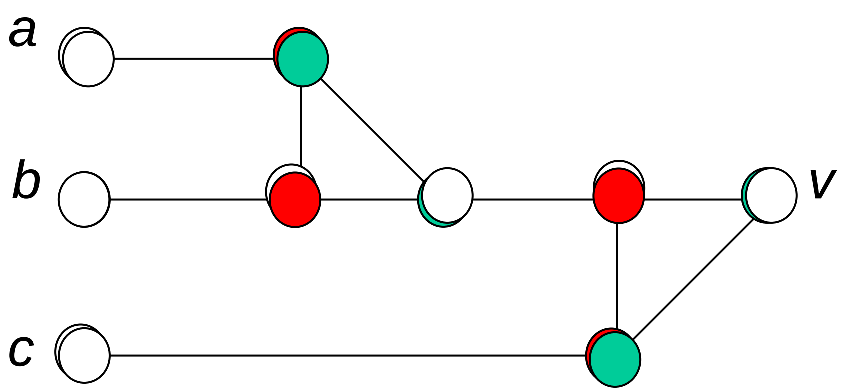
$E=(x'_1+x_2+x_3)(x_1+x_3+x'_4)(x'_2+x_3+x_4)$

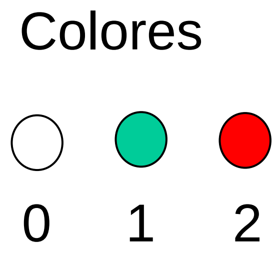
Construcción de G
a partir de E



3-coloración es NP-completo

Construcción de las piezas H



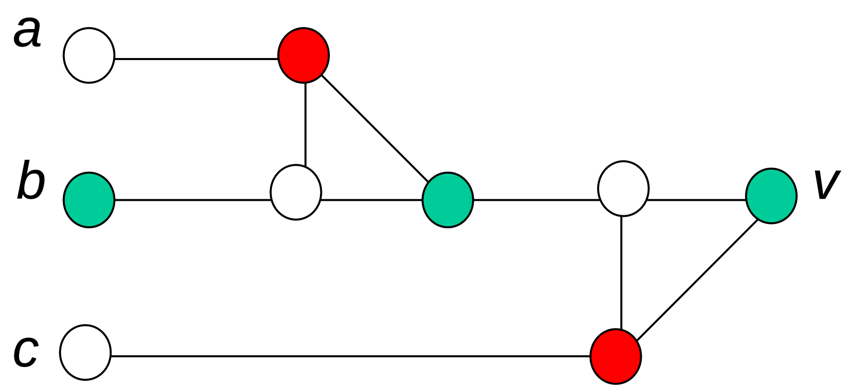


Si a, b, c color ⟹ v color

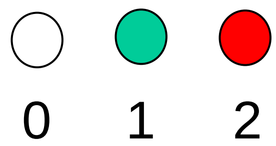
Si alguno de los vértices a, b, ó c tiene color ó ⟹ Se puede colorear v con los colores

3-coloración es NP-completo

Construcción de las piezas H



Colores

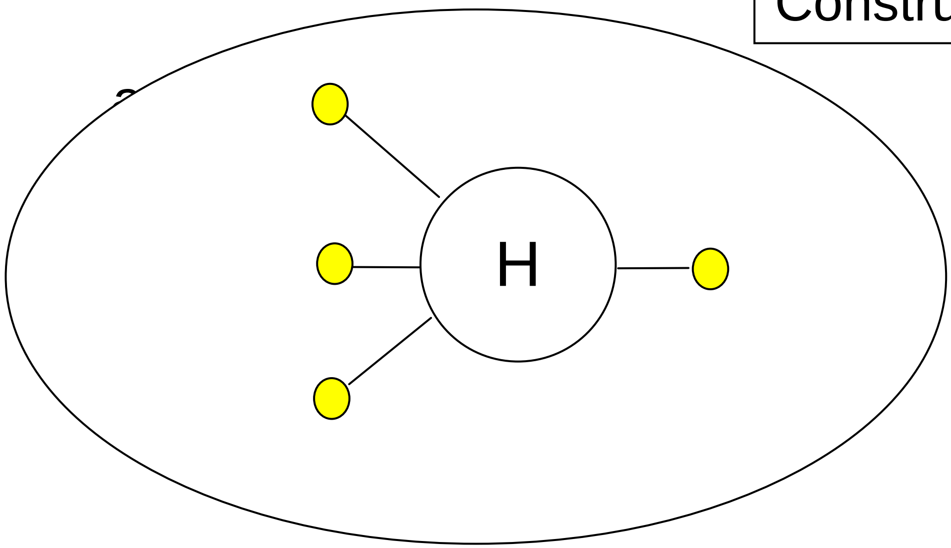


Si a, b, c color ⇒ v color

Si alguno de los vértices a, b, ó c tiene color ó ⇒ Se puede colorear v con los colores

3-coloración es NP-completo

Construcción de las piezas H



Colores

0 1 2

Si a, b, c color

v color

Si alguno de los vértices a, b, ó c tiene color ó

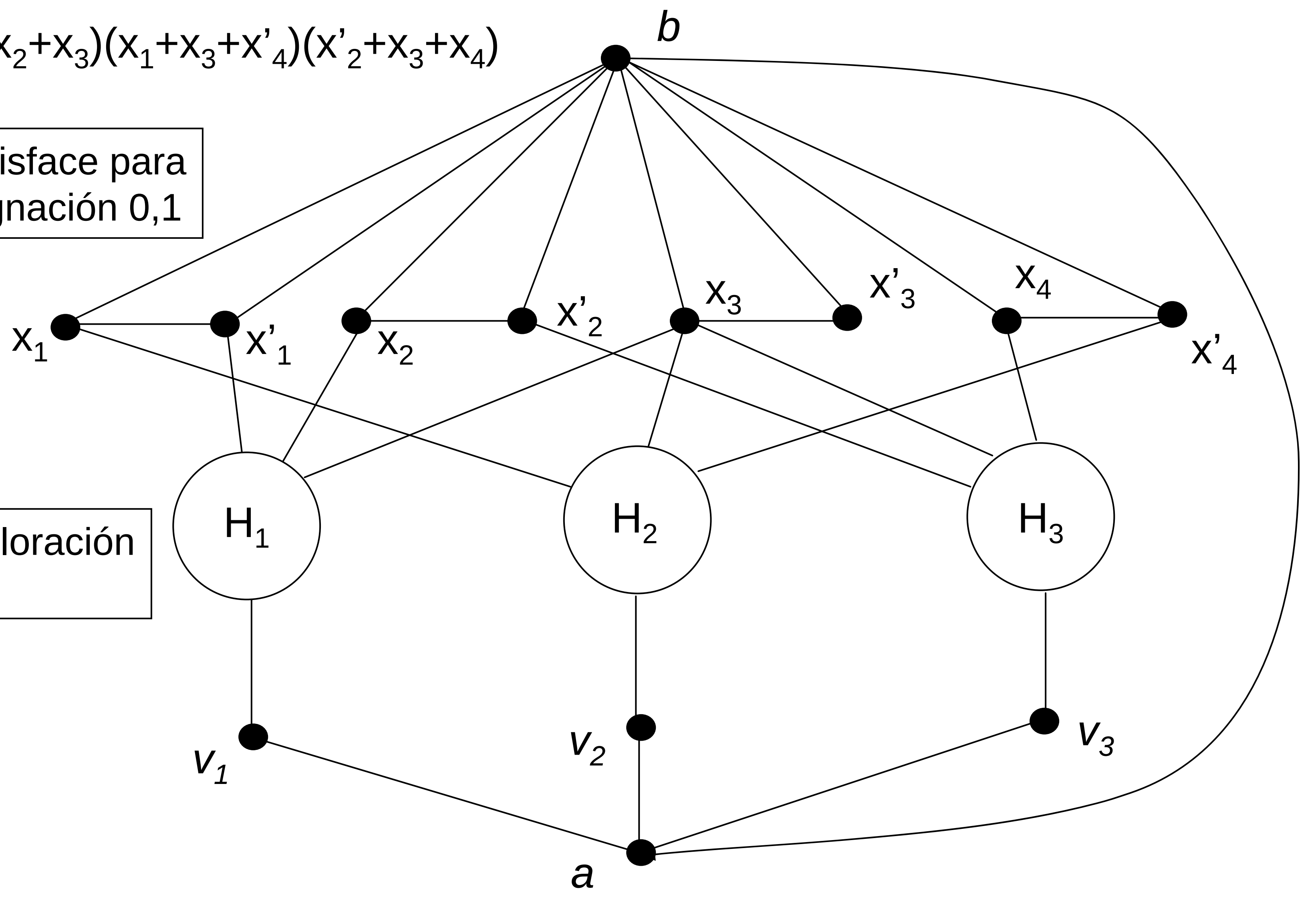
Se puede colorear v con los colores

3-coloración es NP-completo

$E=(x'_1+x_2+x_3)(x_1+x_3+x'_4)(x'_2+x_3+x_4)$

E se satisface para una asignación 0,1

Una 3-coloración de G

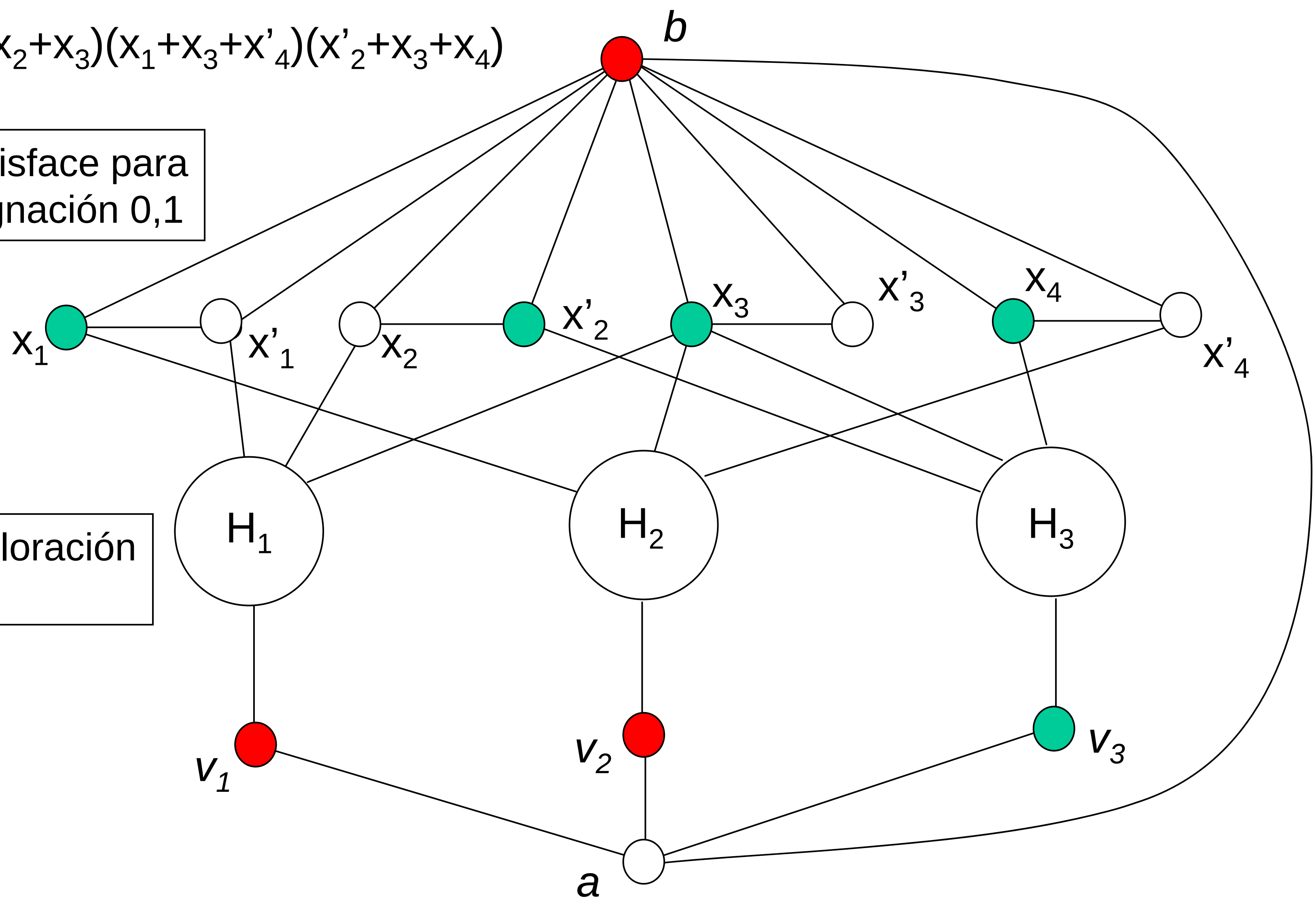


3-coloración es NP-completo

$$E=(x'_1+x_2+x_3)(x_1+x_3+x'_4)(x'_2+x_3+x_4)$$

E se satisface para una asignación 0,1

Una 3-coloración de G



3-coloración es NP-completo

$$E=(x'_1+x_2+x_3)(x_1+x_3+x'_4)(x'_2+x_3+x_4)$$

E se satisface para una asignación 0,1

Si E se satisface con una asignación T de valores 0,1 coloreamos el grafo G así:

- Si $x_i=1$ (verdad) $\Rightarrow$ x_i color , x'_i color
- Si $x_i=0$ (falso) $\Rightarrow$ x_i color , x'_i color

Una 3-coloración de G

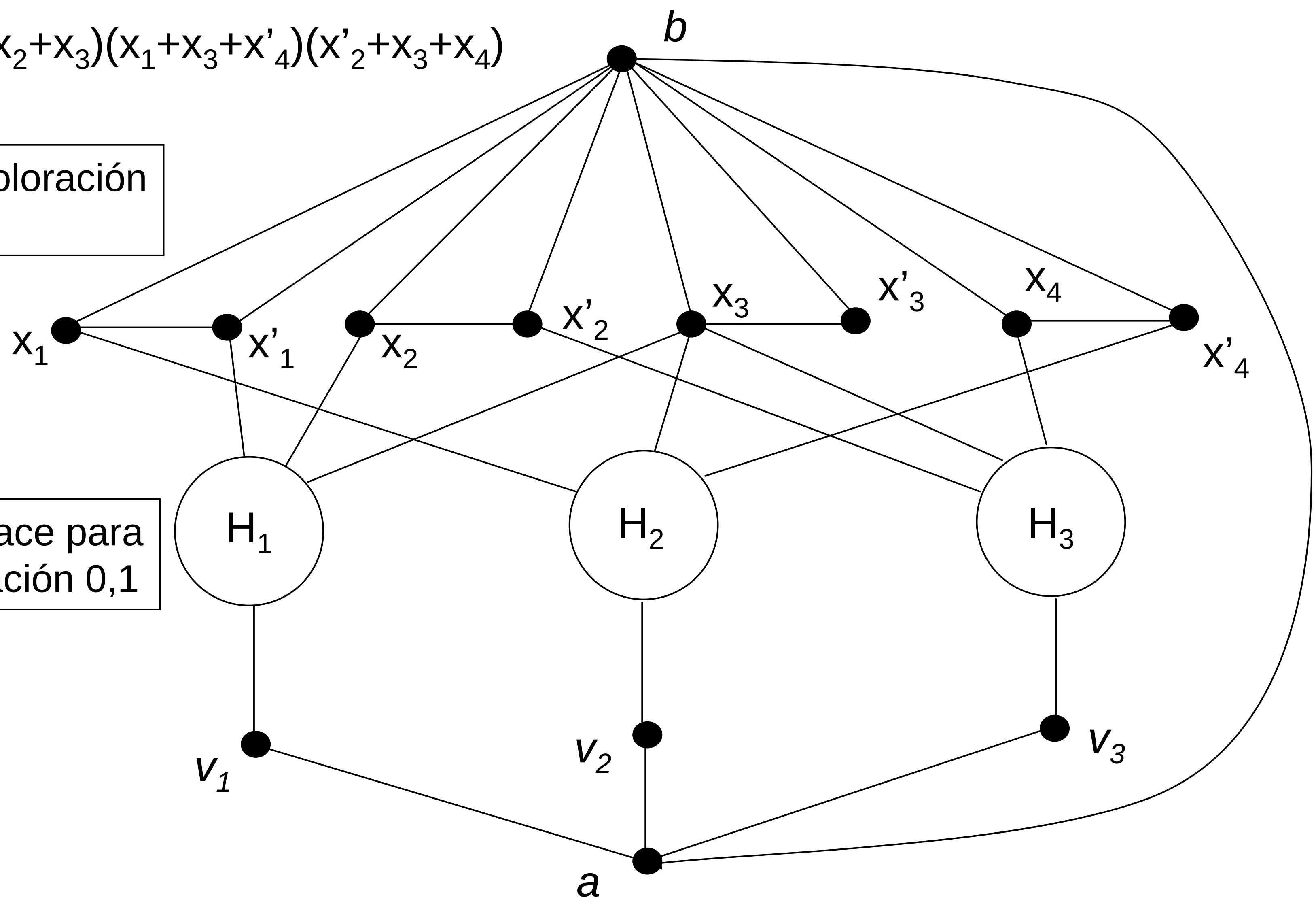
Como la asignación T satisface la expresión E, alguna literal de cada H_k no puede tener color 0, luego cada uno de los vértices v_k o bien tiene color 1 (verde) o color 2 (rojo). Coloreamos el vértice *a* con el color 0 (blanco) y *b* con el color 2 (rojo) y tendremos una 3-coloración de G

3-coloración es NP-completo

$$E=(x'_1+x_2+x_3)(x_1+x_3+x'_4)(x'_2+x_3+x_4)$$

Una 3-coloración de G

E se satisface para una asignación 0,1

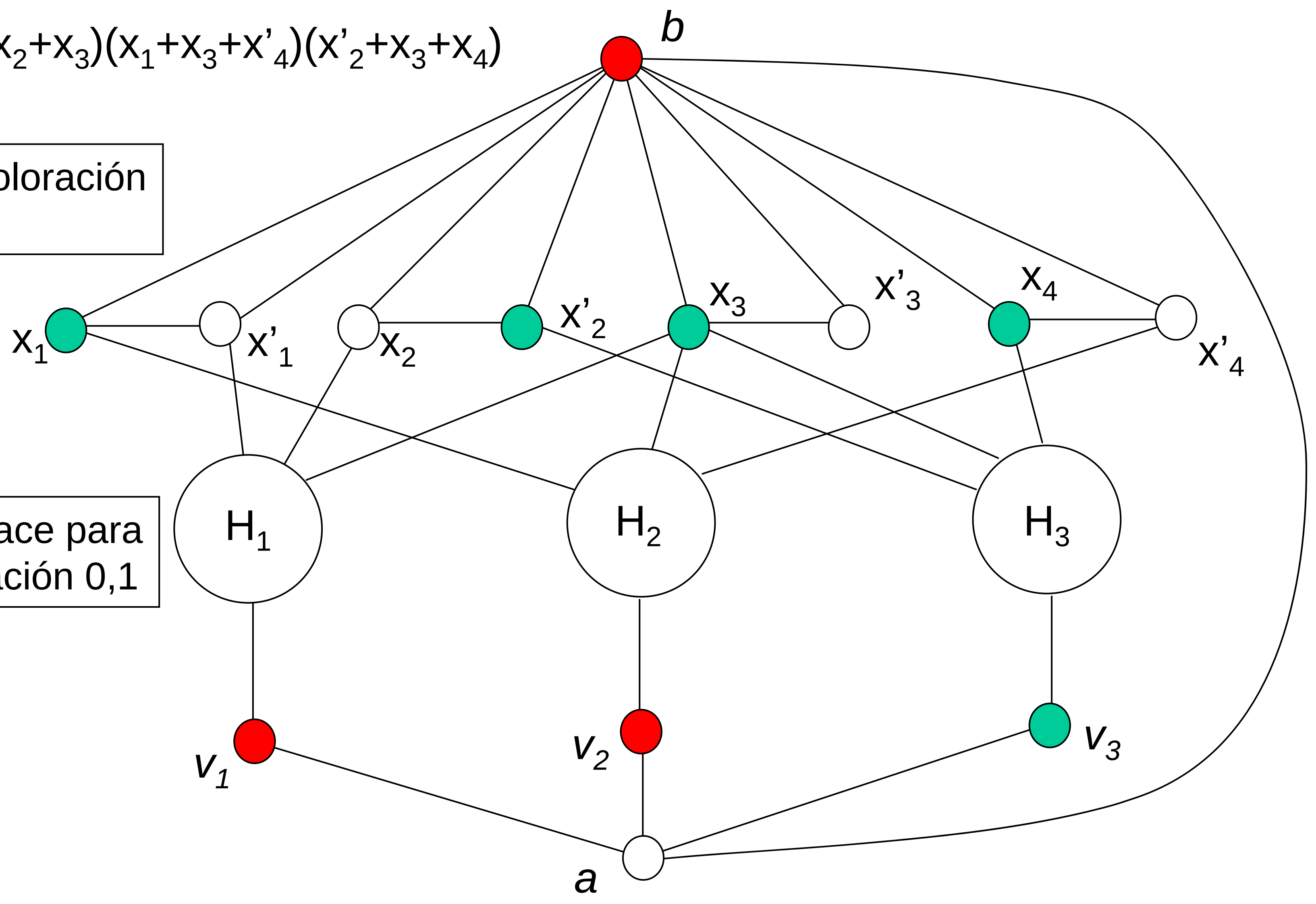


3-coloración es NP-completo

$$E=(x'_1+x_2+x_3)(x_1+x_3+x'_4)(x'_2+x_3+x_4)$$

Una 3-coloración de G

E se satisface para una asignación 0,1



3-coloración es NP-completo

$$E=(x'_1+x_2+x_3)(x_1+x_3+x'_4)(x'_2+x_3+x_4)$$

Una 3-coloración de G

E se satisface para una asignación 0,1

Si G es 3-coloreable, renombrando los colores podemos suponer que *a* color 0 (blanco) y *b* color 2 (rojo)

Así cada variable tiene una literal con 0 (blanco) y otra con 1 (verde).

Construimos una asignación T de verdad (valores 0, 1)

x_j es verdad (toma el valor 1) $\Leftrightarrow$ recibe el color 1 (verde)

Los vértices v's de las piezas H's no tienen color 0, luego cada producto (cláusula) de E debe tener una literal de color 1 (verde), es decir, cada producto toma el valor 1 y, por tanto, la expresión E toma el valor 1 (se satisface) para la asignación de 0,1 realizada.