NP-Completitud

Agustín J. González

ELO320: Estructura de Datos y Algoritmos

1er. Sem. 2002

Introducción

- Hasta ahora todos los algoritmos estudiados han sido algoritmos de tiempo polinomial: para entrada n, el peor caso de tiempo de ejecución es $O(n^k)$.
- Pero no todos los algoritmos son polinomiales.
- Se dice que los problemas solubles en tiempo polinomial son tratables y los otros son intratables.
- Veremos problemas cuyo status es desconocido; es decir, no se ha descubierto algoritmo de tiempo polinomial para ellos, ni se ha podido probar un límite inferior de tiempo super-polinomial.
- Se cree que estos problemas son intratable (no polinomiales)
- Si podemos establecer que un problema es NP-completo, mejor nos dedicamos a encontrar una buena aproximación que a buscar una solución exacta.

Ejemplo: Satisfacción de circuitos

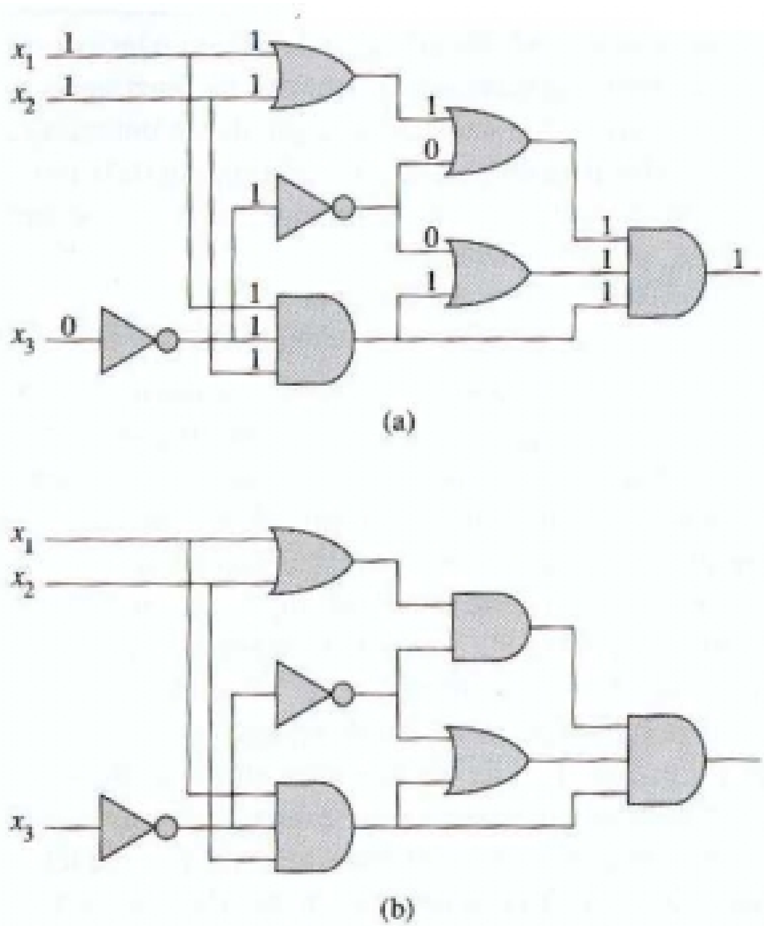


Figure 36.6 Two instances of the circuit-satisfiability problem. **(a)** The assignment $\langle x_1 = 1, x_2 = 1, x_3 = 0 \rangle$ to the inputs of this circuit causes the output of the circuit to be 1. The circuit is therefore satisfiable. **(b)** No assignment to the inputs of this circuit can cause the output of the circuit to be 1. The circuit is therefore unsatisfiable.

- La idea que normalmente se usa es ubicar un problema que sea NP-Completo y luego probamos que otro problema es NP-completo tratando de reducirlo al ya conocido como NP-Completo.
- Un circuito combinacional es “Satisfactorio” si hay una vector de entrada para el cual el circuito arroja 1.
- Hay circuitos para los cuales no hay entrada que produzca salida 1. Éstos son no “satisfactorios”.
- Ejemplo:Figura 36.6

Ejemplo: Satisfacción de circuitos (Cont)

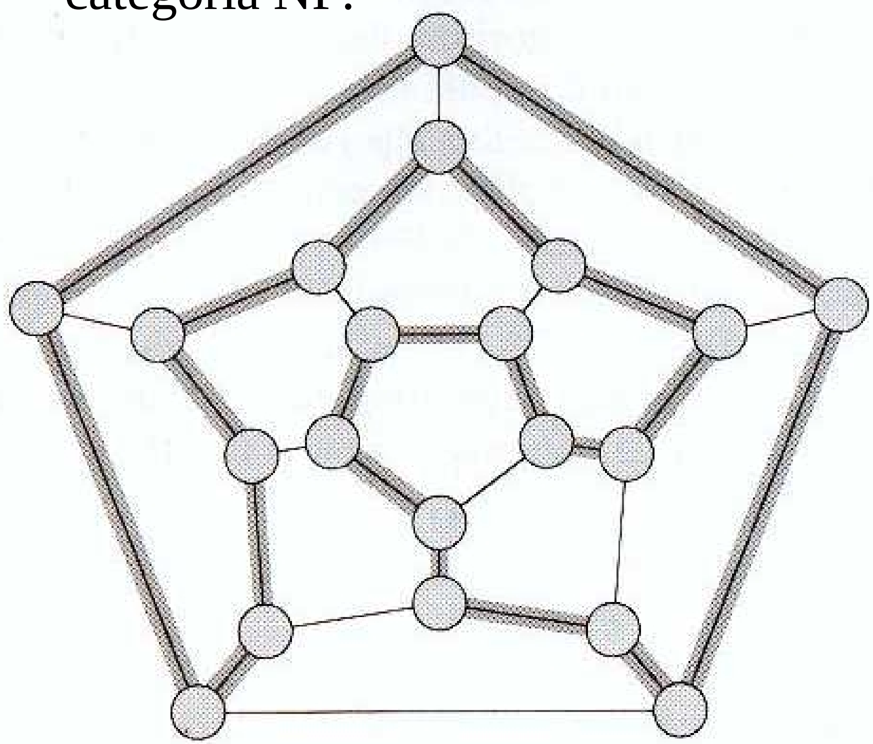
- El problema es: Dado un circuito combinacional compuesto por compuertas AND, OR, y NOT, ¿es éste satisfactorio?
- Se podría intentar la búsqueda de una salida alta, probando todas las entradas posibles.
- Si tenemos k entradas hay 2^k posibles entradas.
- Cuando el tamaño de C es un polinomio en k (k no es constante), el chequeo de todas las entradas toma un tiempo super-polinomial.

P y NP

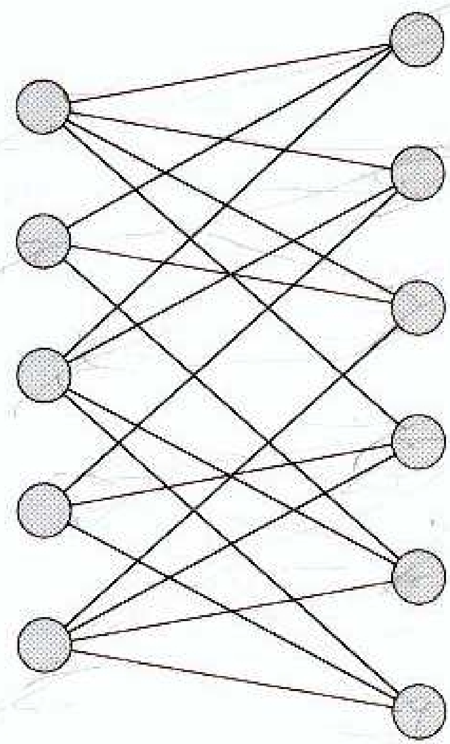
- La clase de complejidad **P** corresponde a todos los problemas que pueden ser resueltos en tiempo polinomial
- La clase de complejidad NP (viene de tiempo polinomial no-determinístico) corresponde a la clase de problemas cuya solución puede ser verificada en tiempo polinomial.
- Por ejemplo: Consideremos el problema del ciclo Hamiltoniano “Tiene un grafo G un ciclo Hamiltoniano? (Aquél que recorre todos los nodos en un camino simple) Un posible candidato para este problema puede ser verificado en tiempo polinomial, luego este problema está en la categoría NP.

Ciclo Hamiltoniano

- Consideremos el problema del ciclo Hamiltoniano "¿Tiene un grafo G un ciclo Hamiltoniano? (Aquél que recorre todos los nodos en un camino simple) Un posible candidato para este problema puede ser verificado en tiempo polinomial, luego este problema está en la categoría NP.



(a)



(b)

Figure 36.1 **(a)** A graph representing the vertices, edges, and faces of a dodecahedron, with a hamiltonian cycle shown by shaded edges. **(b)** A bipartite graph with an odd number of vertices. Any such graph is nonhamiltonian.

¿P = NP?

- Se desconoce si estos dos conjuntos son equivalentes, pero muchos científicos creen que estas dos clases son distintas.
- Científicos creen que la clase NP incluye problemas que no están en P.
- Generalmente es más fácil verificar una posible solución que encontrar la solución de un problema.
- Esta pregunta lleva a la aparición de la clase NP-completo (NPC). Aquellos problemas cuyo status es desconocido pero se cree no tienen soluciones polinomiales. Hasta ahora no se ha encontrado solución polinomial para ninguno de ellos.
- Se puede probar que: si un problema NP-completo tiene solución polinomial, todos la tendrían.

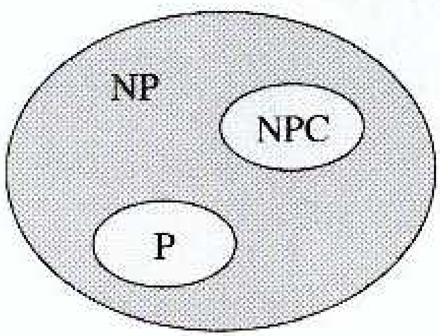


Figure 36.5 How most theoretical computer scientists view the relationships among P, NP, and NPC. Both P and NPC are wholly contained within NP, and $P \cap NPC = \emptyset$.

Problemas NP-Completos

- ***Problema Pandilla (clique)***:
- Una pandilla en un grafo no dirigido G=(V,E) es un subconjunto de vértices que cumplen con tener conexión todos con todos. El tamaño del pandilla es el numero de vértices de el.
- Problema: Existe un pandilla de tamaño k en el grafo?
- Algoritmo simple: buscar todos los grupos de k nodos y probar para cada uno si están todos conectados.

- ***Problema de cubierta de vértices***
- La cubierta de vértices de un grafo no dirigido G=(V,E) es un subconjunto V' de V tal que si (***u,v***) esta en E, ***u*** esta en V' o ***v*** esta en V'.
- Cada vértice cubre su arco incidente y la idea es buscar el conjunto que cubra todos los arcos.
- Problema: encontrar la cubierta de vértices de tamaño mínimo.

Ejemplos de Pandilla (Clique) y cubierta de vértices

- Pandilla (clique) ={u,v,x,y}
- Cubierta de vértices = {w,z}

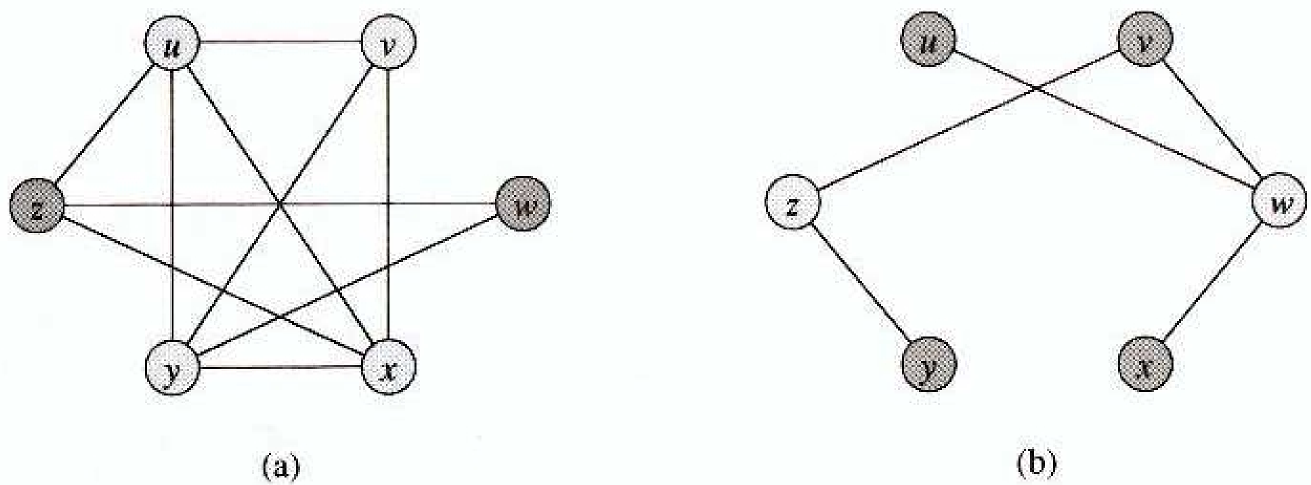


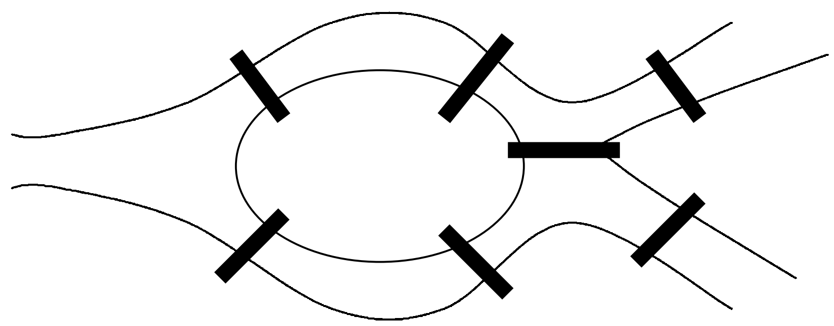
Figure 36.13 Reducing CLIQUE to VERTEX-COVER. **(a)** An undirected graph $G = (V, E)$ with clique $V' = \{u, v, x, y\}$. **(b)** The graph $\overline{G}$ produced by the reduction algorithm that has vertex cover $V - V' = \{w, z\}$.

Más Problemas NP-Completos

- ***Problema del ciclo hamiltoniano***
- Problema: Existe un ciclo simple en G que contenga a todos los vértices?
- ***Vendedor viajero***
- Se tiene un grafo completo (todos los arcos) con costos asociados a cada arco. El vendedor desea hacer un “tour” (ciclo hamiltoniano) tal que el costo sea mínimo.
- ***El camino más largo***
- ¿ Cuál es el camino simple más largo que conecta dos nodos del grafo?
- ***Coloración***
- Existe una forma de asignar k colores a los vértices de un grafo de modo que no hayan arcos conectando vértices del mismo color?
- ***Conjunto independiente***
- Cual es el conjunto más grande de vértices que no están conectados entre sí en el grafo?
- ***Isomorfismo de grafos***
- ¿Podemos hacer dos grafos iguales sólo renombrando los vértices?

Problemas

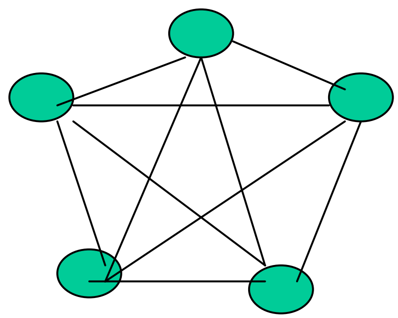
- Hay muchos problemas parecidos que no son NP-completos.
- Ejemplo: Recorrido de Euler (1736)
- ¿Existe un camino que conecte dos vértices y use todos los arcos exactamente una vez?
- Origen los 7 puentes de Konigsberg



- Si lo buscado es un Euler tour (es decir debemos volver al mismo punto de partida, hay solución solo si es un grafo conexo y cada vértice tiene un número par de arcos adyacentes.
- Si lo buscado es un camino Euler (es decir cubrimos todos los arcos una vez, pero no volvemos al punto de partida), hay solución si y sólo si, el grafo es conexo y hay exactamente dos nodos de grado impar.

Problemas

- Planariedad
- Dado un grafo arbitrario: ¿podemos dibujarlo sin arcos que se crucen?
- Kuratowski propuso un teorema en el que demuestra que los únicos grafos no planares son aquellos que luego de remover arcos de grado 2 (con dos arcos adyacentes) tienen un subconjunto isomórfico con



o