

# Complejidad y optimización

## Problemas computacionales, intratabilidad y problemas NP completos

Carlos Andrés Delgado S.

Facultad de Ingeniería. Universidad del Valle

Febrero 2017

Complejidad y  
optimización

Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

## 1 Problemas computacionales y tratabilidad

## 2 Clasificación problemas computacionales

## 3 Problemas NPC

### ■ Problema satisfactibilidad (SAT)

## Introducción

- En ciencias de la computación se utilizan algoritmos para resolver problemas y siempre se busca que esta solución sea la más eficiente.
- Una de las principales características que describe un algoritmo es su complejidad computacional que puede ser en términos de tiempo de ejecución o espacio necesario para hacerlo

Complejidad y  
optimización

Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

## Clases de complejidad

Existen varias clases de complejidad computacional

- Temporal
- Espacial

## Complejidad temporal

Es el tiempo requerido por un algoritmo para solucionar un problema.

- Varios algoritmos pueden solucionar el mismo problema
- Siempre buscamos la solución con menor complejidad temporal

## Complejidad especial

Es el espacio requerido por las estructuras de datos de un algoritmo para solucionar un problema.

- Está directamente relacionada con la complejidad temporal, por ejemplo no es lo mismo recorrer un arreglo unidimensional que uno bidimensional
- Adquiere importancia la **selección de las estructuras de datos** utilizadas en la implementación, ejemplo: arreglos o listas enlazadas.

## Notación

De acuerdo a la complejidad de la solución ante una entrada de tamaño  $n$  se utilizan las siguientes notaciones:

- $O(f(n))$
- $\Omega(f(n))$
- $\Theta(f(n))$

## Notación

Recordemos ¿Que significa?

- $O(1)$
- $O(n^3)$
- $O(2^n)$
- $\Omega(1)$
- $\Omega(n^2)$
- $\Theta(1)$
- $\Theta(n \log(n))$



Complejidad y  
optimización

Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

## General

- En las ciencias de la computación los problemas tienen diferentes niveles de dificultad
- Algunos se pueden resolver y otros no, con la capacidad que contamos en la actualidad
- Se clasifican los problemas con el objetivo de agruparlos de acuerdo a un problema conocido

# Clasificación problemas

Complejidad y  
optimización

Carlos Andrés  
Delgado S.

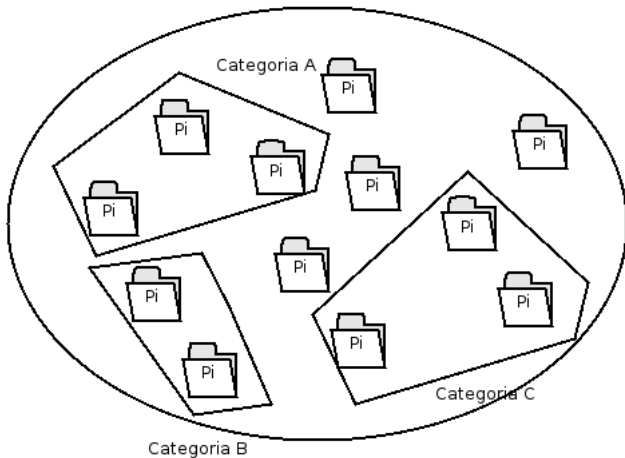
Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

## Universo de los problemas en ciencias de la computación



# Clasificación problemas

Complejidad y  
optimización

Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

## De acuerdo al tipo

Se pueden clasificar en: Problemas de solución y problemas de decisión

## Problemas optimización

Estos consisten en encontrar la mejor solución a un problema de acuerdo a un criterio, estos pueden ser de decisión o solución.

## Ejemplo problemas de solución

- 1 Encontrar todos los caminos en un grafo desde un nodo  $A$  hasta un nodo  $B$
- 2 Hallar las combinaciones de elementos en un conjunto  $S$ , cuya suma sea  $P$

## Ejemplo problemas de decisión

- 1 El camino más corto entre  $A$  y  $B$  está conformado por los nodos  $\{C, D, E\}$
- 2 Los elementos del subconjunto  $S_1, S_2, \dots, S_N$  del conjunto  $S$  suman  $P$

Respuesta: **SI** o **NO**

## Ejemplo problemas de optimización

- 1 Encontrar el camino más corto en un grafo desde un nodo  $A$  hasta un nodo  $B$
- 2 Hallar las combinación con el menor número de elementos en un conjunto  $S$ , cuya suma sea  $P$

Complejidad y  
optimización

Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

## Decibilidad

Los problemas pueden ser decidibles o indecidibles.

## Problemas decidibles

Pueden ser tratables o intratables

## Problemas indecidibles

Puede ser indecidibles o altamente indecidibles.

## Decibilidad

- 1 Los problemas decidibles son aquellos para los cuales se puede construir un algoritmo que lleve a una respuesta correcta
- 2 En sentido más formal, dado un problema  $P$  y una entrada  $s$  esta puede ser procesada por una máquina de Turing
- 3 Estos problemas están clasificados en tratables e intratables.



## Decibilidad

Existen problemas indecidibles conocidos

- Problema de la parada. *Dado un algoritmo  $p$  y una entrada  $s$ , determinar si  $p$  puede procesar  $s$  en un número finito de pasos*
- Problema de la matriz mortal. *Determinar si existe un orden de multiplicaciones de matrices cuadradas de tal forma se pueda obtener la matriz cero*
- Conjetura de Collatz ( $3n + 1$ ) *La sucesión dada por*

$$f(n) = \begin{pmatrix} 3n + 1 & \text{con } n \text{ impar} \\ \frac{n}{2} & \text{con } n \text{ par} \end{pmatrix}$$

¿Es finita para todo  $n$ ?

## Tratabilidad

- 1 Los problemas tratables se pueden solucionar con algoritmos polinómicos  $O(n^k)$
- 2 Los problemas intratables se resuelven en tiempos no polinómicos  $\Omega(a^n)$

# Clasificación Máquinas de Turing

Complejidad y  
optimización

Carlos Andrés  
Delgado S.

Problemas  
computacionales y  
tratabilidad

Clasificación  
problemas  
computacionales

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

En las máquinas de Turing tenemos finitas e infinitas. Dentro de las finitas se tiene:

## Determinista

Esta ejecuta una acción definida ante la lectura de un símbolo de entrada. **Este es el modelo computacional actual**

## No determinista

Ejecuta la mejor acción posible ante la lectura de un símbolo de entrada. **¡Esto nos quitaría el trabajo :(!**

Complejidad y  
optimización

Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

1 Problemas computacionales y tratabilidad

2 Clasificación problemas computacionales

3 Problemas NPC

- Problema satisfactibilidad (SAT)

## Problema P

Es aquel problema de decisión que se puede solucionar en tiempo polinomial por una máquina determinista.

## Problema NP

Son aquellos que **NO** pueden ser solucionados en tiempo polinomial por una máquina determinista. Pero **SI** pueden ser solucionado en tiempo polinomial por una máquina no determinista.

## Problema NPC

NP-Completo, son problemas que cumplen las siguientes características:

- No se ha demostrado que **NO** existe un algoritmo con complejidad polinomial que los pueda resolver
- No se ha encontrado una solución polinomial para ellos.

# ¿P=NP?

Complejidad y  
optimización

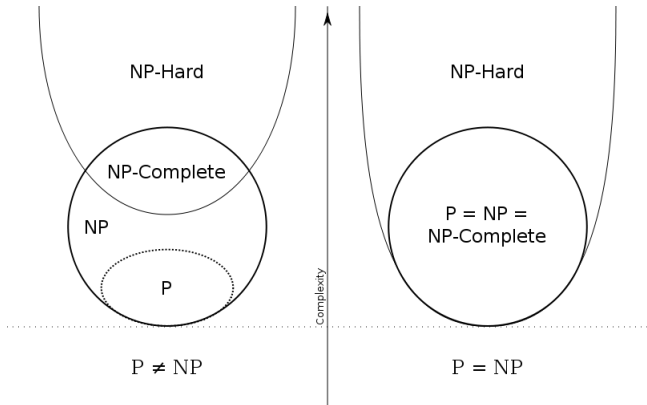
Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)



El problema ¿P=NP?

## Complejidad y optimización

Carlos Andrés  
Delgado S.

Problemas computacionales y tratabilidad

Clasificación problemas computacionales

Problemas NPC

Problema satisfactibilidad (SAT)

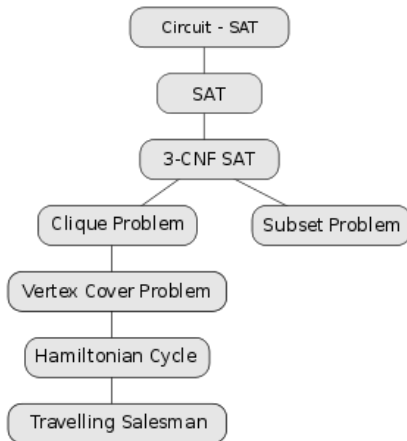
1 Problemas computacionales y tratabilidad

2 Clasificación problemas computacionales

3 Problemas NPC

■ Problema satisfactibilidad (SAT)





## Definición

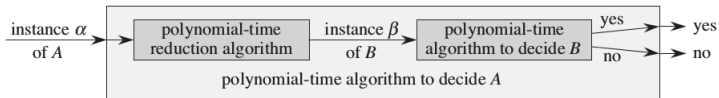
Sean  $A$  y  $B$  dos problemas de decisión pertenecientes a NP. Se puede transformar una instancia  $A$  en una instancia  $B$  en tiempo polinomial.

## Requerimiento

Se debe cumplir:

$$A \leq_p B$$

Es decir  $A$  tiene una complejidad menor o igual a  $B$ .



## Definición

Una reducción es correcta si:

- Instancias negativas de  $A$  resultan en instancias negativas de  $B$
- Instancias positivas de  $A$  resultan en instancias positivas de  $B$

## Introducción

Este es el primer problema NPC completo demostrado. La demostración fue realizada por Stephen Cook en 1971

## Definición

Cualquier problema NPC puede ser reducido desde SAT en tiempo polinomial.

## Definición formal

El problema consiste en un conjunto de variables booleanas  $x_1, x_2, x_3, \dots, x_n$  y un conjunto de cláusulas  $s_1, s_2, s_3, \dots, s_n$  en forma normal conjuntiva, donde existen valores de las variables en la cual la expresión es verdadera.

## Example

$$\begin{aligned} & (x_1 \vee x_2) \wedge (x_2 \vee x_3) \\ & (x_1 \vee x_3 \vee \bar{x}_4) \wedge (x_2 \vee x_3) \\ & (x_1 \vee x_2 \vee x_3) \wedge \bar{x}_1 \end{aligned}$$

## Enunciados demostración

- Se utiliza una máquina de Turing no determinista para el cómputo de este algoritmo.
- Se codifica la entrada de la máquina de Turing de tal forma esta resulta en una fórmula booleana en FNC.
- Se la máquina de turing acepta la entrada, entonces la fórmula es satisfactible.

## Explicación de la demostración

La máquina de Turing no determinista al poder procesar la fórmula booleana, dando los valores correctos a cada variable de cada clausula, conduce a un estado de aceptación, de acuerdo a los valores asignados a cada variable.

## Consecuencias

- Si se llegara a demostrar que SAT puede ser resuelto en tiempo polinomial entonces cualquier problema NPC puede ser resuelto en tiempo polinomial.
- Al reducir cualquier problema NP a SAT, se demuestra que es NPC.



## Complejidad y optimización

Carlos Andrés  
Delgado S.

Problemas  
computaciona-  
les y  
tratabilidad

Clasificación  
problemas  
computaciona-  
les

Problemas  
NPC

Problema  
satisfactibilidad  
(SAT)

