

Investigación de Operaciones

Introducción algoritmos incompletos

Leslie Pérez Cáceres

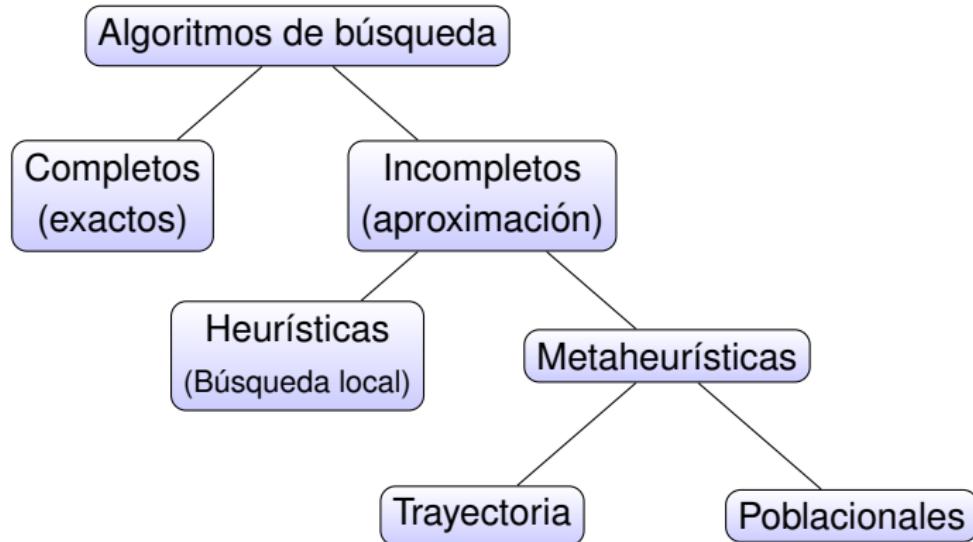
leslie.perez@pucv.cl

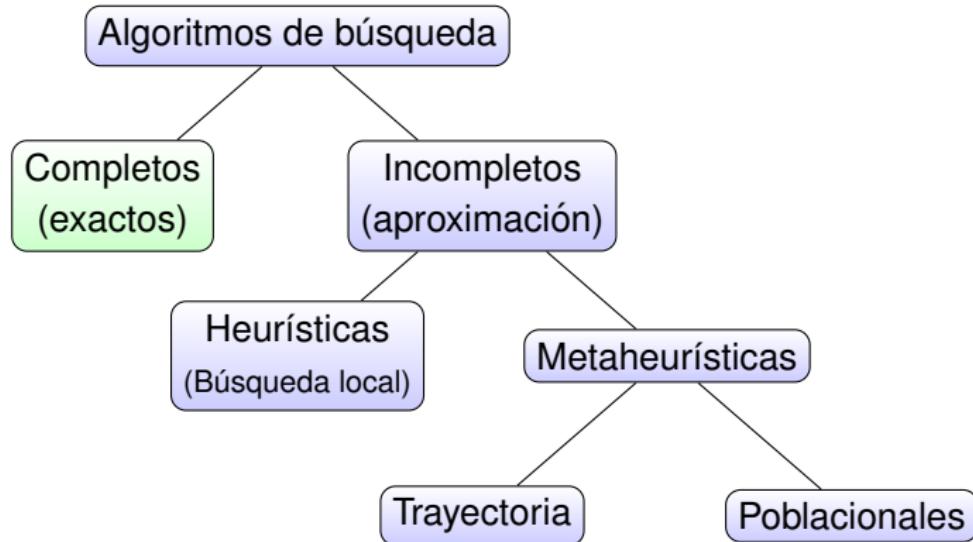
Escuela de Informática

Pontificia Universidad Católica de Valparaíso



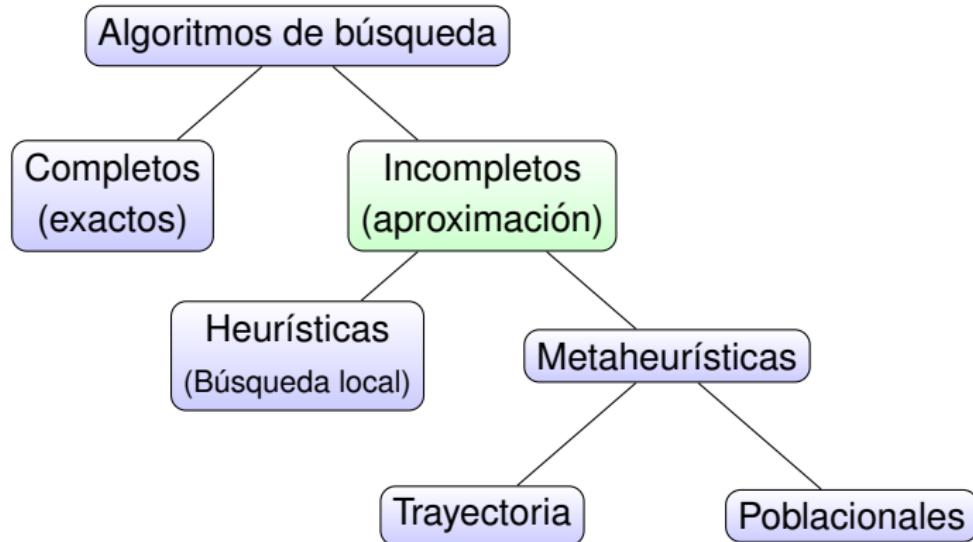
PONTIFICIA
UNIVERSIDAD
CATÓLICA
DE
VALPARAÍSO





- Entregan la **solución óptima**
- Son capaces de averiguar que **NO** existe solución factible
- Su tiempo de ejecución crece rápidamente en relación a
 - 1 la cantidad de variables
 - 2 la cantidad de valores de los dominios

¡Estas técnicas fueron revisadas en la primera parte de este curso!



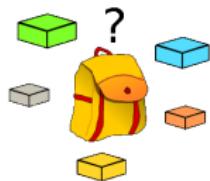
- Normalmente estocásticas
- Entregan la mejor solución encontrada:
sin garantía de optimalidad
- Capaces de manejar en gran espacios de búsqueda
- No necesitan conocimiento a priori del problema

Problema de la Mochila

Dos exploradores se encuentran en una caverna a punto de colapsar.

Encuentran un tesoro de 8 objetos, cada uno con un peso y valor.

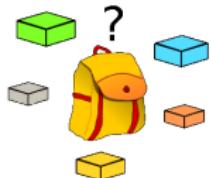
Rápidamente deben seleccionar cuales llevar: no podrán regresar.



	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

- el peso de los objetos seleccionados no debe ser mas de **101 kg**
- el valor de los objetos debe ser el **máximo** posible

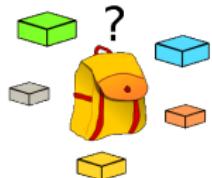
Modelo matemático Mochila



	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

¿Como podría ser el modelo matemático de este problema?

Modelo matemático Mochila



	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

$x_i = 1$ si el objeto i es seleccionado, 0 en otro caso

$$\max (32x_1 + 47x_2 + 18x_3 + 26x_4 + 85x_5 + 33x_6 + 45x_7 + 59x_8)$$

sujeto a:

$$(26x_1 + 48x_2 + 21x_3 + 22x_4 + 95x_5 + 43x_6 + 55x_7 + 52x_8) \leq 101$$

$$x_1, \dots, x_8 \in \{0, 1\}$$

Modelo general:

$x_i = 1$ si el objeto i es seleccionado, 0 en otro caso

maximizar

$$f(X) = \sum_{i=1}^n v_i x_i$$

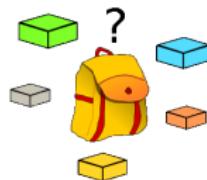
sujeto a:

$$\sum_{i=1}^n p_i x_i \leq W$$

$$x_i \in \{0, 1\} \quad \forall i \in [1, n]$$

Representación Mochila

¿Cómo **representar** soluciones al programar la búsqueda de soluciones?



	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

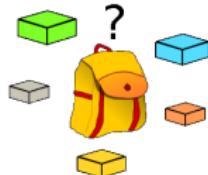
Defina una estructura de datos para representar la solución que solo lleva el objeto 1 (o_1)

Podemos representar la solución como un **arreglo**

$$s = \boxed{1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0} \quad p(s) = 26 \quad f(s) = 32$$

Comparación Mochila

¿Cómo **comparar** soluciones soluciones?



	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

¿Qué solución es mejor?

$$s_1 = \boxed{1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0} \quad p(s_1) = 26 \quad f(s_1) = 32$$

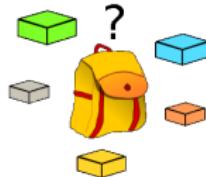
$$s_2 = \boxed{0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0} \quad p(s_2) = 95 \quad f(s_2) = 85$$

¿En qué basamos la comparación de las soluciones?

→ la comparación es basada en la **calidad** (función objetivo)

Comparación Mochila

¿Cómo **comparar** soluciones soluciones?



	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

¿Qué solución es mejor?

$$s_2 = \boxed{0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0} \quad p(s_2) = 95 \quad f(s_2) = 85$$

$$s_3 = \boxed{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1} \quad p(s_3) = 107 \quad f(s_3) = 104$$

¿Qué sucede con la factibilidad y la comparación?

La **función de evaluación** añade una **penalización** a la función objetivo

→ permite incluir la **factibilidad** dentro de la evaluación de calidad

Definimos la función de evaluación $z(s)$:

$$z(s) = f(s) + \Phi(s)$$

Por ejemplo, para soluciones infactibles la penalización $\Phi(s)$ puede ser definida como:

- una constante suficientemente grande

Ej. $\Phi(s) = \pm 1\,000\,000$

- una función basada en la instancia

Ej. $\Phi(s) = \pm 100N$ (N : tamaño instancia)

La penalización suma o resta una cantidad dependiendo si el problema es de maximización o minimización

La **función de evaluación** añade una **penalización** a la función objetivo

→ permite incluir la **factibilidad** dentro de la evaluación de calidad

Definimos la función de evaluación $z(s)$:

$$z(s) = f(s) + \Phi(s)$$

Por ejemplo, para soluciones infactibles la penalización $\Phi(s)$ puede ser definida como:

- una constante suficientemente grande

Ej. $\Phi(s) = \pm 1\,000\,000$

- una función basada en la instancia

Ej. $\Phi(s) = \pm 100N$ (N : tamaño instancia)

La penalización suma o resta una cantidad dependiendo si el problema es de maximización o minimización

Función de evaluación: penalización constante

Volvamos a nuestro ejemplo:

	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

$$s_3 = \boxed{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1} \quad p(s_3) = \textcolor{red}{107} \quad f(s_3) = 104$$

Podemos definir la penalización

$$\Phi(s) = \begin{cases} 0, & \text{si } s \text{ es factible} \\ -1000, & \text{si } s \text{ es infactible} \end{cases}$$

La función de evaluación de la solución s_3 quedaría:

$$\begin{aligned} z(s_3) &= f(s_3) + \Phi(s_3) \\ &= 104 - 1000 \\ &= -896 \end{aligned}$$

La **función de evaluación** añade una **penalización** a la función objetivo

→ permite incluir la **factibilidad** dentro de la evaluación de calidad

Definimos la función de evaluación $z(s)$:

$$z(s) = f(s) + \Phi(s)$$

Por ejemplo, para soluciones infactibles la penalización $\Phi(s)$ puede ser definida como:

- una constante suficientemente grande
Ej. $\Phi(s) = \pm 1\,000\,000$
- una función basada en la instancia

Ej. $\Phi(s) = \pm 100N$ (N : tamaño instancia)

La penalización suma o resta una cantidad dependiendo si el problema es de maximización o minimización

Función de evaluación: penalización función

	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52

$$s_3 = \boxed{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1} \quad p(s_3) = \textcolor{red}{107} \quad f(s_3) = 104$$

Podemos definir la penalización

$$\Phi(s) = \begin{cases} 0, & \text{si } s \text{ es factible} \\ -\rho \cdot (p(s) - W), & \text{si } s \text{ es infactible} \end{cases}$$

ρ : maxima razón valor/peso (*valor/peso*)

W : parámetro del máximo peso permitido

Función de evaluación: penalización función

ρ : **maxima** razón valor peso (*valor/peso*)

	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8
valor	32	47	18	26	85	33	45	59
peso	26	48	21	22	95	43	55	52
valor/ peso	1.23	0.98	0.86	1.18	0.89	0.77	0.82	1.13

La razón *valor/peso* del primer objeto o_1 :

$$v_1/p_1 = 32/26 = 1.23$$

La razón *valor/peso* del primer objeto o_2 :

$$v_2/p_2 = 47/48 = 0.98$$

...

Calculamos la máxima razón:

$$\rho = 1.23$$

Función de evaluación: penalización función

La función de evaluación queda:

$$\Phi(s) = \begin{cases} 0, & \text{si } s \text{ es factible} \\ -1.23(p(s) - 101), & \text{si } s \text{ es infactible} \end{cases}$$

Calculemos la función de evaluación para s_3 :

$$s_3 = \boxed{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1} \quad p(s_3) = \textcolor{red}{107} \quad f(s_3) = 104$$

$$\begin{aligned} z(s_3) &= f(s_3) + \Phi(s_3) \\ &= 104 - 1.23(p(s_3) - 101) \\ &= 104 - 1.23(107 - 101) \\ &= 104 - 1.23(6) \\ &= 96.62 \end{aligned}$$

Al usar una **función de evaluación**, la búsqueda de soluciones factibles se realiza durante la optimización de la función objetivo

- debe elegirse una función de evaluación adecuada para permitir la búsqueda de soluciones
- no siempre es necesario/posible usar una función de evaluación
- es aconsejado cuando encontrar soluciones factibles no es una tarea fácil (incluso imposible)

En las próximas clases, aprenderemos como buscar soluciones en **grandes espacios** de soluciones

→ y la necesidad de aplicar una función de evaluación