

Metaheurísticas

Unidad 1
Introducción a las Metaheurísticas

Tema 1: Introducción

Objetivos

- Entender el concepto de metaheurísticas
- Conocer los elementos más importantes en el diseño de una metaheurística
- Conocer diferentes criterios de clasificación de metaheurísticas

Bibliografía

[Tal09] E.-G. Talbi. **Metaheuristics. From design to implementation.** Wiley, 2009

[Meli03] B. Melián, J.A. Moreno, J.M. Moreno. **Metaheurísticas: una visión global.** Revista Iberoamericana de Inteligencia Artificial, 9, 2003, 7-28

Motivación

- Múltiples problemas de optimización de ciencia, ingeniería, economía, etc. Estamos hablando de problemas complejos y difíciles de resolver
 - No se pueden resolver de forma exacta en un tiempo razonable
 - La alternativa es el uso de algoritmos aproximados
- Tipos de algoritmos aproximados

Motivación

- Múltiples problemas de optimización de ciencia, ingeniería, economía, etc. Estamos hablando de problemas complejos y difíciles de resolver
- Tipos de algoritmos aproximados
 - Heurísticas: Dependientes del problema
 - Metaheurísticas: Algoritmos aproximados más generales y aplicables a una gran variedad de problemas de optimización. Resuelven problemas:
 - ▶ De forma más rápida
 - ▶ Más complejos
 - ▶ Con algoritmos más robustos

Motivación

- Múltiples problemas de optimización de ciencia, ingeniería, economía, etc. Estamos hablando de problemas complejos y difíciles de resolver
- Tipos de algoritmos aproximados
 - Heurísticas: Dependientes del problema
 - Metaheurísticas: Algo que se aplica a una gran variedad de problemas
 - ▶ De forma más rápida
 - ▶ Más complejos
 - ▶ Con algoritmos más

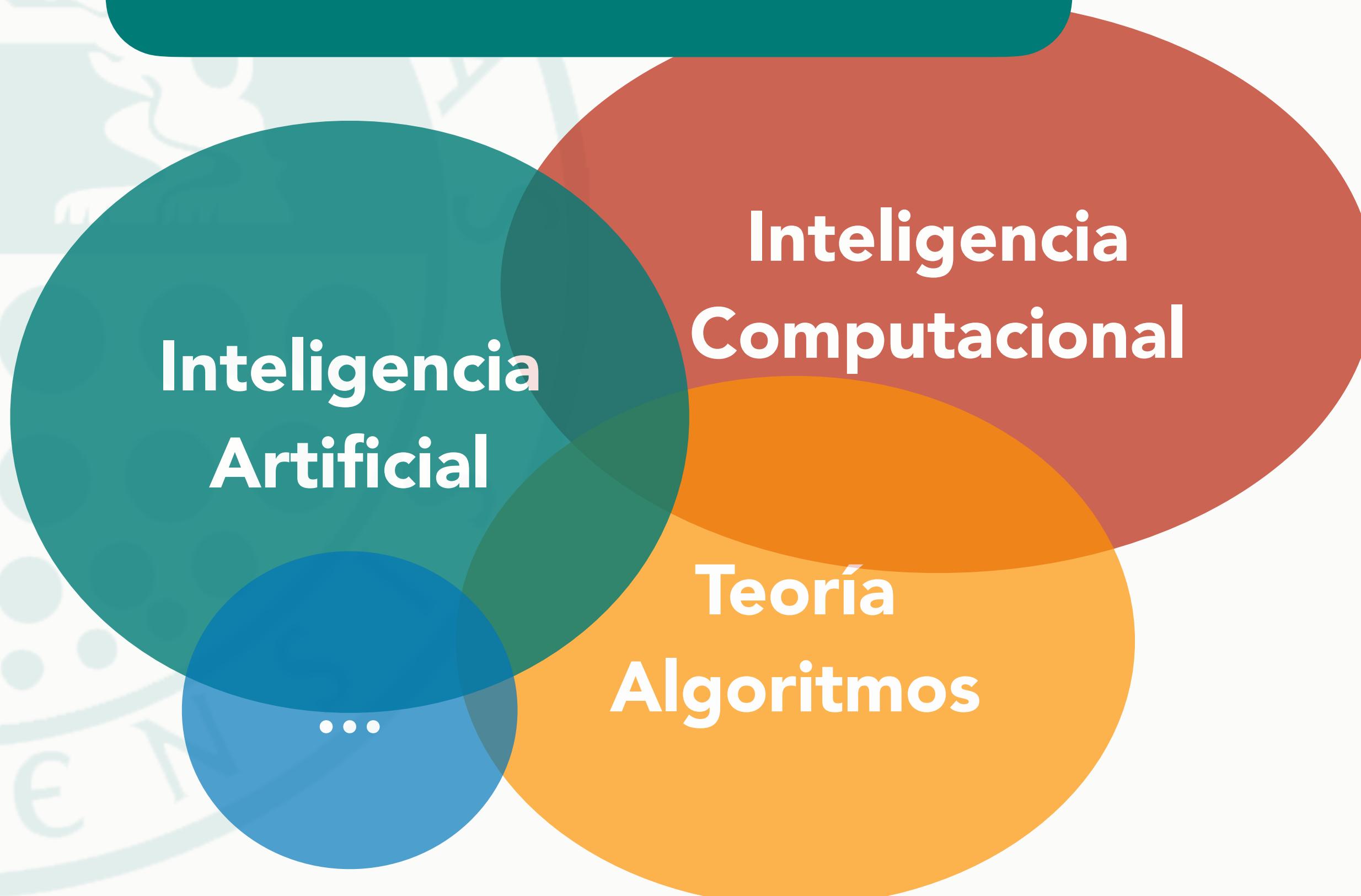
Cuando se usa como sustantivo, se refiere a la disciplina, el arte o la ciencia del descubrimiento. Cuando aparece como adjetivo, se refiere a cosas más concretas, como estrategias, reglas, silogismos y conclusiones heurísticas

Motivación

Metaheurística = Optimización/Búsqueda

Intersección de campos

Metáforas naturales



- Evolución de especies
- Nubes de partículas
- Procesos físicos
- Sociedades de insectos
- etc.

Índice

1. Algoritmos para resolución de problemas de búsqueda
2. Algoritmos aproximados
3. Definición, clasificación y conceptos básicos
4. Evaluación
5. Paralelización
6. Aplicaciones

1

Algoritmos para resolución de problemas de búsqueda

1. Algoritmos para resolución de problemas de búsqueda

algoritmos ya estudiados

- Algoritmos no informados
- Algoritmos informados

1. Algoritmos para resolución de problemas de búsqueda

algoritmos ya estudiados

- Algoritmos no informados
 - Primero en profundidad o anchura
- Algoritmos informados
 - Heurísticos: Estrella
 - Con adversario

1. Algoritmos para resolución de problemas de búsqueda

conceptos básicos

- **Objetivo general:** resolución de problemas mediante procesos de cómputo
- **Solución:** sistema informático implementando un algoritmo
- Solución en abstracto: **ALGORITMO**
- **Computabilidad:** ¿es resoluble mediante modelos de cómputo o no?
- **Complejidad:** ¿es fácil de resolver o no?
- **Exactitud:** ¿se necesita la mejor solución o es bastante con una suficientemente buena?

1. Algoritmos para resolución de problemas de búsqueda

conceptos básicos

	n=5	n=10	n=100	n=1000
n	5	10	100	1000
n^2	25	100	10000	10^6
2^n	32	1024	1.2×10^{30}	1.1×10^{301}
$n!$	120	3.6×10^6	9.3×10^{157}	4.0×10^{2567}

1. Algoritmos para resolución de problemas de búsqueda

conceptos básicos

	n=5	n=10	n=100	n=1000
n	5	10	100	1000
n^2	25	100	10000	10^6
2^n	32	1024	1.2×10^{30}	1.1×10^{301}
$n!$	120	3.6×10^6	9.3×10^{157}	4.0×10^{2567}

¡Necesitamos buenos algoritmos y eficientes!

Algoritmos que proporcionen una buena solución en un tiempo razonable

1. Algoritmos para resolución de problemas de búsqueda

complejidad de los algoritmos

- **Complejidad** en tiempo más rápido para resolver un problema
- Los problemas se pueden clasificar en:
 - **P**, la clase de problemas con complejidad polinomial
 - **NP**, la clase de problemas que se pueden resolver con un algoritmo (No determinístico Polinomial)
 - **NP-completos**, los problemas más duros en la clase NP
 - **NP-duros**, problemas “al menos tan duros” como los pertenecientes a la clase NP-completos

1. Algoritmos para resolución de problemas de búsqueda

complejidad de los algoritmos



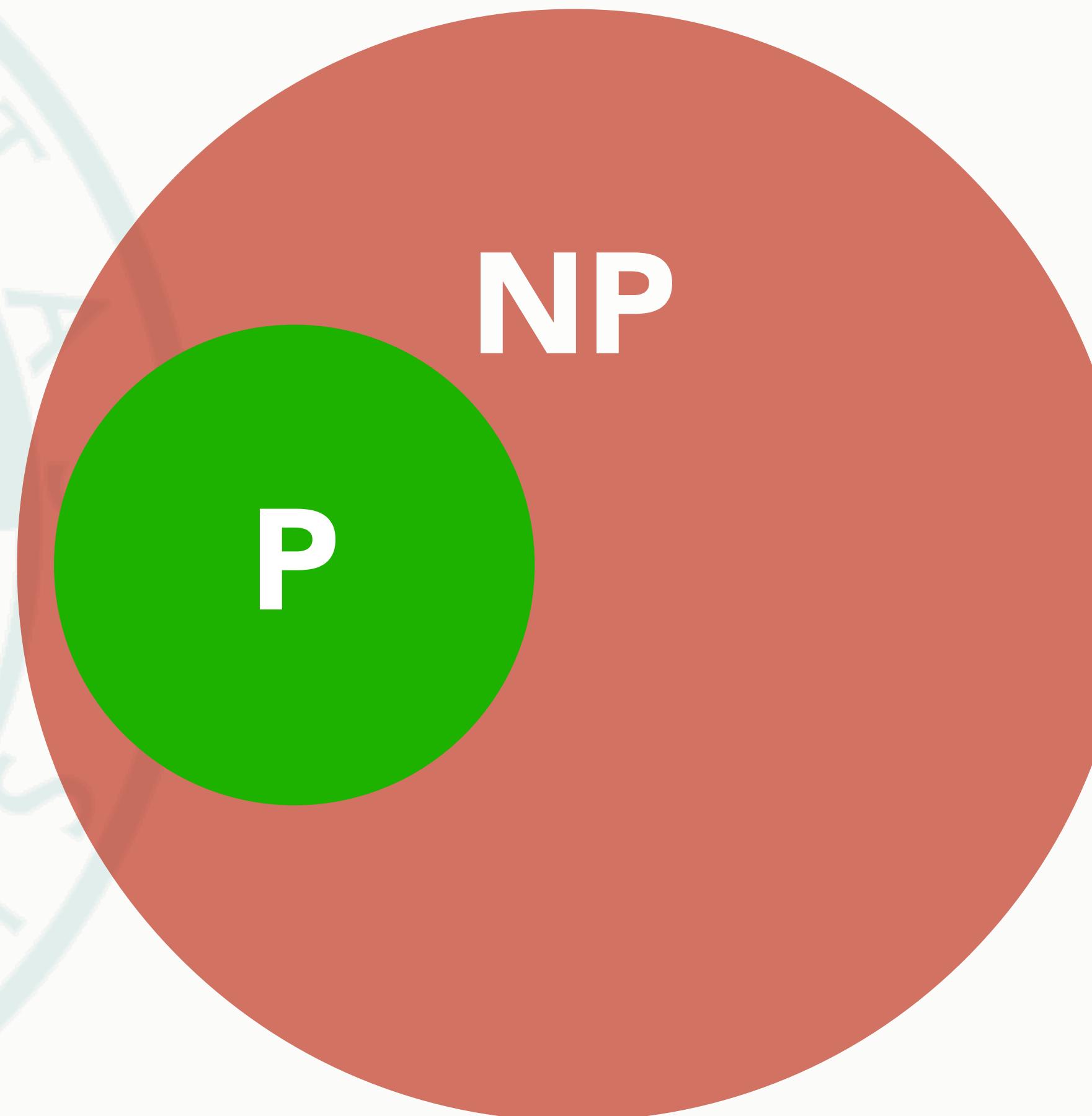
1. Algoritmos para resolución de problemas de búsqueda

complejidad de los algoritmos



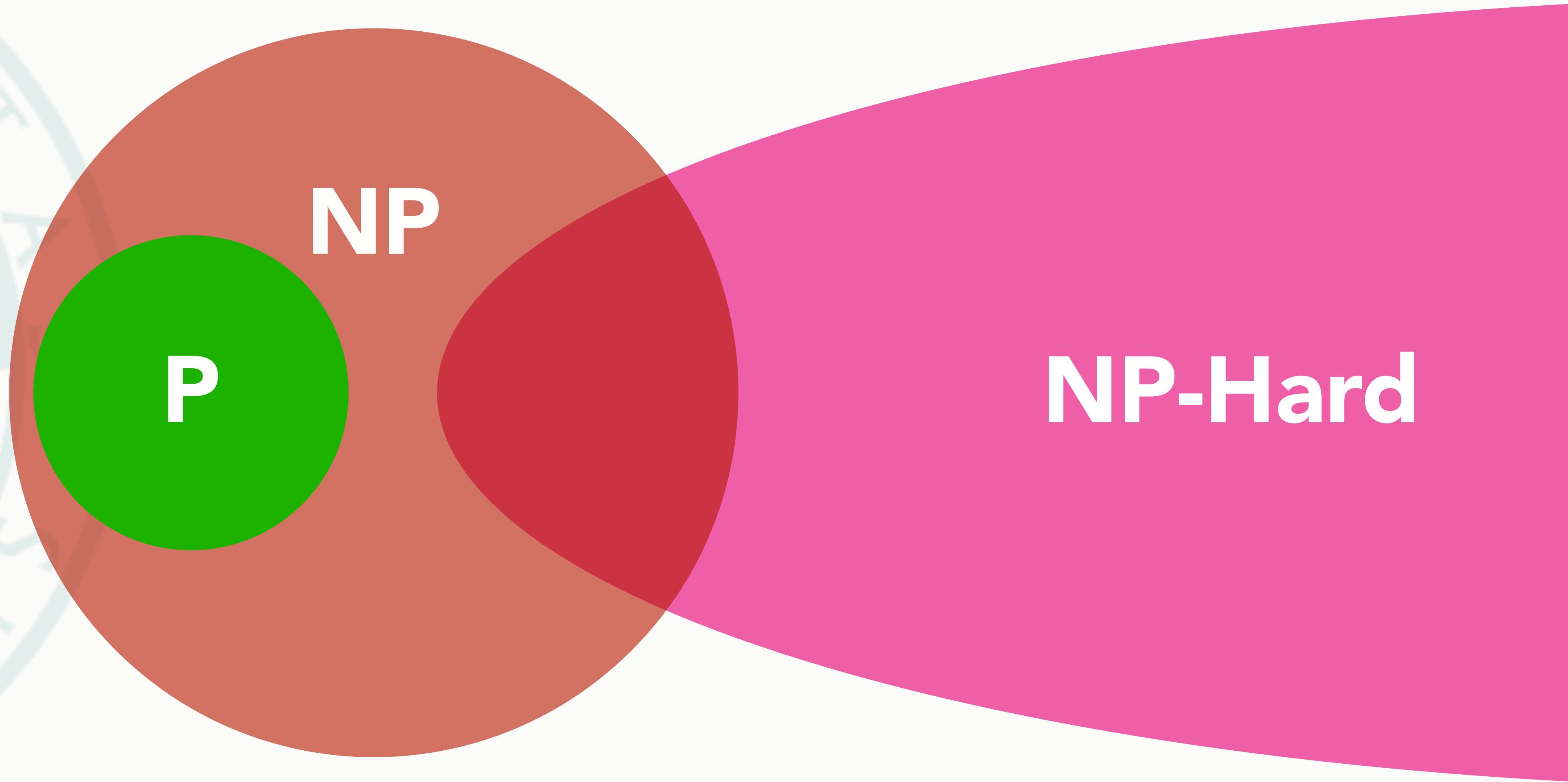
1. Algoritmos para resolución de problemas de búsqueda

complejidad de los algoritmos



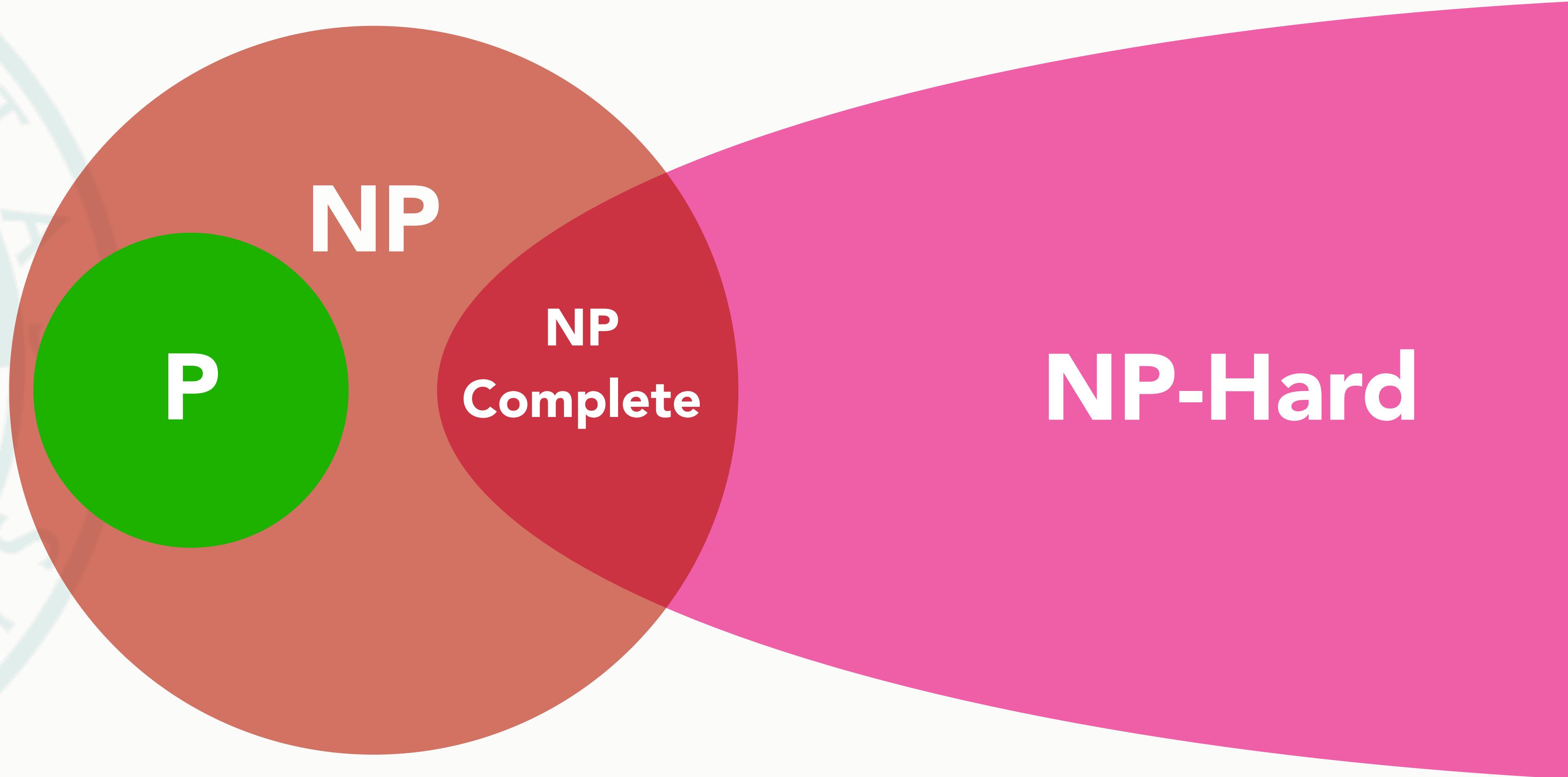
1. Algoritmos para resolución de problemas de búsqueda

complejidad de los algoritmos



1. Algoritmos para resolución de problemas de búsqueda

complejidad de los algoritmos



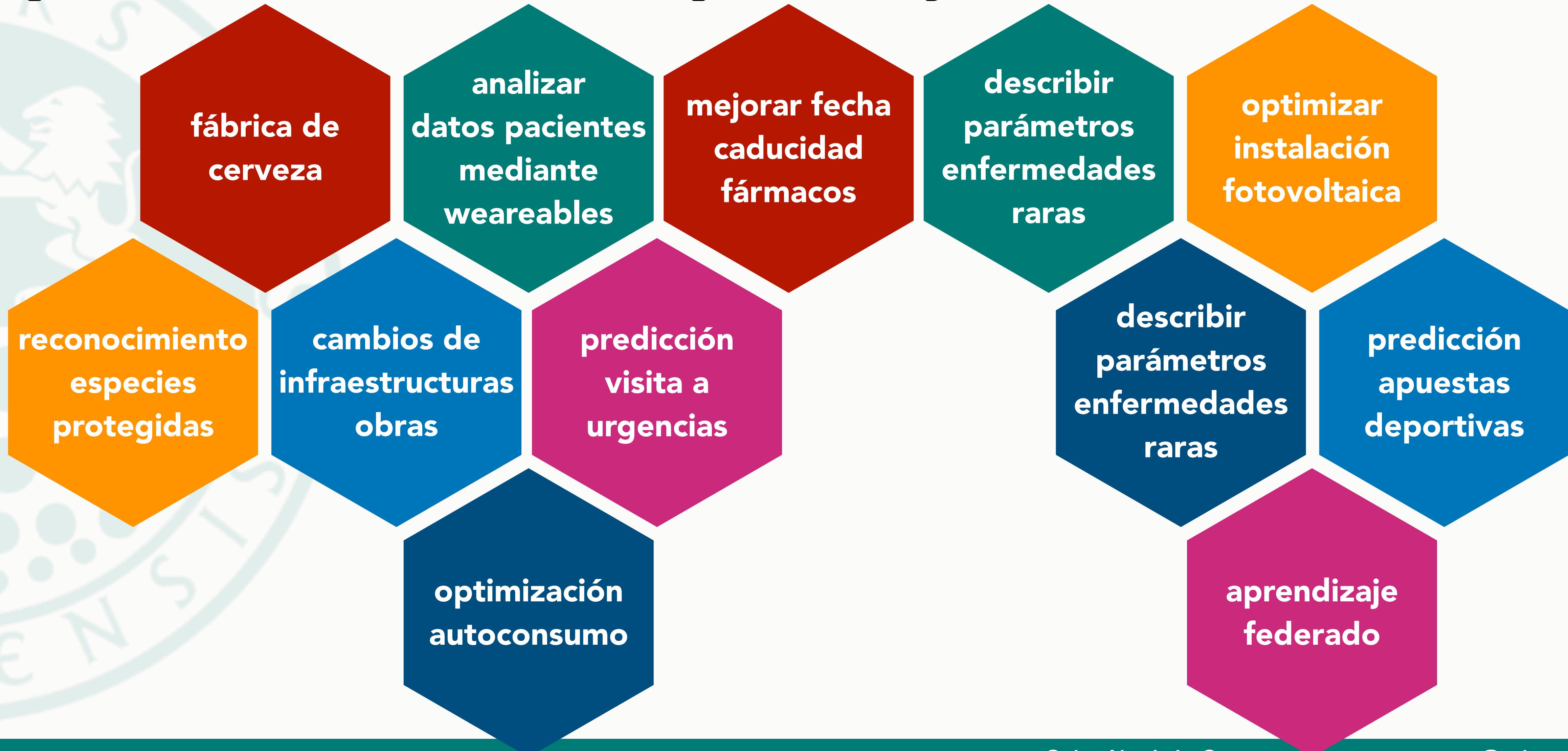
1. Algoritmos para resolución de problemas de búsqueda

problemas reales

- el camino más corto entre varios puntos,
- un plan de mínimo coste para repartir mercancías a clientes,
- una asignación óptima de trabajadores a tareas a realizar,
- una secuencia óptima de proceso de trabajos en una cadena de producción,
- una distribución de tripulaciones de aviones con mínimo coste,
- el mejor enrutamiento de un paquete de datos en Internet,
- ...

1. Algoritmos para resolución de problemas de búsqueda

problemas reales en los que trabajamos en SIMIDAT



1. Algoritmos para resolución de problemas de búsqueda

problemas reales

- Problemas de una gran complejidad computacional (NP-duros)
- Algoritmos exactos son ineficientes o imposibles de aplicar
- Se encuentran en muchas áreas de aplicación
- En la práctica se resuelven mediante algoritmos aproximados que proporcionan buenas soluciones. No es necesariamente la óptima al problema en un tiempo razonable

1. Algoritmos para resolución de problemas de búsqueda

¿con qué problemas vamos a trabajar?

- **Problemas combinatorios**

- Buscar una agrupación, ordenación o asignación de un conjunto discreto y finito que verifiquen restricciones
- Frecuentes en informática y áreas de aplicación
- NP-duros
- Búsqueda en un espacio exponencialmente grande

1. Algoritmos para resolución de problemas de búsqueda

¿con qué problemas vamos a trabajar?

- **Problemas combinatorios**

Solución candidata: combinación de componentes de solución que no necesariamente cumple las condiciones

Solución: solución candidata que verifica todas las restricciones

- Búsqueda en un espacio exponencialmente grande

1. Algoritmos para resolución de problemas de búsqueda

¿con qué problemas vamos a trabajar?

El problema de la máxima diversidad

$$\binom{n}{m} = 10^{69364}$$

Imaginemos un problema:
500 elementos
50 más diversos

1. Algoritmos para resolución de problemas de búsqueda

¿con qué problemas vamos a trabajar?

• Problemas de optimización

- Una función $f(x)$ nos mide la calidad de la solución
- El objetivo es encontrar la solución de calidad óptima
 - Minimización donde la solución sea el menor valor para $f(x)$
 - Maximización donde la solución sea el mayor valor para $f(x)$
- La solución candidata es buena si cumple las restricciones

1. Algoritmos para resolución de problemas de búsqueda

**Crear
soluciones
candidatas**

- Función objetivo:
 - Max(Min) con variables
 - Restricciones (=) ó ($<$, $>$, ...)
- Forma de representación para determinar el espacio de búsqueda

**Evaluar soluciones
candidatas es
computacionalmente
menos costoso que
encontrar soluciones
óptimas**

2

Algoritmos aproximados



2. Algoritmos aproximados

Los **algoritmos aproximados** aportan soluciones cercanas a la óptima en problemas complejos (NP-duros) en un tiempo razonable

¿Qué factores los hacen interesantes?

- Cuando no hay un método exacto de resolución, o éste requiere mucho tiempo de cálculo y memoria
- Cuando no se necesita la solución óptima sino que basta con una buena calidad en un tiempo aceptable

2. Algoritmos aproximados

Búsqueda es un término utilizado para construir/mejorar soluciones y obtener el óptimo o soluciones casi-óptimas

Los elementos clave son

2. Algoritmos aproximados

Búsqueda es un término utilizado para construir/mejorar soluciones y obtener el óptimo o soluciones casi-óptimas

Los elementos clave son

Solución

Representación de la solución del problema

2. Algoritmos aproximados

Búsqueda es un término utilizado para construir/mejorar soluciones y obtener el óptimo o soluciones casi-óptimas

Los elementos clave son

Solución

Representación de la solución del problema

Entorno

Soluciones cercanas

2. Algoritmos aproximados

Búsqueda es un término utilizado para construir/mejorar soluciones y obtener el óptimo o soluciones casi-óptimas

Los elementos clave son

Solución

Representación de la solución del problema

Entorno

Soluciones cercanas

Movimiento

Transformación de la solución actual en otra

2. Algoritmos aproximados

Búsqueda es un término utilizado para construir/mejorar soluciones y obtener el óptimo o soluciones casi-óptimas

Los elementos clave son

Solución

Representación de la solución del problema

Entorno

Soluciones cercanas

Movimiento

Transformación de la solución actual en otra

Evaluación

Factibilidad de la solución y la función objetivo

2. Algoritmos aproximados

la búsqueda por entornos



2. Algoritmos aproximados

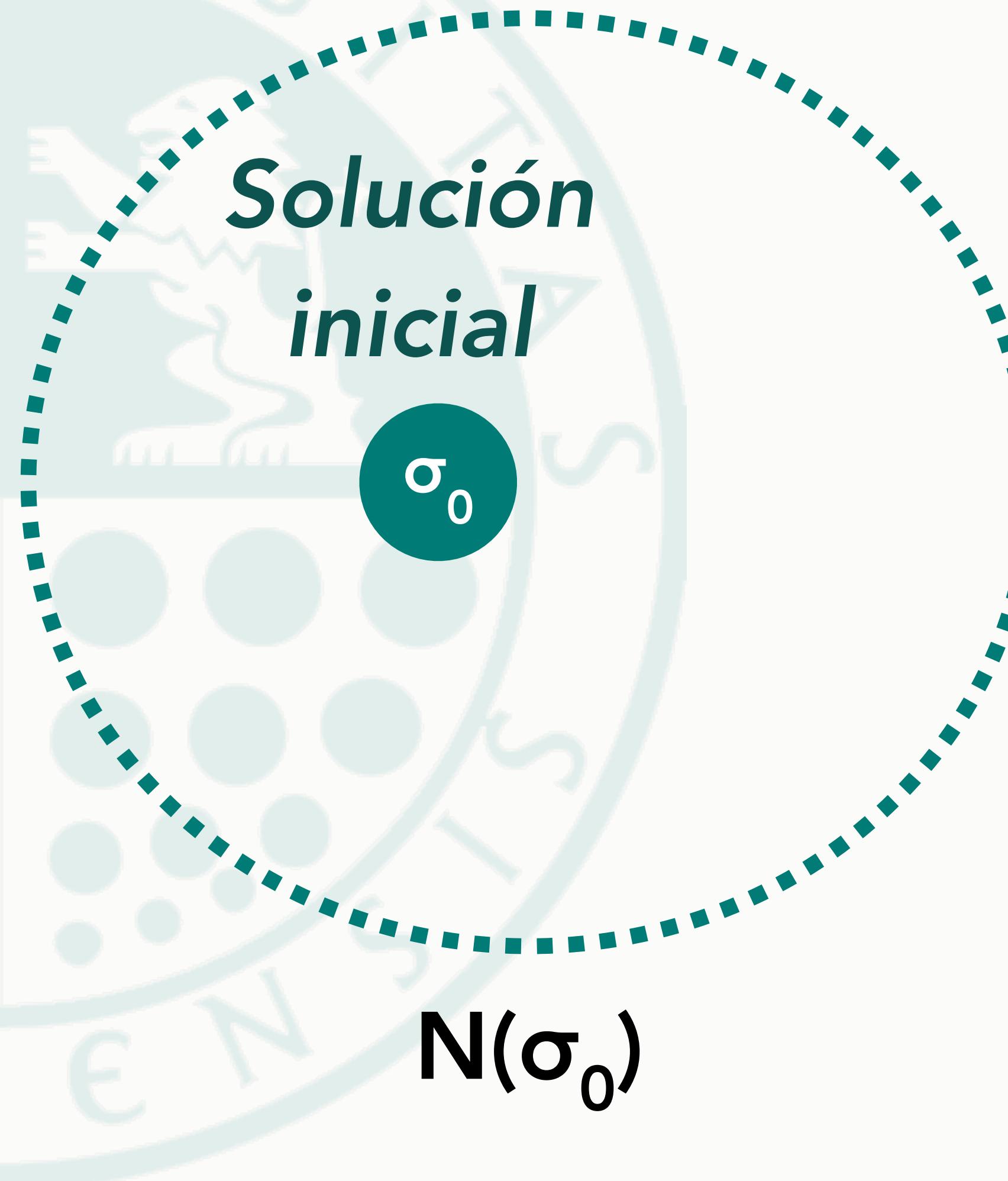
la búsqueda por entornos

**Solución
inicial**

$$\sigma_0$$

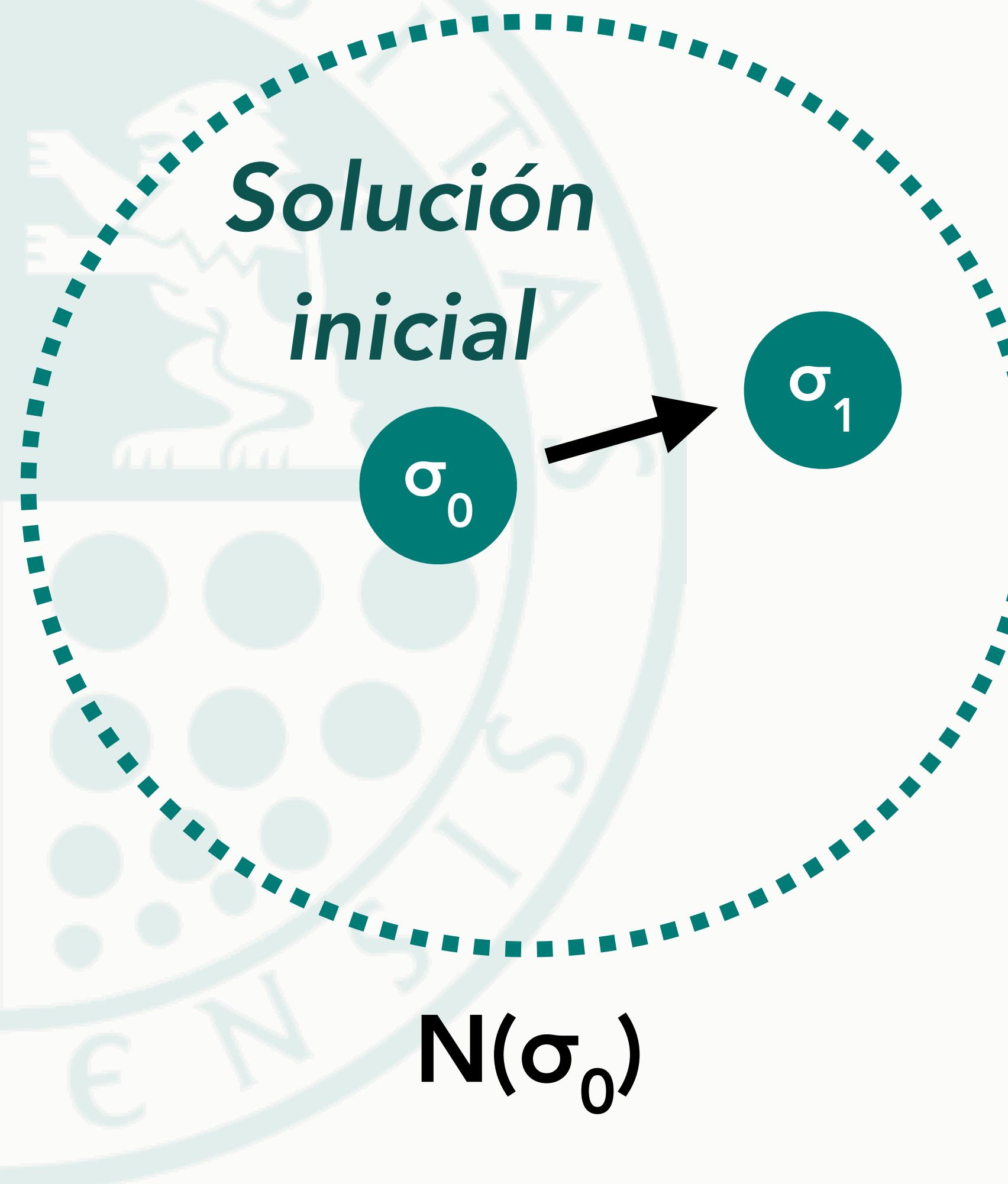
2. Algoritmos aproximados

la búsqueda por entornos



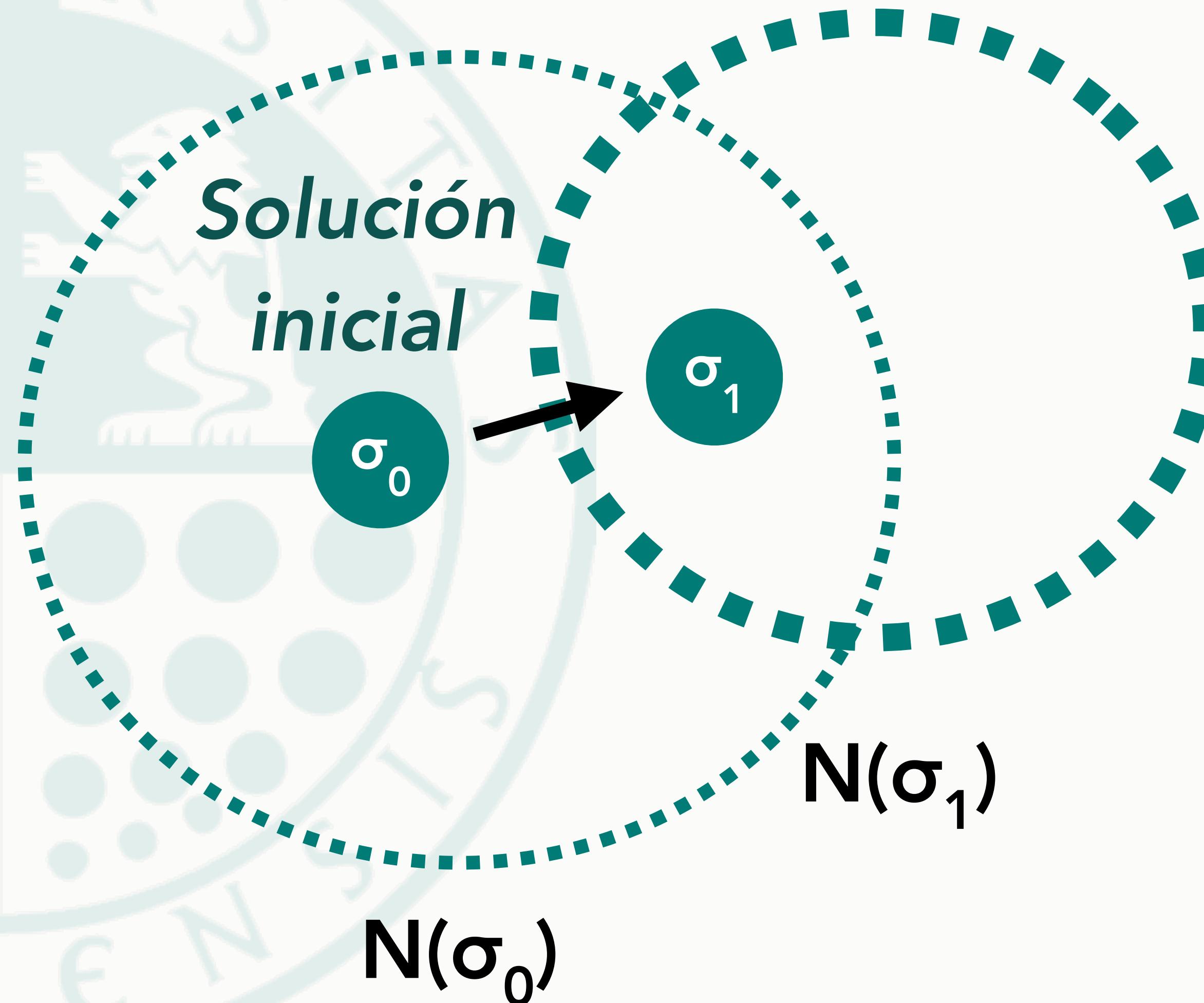
2. Algoritmos aproximados

la búsqueda por entornos



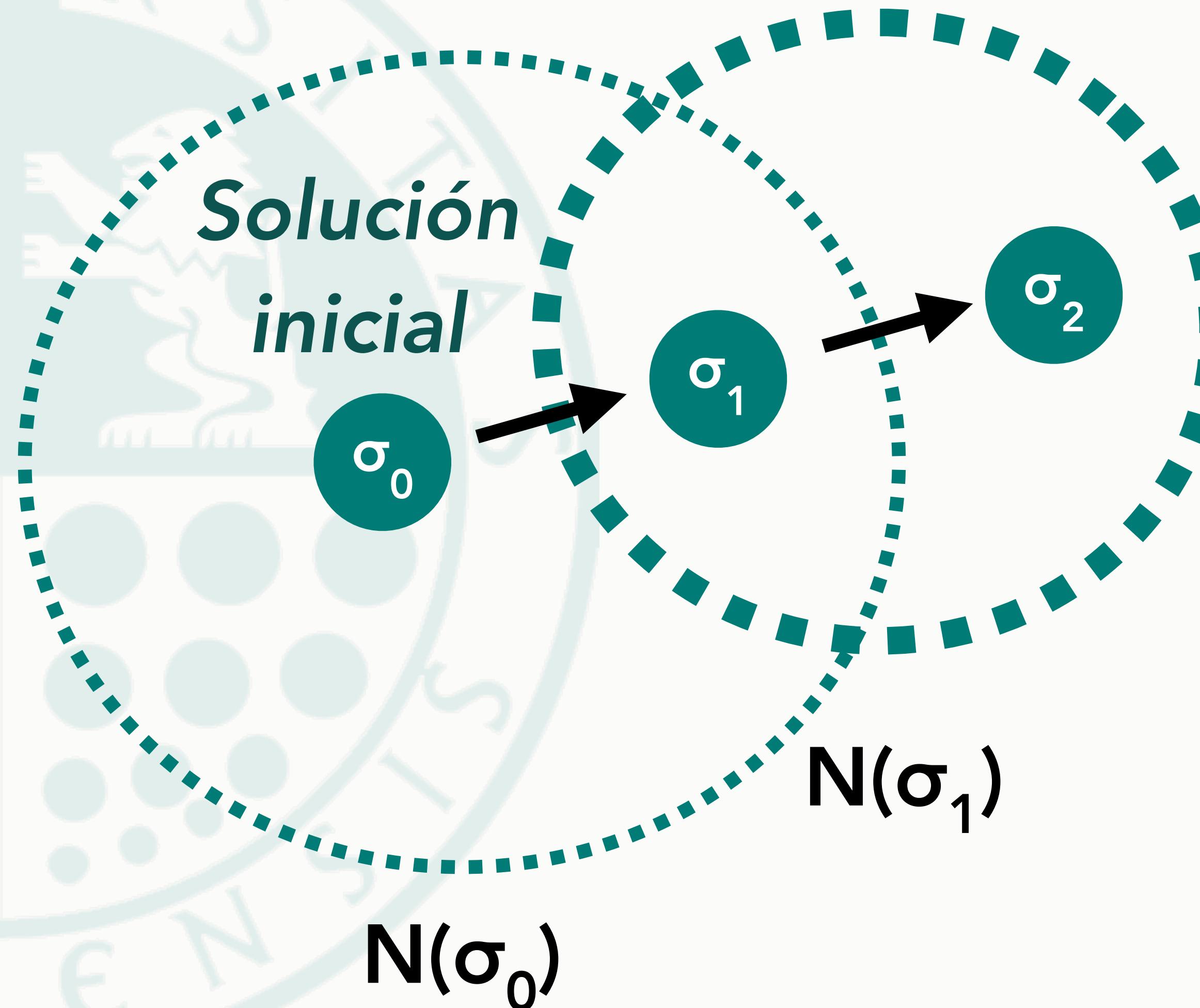
2. Algoritmos aproximados

la búsqueda por entornos



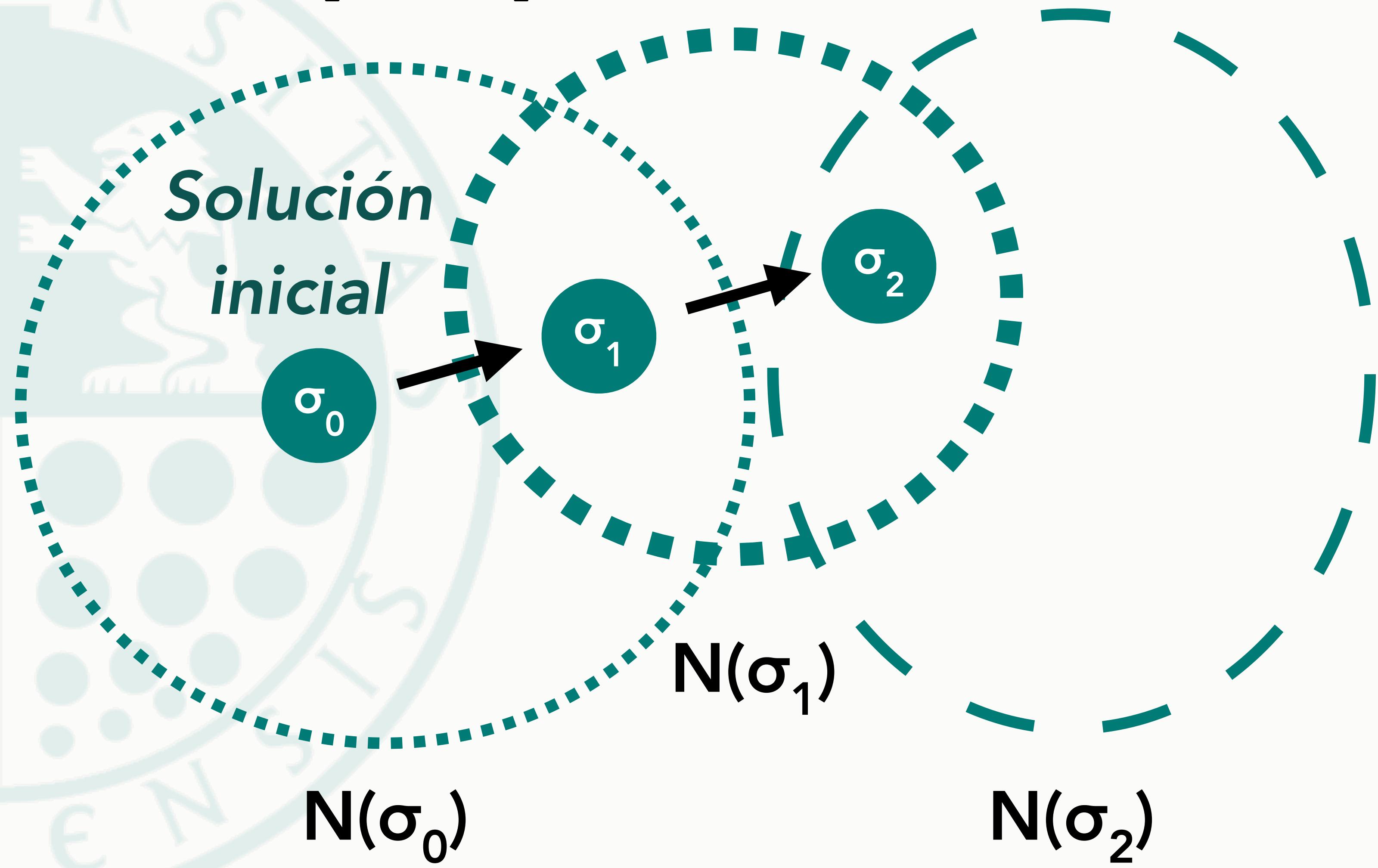
2. Algoritmos aproximados

la búsqueda por entornos



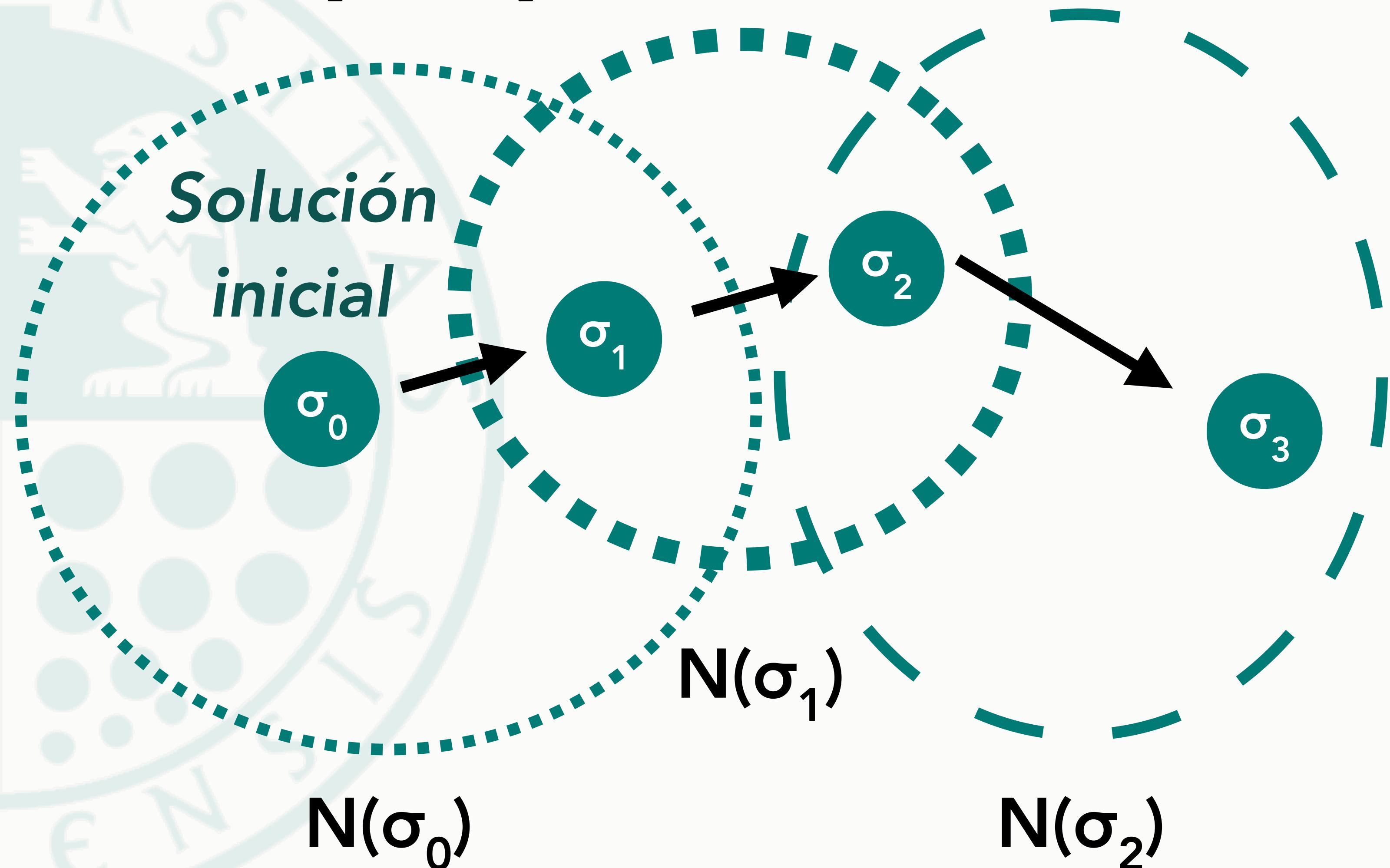
2. Algoritmos aproximados

la búsqueda por entornos



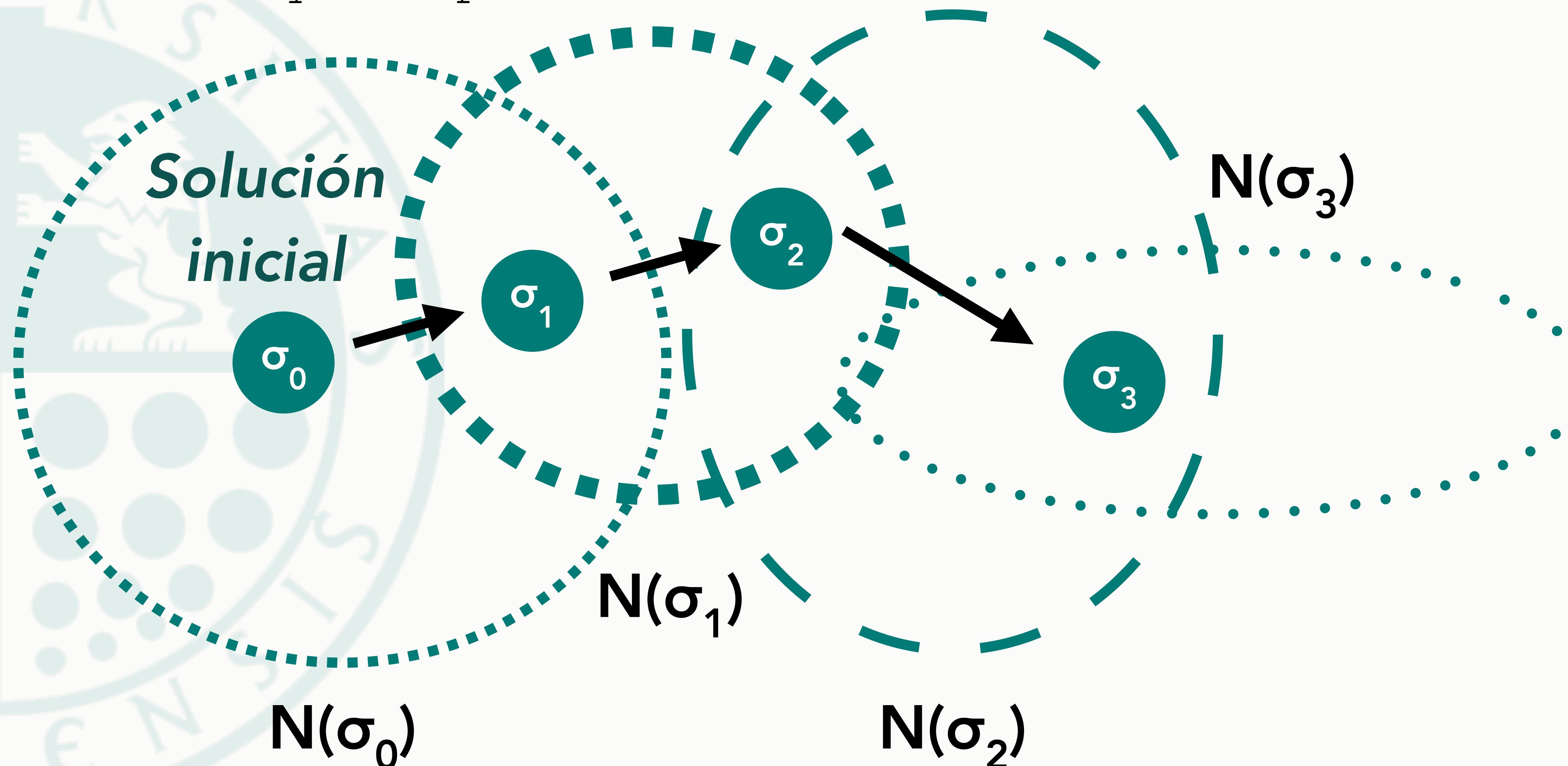
2. Algoritmos aproximados

la búsqueda por entornos



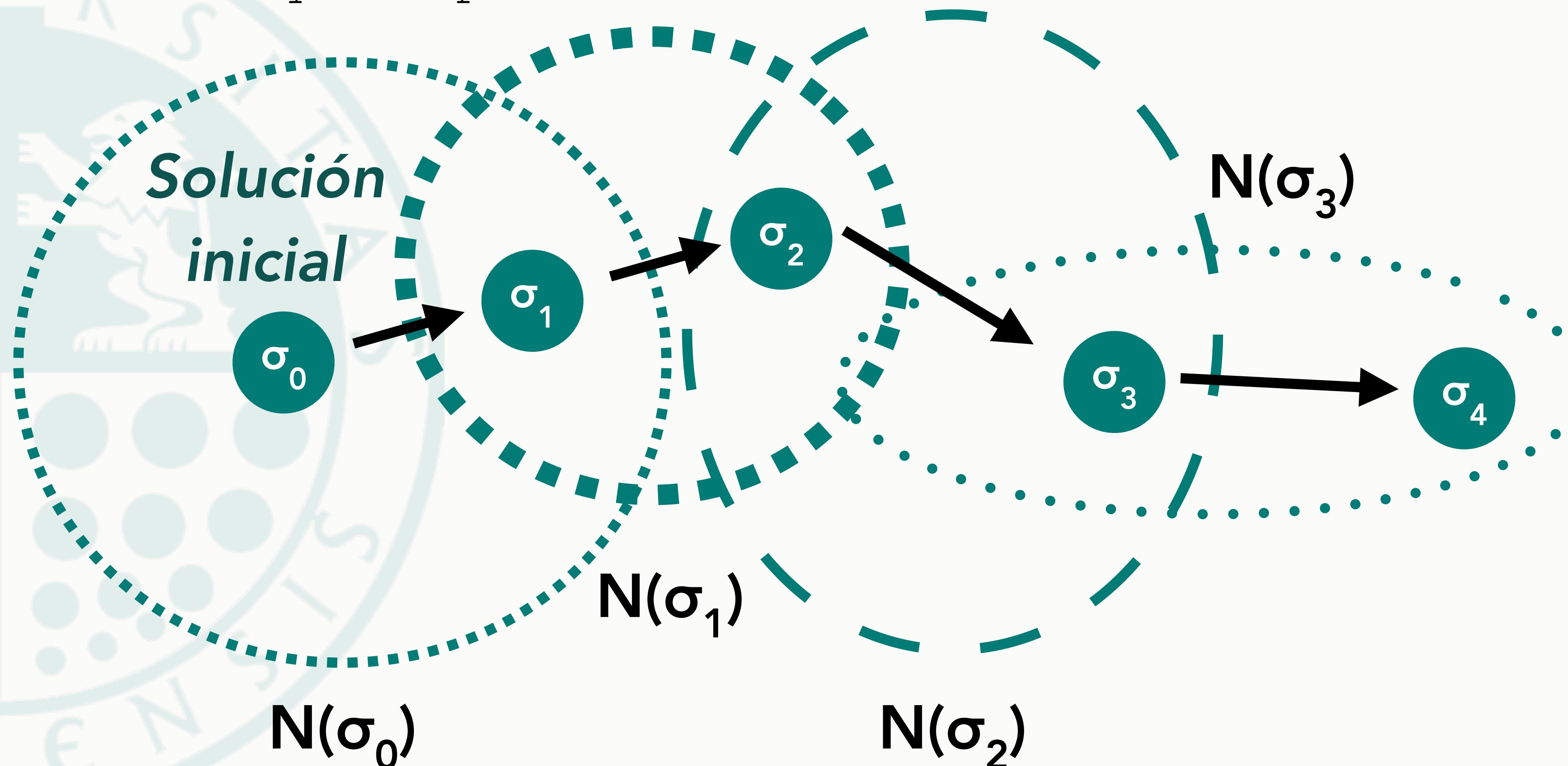
2. Algoritmos aproximados

la búsqueda por entornos



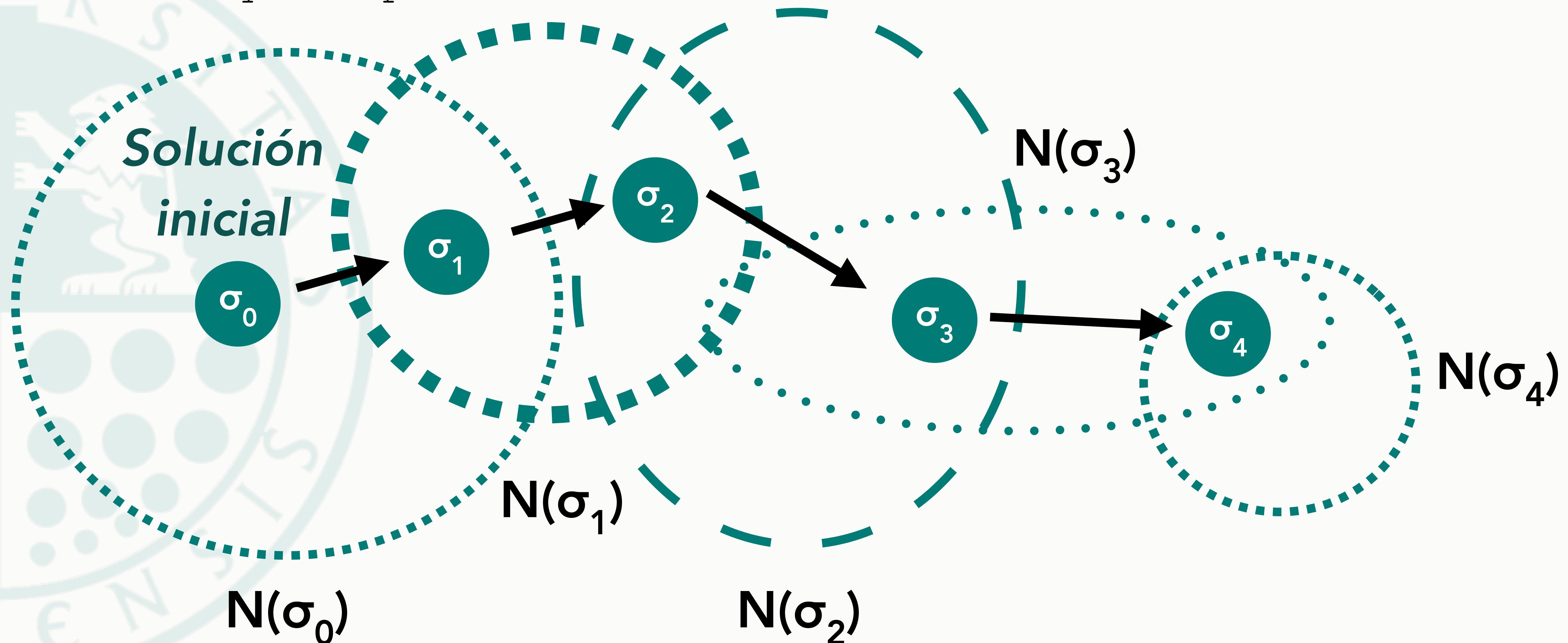
2. Algoritmos aproximados

la búsqueda por entornos



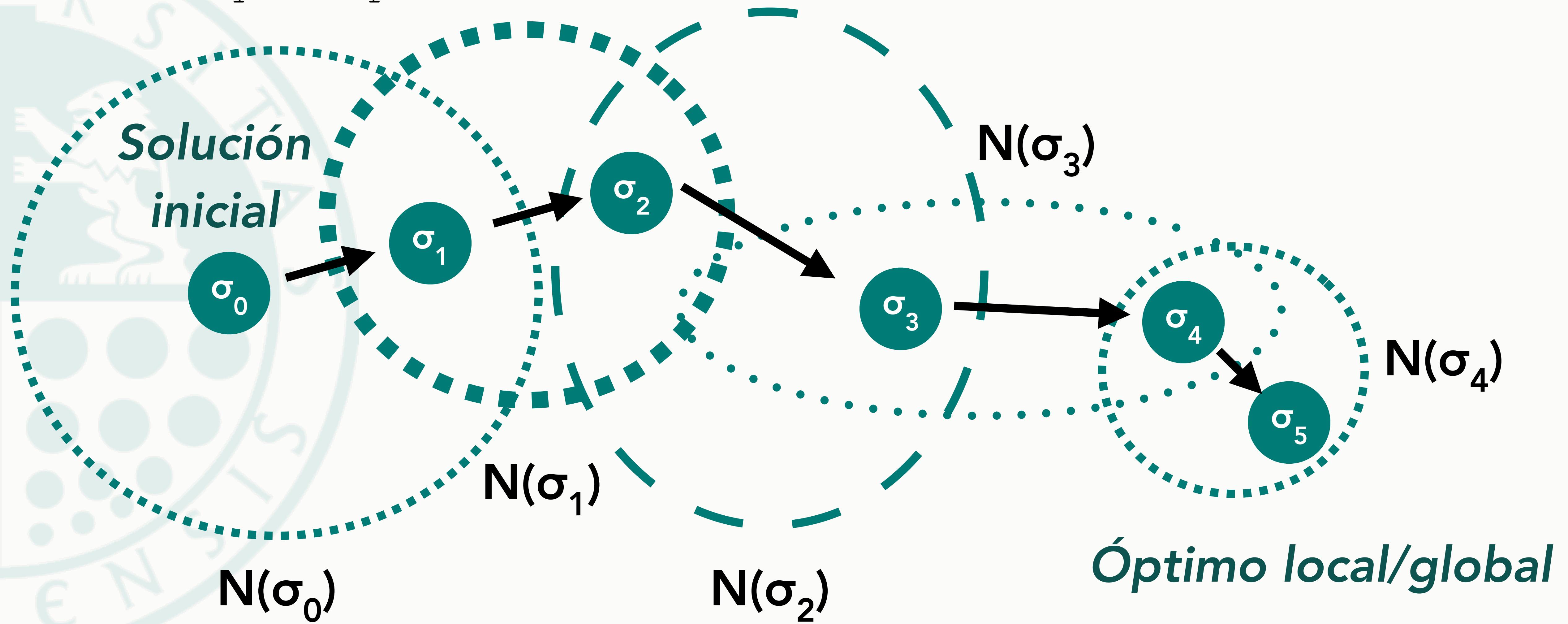
2. Algoritmos aproximados

la búsqueda por entornos

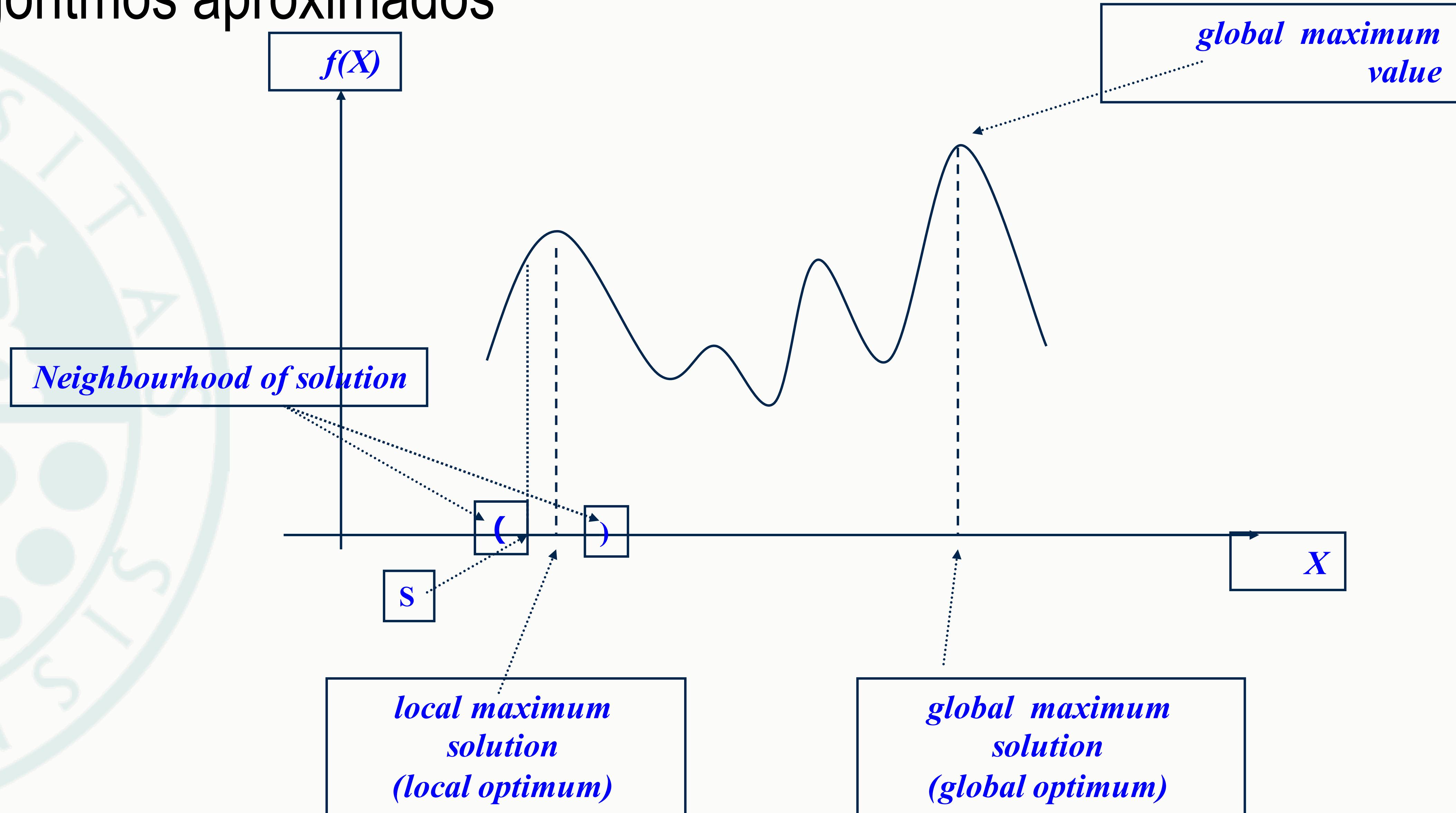


2. Algoritmos aproximados

la búsqueda por entornos



2. Algoritmos aproximados



3

Definición Clasificación Conceptos básicos

3. Definición

Son una familia de algoritmos aproximados de propósito general

No existe una única definición para este concepto

Suelen ser procedimientos iterativos que guían una heurística subordinada de búsqueda, combinando de forma inteligente distintos conceptos para explorar y explotar adecuadamente el espacio de búsqueda

3. Definición

Son una familia de algoritmos aproximados de propósito general

Ventajas

- Algoritmos de propósito general
- Gran éxito en la práctica
- Fácilmente implementables
- Fácilmente paralelizables

Inconvenientes

- Son algoritmos aproximados, no exactos
- Son no determinísticos (probabilísticos)
- No siempre existe una base teórica establecida

3. Definición

“A metaheuristic is formally defined as an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space, learning strategies are used to structure information in order to find efficiently near-optimal solutions”

“A metaheuristic is an iterative master process that guides and modifies the operations of subordinate heuristics to efficiently produce high-quality solutions. It may manipulate a complete (or incomplete) single solution or a collection of solutions at each iteration. The subordinate heuristics may be high (or low) level procedures, or a simple local search, o just a construction method”

3. Clasificación

Existen tres grandes grupos en función de conceptos:

3. Clasificación

Existen tres grandes grupos en función de conceptos:

**Seguimiento de trayectorias considerando
Trayectorias simples y múltiples**

3. Clasificación

Existen tres grandes grupos en función de conceptos:

**Seguimiento de trayectorias considerando
Trayectorias simples y múltiples**

Uso de poblaciones de soluciones

3. Clasificación

Existen tres grandes grupos en función de conceptos:

**Seguimiento de trayectorias considerando
Trayectorias simples y múltiples**

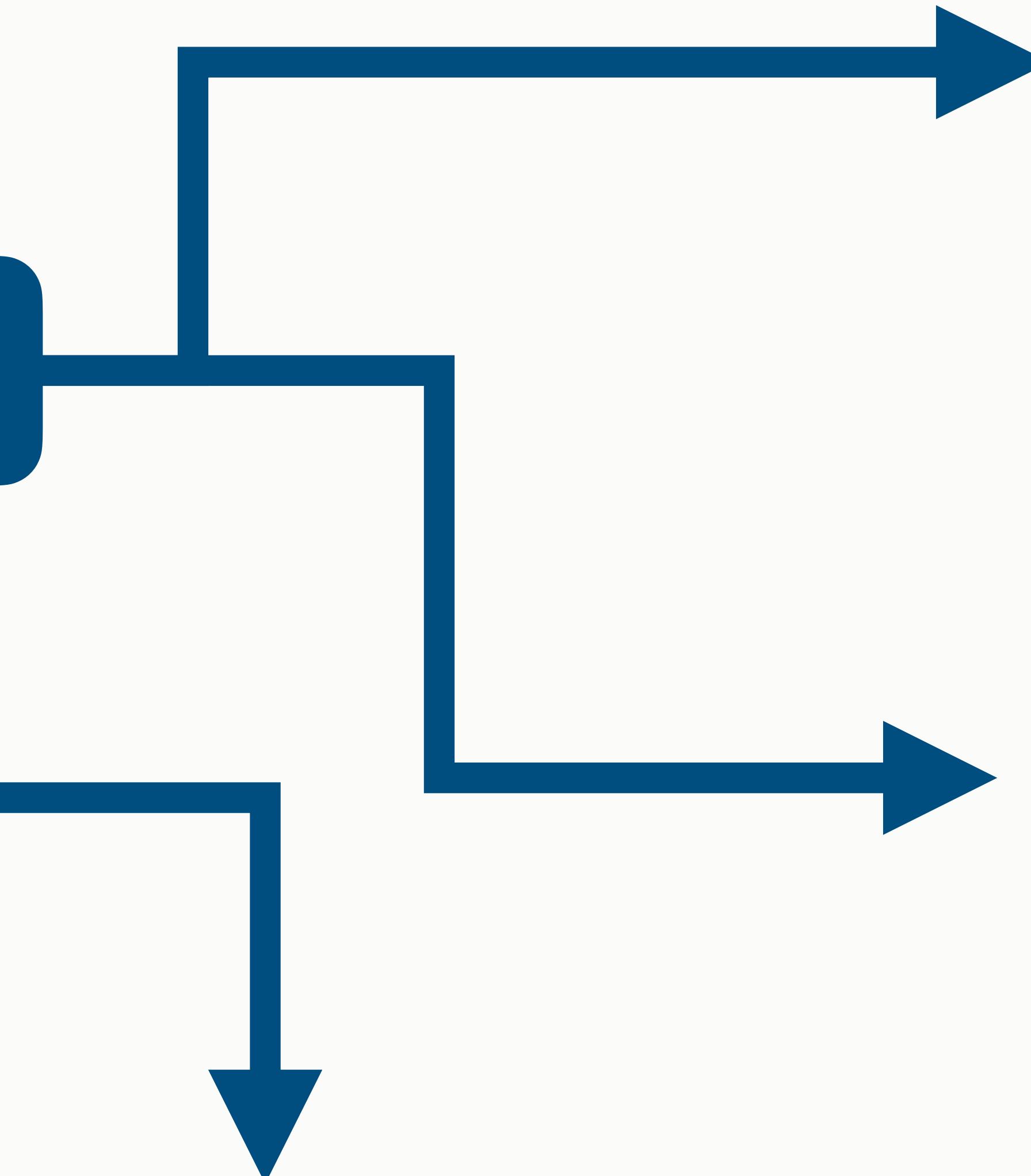
Uso de poblaciones de soluciones

**Fuente de inspiración
Bioinspirada: genéticos, colonias de hormigas, ...**

3. Clasificación

Una posible clasificación:

Basados en métodos constructivos



3. Clasificación

Una posible clasificación:

Basados en métodos constructivos

Basadas en trayectorias

La heurística subordinada es un algoritmo de búsqueda local que sigue una trayectoria en el espacio de búsqueda

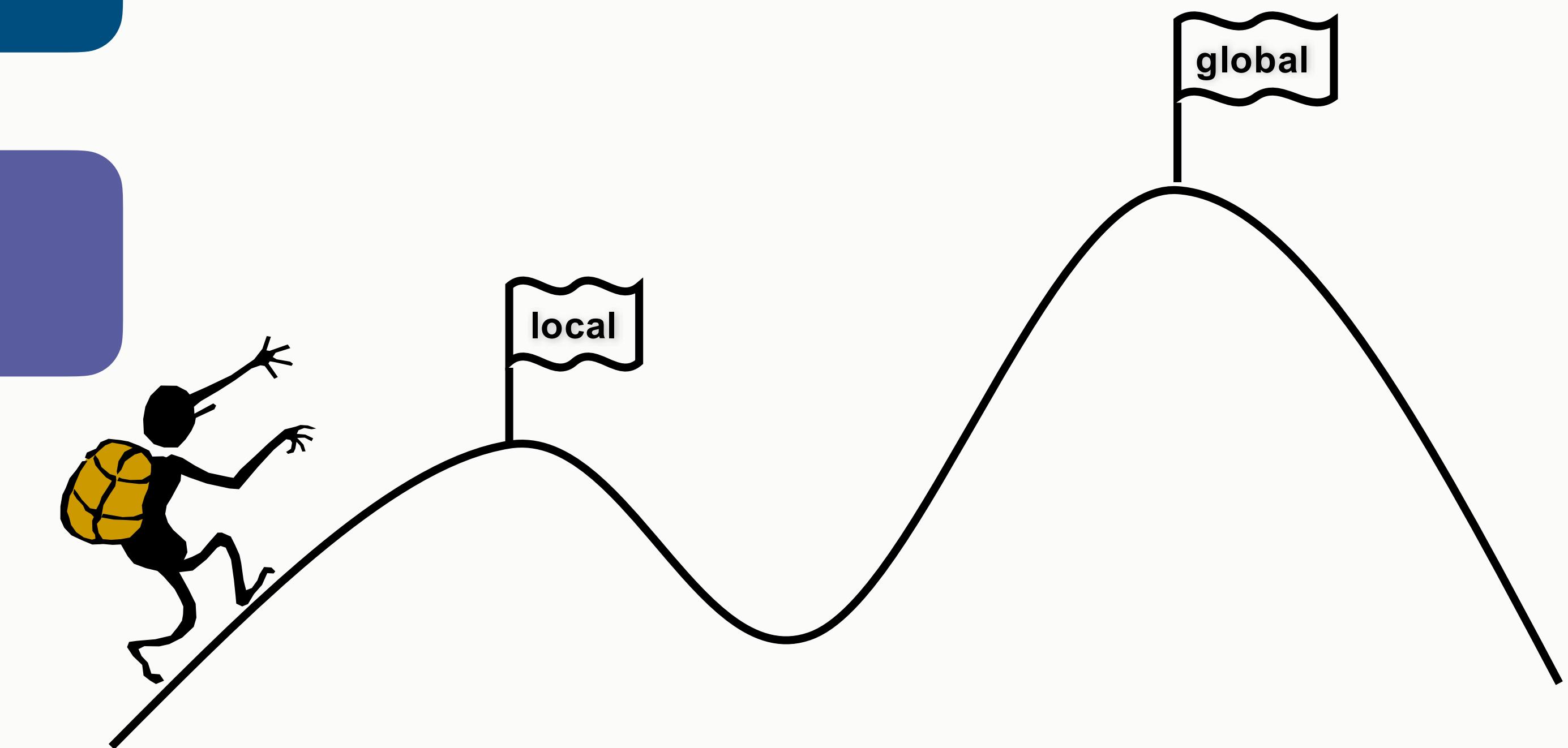
- Búsqueda local
- Enfriamiento simulado
- Búsqueda tabú

3. Clasificación

Una posible clasificación:

Basados en métodos constructivos

Basadas en trayectorias



3. Clasificación

Una posible clasificación:

Basados en métodos constructivos

Basadas en trayectorias

Basadas en poblaciones

El proceso considera múltiples puntos de búsqueda en el espacio

- Genéticos
- Meméticos
- Evolución diferencial

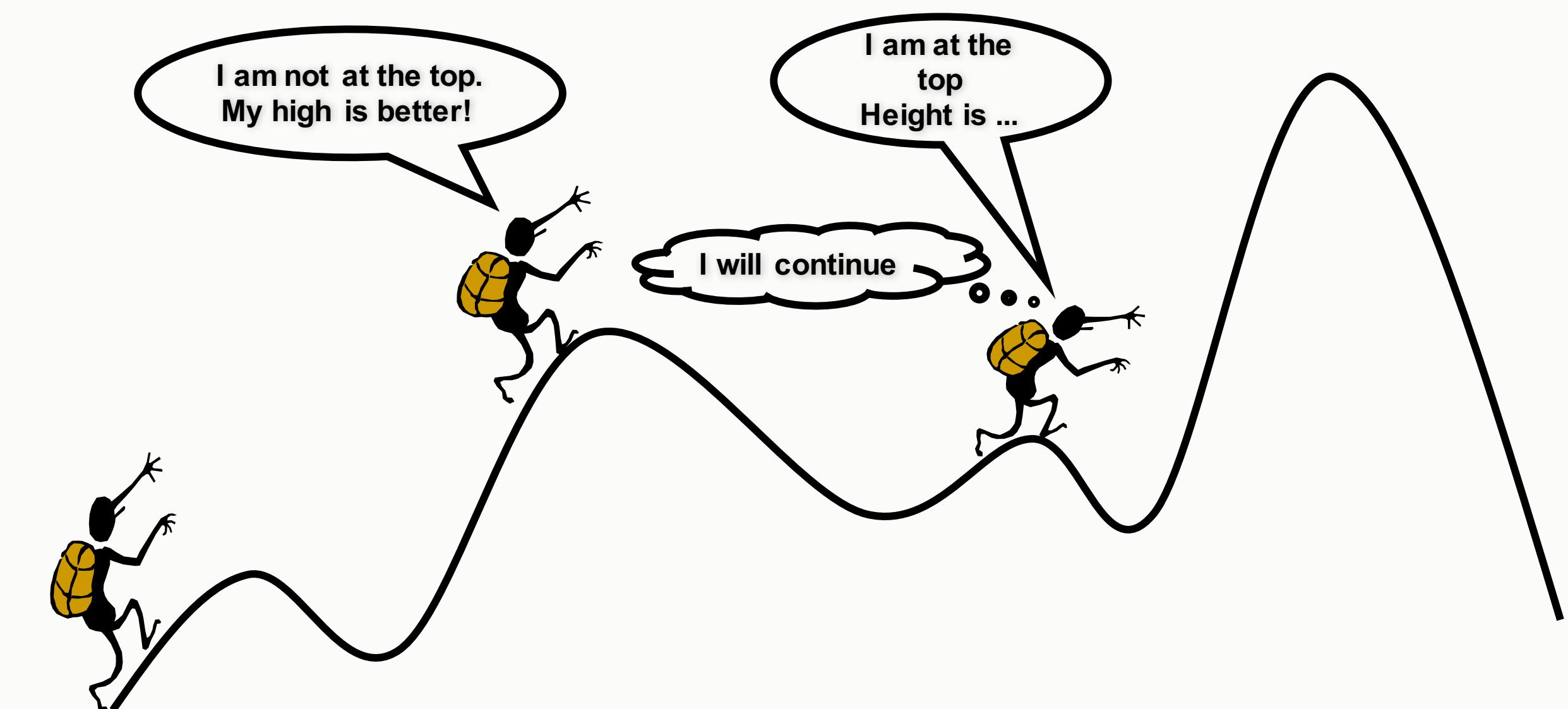
3. Clasificación

Una posible clasificación:

Basados en métodos constructivos

Basadas en trayectorias

Basadas en poblaciones



3. Clasificación

Una posible clasificación:

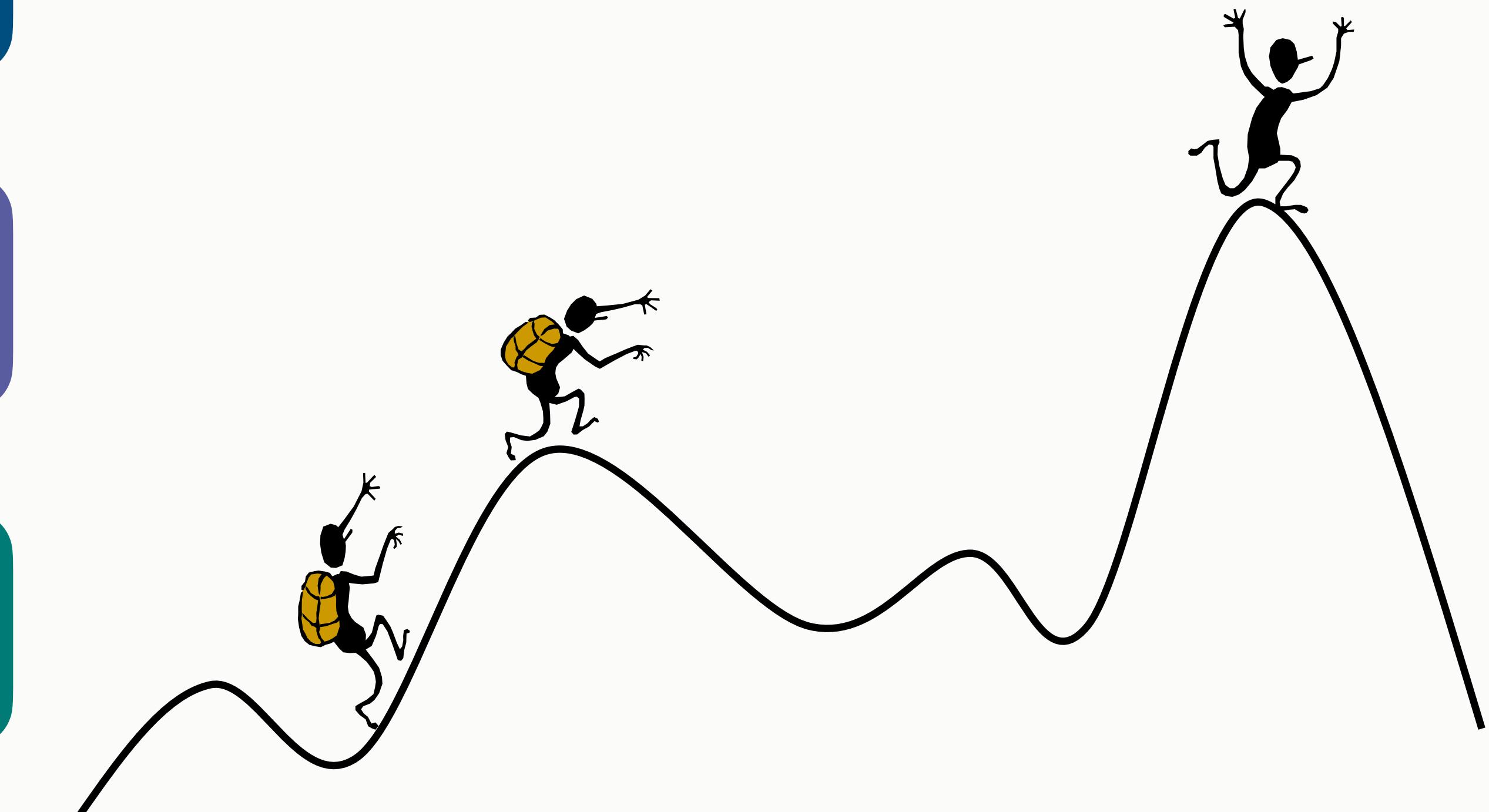
Basados en métodos constructivos

Basadas en trayectorias

Basadas en poblaciones

Un conjunto de soluciones se combinan para obtener nuevas soluciones que heredan las propiedades de las primeras

Secuencia de poblaciones que mejoran la calidad media



3. Clasificación

Evolutivo

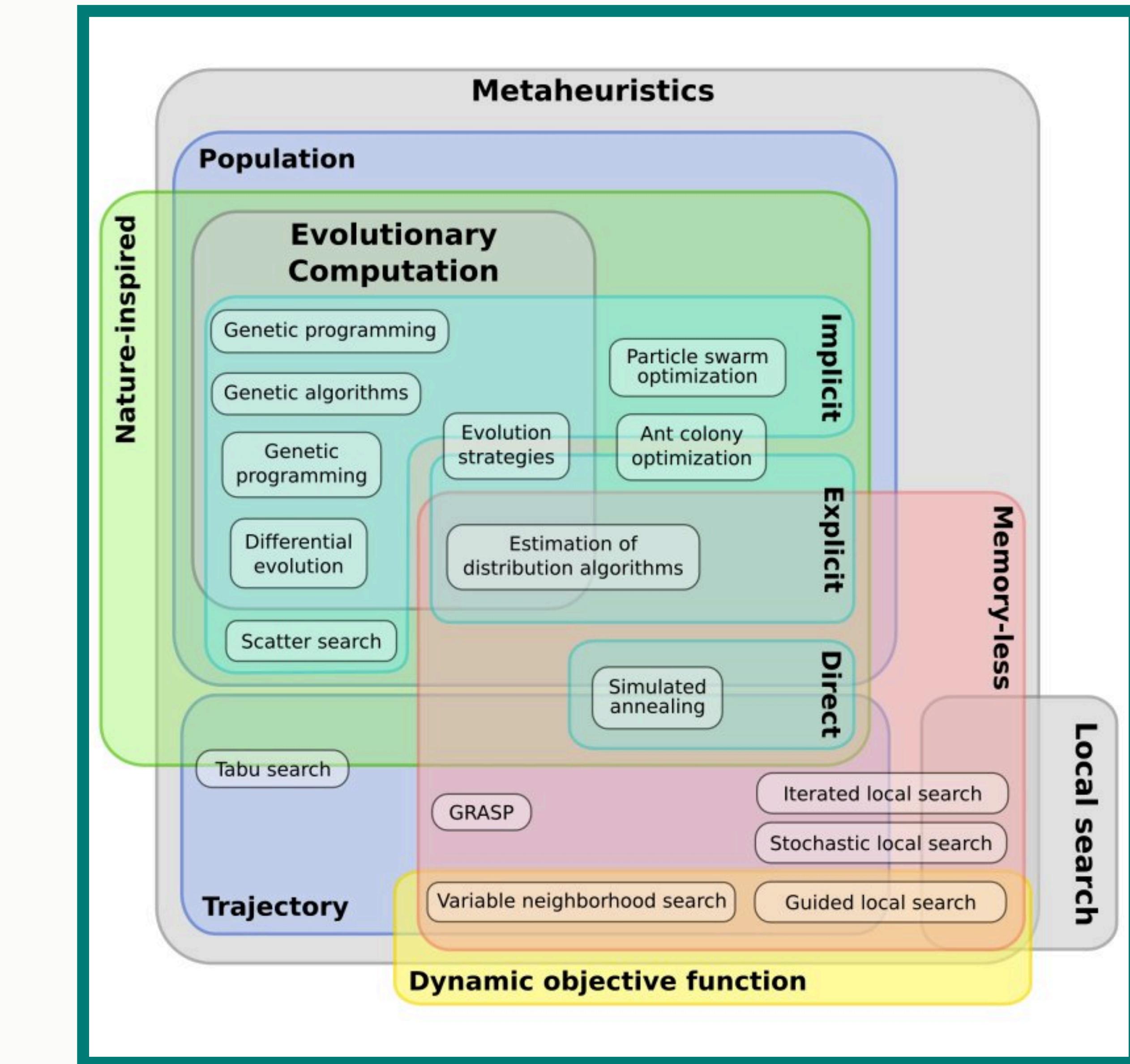
Computación

Inspirada en la naturaleza

Memorias

Búsqueda local

Función dinámica



3. Conceptos básicos

Las metaheurísticas son estrategias de alto nivel para explorar espacios de búsqueda utilizando diferentes métodos

Deben proporcionar un balance dinámico entre

Diversificación

se refiere a la

EXPLORACIÓN

Intensificación

hace referencia a la

EXPLOTACIÓN

3. Conceptos básicos

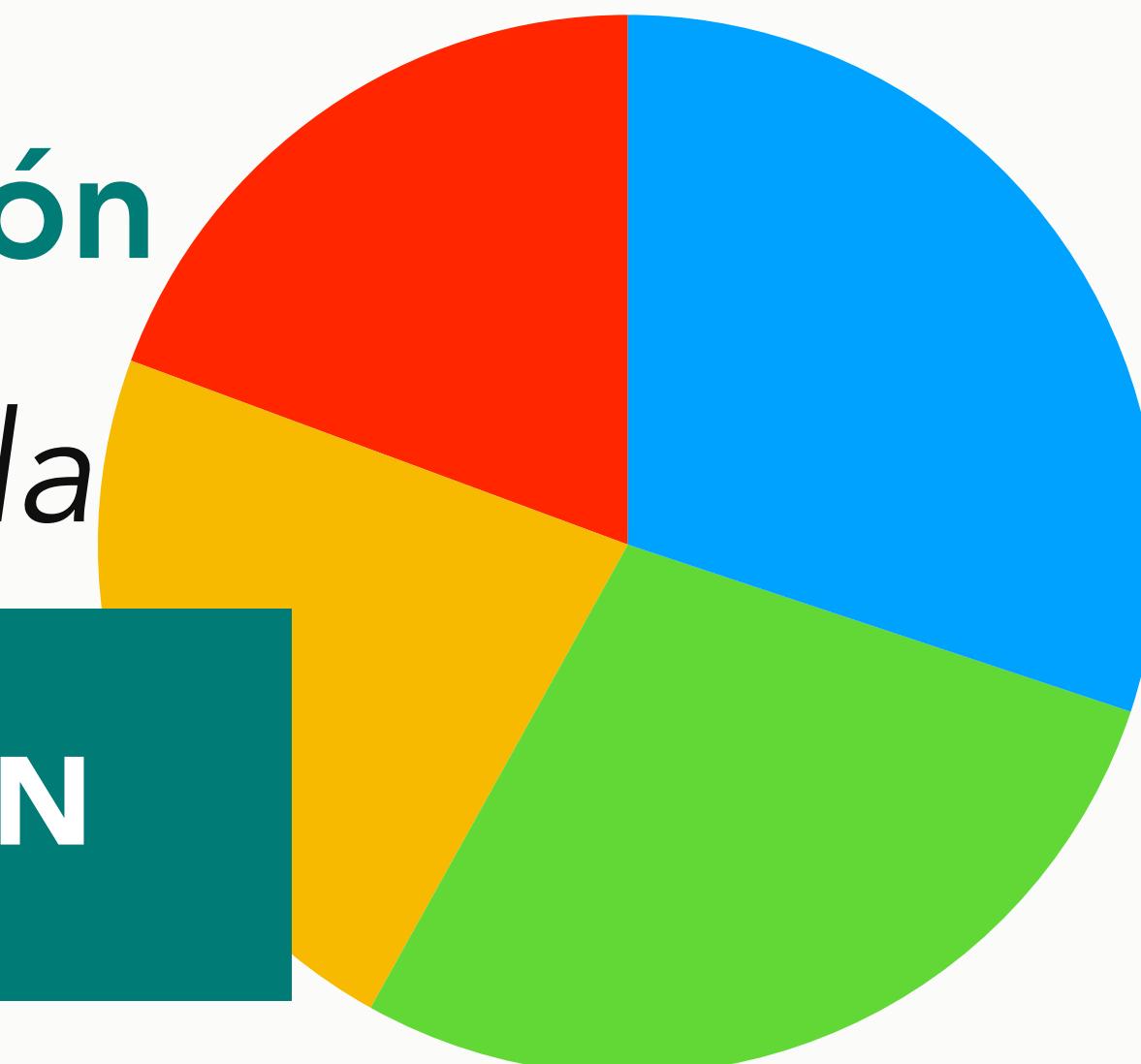
Las metaheurísticas son estrategias de alto nivel para explorar espacios de búsqueda utilizando diferentes métodos

Deben proporcionar un balance dinámico entre

Diversificación

se refiere a la

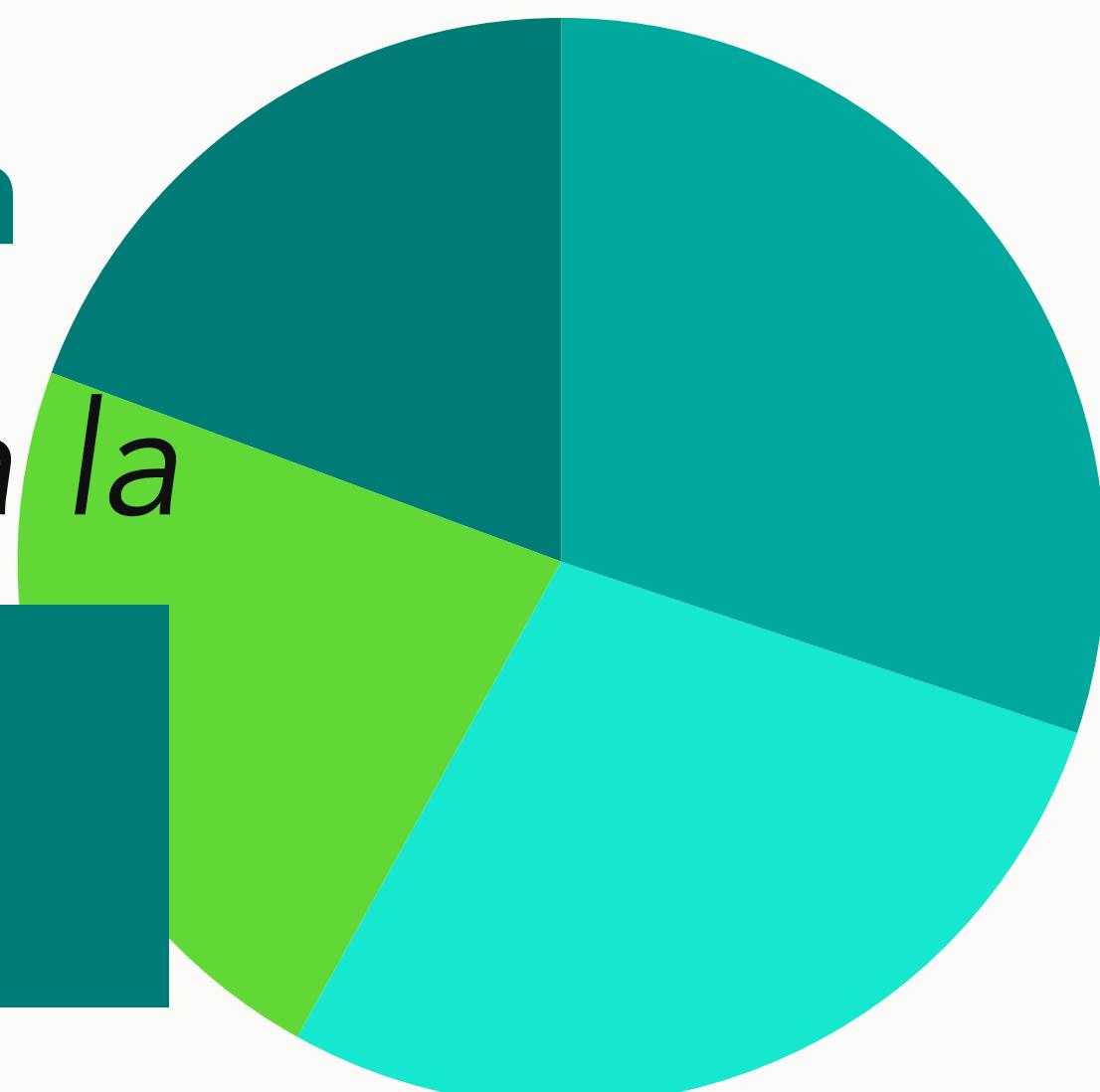
EXPLORACIÓN



Intensificación

hace referencia a la

EXPLOTACIÓN

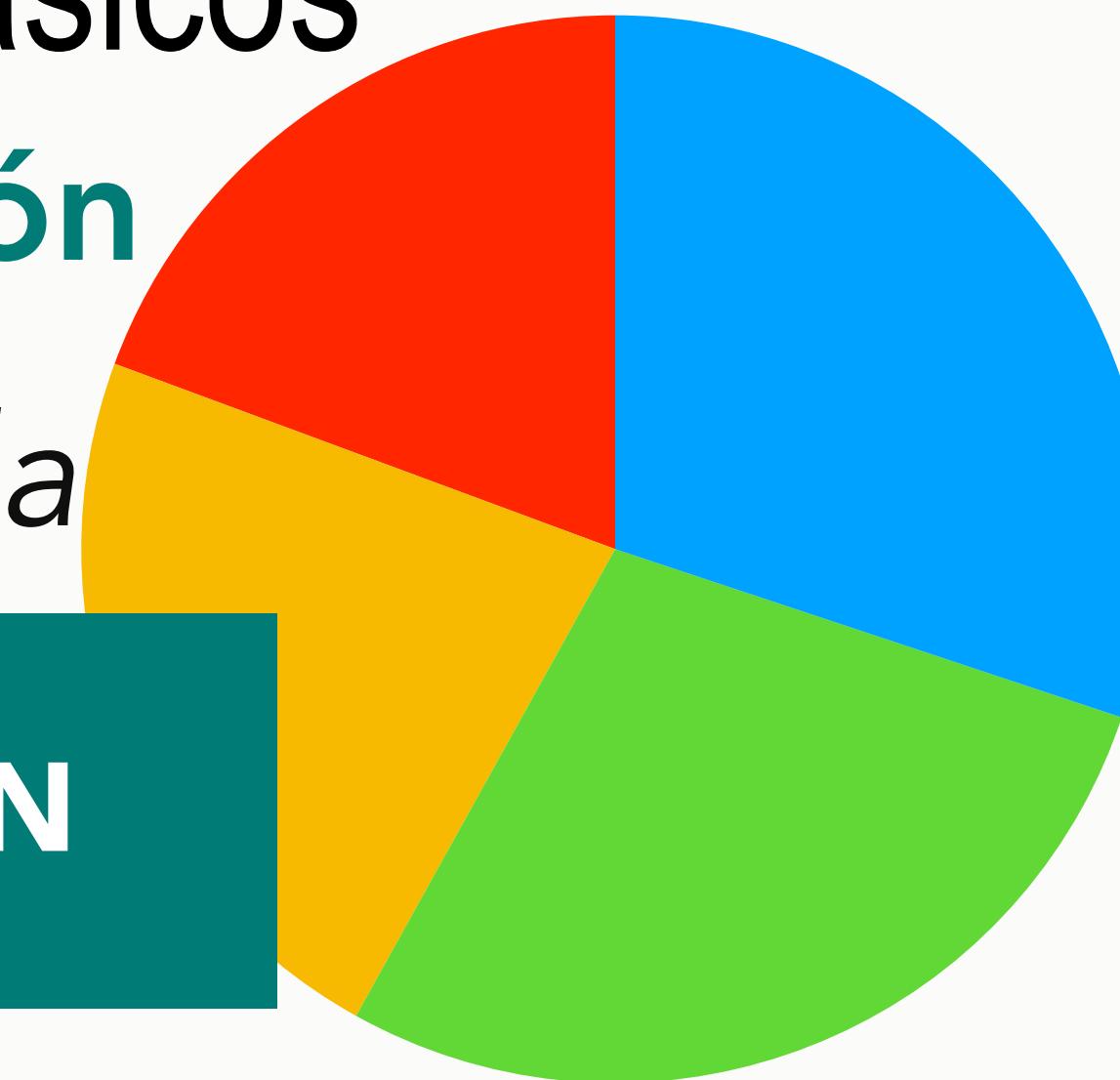


3. Conceptos básicos

Diversificación

se refiere a la

EXPLORACIÓN



Intensificación

hace referencia a la

EXPLOTACIÓN



El equilibrio entre ambos conceptos es necesario para

- Identificar rápidamente regiones con soluciones de buena calidad
- No consumir tiempo en regiones no prometedoras o ya exploradas

3. Conceptos básicos

Diversificación

se refiere a la

EXPLORACIÓN



Intensificación

hace referencia a la

EXPLOTACIÓN



El equilibrio entre ambos conceptos es necesario para

- Identificar rápidamente regiones con soluciones de buena calidad
- No consumir tiempo en regiones no prometedoras o ya exploradas

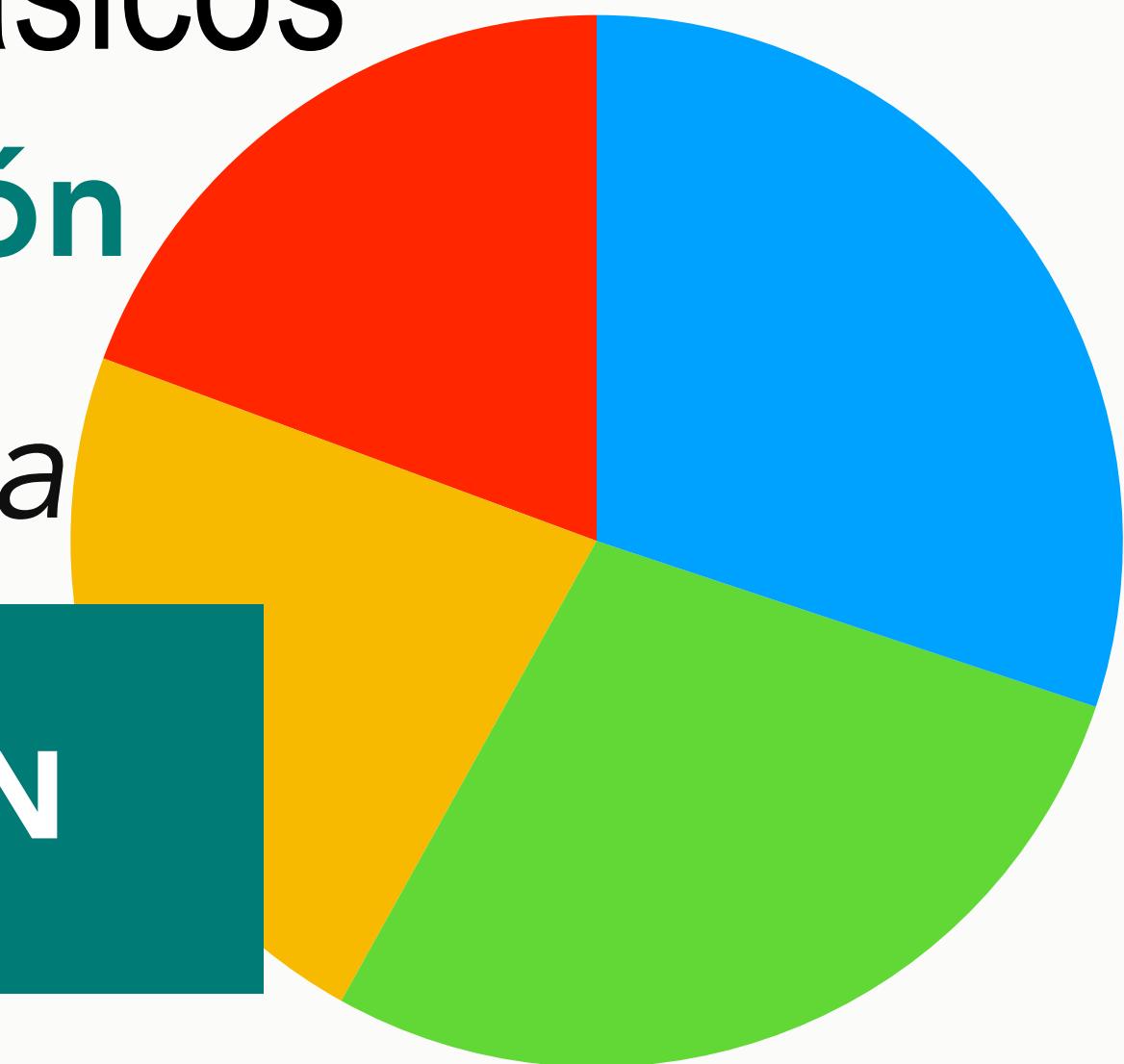


3. Conceptos básicos

Diversificación

se refiere a la

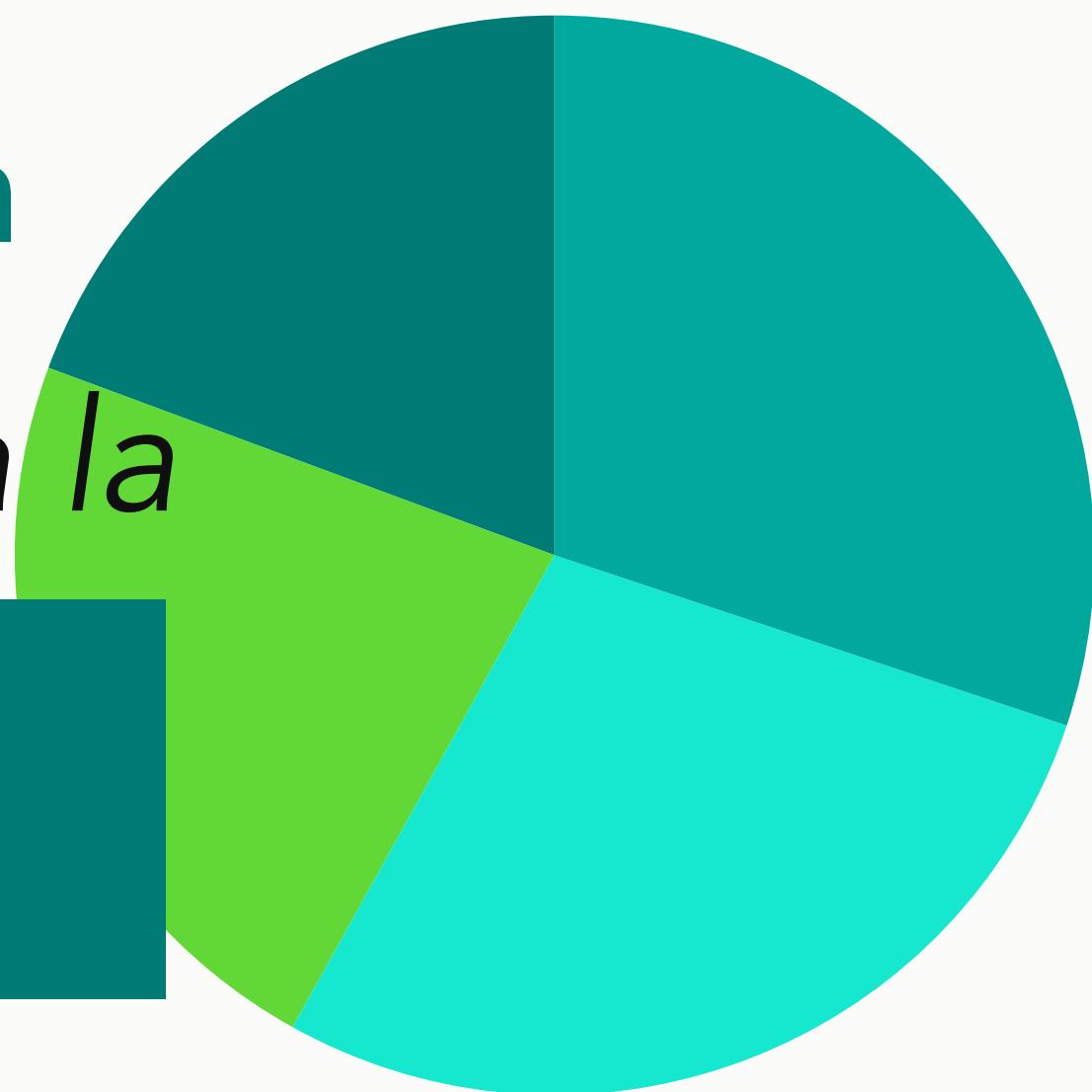
EXPLORACIÓN



Intensificación

hace referencia a la

EXPLOTACIÓN



El equilibrio entre ambos conceptos es necesario para

- Identificar rápidamente regiones con soluciones de buena calidad
- No consumir tiempo en regiones no prometedoras o ya exploradas

Búsqueda
aleatoria



Búsqueda
local

3. Conceptos básicos

representación

Diseñar una metaheurística implica una codificación (representación) de una solución

La codificación es clave en la eficiencia y efectividad

Pueden existir diferentes representaciones

Deben cumplir las siguientes características

- **Completitud**: poder representar todas las soluciones
- **Conexión**: existir un camino de búsqueda entre cualquier par
- **Eficiencia**: fácil de manipular mediante operadores

3. Conceptos básicos

representación

Diseñar una metaheurística implica una codificación (representación) de una solución

La codificación es clave en la eficiencia y efectividad

Pueden existir diferentes representaciones

Deben cumplir las siguientes características

- **Completitud**: poder representar todas las soluciones
- **Conexión**: existir un camino de búsqueda entre cualquier par
- **Eficiencia**: fácil de manipular mediante operadores

Adeuada y
relevante
Será clave la
evaluación y los
operadores

3. Conceptos básicos

representación



3. Conceptos básicos

representación

one-to-one

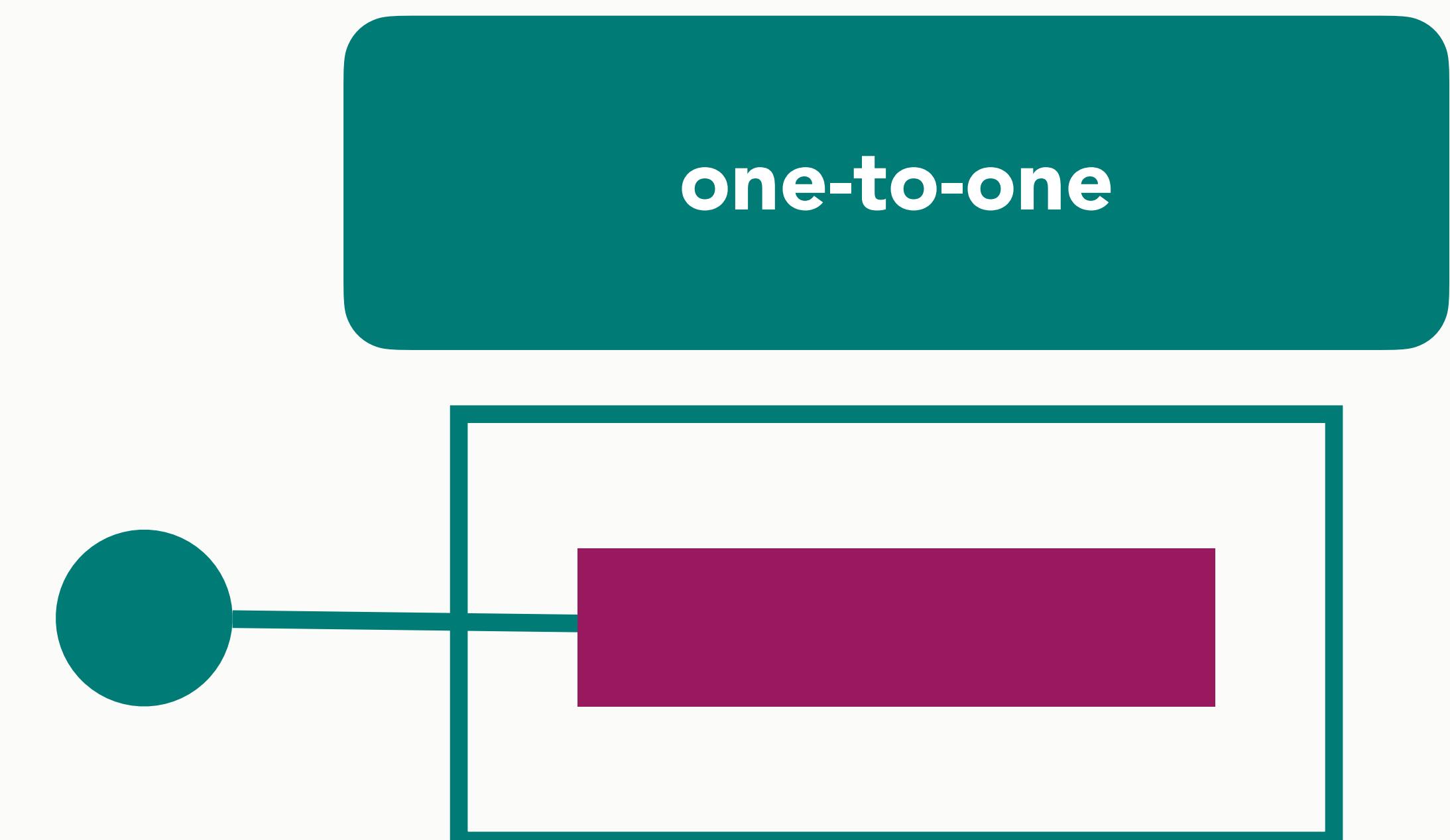


3. Conceptos básicos

representación



one-to-multiple

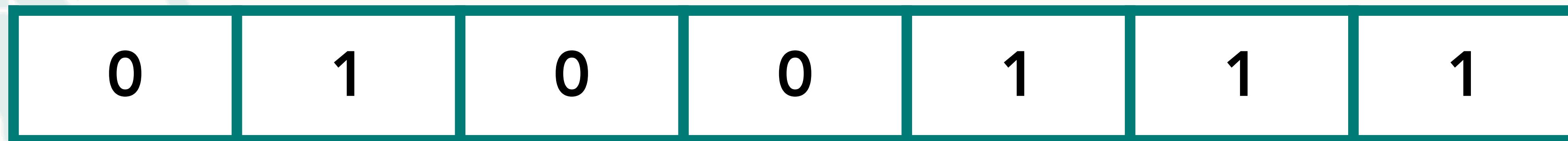


one-to-one

3. Conceptos básicos

representación

BINARIA



Mochila

SAT

Balanza

3. Conceptos básicos

representación

VALORES DISCRETOS



Asignación

Localización

3. Conceptos básicos

representación

REALES



**Optimización
continua**

Parametrización

**Optimización
global**

3. Conceptos básicos

representación

PERMUTACIÓN



Secuenciación

TSP

Programación

3. Conceptos básicos

función objetivo

Describe el objetivo a conseguir

Asocia a cada solución un valor real

Elemento clave de toda metaheurística para guiar el proceso de búsqueda hacia “buenas” soluciones

Ejemplos

- **Autosuficientes:** derivadas directamente de la definición del problema
- **Orientativas:** se transforman para mejorar la convergencia
- **Interactivas:** implican al usuario para su resolución
- **Relativas/Competitivas:** con ranking o mediante competición entre soluciones

3. Conceptos básicos

manejo de restricciones

Son un elemento importante a la hora de abordar un problema

Formas habituales:

- **Estrategias de rechazo**
- **Estrategias de penalización**
- **Estrategias de reparación**
- **Estrategias conservadoras**

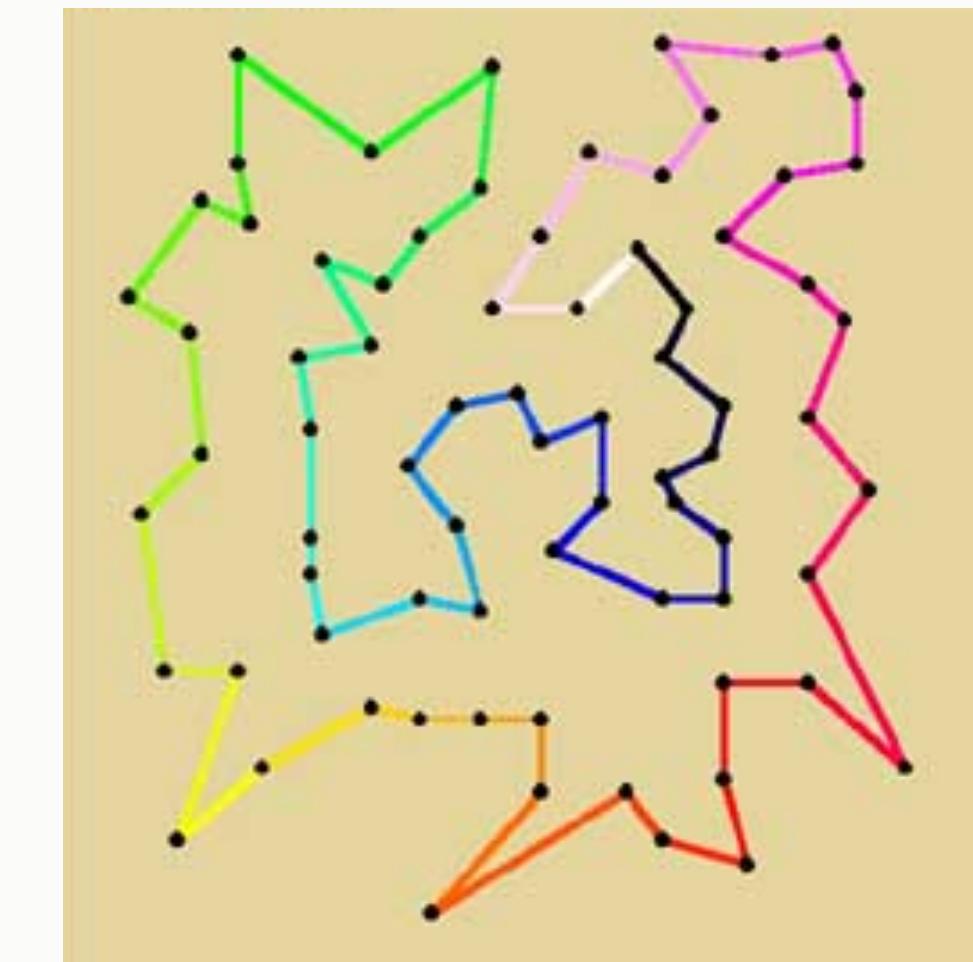
Ejemplo

El problema del viajante de comercio

Representación como secuencia de ciudades (1 a n), $n!$ soluciones

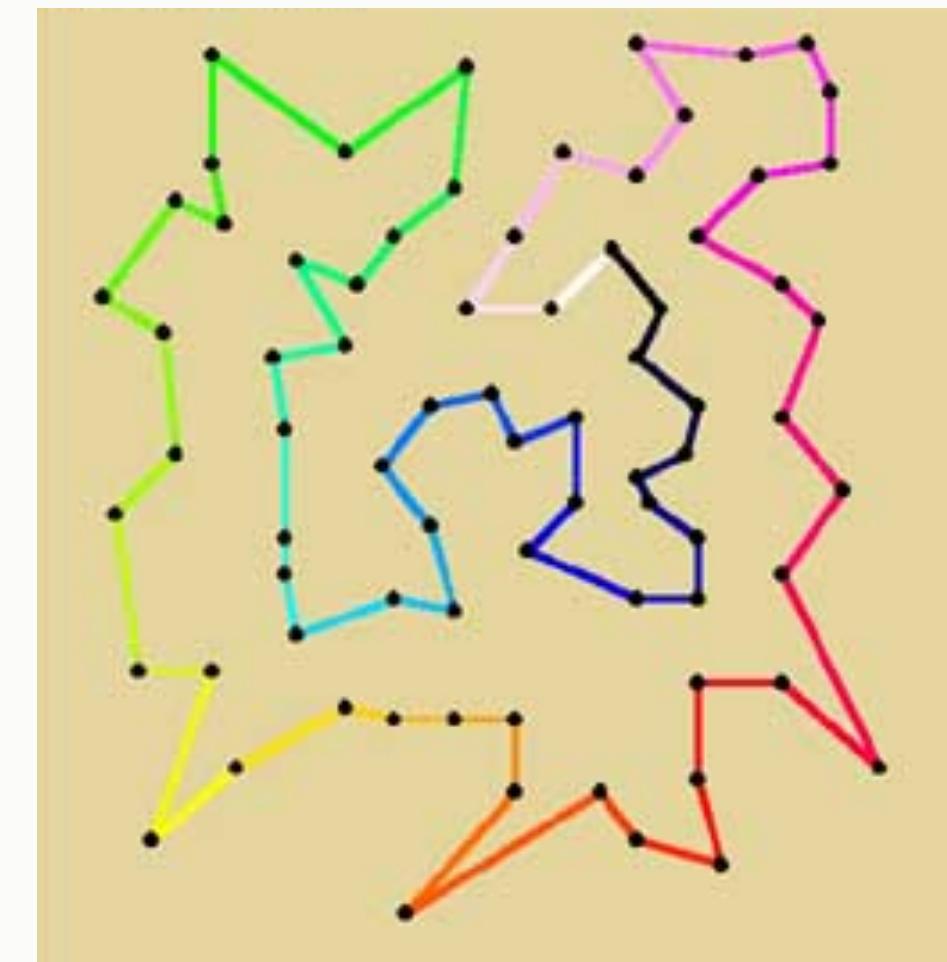
Ejemplo: El problema del viajante de comercio

Problema muy estudiado al presentar aplicaciones reales tales como la fabricación en serie de tarjetas de ordenador (impresión de los buses de estaño)



Ejemplo: El problema del viajante de comercio

Problema muy estudiado al presentar aplicaciones reales tales como la fabricación en serie de tarjetas de ordenador (impresión de los buses de estaño)



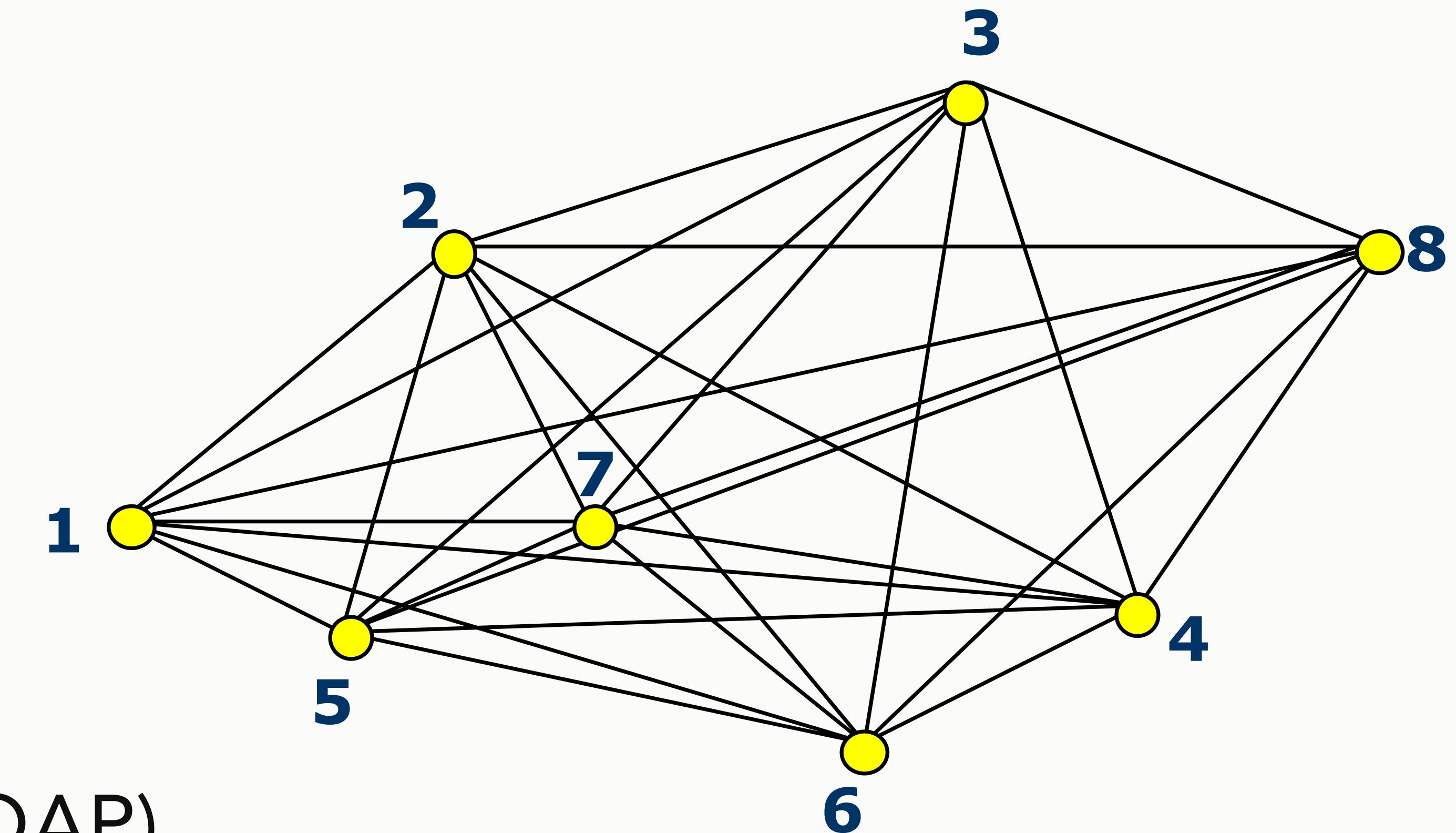
En el viajante de comercio, se tiene una red de nodos, que pueden ser ciudades o simplemente lugares de una ciudad. Se parte de un lugar inicial, y deben recorrerse todos sin pasar más de una vez por cada lugar, volviendo al lugar inicial. Para cada arco, se tiene un valor C_{ij} , que indica la distancia o el costo de ir del nodo i al nodo j

Ejemplo: El problema del viajante de comercio

- ¿Representación?
- Utilización mediante permutaciones de 1 a N
- Aplicaciones
 - Coloreo de grafos
 - Secuenciación de tareas
 - Asignación cuadrática (QAP)

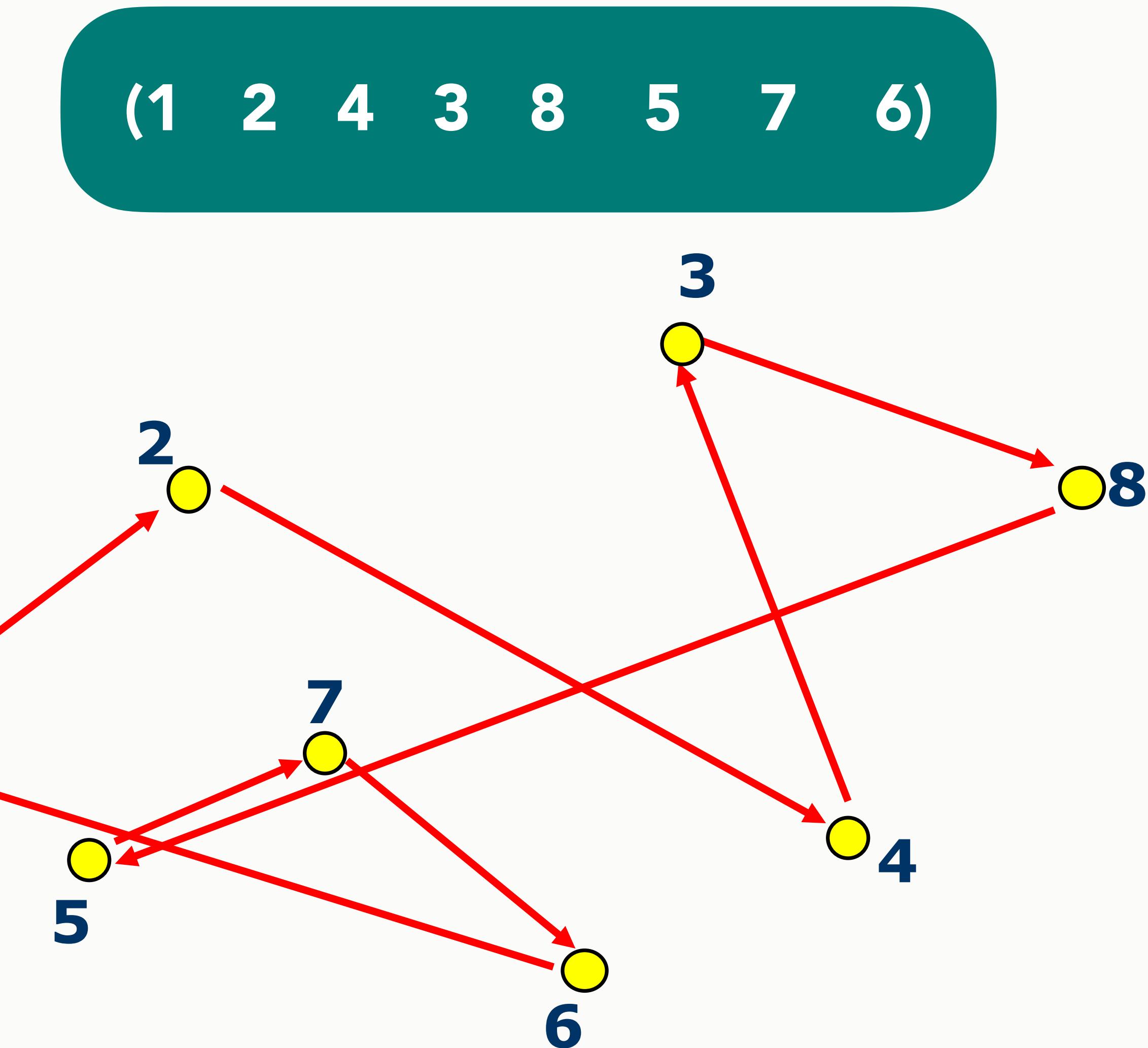
$$X = (x_1, \dots, x_n)$$

$$x_i \in \{1, \dots, N\}$$



Ejemplo: El problema del viajante de comercio

- Representación de orden mediante un camino
- ¿Función objetivo?
 - Minimizar el camino para recorrer todos los puntos pasando únicamente una vez por cada uno de ellos



Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

**Matriz
de
distancias**

	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	82	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	22	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-

Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	82	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	22	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-

0

●

Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	82	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	22	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-

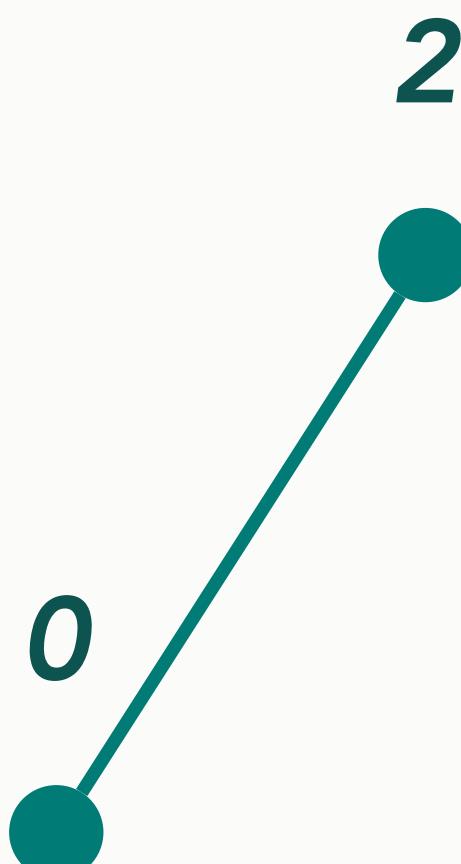
0

●

Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

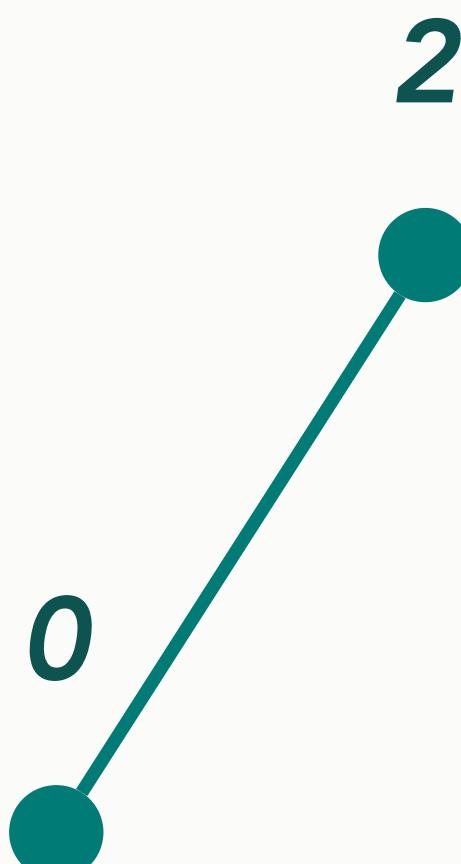
	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	82	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	22	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-



Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

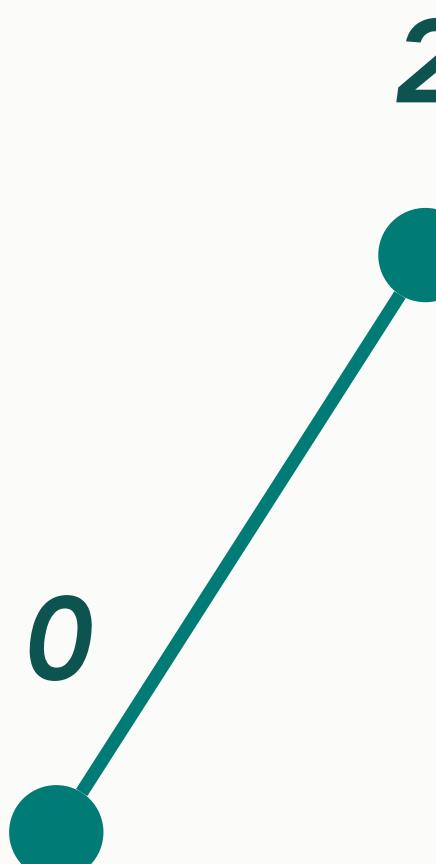
	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	82	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	22	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-



Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

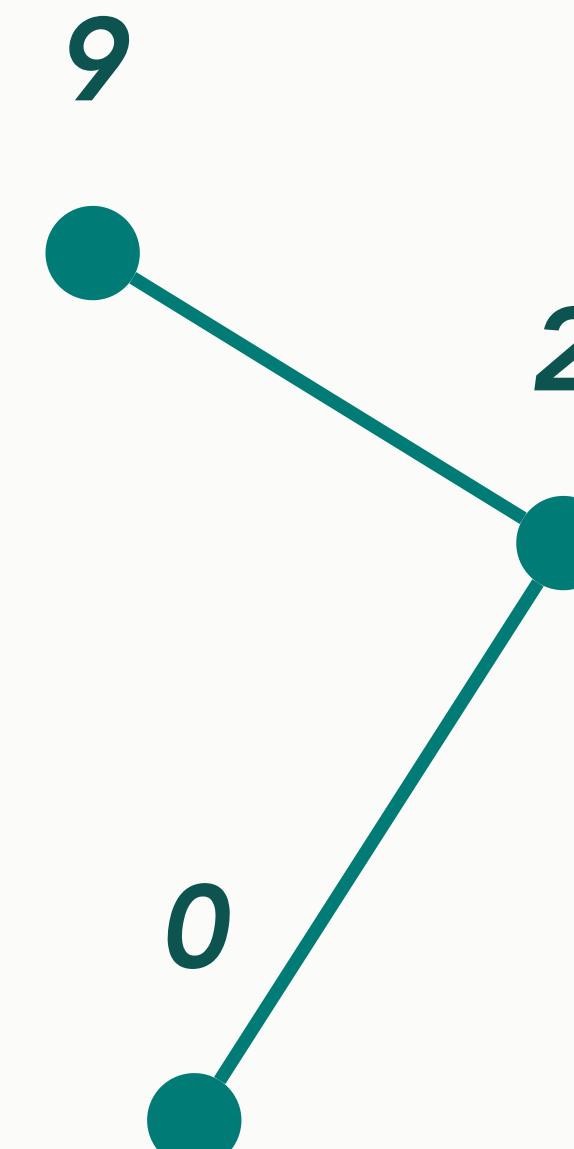
	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	-	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	22	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-



Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

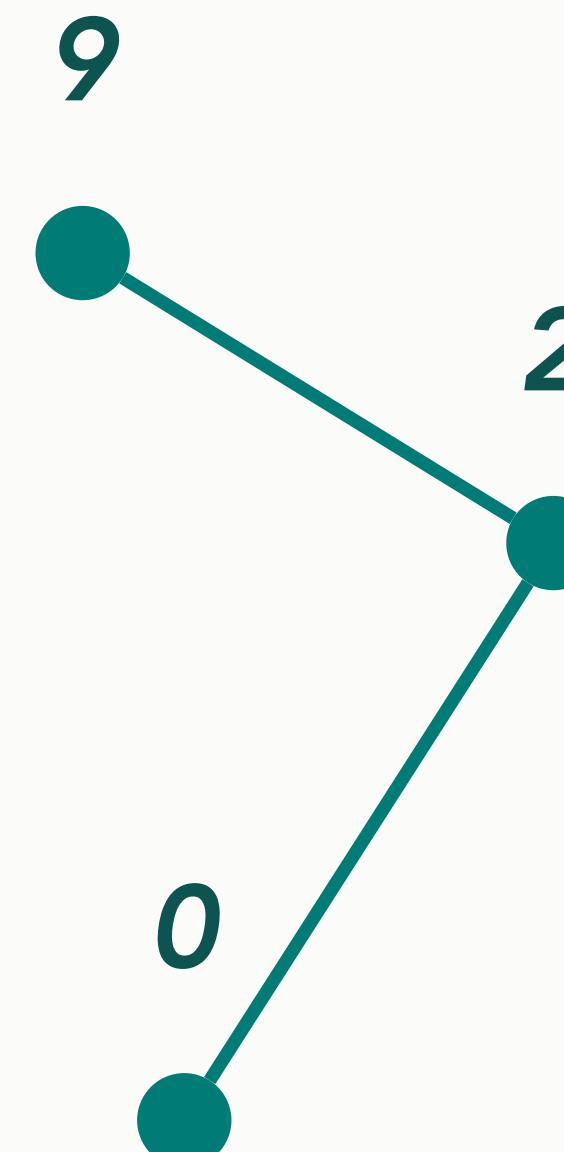
	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	-	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	22	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-



Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	-	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	72	-	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-



Ejemplo: El problema del viajante de comercio

resolución mediante una heurística constructiva

	1	2	3	4	5	6	7	8	9	10
0	76	7	73	69	79	54	58	38	23	39
1	-	82	108	16	52	99	51	42	72	59
2	-	-	69	74	81	50	60	42	22	40
3	108	69	-	92	68	19	57	73	52	52
4	16	74	92	-	36	84	35	31	60	45
5	52	81	68	36	-	68	21	46	60	41
6	99	50	19	847	68	-	52	60	35	40
7	51	60	57	35	21	52	-	28	39	20
8	42	42	73	31	46	60	28	-	30	21
9	-	-	52	60	60	35	39	30	-	19
10	59	40	52	45	41	40	20	21	19	-

Mejor coste = 308

Sobre un 8-9% mejor

Possible mejora con
heurísticas de
vecindad o búsqueda
local

SOLUCIÓN 0 2 9 10 7 5 4 1 8 6 3 ¿cuál es el coste?

4

Evaluación



4. Evaluación

Es complicado definir una metaheurística como “la mejor metaheurística”

Hay aspectos evaluables:

- ¿Está bien diseñada?
- ¿Su funcionamiento es correcto?

Análisis cualitativo

Análisis cuantitativo

4. Evaluación

análisis cualitativo, ¿está bien diseñada?

Se recomienda:

1. Claridad, modularidad, simplicidad
2. Intensificación
3. Diversificación

4. Evaluación

análisis cualitativo, ¿está bien diseñada?

Se recomienda:

1. Claridad, modularidad, simplicidad

Debería ser fácil de aplicar a cualquier problema difícil

2. Intensificación

3. Diversificación

4. Evaluación

análisis cualitativo, ¿está bien diseñada?

Se recomienda:

1. Claridad, modularidad, simplicidad
2. Intensificación

**El óptimo global también es óptimo local
Una buena metaheurística debería tener la habilidad de localizar
óptimos locales**

3. Diversificación

4. Evaluación

análisis cualitativo, ¿está bien diseñada?

Se recomienda:

1. Claridad, modularidad, simplicidad
2. Intensificación
3. Diversificación

**Las soluciones interesantes podrían estar en regiones diferentes
Una buena metaheurística debería proporcionar una exploración amplia del espacio de soluciones**

4. Evaluación

análisis cuantitativo, ¿funciona bien en un problema?

Evaluarla sobre un problema una vez implementada

Criterios

- Habilidad para encontrar buenas soluciones
- Tiempo computacionalmente razonable
- Robustez y estabilidad sobre diferentes ejecuciones

4. Evaluación

análisis cuantitativo, ¿funciona bien en un problema?

Evaluarla sobre un problema una vez implementada

Criterios

- Habilidad para encontrar buenas soluciones

Comparación con soluciones óptimas, con cotas inferiores y/o superiores, con otras metaheurísticas

- Tiempo computacionalmente razonable
- Robustez y estabilidad sobre diferentes ejecuciones

4. Evaluación

análisis cuantitativo, ¿funciona bien en un problema?

Evaluarla sobre un problema una vez implementada

Criterios

- Habilidad para encontrar buenas soluciones
- Tiempo computacionalmente razonable

La palabra razonable es subjetiva

Tiene una dependencia directa del contexto del problema

- Robustez y estabilidad sobre diferentes ejecuciones

4. Evaluación

análisis cuantitativo, ¿funciona bien en un problema?

Evaluarla sobre un problema una vez implementada

Criterios

- Habilidad para encontrar buenas soluciones
- Tiempo computacionalmente razonable
- Robustez y estabilidad sobre diferentes ejecuciones

Análisis estadístico

4. Evaluación

el análisis experimental

Diseño experimental

Definir

- objetivos de los experimentos
- instancias
- factores(parámetros) a considerar

Medida

- Seleccionar medidas
- Analizar estadísticamente los resultados

Informe

- Presentación de resultados de forma clara
- Uso de técnicas de visualización

De vital importancia la **reproducibilidad** de los resultados

4. Evaluación

el análisis experimental

Diseño experimental

Definir

- objetivos de los experimentos
- instancias
- factores(parámetros) a considerar

Objetivos:

- tiempo de búsqueda
- calidad de las soluciones
- robustez
- resolución de problemas a gran escala
- procesamiento paralelo escalable respecto al número de procesadores
- simplicidad de la implementación
- facilidad para combinar con otros algoritmos
- ajuste automático de parámetros

4. Evaluación

el análisis experimental

Diseño experimental

Definir

- objetivos de los experimentos
- instancias
- factores(parámetros) a considerar

Instancias: reales/construídas

Algunas librerías públicas de instancias estándar

- <http://people.brunel.ac.uk/~mastjjb/jeb/info.html>
- <http://dimacs.rutgers.edu/Challenges/>
- <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>
- <http://www.opt.math.tu-graz.ac.at/qaplib/inst.html>

4. Evaluación

el análisis experimental

Calidad de las soluciones:



- Seleccionar medidas
- Analizar estadísticamente los resultados

Esfuerzo computacional mediante análisis teórico y análisis empírico (tiempos, SO, lenguaje, etc.)

Robustez sin variabilidad respecto a pequeñas desviaciones

4. Evaluación

el análisis experimental

Medida

- Seleccionar medidas
- Analizar estadísticamente los resultados

- Análisis estadístico
 - Paramétricos (t-test, ANOVA)
 - No-paramétricos (Wilcoxon, Friedman, etc.)
 - Análisis simplificado con medias, desviaciones típicas, mayor, menor, etc.

4. Evaluación

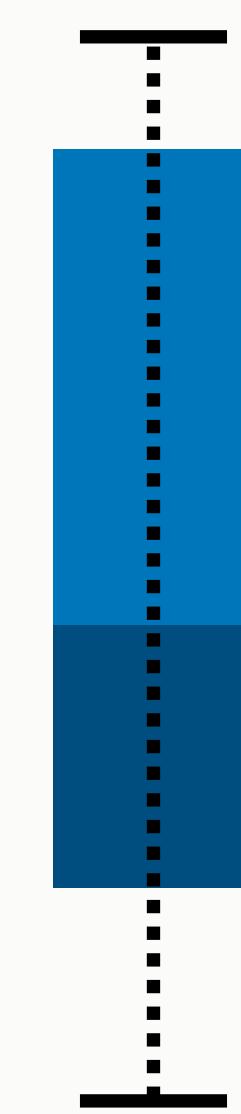
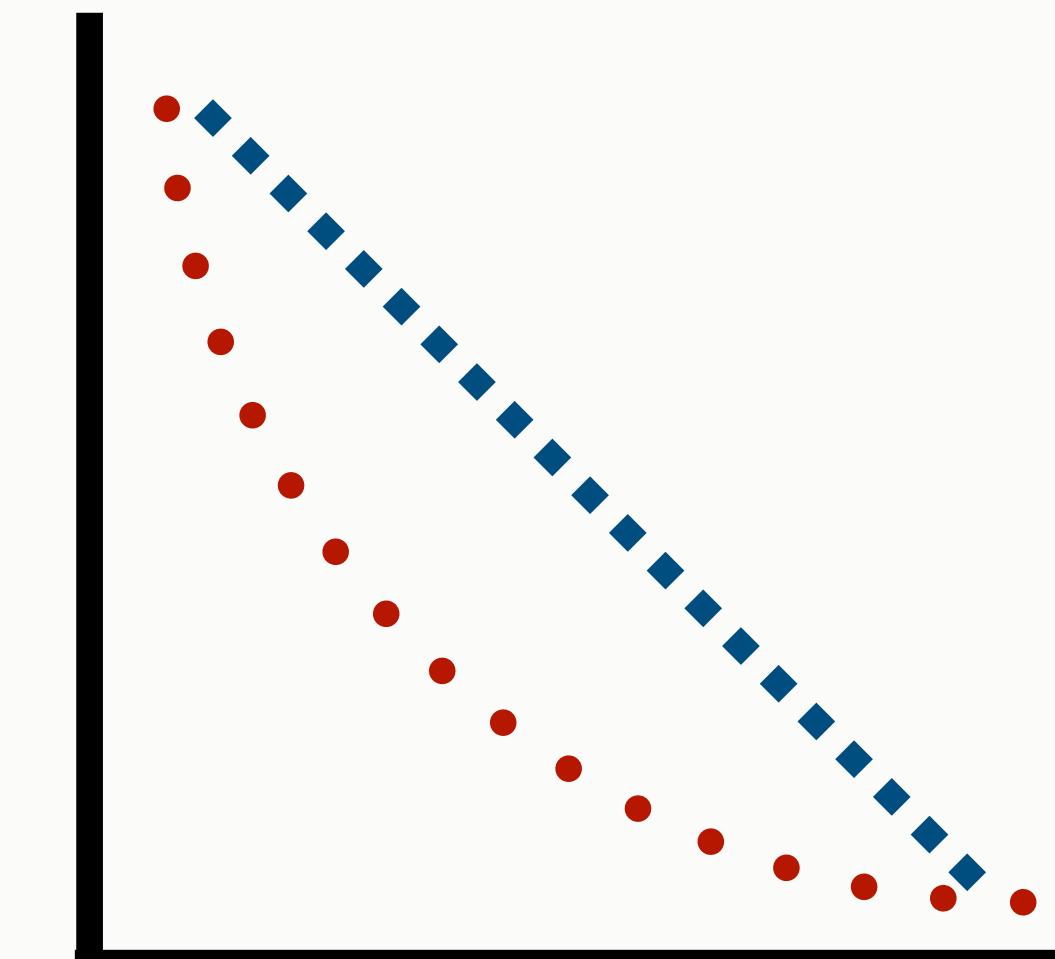
el análisis experimental

Informe

- Presentación de resultados de forma clara
- Uso de técnicas de visualización

Debe estar guiada por los objetivos y las medidas de calidad consideradas

Las herramientas de visualización ayudan



Mayor
Cuartil superior

Mediana

Cuartil inferior

Menor

5

Paralelización

5. Paralelización

objetivos

1. Preservar la calidad de las soluciones reduciendo el tiempo de ejecución
2. Incrementar la calidad de las soluciones sin aumentar el tiempo de cálculo
3. Obtener soluciones de mayor calidad debido al efecto sinérgico de la distribución espacial de la búsqueda

5. Paralelización

objetivos

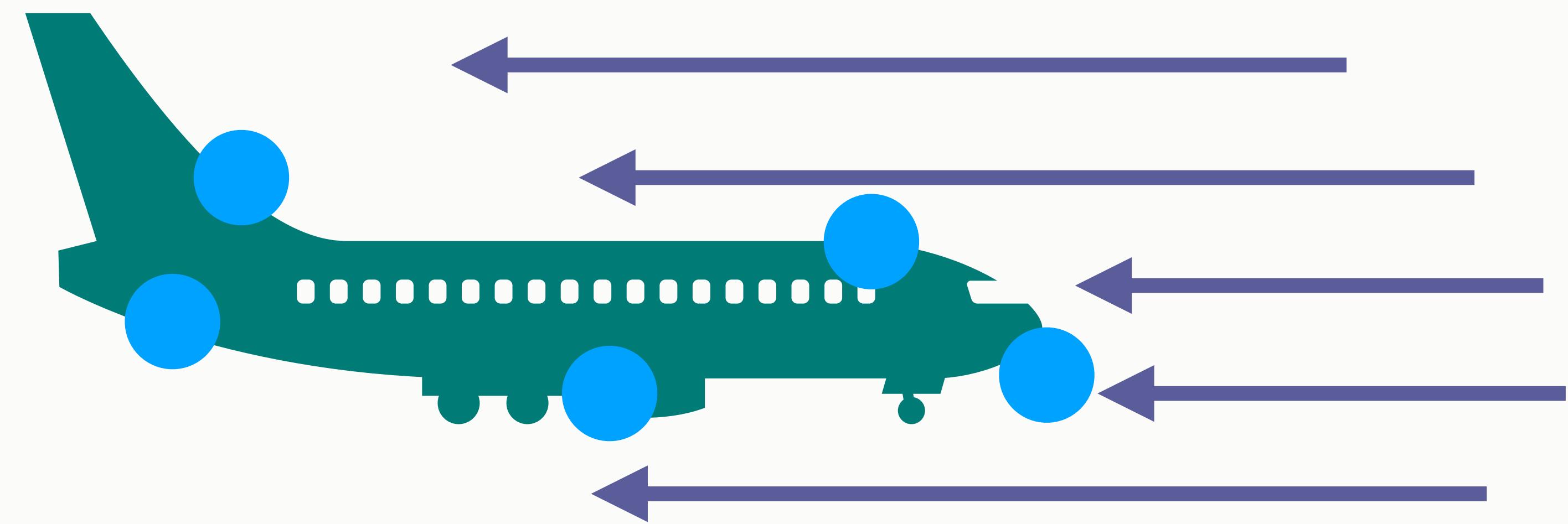
Todas las tendencias de arquitecturas de cómputo actuales llevan a sistemas paralelos y/o distribuidos:

- Arquitecturas de procesadores
(procesadores multinúcleo como Core 2 Quad, multihebra (i7), multiprocesadores como OpenMP)
- Construcción de sistemas paralelos de bajo coste
(clusters, OpenMPI)
- Interconexión de centros de supercomputación como Grid

6

Aplicaciones

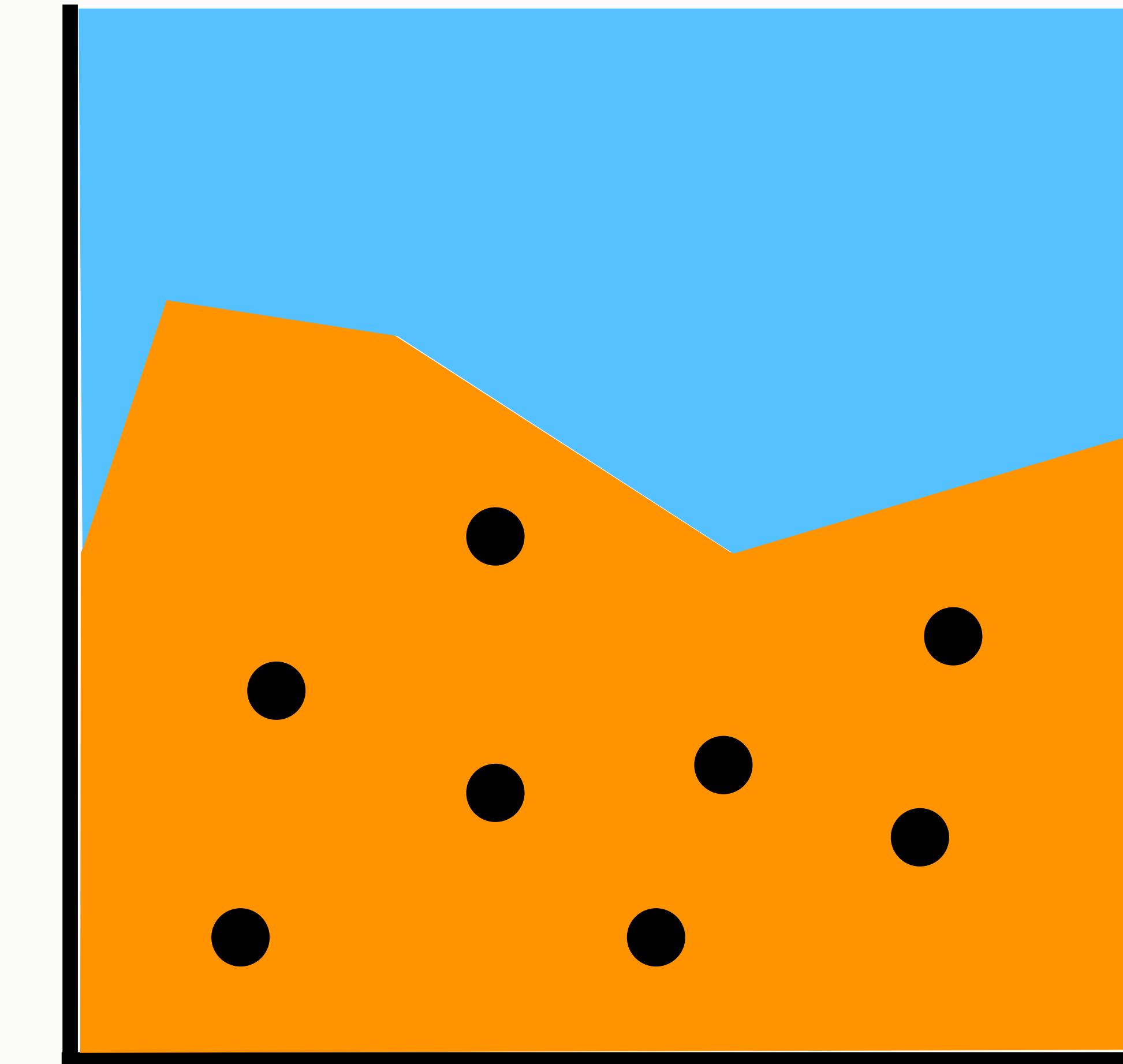
6. Aplicaciones



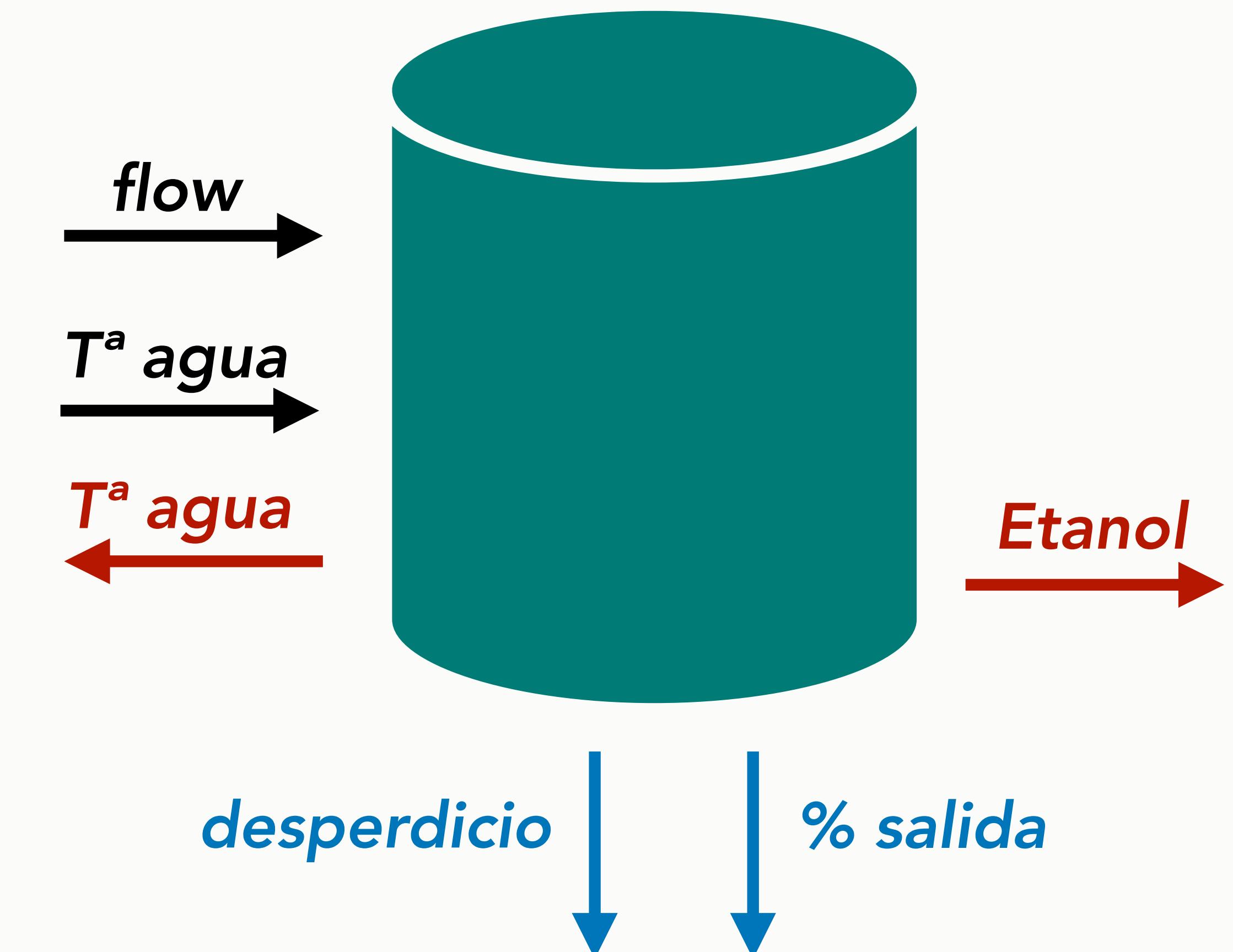
6. Aplicaciones

Optimización
estructural

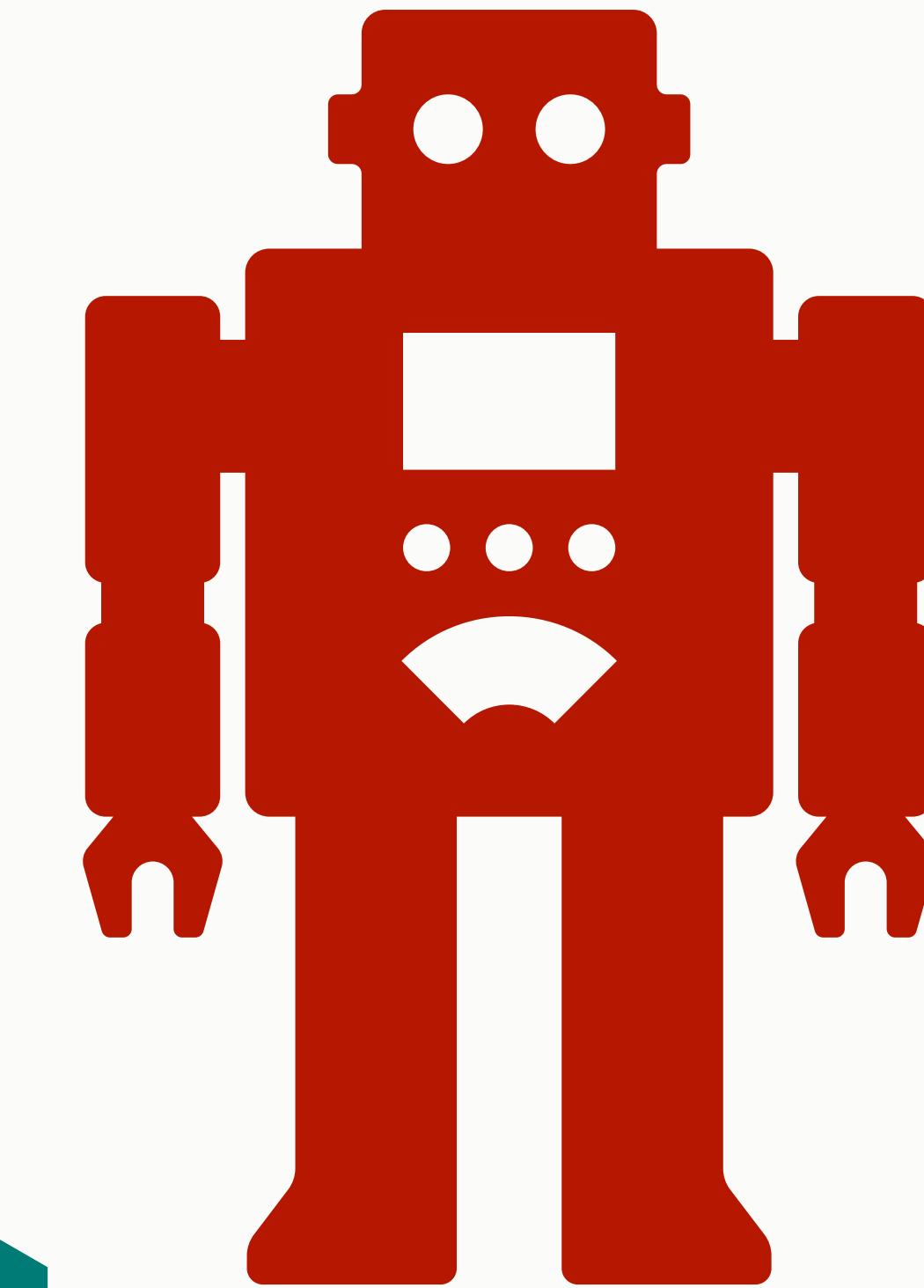
Aprendizaje
Clasificación



6. Aplicaciones



6. Aplicaciones



6. Aplicaciones



6. Aplicaciones

Optimización
estructural

Aprendizaje
Clasificación

Control
procesos
químicos

Sistemas de
producción

Trayectorias

Edge
Computing



Resumen

Resumen

pasos a seguir en la resolución de problemas de optimización

1. Modelar el problema (inspirándonos en modelos similares)
2. Identificar si debería resolverse con metaheurísticas
 - Complejidad y dificultad del problema (NP-completitud, tamaño y estructura de las instancias de entrada...)
 - Requerimientos (tiempo de búsqueda, calidad de la solución, ...)
 - Realizar una revisión del estado del arte en algoritmos de optimización para resolver el problema (exactos y aproximados)

Resumen

pasos a seguir en la resolución de problemas de optimización

3. Si se va a diseñar una metaheurística, se debe determinar:

- Representación de las soluciones del problema, consistente con respecto a la función de evaluación y operadores.
- Función objetivo, que guíe la búsqueda hacia soluciones “buenas”
- Manejo de restricciones sobre el espacio de soluciones y los valores de las variables

Resumen

pasos a seguir en la resolución de problemas de optimización

4. Elegir un entorno software para la implementación
5. Toda metaheurística tiene parámetros que se deben ajustar para cada problema y que tienen influencia en la eficiencia y eficacia de la búsqueda

No existe un conjunto universal de parámetros

6. Evaluación del rendimiento de la metaheurística



Metaheurísticas

Grado en Ingeniería Informática

Universidad de Jaén

Cristóbal J. Carmona

Curso 2023/2024

Esta obra está protegida con licencia
Creative Commons Atribución-NoComercial 4.0 Internacional

