

METAHEURÍSTICAS

2024-2025



- **Tema 1. Introducción a las Metaheurísticas**
- **Tema 2. Modelos de Búsqueda: Entornos y Trayectorias vs Poblaciones**
- **Tema 3. Metaheurísticas Basadas en Poblaciones**
- **Tema 4: Algoritmos Meméticos**
- **Tema 5. Metaheurísticas Basadas en Trayectorias**
- **Tema 6. Metaheurísticas Basadas en Adaptación Social**
- **Tema 7. Aspectos Avanzados en Metaheurísticas**
- **Tema 8. Metaheurísticas Paralelas**
- **Tema 9. Modelos de IA Evolutivos. Aprendizaje Evolutivo**

Objetivos

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

Motivación

- Múltiples problemas de optimización de ciencia, ingeniería, economía, etc. son 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:
 - **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
 - Resuelven problemas más complejos
 - Obtienen algoritmos más robustos

Motivación

- **Metaheurísticas:** Optimización/búsqueda
- Intersección de campos:
 - Inteligencia Artificial
 - Teoría de Algoritmos, etc.
- Existen metaheurísticas que son metáforas naturales para resolver problemas:
 - Evolución de especies
 - Procesos físicos: enfriamiento de partículas, ...
 - Sociedades de insectos: Colonias de hormigas, abejas, ...
 - Comportamiento de especies, ...

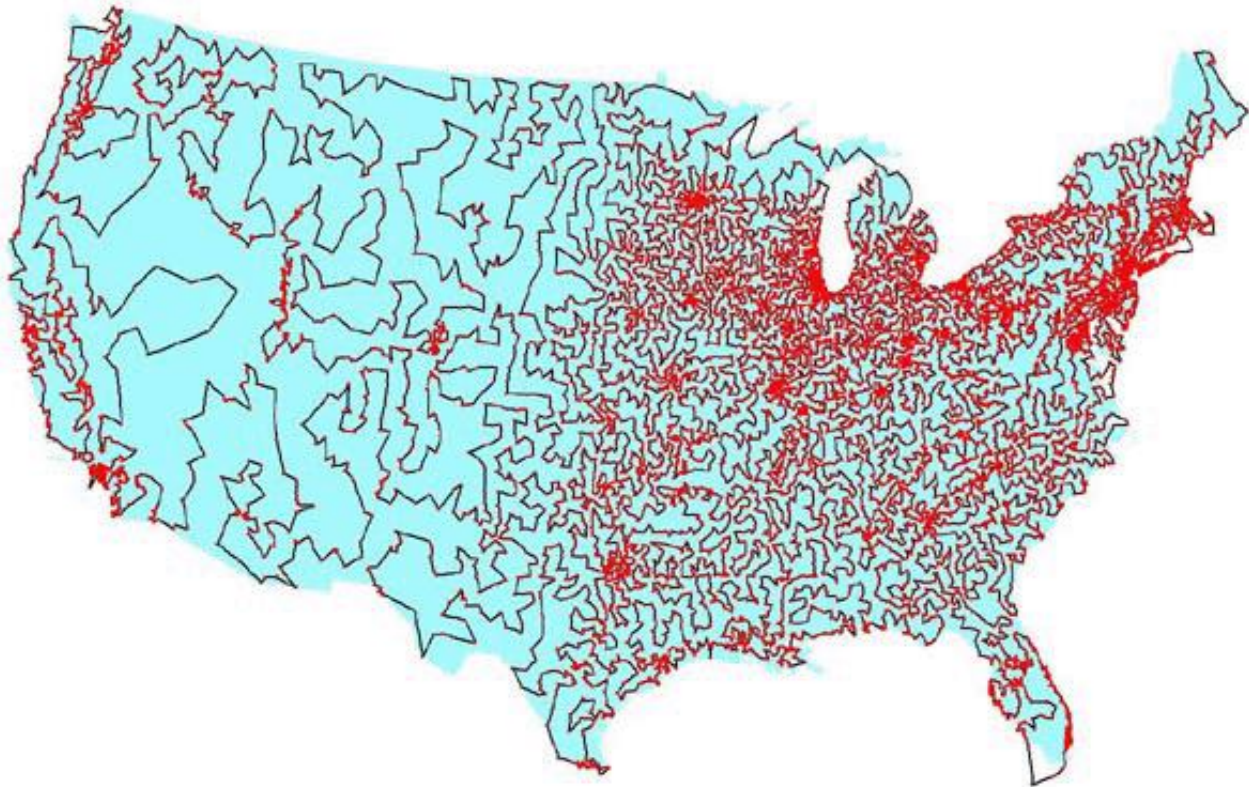
METAHEURÍSTICAS

TEMA 1. Introducción a las Metaheurísticas

1. Resolución de problemas mediante algoritmos de búsqueda
3. Metaheurísticas: definición y clasificación
4. Metaheurísticas: Paralelización
5. Aplicaciones

Ejemplo: El problema del viajante de comercio

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



¿Necesitamos una solución exacta o una Buena solución para el problema?

1. RESOLUCIÓN DE PROBLEMAS MEDIANTE ALGORITMOS DE BÚSQUEDA

Complejidad algorítmica: Algoritmos en tiempo polinomial y no polinomial

	n=5	n=10	n=100	n=1000
n	5	10	100	1000
n^2	25	100	10000	1000000
n^3	125	1000	1000000	10^9
2^n	32	1024	1.27×10^{30}	1.07×10^{301}
n!	120	3.6×10^6	9.33×10^{157}	4.02×10^{2567}

¡Necesitamos buenos algoritmos y eficientes!

Algoritmos que proporcionen una buena solución en un tiempo razonable

1. RESOLUCIÓN DE PROBLEMAS MEDIANTE ALGORITMOS DE BÚSQUEDA

- Existen problemas reales (de optimización o búsqueda) de difícil solución que requieren de tareas tales como encontrar:
 - 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. RESOLUCIÓN DE PROBLEMAS MEDIANTE ALGORITMOS DE BÚSQUEDA

- Estos problemas se caracterizan porque:
 - presentan una gran complejidad computacional (son NP-duros)
 - los algoritmos exactos (Programación Dinámica, Backtracking, Branch and Bound, ...) son ineficientes o simplemente imposibles de aplicar,
 - se encuentran en muchas áreas de aplicación,
 - en la práctica se resuelven mediante **algoritmos aproximados** que proporcionan **buenas soluciones** (no necesariamente la óptima) al problema **en un tiempo razonable**

1. RESOLUCIÓN DE PROBLEMAS MEDIANTE ALGORITMOS DE BÚSQUEDA

Elementos del problema de búsqueda

■ Función objetivo

Max (Min) una función con variables de decisión

Subject to (s.t.)

igualdad (=) restricciones

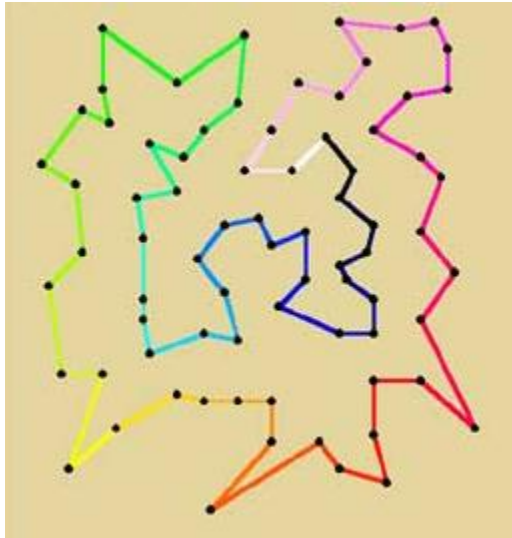
desigualdad (<, >, ≤, ≥) restricciones

■ Espacio de búsqueda

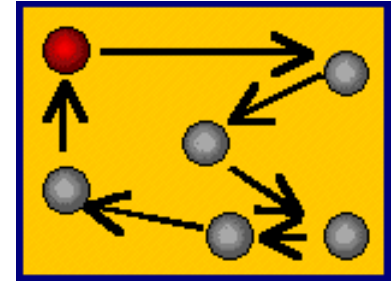
Valores de las variables de decisión que serán evaluados durante el proceso de optimización.

Puede ser discreto, contable o continuo e incontable.

Ejemplo: El problema del viajante de comercio



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

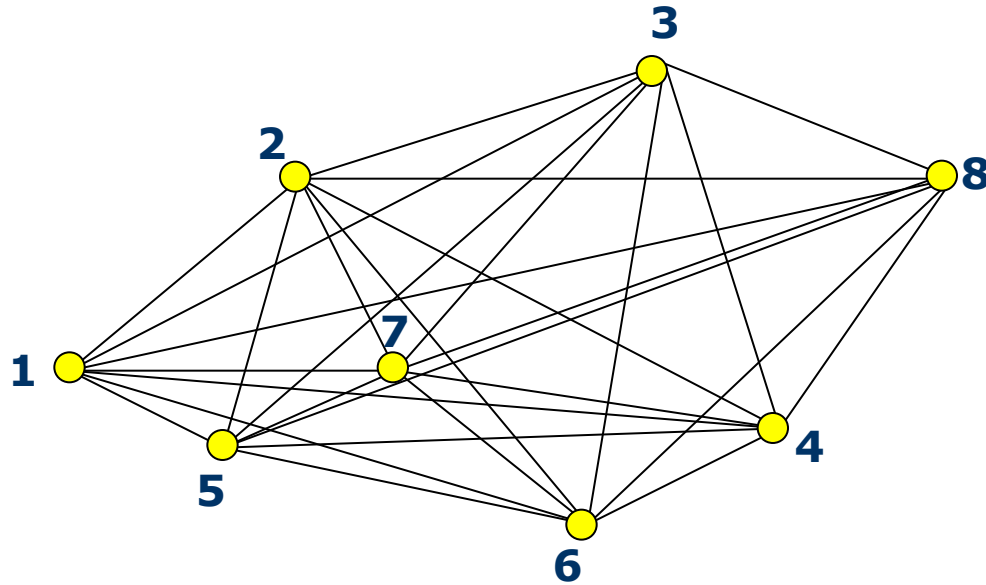


En el viajante de comercio, se tiene una red de nodos, que pueden ser ciudades o simplemente lugares de una ciudad. Se parte de un lugar inicial, y deben recorrerse todos sin pasar más de una vez por cada lugar, volviendo al lugar inicial. Para cada arco, se tiene un valor C_{ij} , que indica la distancia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

Ejercicio: Analizar el espacio de búsqueda **¿Cómo representar una solución al problema?**

Ejemplo: El problema del viajante de comercio

■ Ejemplo: Viajante de Comercio



Ejemplo: El problema del viajante de comercio

Representación de Orden

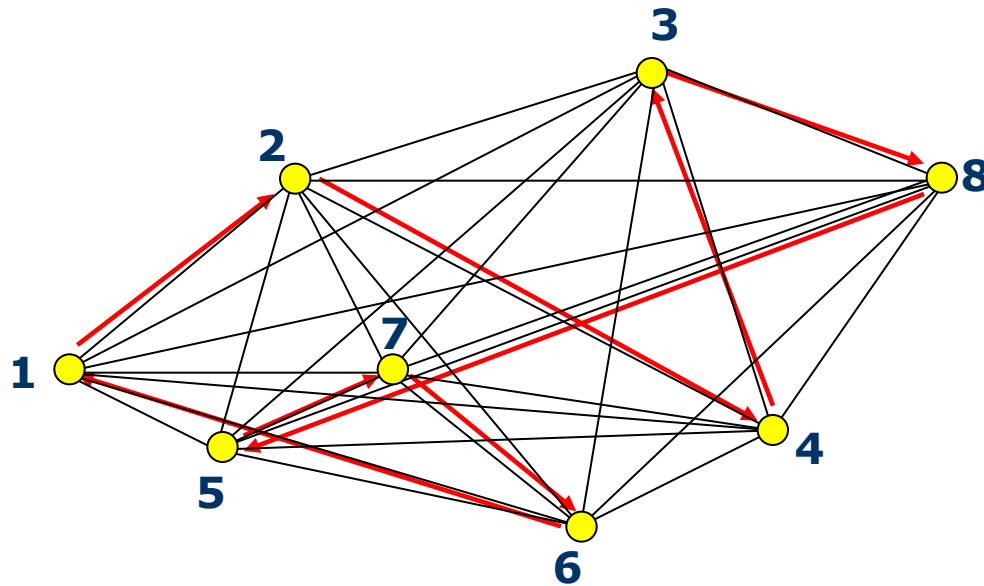
- Se utiliza para problemas donde la solución se representa como una permutación de $1, \dots, N$

$$X = (x_1, \dots, x_n) \quad x_i \in \{1, \dots, N\}$$

- Aplicaciones: Viajante de Comercio (TSP), Coloreo de Grafos, Secuenciación de tareas, QAP (asignación cuadrática),

Ejemplo: El problema del viajante de comercio

■ Ejemplo: Viajante de Comercio



■ Representación de una solución: Camino (1 2 4 3 8 5 7 6)

Ejemplo: El problema del viajante de comercio

Espacio de búsqueda y función objetivo

1. Esquema de representación:
Permutación de $\{1, \dots, n\}$.

2. Función objetivo:

$$\mathit{Min} \, C(S) = \sum_{i=1}^{n-1} (D[S[i], S[i + 1]]) + D[S[n], S[1]]$$

1. RESOLUCIÓN DE PROBLEMAS MEDIANTE ALGORITMOS DE BÚSQUEDA

Representación del espacio de búsqueda

- Knapsack problem
- SAT problem
- 0/1 IP problems

1 0 0 0 1 1 0 1 1 1 0 1

Binary encoding

- Location problem
- Assignment problem

5 7 6 6 4 3 8 4 2

Vector of discrete values

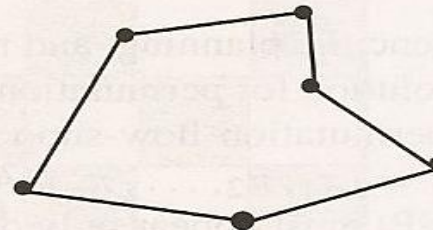
- Continuous optimization
- Parameter identification
- Global optimization

$$f(x) = 2x + 4x \cdot y - 2x \cdot z$$

1.23 5.65 9.45 4.76 8.96

Vector of real values

- Sequencing problems
- Traveling salesman problem
- Scheduling problems



1 4 8 9 3 6 5 2 7

Permutation

METAHEURÍSTICAS

TEMA 1. Introducción a las Metaheurísticas

1. Resolución de problemas mediante algoritmos de búsqueda
3. Metaheurísticas: definición y clasificación
4. Metaheurísticas: Paralelización
5. Aplicaciones

N. Xiong, D. Molina, M. Leon-Ortiz, F. Herrera. A walk into Metaheuristics for Engineering Optimization: Principles, Methods and Recent Trends. International Journal of Computational Intelligent Systems (IJCIS), 8, 2015, 606-636.

3. Metaheurísticas: Definición

ALGORITMOS APROXIMADOS

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

Factores que pueden hacer interesante su uso

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

3. Metaheurísticas: Definición

Son una familia de algoritmos aproximados de propósito general. 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.

■ **Ventajas:**

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

■ **Inconvenientes:**

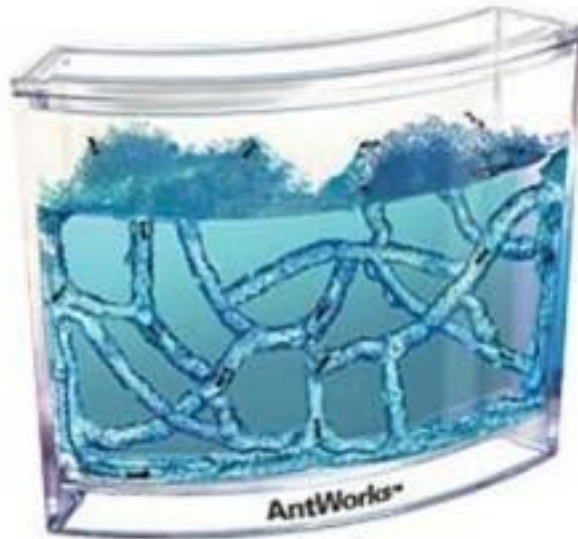
- **Son algoritmos aproximados, no exactos**
- **Son no determinísticos (probabilísticos)**
- **No siempre existe una base teórica establecida**

3. Metaheurísticas: Taxonomía

- Una posible taxonomía:
 - ***Basadas en métodos constructivos***
 - ***Basadas en trayectorias***
 - ***Basadas en poblaciones***

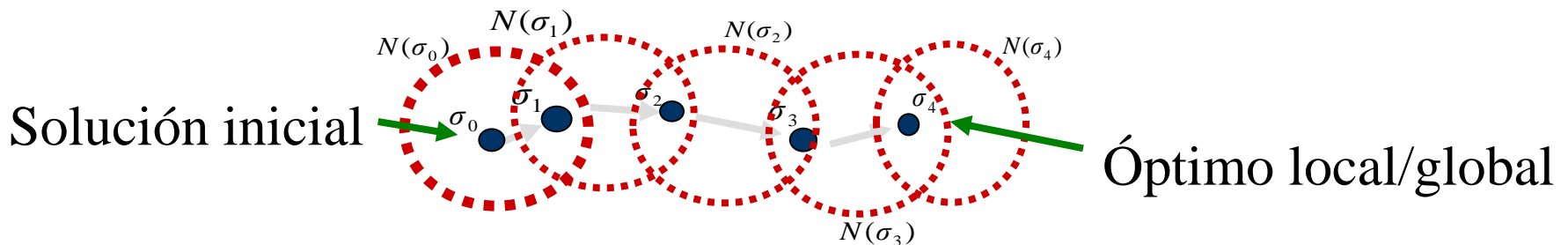
3. Metaheurísticas: Taxonomía

- Una posible taxonomía:
 - ***Basadas en métodos constructivos***: (mecanismos para construir soluciones) GRASP, Optimización Basada en Colonias de Hormigas



3. Metaheurísticas: Taxonomía

- Una posible taxonomía:
 - **Basadas en métodos constructivos:** GRASP, Optimización Basada en Colonias de Hormigas
 - **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, BL Iterativa, ...

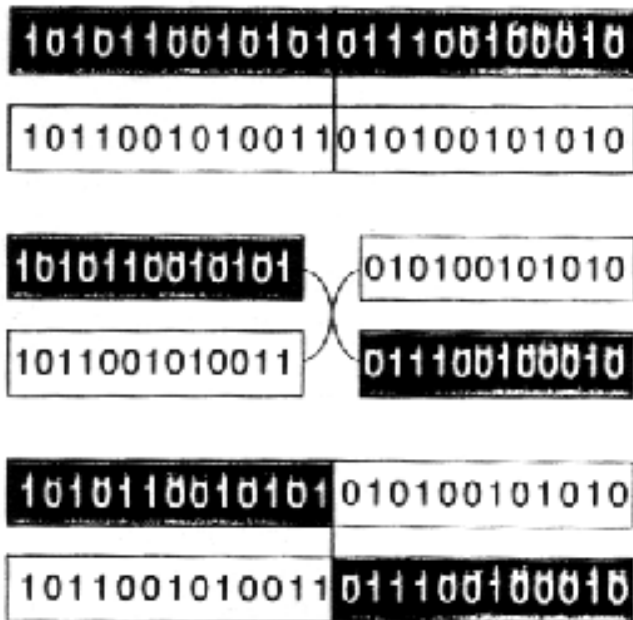


3. Metaheurísticas: Taxonomía

- Una posible taxonomía:
 - **Basadas en métodos constructivos:** GRASP, Optimización Basada en Colonias de Hormigas
 - **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, BL Iterativa, ...
 - **Basadas en poblaciones** (*el proceso considera múltiples puntos de búsqueda en el espacio*): Algoritmos Genéticos, Algoritmos Meméticos, Differential Evolution, ...

3. Metaheurísticas: Definición

Basadas en poblaciones

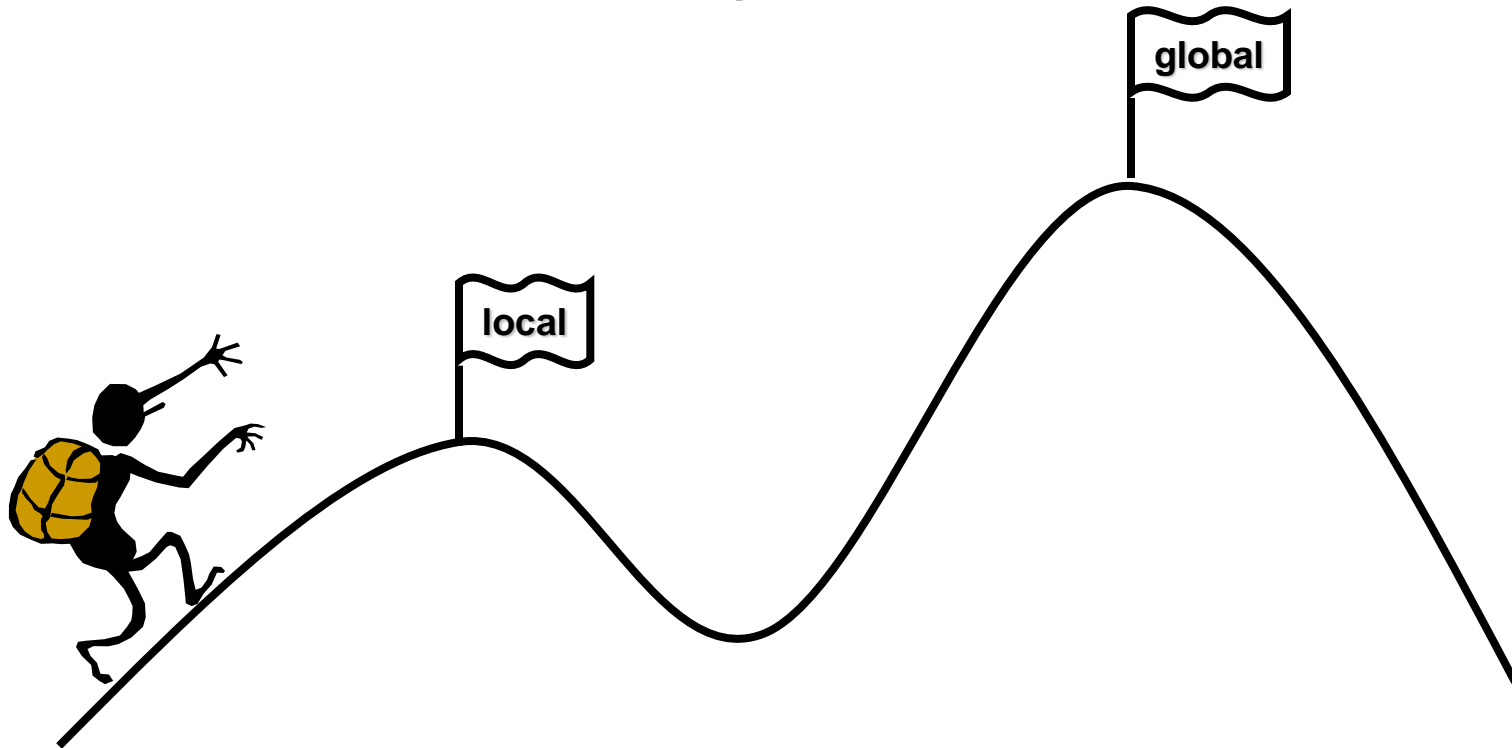


CROSSOVER is the fundamental mechanism of genetic rearrangement for both real organisms and genetic algorithms.

Chromosomes line up and then swap the portions of their genetic code beyond the crossover point.

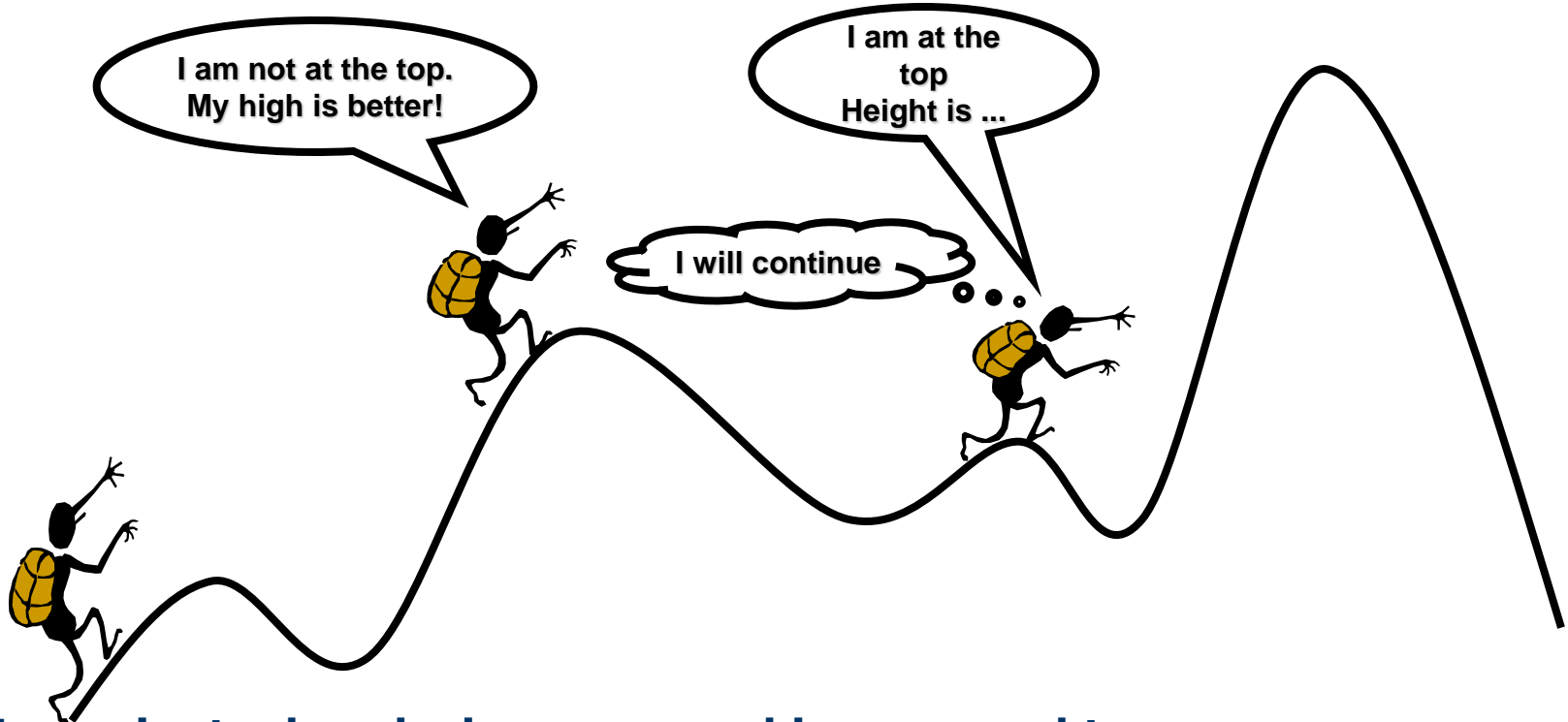
3. Metaheurísticas: Taxonomía

- Basadas en trayectorias



3. Metaheurísticas: Clasificación

- 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. Metaheurísticas: Clasificación

- 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.

Revisión de propuestas de Metaheurísticas

Published: 05 July 2020

Comprehensive Taxonomies of Nature- and Bio-inspired Optimization: Inspiration Versus Algorithmic Behavior, Critical Analysis Recommendations

[Daniel Molina](#) , [Javier Poyatos](#), [Javier Del Ser](#), [Salvador García](#), [Amir Hussain](#) & [Francisco Herrera](#)

[Cognitive Computation](#) **12**, 897–939(2020) | [Cite this article](#)

6 Citations | 0 Altmetric | [Metrics](#)



We gratefully acknowledge support from the
In

arXiv > cs > arXiv:2002.08136

Search...

Help | Ad

Computer Science > Artificial Intelligence

[Submitted on 19 Feb 2020 (v1), last revised 17 Apr 2024 (this version, v5)]

Comprehensive Taxonomies of Nature- and Bio-inspired Optimization: Inspiration versus Algorithmic Behavior, Critical Analysis and Recommendations (from 2020 to 2024)

[Daniel Molina](#), [Javier Poyatos](#), [Javier Del Ser](#), [Salvador García](#), [Amir Hussain](#), [Francisco Herrera](#)

In recent years, bio-inspired optimization methods, which mimic biological processes to solve complex problems, have gained popularity in recent literature. The proliferation of proposals prove the growing interest in this field. The increase in nature- and bio-inspired algorithms, applications, and guidelines highlights growing interest in this field. However, the exponential rise in the number of bio-inspired algorithms poses a challenge to the future trajectory of this research domain. Along the five versions of this document, the number of approaches grows incessantly, and where having a new biological description takes precedence over real problem-solving. This document presents two comprehensive taxonomies. One based on principles of biological similarity, and the other one based on operational aspects associated with the iteration of population models that initially have a biological inspiration. Therefore, these taxonomies enable researchers to categorize existing algorithmic developments into well-defined classes, considering two criteria: the source of inspiration, and the behavior exhibited by each algorithm. Using these taxonomies, we classify 518 algorithms based on nature-inspired and bio-inspired principles. Each algorithm within these categories is thoroughly examined, allowing for a critical synthesis of design trends and similarities, and identifying the most analogous classical algorithm for each proposal. From our analysis, we conclude that a poor relationship is often found between the natural inspiration of an algorithm and its behavior. Furthermore, similarities in terms of behavior between different algorithms are greater than what is claimed in their public disclosure: specifically, we show that more than one-fourth of the reviewed solvers are versions of classical algorithms. The conclusions from the analysis of the algorithms lead to several learned lessons.

METAHEURÍSTICAS

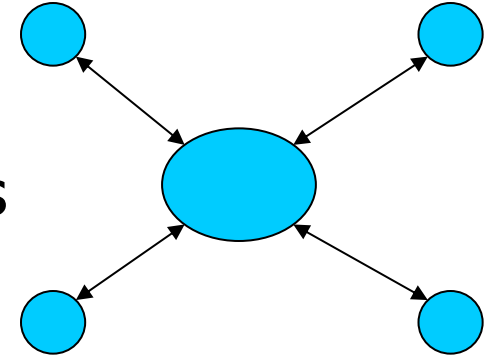
TEMA 1. Introducción a las Metaheurísticas

1. Resolución de problemas mediante algoritmos de búsqueda
3. Metaheurísticas: definición y clasificación
4. Metaheurísticas: Paralelización
5. Aplicaciones

4. Metaheurísticas: 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



METAHEURÍSTICAS

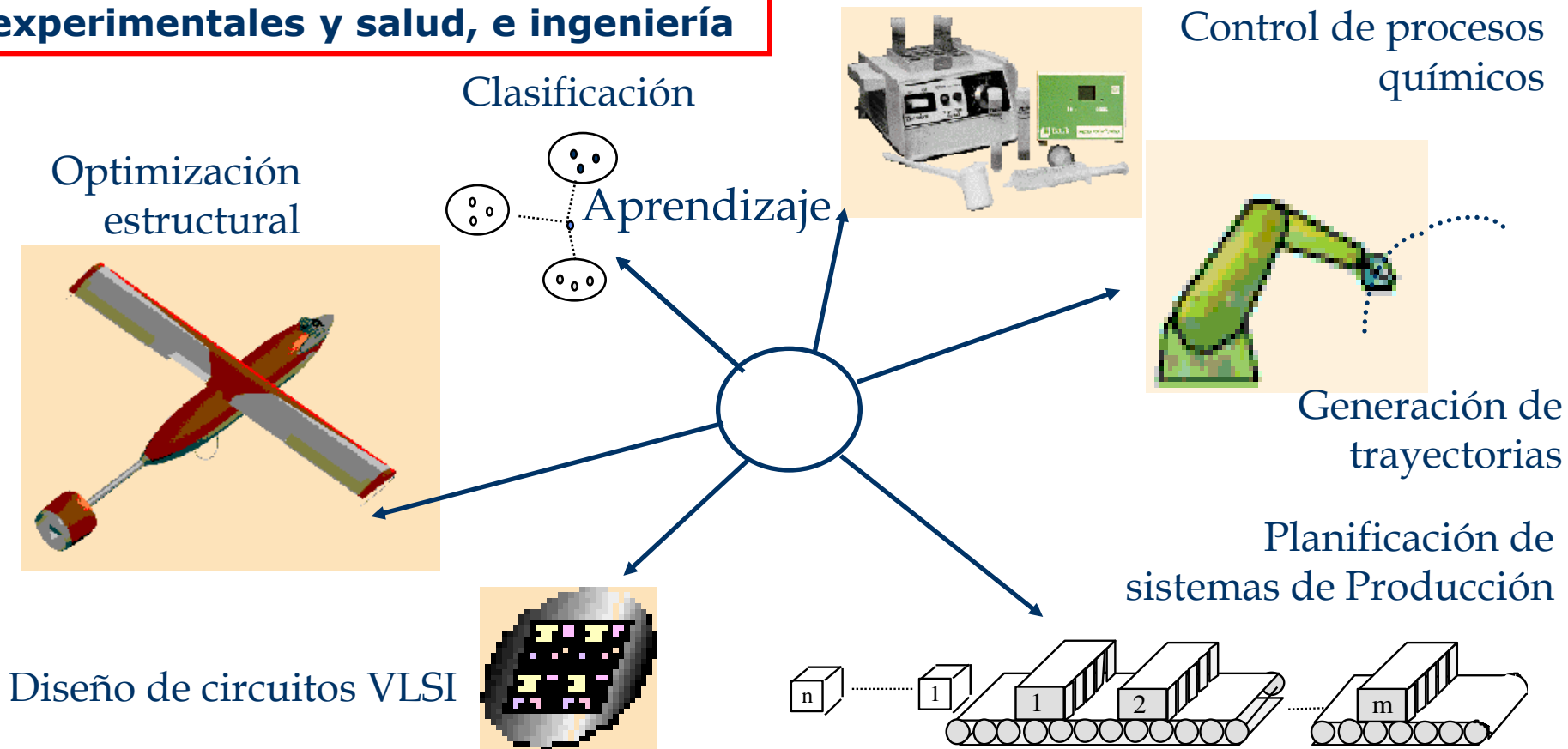
TEMA 1. Introducción a las Metaheurísticas

1. Resolución de problemas mediante algoritmos de búsqueda
3. Metaheurísticas: definición y clasificación
4. Metaheurísticas: Paralelización
5. Aplicaciones

N. Xiong, D. Molina, M. Leon-Ortiz, F. Herrera. A walk into Metaheuristics for Engineering Optimization: Principles, Methods and Recent Trends. International Journal of Computational Intelligent Systems (IJCIS), 8, 2015, 606-636.

5. Metaheurísticas: Aplicaciones

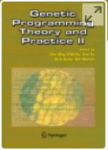
Múltiples aplicaciones en todos los ámbitos de las ciencias experimentales y salud, e ingeniería



Ejemplo Real: Diseño de antenas

Evolved antenna

https://en.wikipedia.org/wiki/Evolved_antenna



Genetic Programming Theory and Practice II pp 301–315 | [Cite as](#)

[Home](#) > [Genetic Programming Theory and Practice II](#) > [Chapter](#)

An Evolved Antenna for Deployment on Nasa's Space Technology 5 Mission

[Jason D. Lohn](#), [Gregory S. Hornby](#) & [Derek S. Linden](#)

Chapter

1515 Accesses | 29 Citations | 1 Altmetric

Part of the [Genetic Programming](#) book series (GPEM, volume 8)



The 2006 NASA [ST5](#) spacecraft antenna. This complicated shape was found by an evolutionary computer design program to create the best radiation pattern.

Ejemplo Real: Organización de equipos médicos

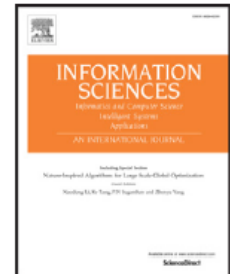
Information Sciences 326 (2016) 215–226



Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins



An alternative artificial bee colony algorithm with destructive–constructive neighbourhood operator for the problem of composing medical crews



José A. Delgado-Osuna^a, Manuel Lozano^b, Carlos García-Martínez^{c,*}

^a Provincial TIC Subdirectorate of Córdoba, U.H. Reina Sofía, Andalusian Health Service, Córdoba 14004, Spain

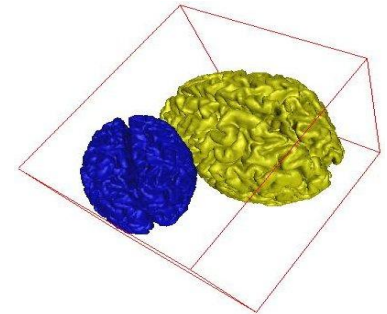
^b Department of Computer Science and Artificial Intelligence, University of Granada, Granada 18071, Spain

^c Computing and Numerical Analysis Department, University of Córdoba, Córdoba 14071, Spain

Proyecto grupo
investigación SCI²S

Ejemplo Real: Registrado de imágenes. Aplicación a la Superposición craneofacial

Algoritmos Genéticos para Registrado de Imágenes



Ejemplo Real: Registrado de imágenes. Aplicación a la Superposición craniofacial

Búsqueda de la mejor superposición
(Algoritmo Evolutivo)

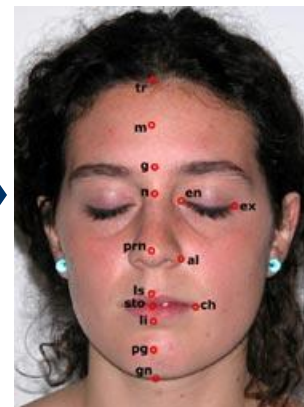
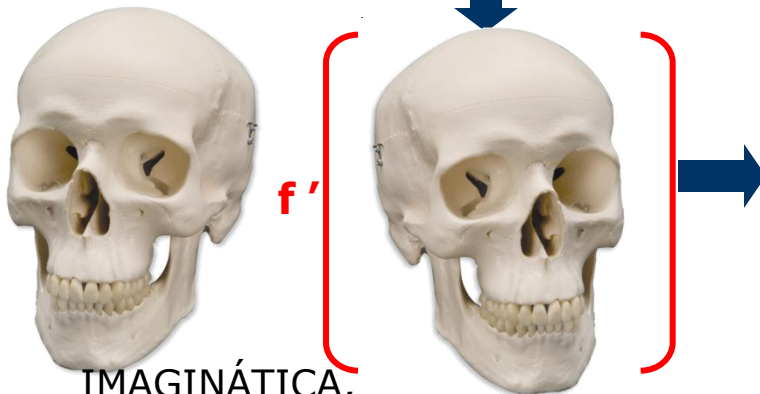
Error de Registrado

$$f' \cong f^*$$

Evaluación f'

Rotación = $\{60^\circ, (0, 1, 0)\}$
Traslación = $\{2, 0, 1\}...$

Medir la distancia
entre cada par de
puntos de referencia

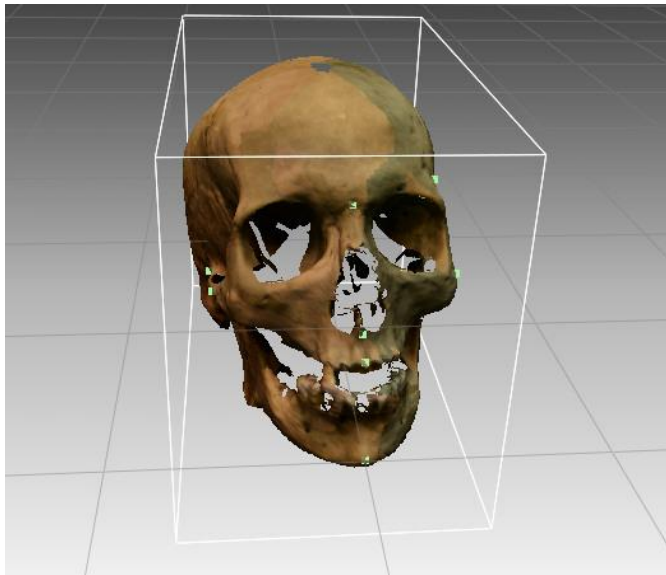


IMAGINÁTICA,
Sevilla, 4 de Marzo
de 2009

Proyecto grupo
investigación SCI²S

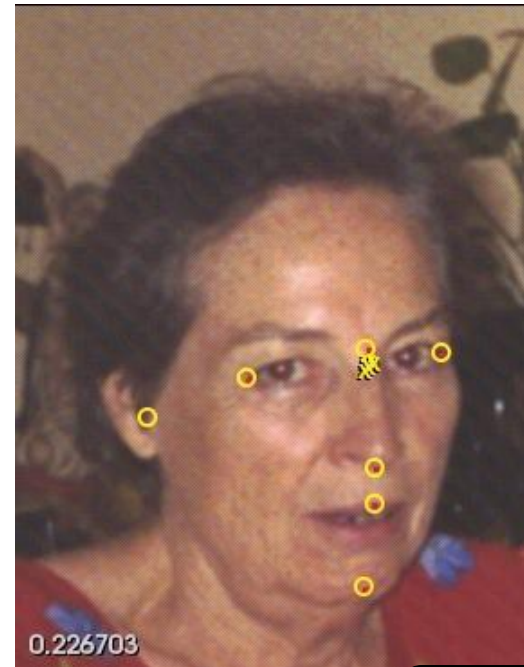
Ejemplo Real: Registrado de imágenes. Aplicación a la Superposición craniofacial

- **Caso real estudiado en el Lab. de Antropología Física de la Universidad de Granada**



Ejemplo Real: Registrado de imágenes. Aplicación a la Superposición craniofacial

- **Resultados iniciales, usando métodos que no aprovechan la potencia del Soft Computing:**



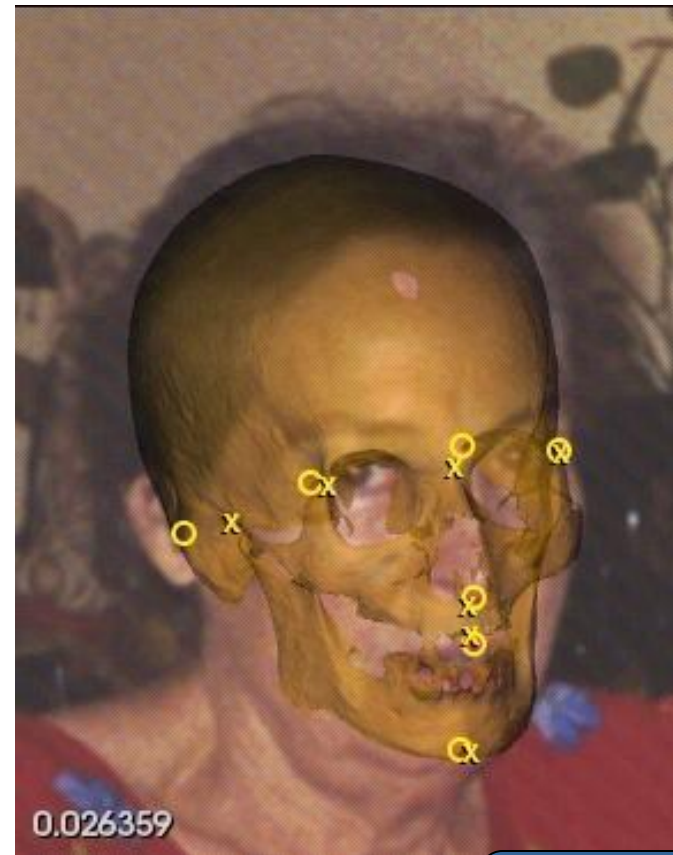
Ejemplo Real: Registrado de imágenes. Aplicación a la Superposición craniofacial

Superposición manual



24 horas

Superposición automática



25 segundos

**Proyecto grupo
investigación SCI²S**

METAHEURÍSTICAS

TEMA 1. Introducción a las Metaheurísticas

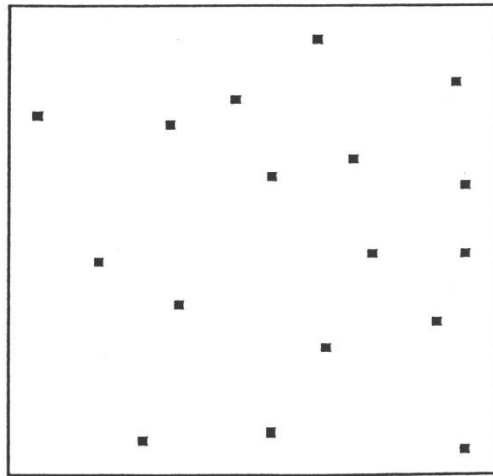
1. Resolución de problemas mediante algoritmos de búsqueda
3. Metaheurísticas: definición y clasificación
4. Metaheurísticas: Paralelización
- 5. Aplicaciones. Ejemplo de uso en el Viajante de Comercio**

3. Metaheurísticas: Ej. (Alg. Basado en Poblaciones)

Ejemplo: El problema del viajante de comercio

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	0	12.87	19.71	31.56	22.70	17.26	23.33	12.16	24.71	34.51	12.58	21.38	42.37	27.43	36.51	19.10	1.18
2	0	15.80	37.51	21.52	28.57	35.43	22.70	16.78	28.57	11.13	25.26	50.62	38.16	35.97	9.04	34.56	
3	0	50.18	36.56	35.86	35.51	21.60	31.50	43.51	25.58	38.78	61.57	46.15	51.10	23.50	48.52		
4	0	20.90	21.52	37.62	38.14	33.26	31.90	27.13	13.03	15.53	18.39	19.37	35.84	8.12			
5	0	26.00	40.72	33.74	12.87	14.71	11.68	9.72	35.86	30.96	15.06	16.78	15.27				
6	0	16.99	18.53	34.51	40.20	22.34	18.53	27.70	10.80	34.94	32.08	25.24					
7	0	14.54	46.60	54.54	33.80	34.52	40.35	22.09	51.20	41.84	41.73						
8	0	36.31	46.12	24.21	30.50	45.72	28.09	46.77	30.20	39.71							
9	0	12.54	13.31	21.52	48.18	41.50	23.85	8.50	27.43								
10	0	22.43	23.33	46.67	44.80	16.31	20.53	24.58									
11	0	14.71	40.81	30.52	26.21	10.50	23.93										
12	0	27.43	21.97	17.20	23.35	10.35											
13	0	18.89	32.78	50.15	22.59												
14	0	35.88	40.51	24.71													
15	0	30.18	11.90														
16	0	31.31															
17	0																

Ejemplo: 17 ciudades

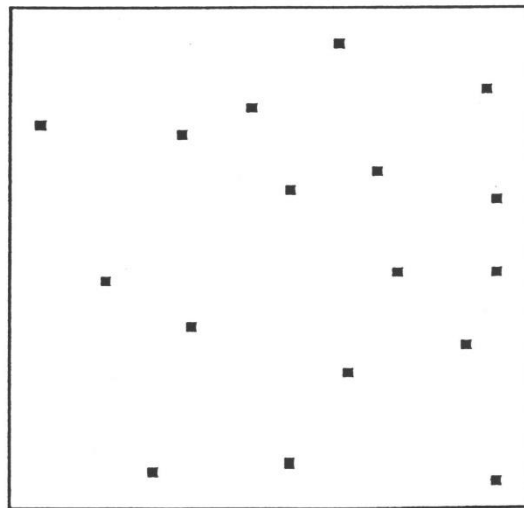


Representación de orden

(3 5 1 13 6 15 8 2 17 11 14 4 7 9 10 12 16)

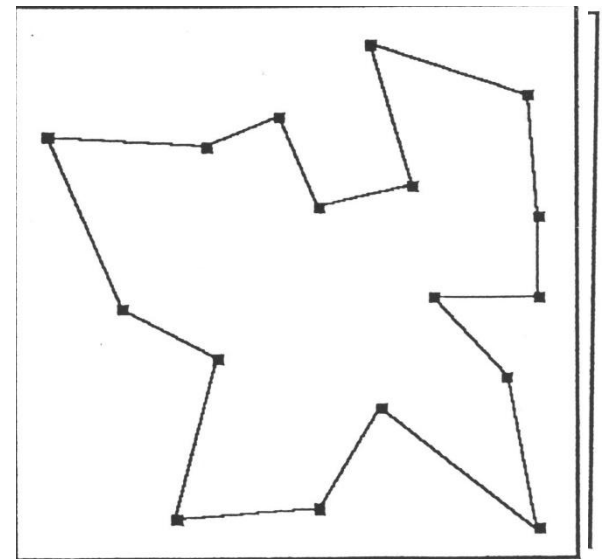
Ejemplo: El problema del viajante de comercio

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	0	12.87	19.71	31.56	22.70	17.26	23.33	12.16	24.71	34.51	12.58	21.38	42.37	27.43	36.51	19.10	1.18
2	0	15.80	37.51	21.52	28.57	35.43	22.70	16.78	28.57	11.13	25.26	50.62	38.16	35.97	9.04	34.56	
3	0	50.18	36.56	35.86	35.51	21.60	31.50	43.51	25.58	38.78	61.57	46.15	51.10	23.50	48.52		
4	0	20.90	21.52	37.62	38.14	33.26	31.90	27.13	13.03	15.53	18.39	19.37	35.84	8.12			
5	0	26.00	40.72	33.74	12.87	14.71	11.68	9.72	35.86	30.96	15.06	16.78	15.27				
6	0	16.99	18.53	34.51	40.20	22.34	18.53	27.70	10.80	34.94	32.08	25.24					
7	0	14.54	46.60	54.54	33.80	34.52	40.35	22.09	51.20	41.84	41.73						
8	0	36.31	46.12	24.21	30.50	45.72	28.09	46.77	30.20	39.71							
9	0	12.54	13.31	21.52	48.18	41.50	23.85	8.50	27.43								
10	0	22.43	23.33	46.67	44.80	16.31	20.53	24.58									
11	0	14.71	40.81	30.52	26.21	10.50	23.93										
12	0	27.43	21.97	17.20	23.35	10.35											
13	0	18.89	32.78	50.15	22.59												
14	0	35.88	40.51	24.71													
15	0	30.18	11.90														
16	0	31.31															
17	0																

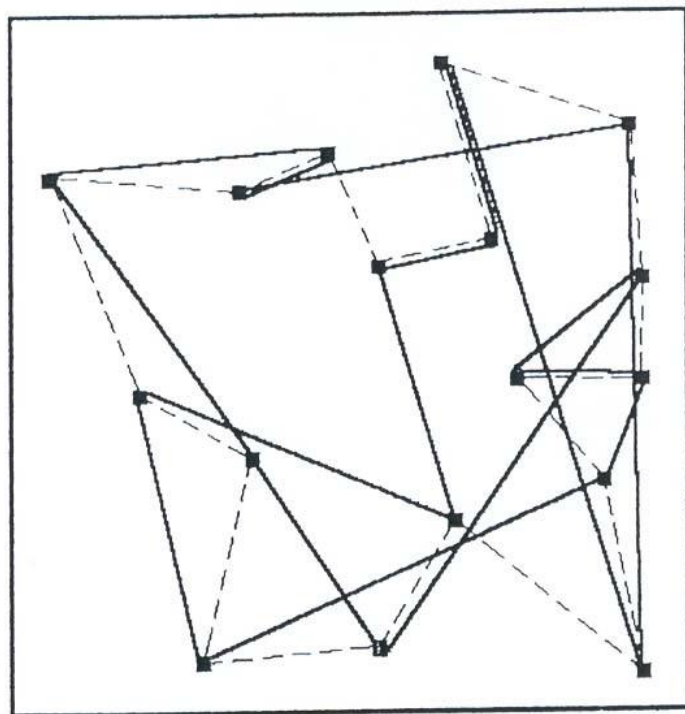


**17! (3.5568734e14)
soluciones posibles**

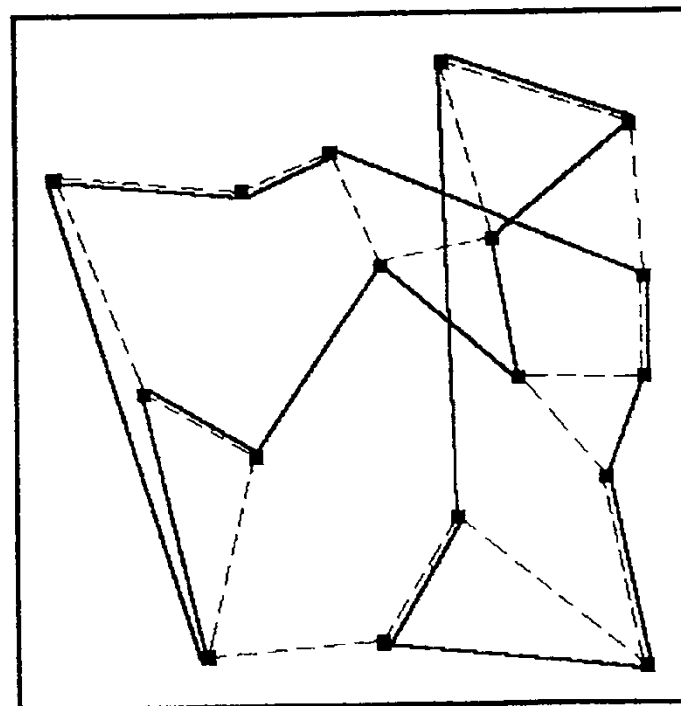
**Solución óptima:
Coste=226.64**



Viajante de Comercio



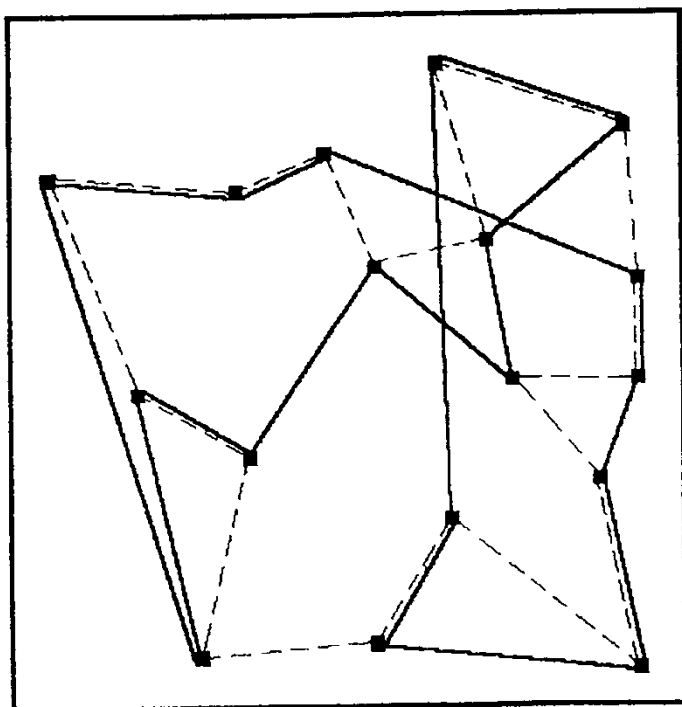
Iteración: 0 Costo: 403.7



Iteración: 25 Costo: 303.86

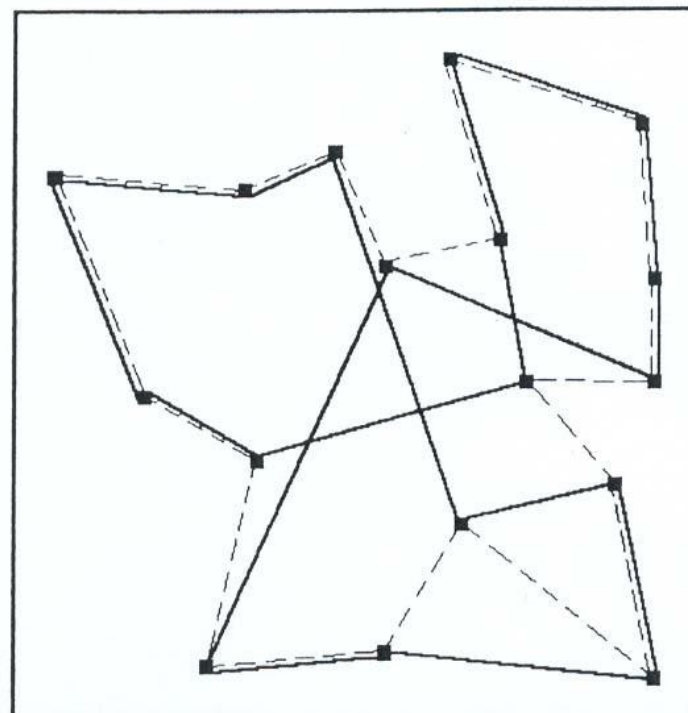
Solución óptima: 226.64

Viajante de Comercio



— Mejor solución
- - - Solución optimal

Iteración: 25 Costo: 303.86

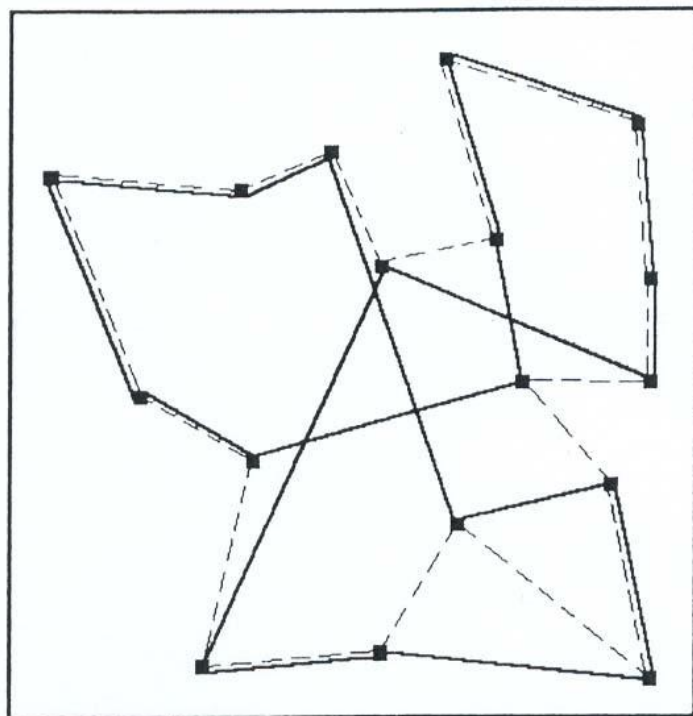


— Mejor solución
- - - Solución optimal

Iteración: 50 Costo: 293.6

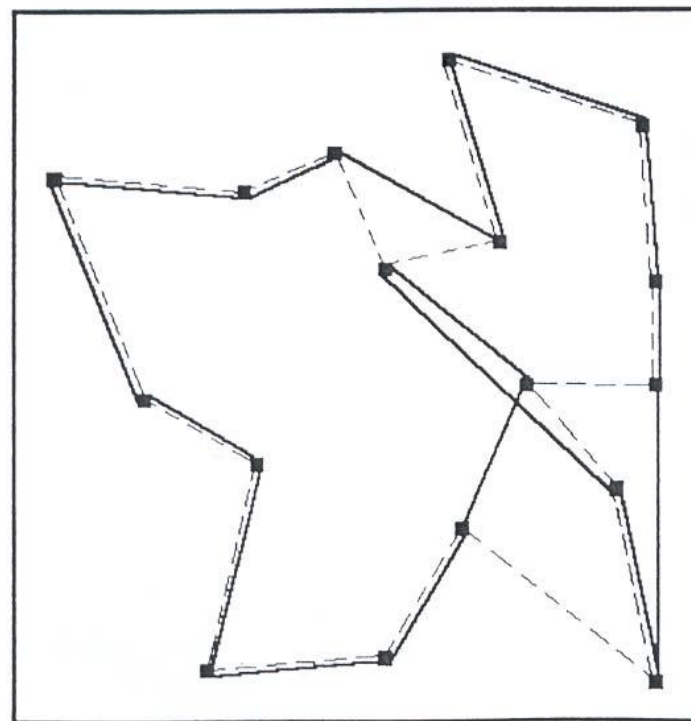
Solución óptima: 226.64

Viajante de Comercio



— Mejor solución
- - - Solución optimal

Iteración: 50 Costo: 293.6

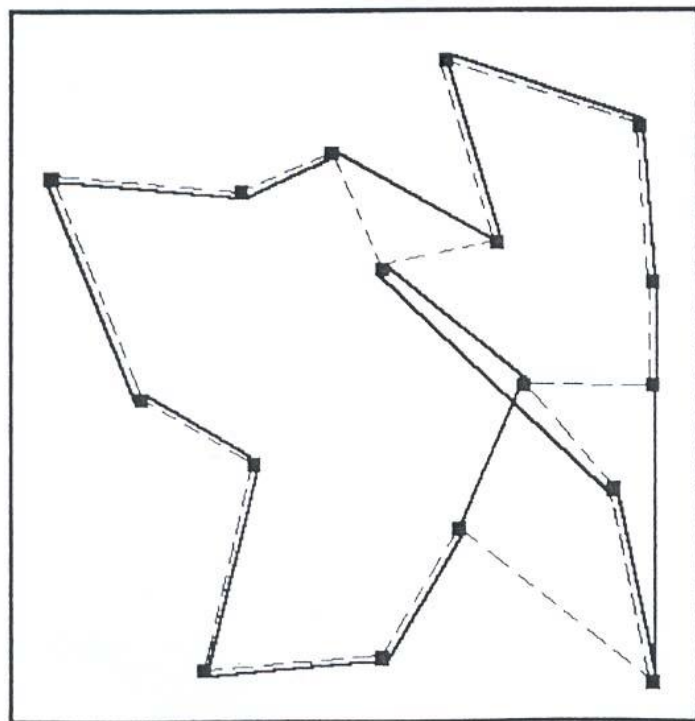


— Mejor solución
- - - Solución optimal

Iteración: 100 Costo: 256.55

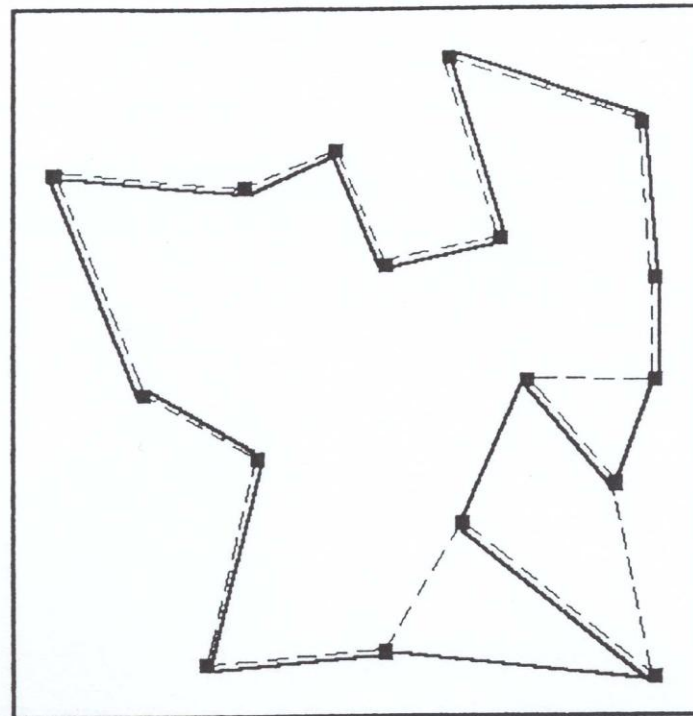
Solución óptima: 226.64

Viajante de Comercio



— Mejor solución
- - - Solución optimal

Iteración: 100 Costo: 256.55

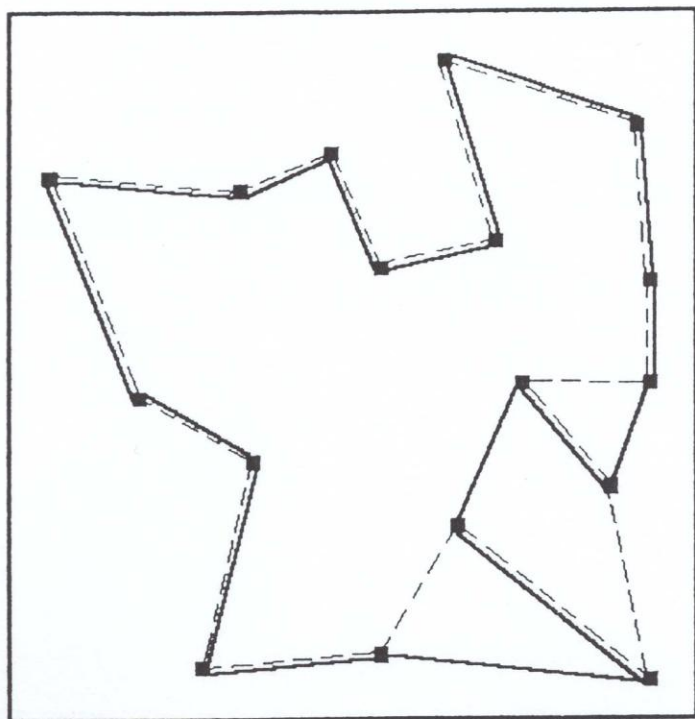


— Mejor solución
- - - Solución optimal

Iteración: 200 Costo: 231.4

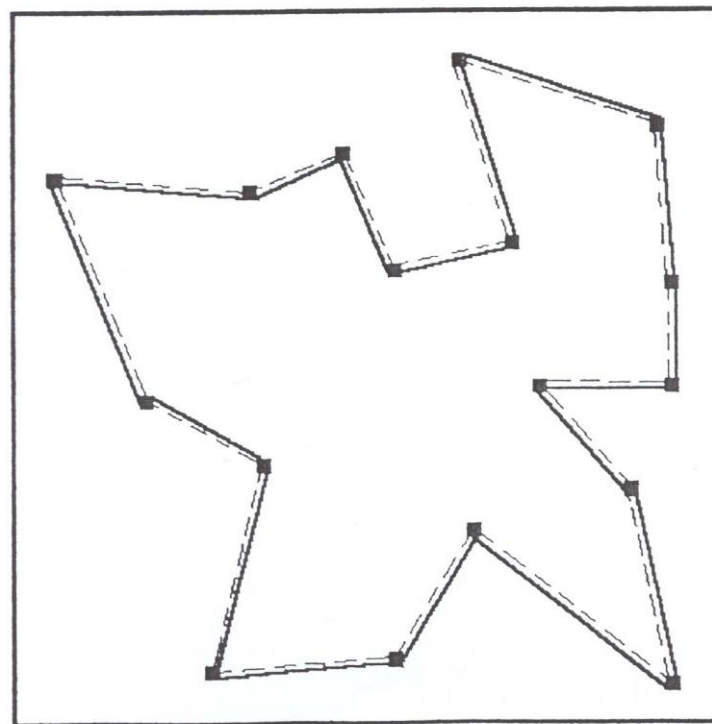
Solución óptima: 226.64

Viajante de Comercio



— Mejor solución
- - - Solución optimal

Iteración: 200 Costo: 231.4

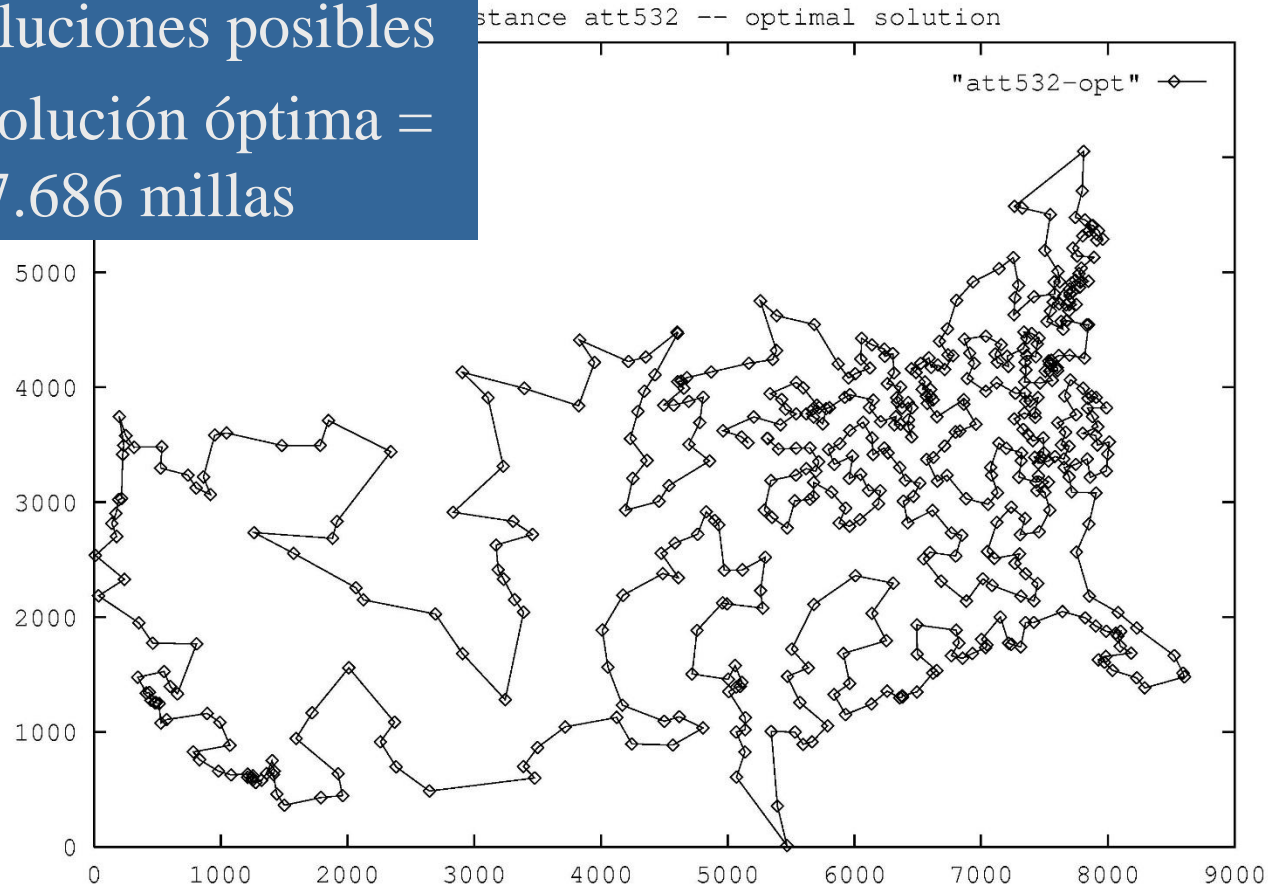


— Mejor solución
- - - Solución optimal

Iteración: 250 Solución
óptima: 226.64

Ejemplo: El problema del viajante de comercio

532! soluciones posibles
Coste solución óptima =
27.686 millas



METAHEURÍSTICAS

2024-2025



- Tema 1. Introducción a las Metaheurísticas
- Tema 2. Modelos de Búsqueda: Entornos y Trayectorias vs Poblaciones
- Tema 3. Metaheurísticas Basadas en Poblaciones
- Tema 4: Algoritmos Meméticos
- Tema 5. Metaheurísticas Basadas en Trayectorias
- Tema 6. Metaheurísticas Basadas en Adaptación Social
- Tema 7. Aspectos Avanzados en Metaheurísticas
- Tema 8. Metaheurísticas Paralelas
- Tema 9. Modelos de IA Evolutivos. Aprendizaje Evolutivo