



# Cos'hanno in comune una formica e un navigatore?

Gli algoritmi ispirati a metafore biologiche o naturali

**Francesca Da Ros, Luca Di Gaspero**

**IOLab – <https://iolab.uniud.it>**

**Dipartimento Politecnico di Ingegneria e Architettura**

**Università degli Studi di Udine**





- La natura è un **laboratorio di innovazione**



- La natura è un **laboratorio di innovazione**
- In natura si sono evoluti **metodi** e **soluzioni efficienti** per affrontare problemi



- La natura è un **laboratorio di innovazione**
- In natura si sono evoluti **metodi** e **soluzioni efficienti** per affrontare problemi
  - Gli **unici sistemi intelligenti** su questo pianeta sono biologici



- La natura è un **laboratorio di innovazione**
- In natura si sono evoluti **metodi** e **soluzioni efficienti** per affrontare problemi
  - Gli **unici sistemi intelligenti** su questo pianeta sono biologici
  - Sono stati **progettati** dai processi di evoluzione naturale, come adattamento al proprio ambiente



- La natura è un **laboratorio di innovazione**
- In natura si sono evoluti **metodi** e **soluzioni efficienti** per affrontare problemi
  - Gli **unici sistemi intelligenti** su questo pianeta sono biologici
  - Sono stati **progettati** dai processi di evoluzione naturale, come adattamento al proprio ambiente
  - **Non usano** gli strumenti tipici del problem solving: **logica**, **matematica**, una sofisticata **pianificazione**, o **modelli** complicati del loro ambiente



- La natura è un **laboratorio di innovazione**
- In natura si sono evoluti **metodi** e **soluzioni efficienti** per affrontare problemi
  - Gli **unici sistemi intelligenti** su questo pianeta sono biologici
  - Sono stati **progettati** dai processi di evoluzione naturale, come adattamento al proprio ambiente
  - **Non usano** gli strumenti tipici del problem solving: **logica**, **matematica**, una sofisticata **pianificazione**, o **modelli** complicati del loro ambiente
  - Sono basati sulla **condivisione** e lo **scambio** di informazioni

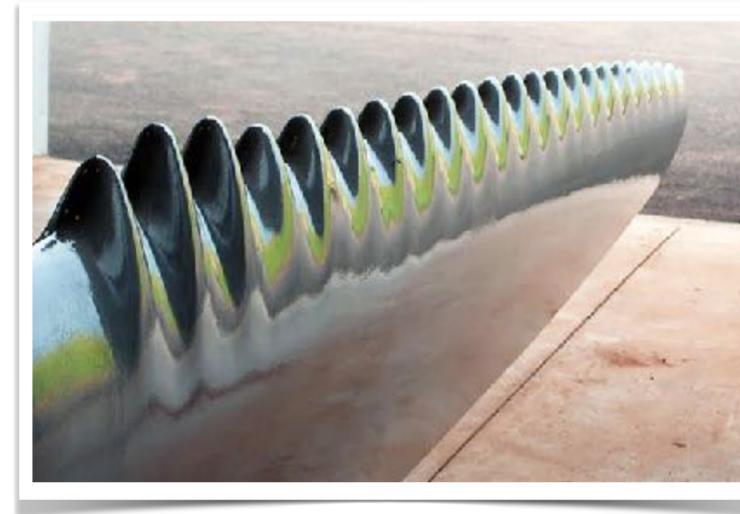


- La natura è un **laboratorio di innovazione**
- In natura si sono evoluti **metodi** e **soluzioni efficienti** per affrontare problemi
  - Gli **unici sistemi intelligenti** su questo pianeta sono biologici
  - Sono stati **progettati** dai processi di evoluzione naturale, come adattamento al proprio ambiente
  - **Non usano** gli strumenti tipici del problem solving: **logica**, **matematica**, una sofisticata **pianificazione**, o **modelli** complicati del loro ambiente
  - Sono basati sulla **condivisione** e lo **scambio** di informazioni
  - Sono in grado di **svolgere agevolmente** dei **compiti** di elaborazione dell'informazione e di calcolo che l'intelligenza artificiale attualmente non è in grado di affrontare

# **Imparare dalla natura: alcuni esempi**

# Imparare dalla natura: alcuni esempi

Il progetto aerodinamico del muso dello Shinkansen è ispirato dal becco del Martin Pescatore (**Popular Mechanics**)



Il progetto delle pale eoliche di nuova generazione è ispirato dalle pinne delle balene, e consente di guadagnare il 40% in efficienza (**Popular Mechanics**)

# LA SETTIMANA ENIGMISTICA

---

Una gita a...



# LA SETTIMANA ENIGMISTICA

Una gita a...



- È una **città** del Nord-Europa
- Il nome della sua squadra di calcio è quello di un **eroe greco**
- I suoi abitanti si spostano quasi esclusivamente in **bicicletta**
- È chiamata **Venezia del Nord**

# LA SETTIMANA ENIGMISTICA

Una gita a...

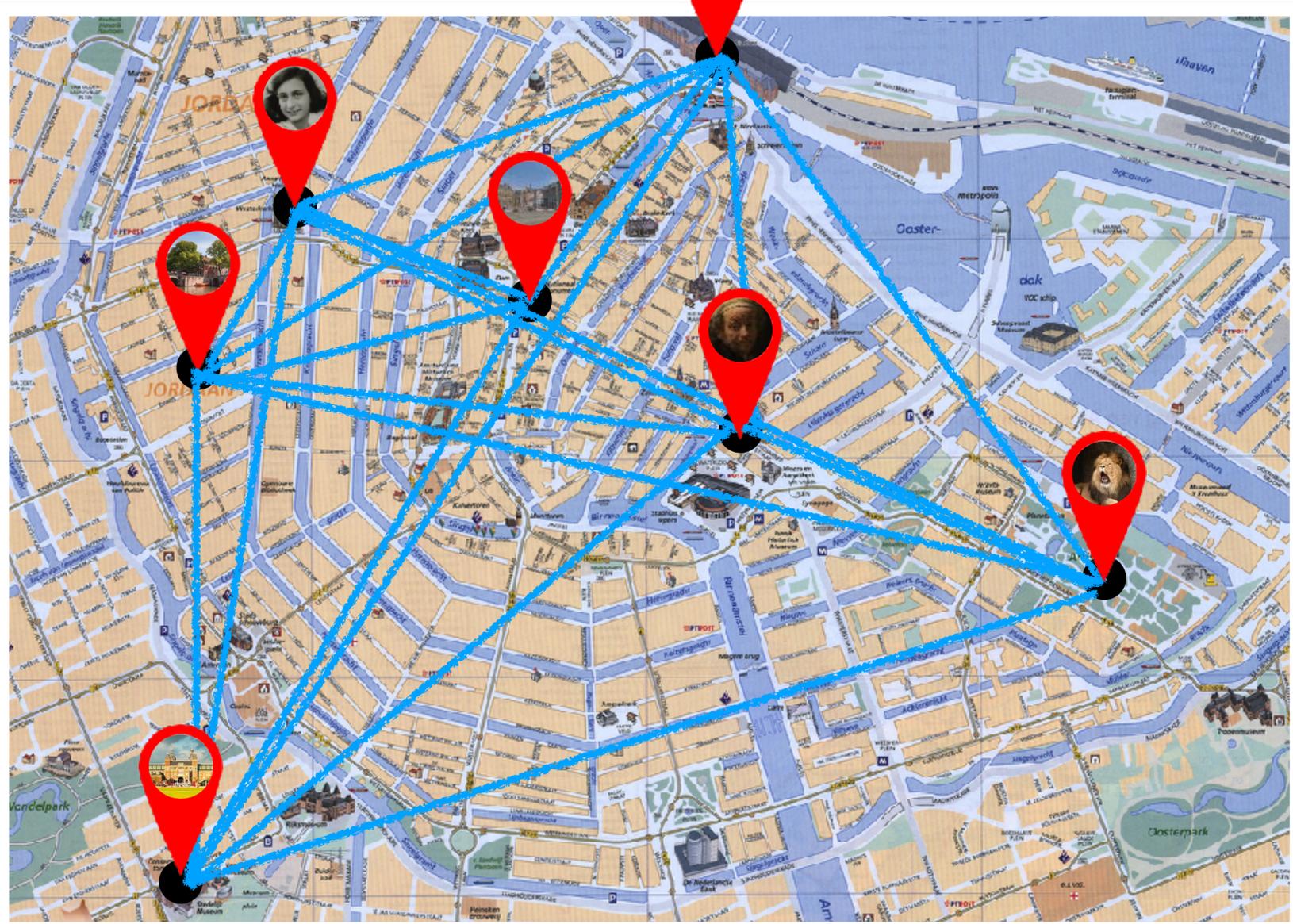


- È una città del Nord-Europa
- Il nome della sua squadra di calcio è quello di un eroe greco
- I suoi abitanti si spostano quasi esclusivamente in bicicletta
- È chiamata Venezia del Nord

# Una gita ad Amsterdam



# Una gita ad Amsterdam



**ottimizzazióne** s. f. [der. di *ottimizzare*]. **1.** Il raggiungimento di una posizione di ottimo, ossia del massimo risultato possibile con i termini dati o in relazione a un determinato fine: *gli sforzi dell'amministrazione per l'o. dei pubblici servizi*. Per estens., l'impiego di una macchina al meglio delle sue capacità tecniche. **2.** In matematica applicata, e in partic. nella teoria delle decisioni, *problemi di o.*, le questioni

attinenti alla ricerca dei criteri di scelta tra diverse opzioni o di determinazione del valore di particolari parametri, di solito riconducibile alla ricerca del massimo o del minimo di funzioni che costituiscono la rappresentazione matematica del problema.

Vocabolario Treccani

$$\min F(x)$$
$$x \in \mathcal{S} \subseteq \mathbb{R}^n$$

# **Caratteristiche dei problemi di ottimizzazione**

# Caratteristiche dei problemi di ottimizzazione

- Obiettivo e vincoli sono, in generale, **non differenziabili** (derivabili)

# Caratteristiche dei problemi di ottimizzazione

- Obiettivo e vincoli sono, in generale, **non differenziabili** (derivabili)
- I vincoli possono essere **non lineari**

# Caratteristiche dei problemi di ottimizzazione

- Obiettivo e vincoli sono, in generale, **non differenziabili** (derivabili)
- I vincoli possono essere **non lineari**
- Lo spazio di ricerca può essere misto **continuo** e **discreto**

# Caratteristiche dei problemi di ottimizzazione

- Obiettivo e vincoli sono, in generale, **non differenziabili** (derivabili)
- I vincoli possono essere **non lineari**
- Lo spazio di ricerca può essere misto **continuo** e **discreto**
- La **dimensionalità** del problema può essere **elevata** (in termini di numero di variabili e vincoli)

# Caratteristiche dei problemi di ottimizzazione

- Obiettivo e vincoli sono, in generale, **non differenziabili** (derivabili)
- I vincoli possono essere **non lineari**
- Lo spazio di ricerca può essere misto **continuo** e **discreto**
- La **dimensionalità** del problema può essere **elevata** (in termini di numero di variabili e vincoli)
- La funzione obiettivo può essere **multimodale** con più di un ottimo

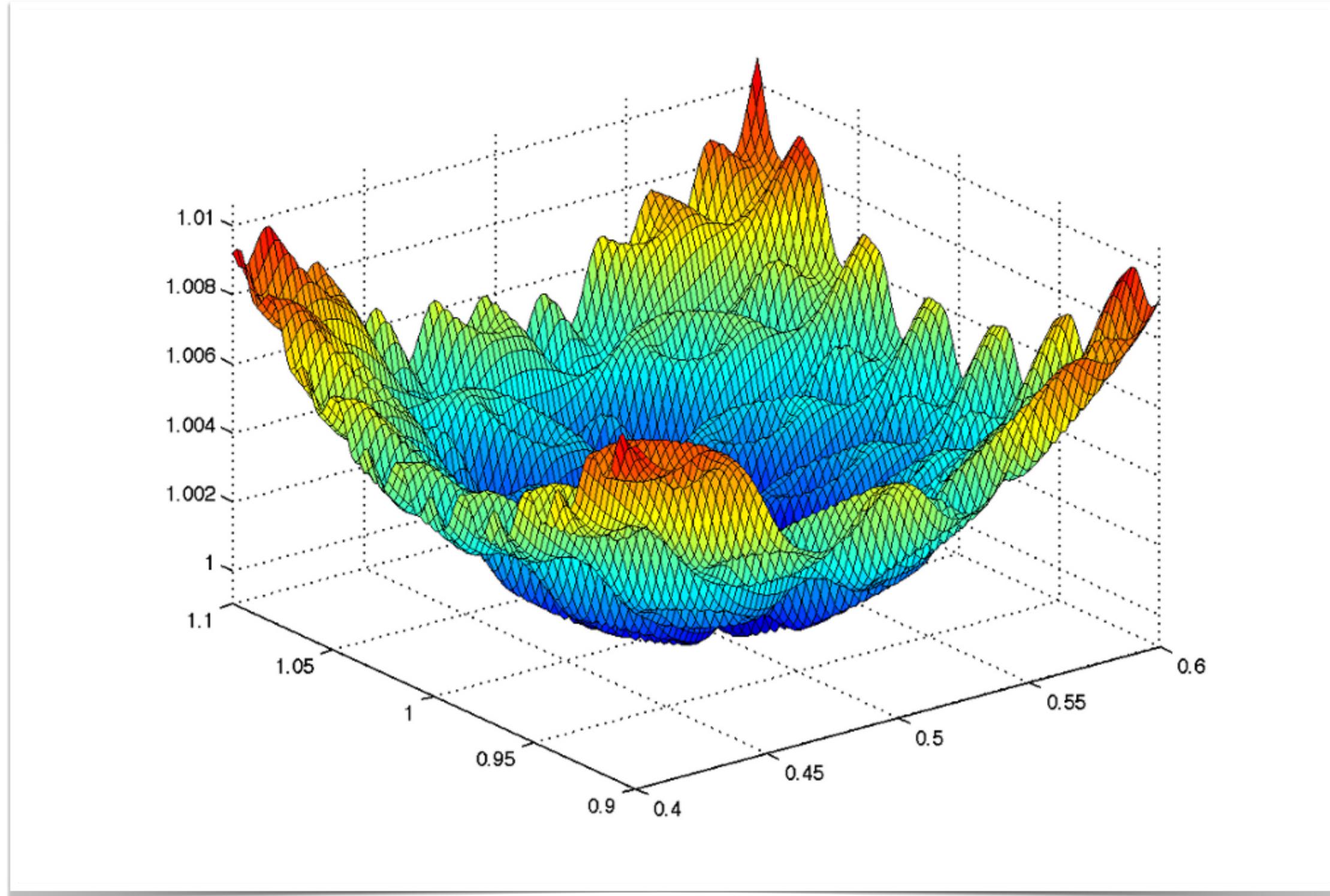
# Caratteristiche dei problemi di ottimizzazione

- Obiettivo e vincoli sono, in generale, **non differenziabili** (derivabili)
- I vincoli possono essere **non lineari**
- Lo spazio di ricerca può essere misto **continuo** e **discreto**
- La **dimensionalità** del problema può essere **elevata** (in termini di numero di variabili e vincoli)
- La funzione obiettivo può essere **multimodale** con più di un ottimo
- Il problema può essere **multiobiettivo**

# Caratteristiche dei problemi di ottimizzazione

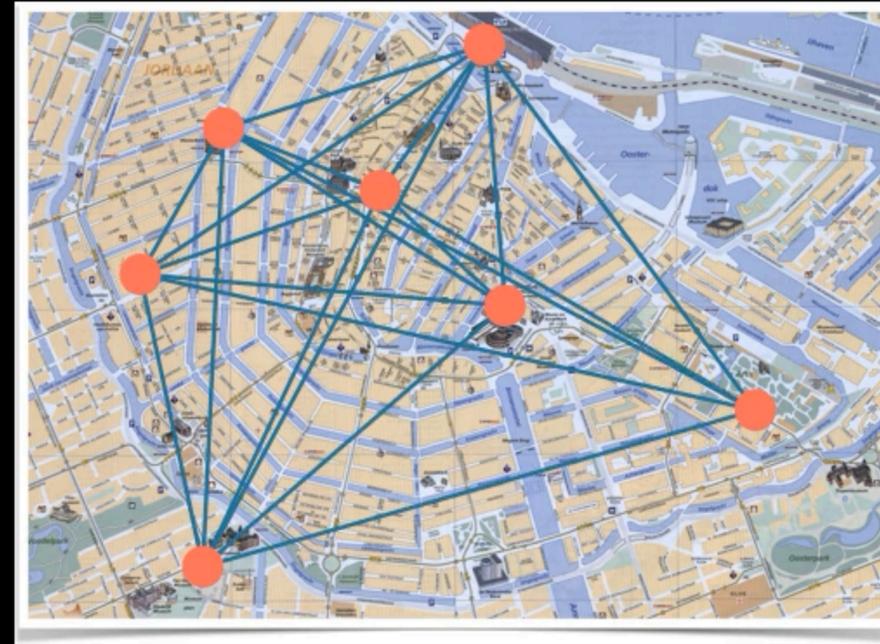
- Obiettivo e vincoli sono, in generale, **non differenziabili** (derivabili)
- I vincoli possono essere **non lineari**
- Lo spazio di ricerca può essere misto **continuo** e **discreto**
- La **dimensionalità** del problema può essere **elevata** (in termini di numero di variabili e vincoli)
- La funzione obiettivo può essere **multimodale** con più di un ottimo
- Il problema può essere **multiobiettivo**
- Il calcolo del problema in forma chiusa potrebbe avere un **elevato costo computazionale**

# Caratteristiche dei problemi di ottimizzazione



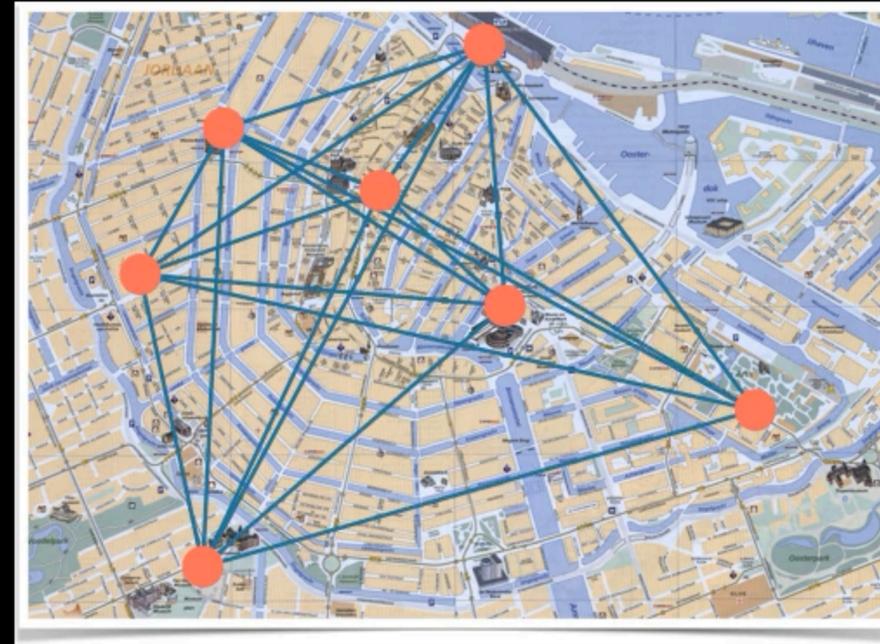
# Problemi di ottimizzazione misti

- Il **problema del commesso viaggiatore** (TSP): dato un insieme di luoghi da visitare, trovare un percorso (circuito Hamiltoniano) di distanza minima che visiti tutti i luoghi (una e una sola volta)



# Problemi di ottimizzazione misti

- Il **problema del commesso viaggiatore** (TSP): dato un insieme di luoghi da visitare, trovare un percorso (circuito Hamiltoniano) di distanza minima che visiti tutti i luoghi (una e una sola volta)



# Problemi di

- Il **problema del commesso viaggiatore** (TSP): dato un insieme di luoghi da visitare, trovare un percorso (Hamiltoniano) di distanza minima (che visiti tutti i luoghi una sola volta)

$$\min \sum_{i=1}^n x_{ij} d_{ij}$$

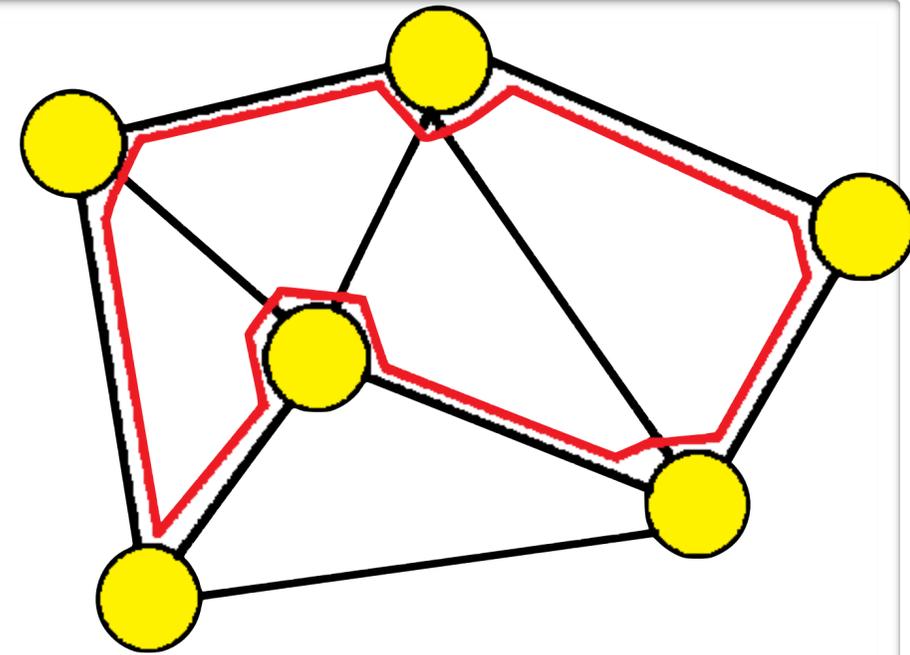
s.t.

$$\sum_{i=1}^n x_{ij} = 1 \quad \forall j \in 1, \dots, n$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \in 1, \dots, n$$

$x_{ij}$  determinano un circuito Hamiltoniano

$$x_{ij} = 0 \quad \text{oppure} \quad x_{ij} = 1 \quad \forall i, j \in 1, \dots, n$$



# Problemi di ottimizzazione misti

- Il **problema del commesso viaggiatore** (TSP): dato un insieme di luoghi da visitare, trovare un percorso (circuito Hamiltoniano) di distanza minima che visiti tutti i luoghi (una e una sola volta)

# Problemi di ottimizzazione misti

- Il **problema del commesso viaggiatore** (TSP): dato un insieme di luoghi da visitare, trovare un percorso (circuito Hamiltoniano) di distanza minima che visiti tutti i luoghi (una e una sola volta)
- Dal punto di vista teorico il problema è difficile: NP-hard
- Quanti sono i possibili percorsi che visitano  $n$  luoghi una e una sola volta?  
*Sono circa  $n!$ , il numero di permutazioni di  $n$  valori.  $7! = 5.040$ ,  $10! = 3.628.800$ ,  $15! = 1.307.674.368.000$  (mille miliardi),  $20! = 2.432.902.008.176.640.000$  (2 miliardi di miliardi)*

# Algoritmi classici

# Algoritmi classici

- Gli algoritmi classici **funzionano** bene per:

# Algoritmi classici

- Gli algoritmi classici **funzionano** bene per:
  - Number **crunching** (es. modelli di fisica nucleare)

# Algoritmi classici

- Gli algoritmi classici **funzionano** bene per:
  - Number **crunching** (es. modelli di fisica nucleare)
  - Ragionamento basato su **regole** (se-allora)

# Algoritmi classici

- Gli algoritmi classici **funzionano** bene per:
  - Number **crunching** (es. modelli di fisica nucleare)
  - Ragionamento basato su **regole** (se-allora)
  - Elaborazione di **dati strutturati** (database)

# Algoritmi classici

- Gli algoritmi classici **funzionano** bene per:
  - Number **crunching** (es. modelli di fisica nucleare)
  - Ragionamento basato su **regole** (se-allora)
  - Elaborazione di **dati strutturati** (database)
  - **Ripetizione** costante di azioni **predefinite** (automazione)

# Algoritmi classici

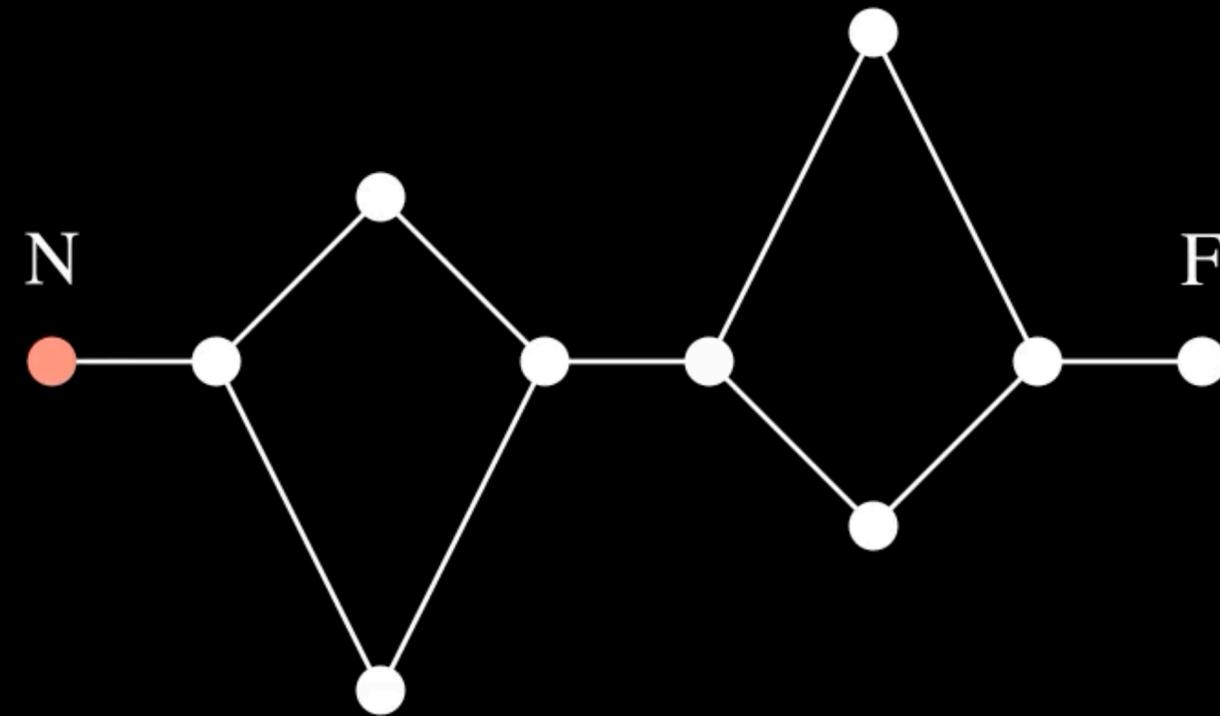
- Gli algoritmi classici **funzionano** bene per:
  - Number **crunching** (es. modelli di fisica nucleare)
  - Ragionamento basato su **regole** (se-allora)
  - Elaborazione di **dati strutturati** (database)
  - **Ripetizione** costante di azioni **predefinite** (automazione)
- Al di fuori di questi casi (che sono però rilevanti), non generalizzano così bene

# Anche le formiche hanno problemi di ottimizzazione

## Quando?

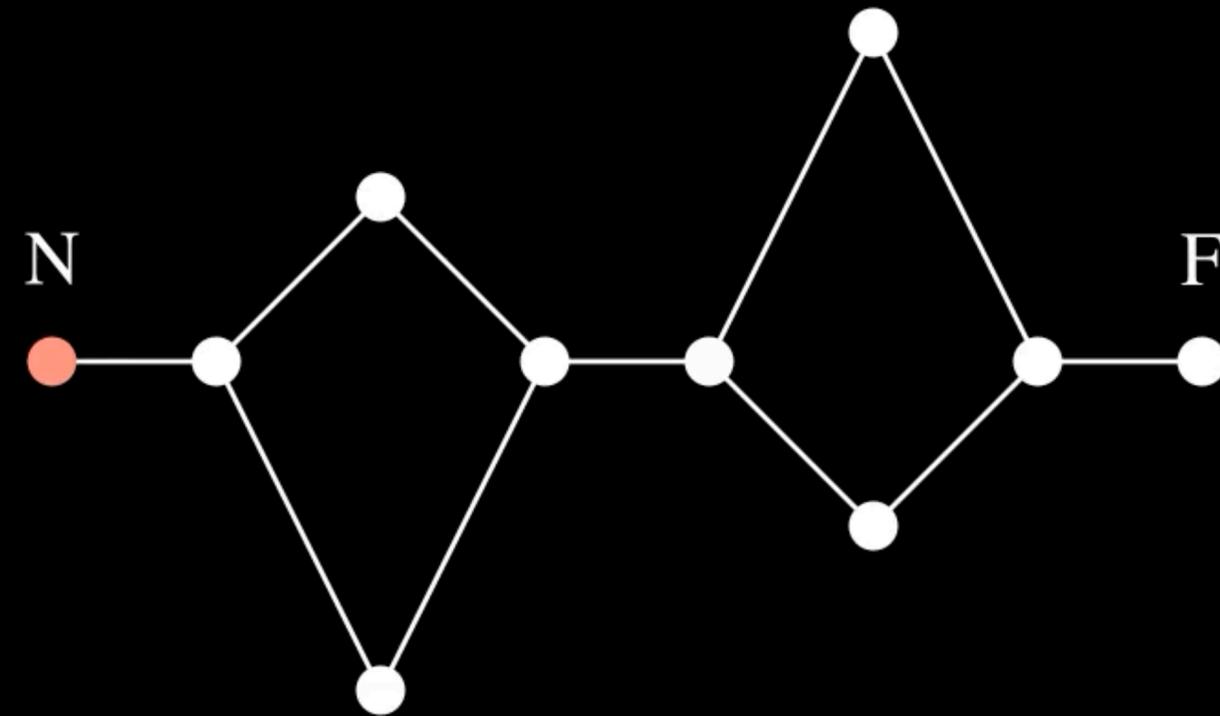


# Alla ricerca del cibo



Foraging

# Alla ricerca del cibo



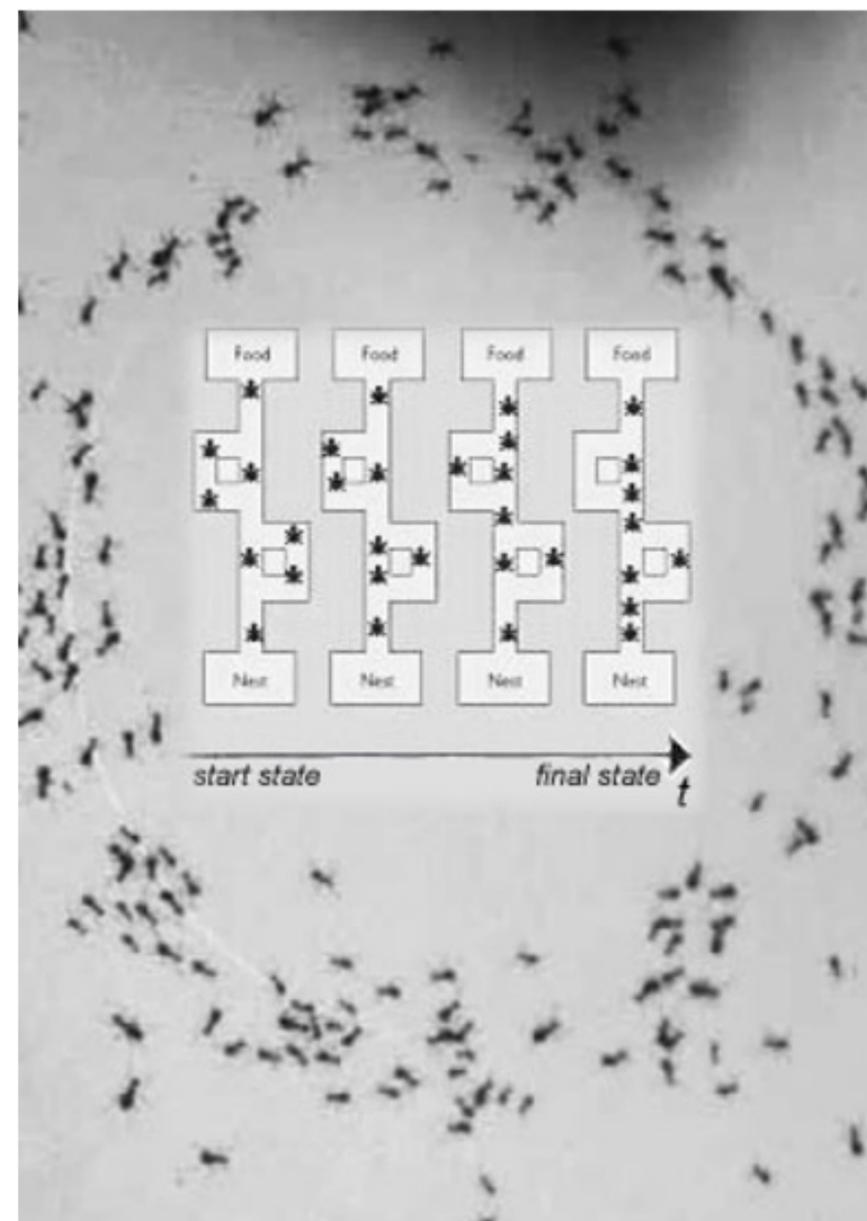
Foraging

**Come fanno a coordinarsi?**

# Come fanno a coordinarsi?

**stigmergia** s. f. – particolare forma di comunicazione indiretta impiegata dagli insetti sociali per coordinare le loro attività mediante la modifica dell'ambiente in cui operano

Lessico del XXI secolo, Treccani



The main quality of the colonies of insects, ants or bees lies in the fact that they are part of a self-organized group in which the keyword is simplicity.

Every day, ants solve complex problems due to a sum of simple interactions, which are carried out by individuals.

The ant is, for example, able to use the quickest way from the anthill to its food simply by following the way marked with pheromones.

Quella delle formiche è un'intelligenza collettiva che emerge dal comportamento di un insieme di individui (Herbert Simon Premio Nobel 1978, Allen Newall)

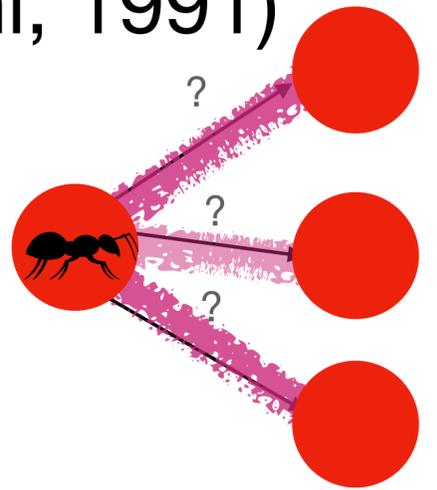
# Stigmergia artificiale

# Stigmergia artificiale

- **Distributed Optimization by Ant Colony** (Dorigo, Maniezzo e Colorni, 1991)

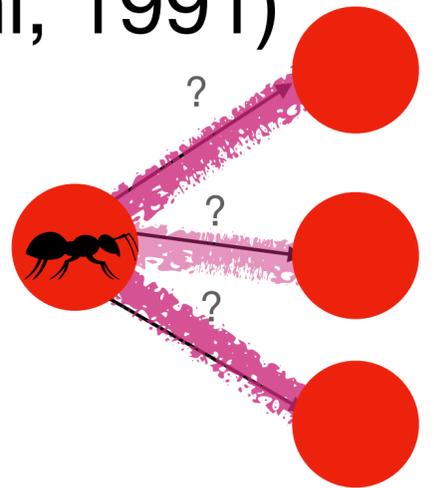
# Stigmergia artificiale

- **Distributed Optimization by Ant Colony** (Dorigo, Maniezzo e Colorni, 1991)
- Problemi modellati come **ricerca di percorsi** su grafi



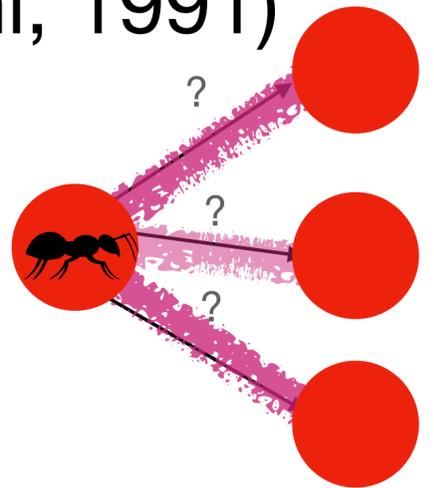
# Stigmergia artificiale

- **Distributed Optimization by Ant Colony** (Dorigo, Maniezzo e Colorni, 1991)
- Problemi modellati come **ricerca di percorsi** su grafi



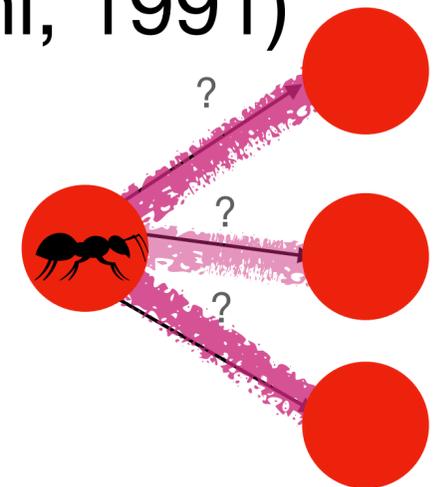
# Stigmergia artificiale

- **Distributed Optimization by Ant Colony** (Dorigo, Maniezzo e Colorni, 1991)
- Problemi modellati come **ricerca di percorsi** su grafi
- **$k$  formiche** (artificiali) sono libere di costruire il loro percorso effettuando le scelte attraverso una distribuzione di **probabilità**



# Stigmergia artificiale

- **Distributed Optimization by Ant Colony** (Dorigo, Maniezzo e Colorni, 1991)
- Problemi modellati come **ricerca di percorsi** su grafi
- **$k$  formiche** (artificiali) sono libere di costruire il loro percorso effettuando le scelte attraverso una distribuzione di **probabilità**
- la scelta non è deterministica per lasciare aperte le possibilità di **esplorazione**



# Stigmergia artificiale

# Stigmergia artificiale

- Una volta che tutte le formiche hanno completato il percorso, viene **aggiorna** il feromone

artificiale  $\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^a \Delta_{ij}^k$

Tasso di evaporazione

# Stigmergia artificiale

- Una volta che tutte le formiche hanno completato il percorso, viene **aggiorna** il feromone

artificiale  $\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^a \Delta_{ij}^k$

Tasso di evaporazione

- $\Delta_{ij}^k = \begin{cases} \text{merit}(L_k) & \text{se la formica } k \text{ ha usato l'arco } (i, j) \\ 0 & \text{altrimenti} \end{cases}$   
merit( $L_k$ ) è una misura legata alla qualità della soluzione

# Stigmergia artificiale

- Una volta che tutte le formiche hanno completato il percorso, viene **aggiorna** il feromone

artificiale  $\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^a \Delta_{ij}^k$

Tasso di evaporazione

- $\Delta_{ij}^k = \begin{cases} \text{merit}(L_k) & \text{se la formica } k \text{ ha usato l'arco } (i, j) \\ 0 & \text{altrimenti} \end{cases}$   
merit( $L_k$ ) è una misura legata alla qualità della soluzione

- Il metodo è iterativo: a ogni iterazione tutte le formiche partono per la ricerca del cibo

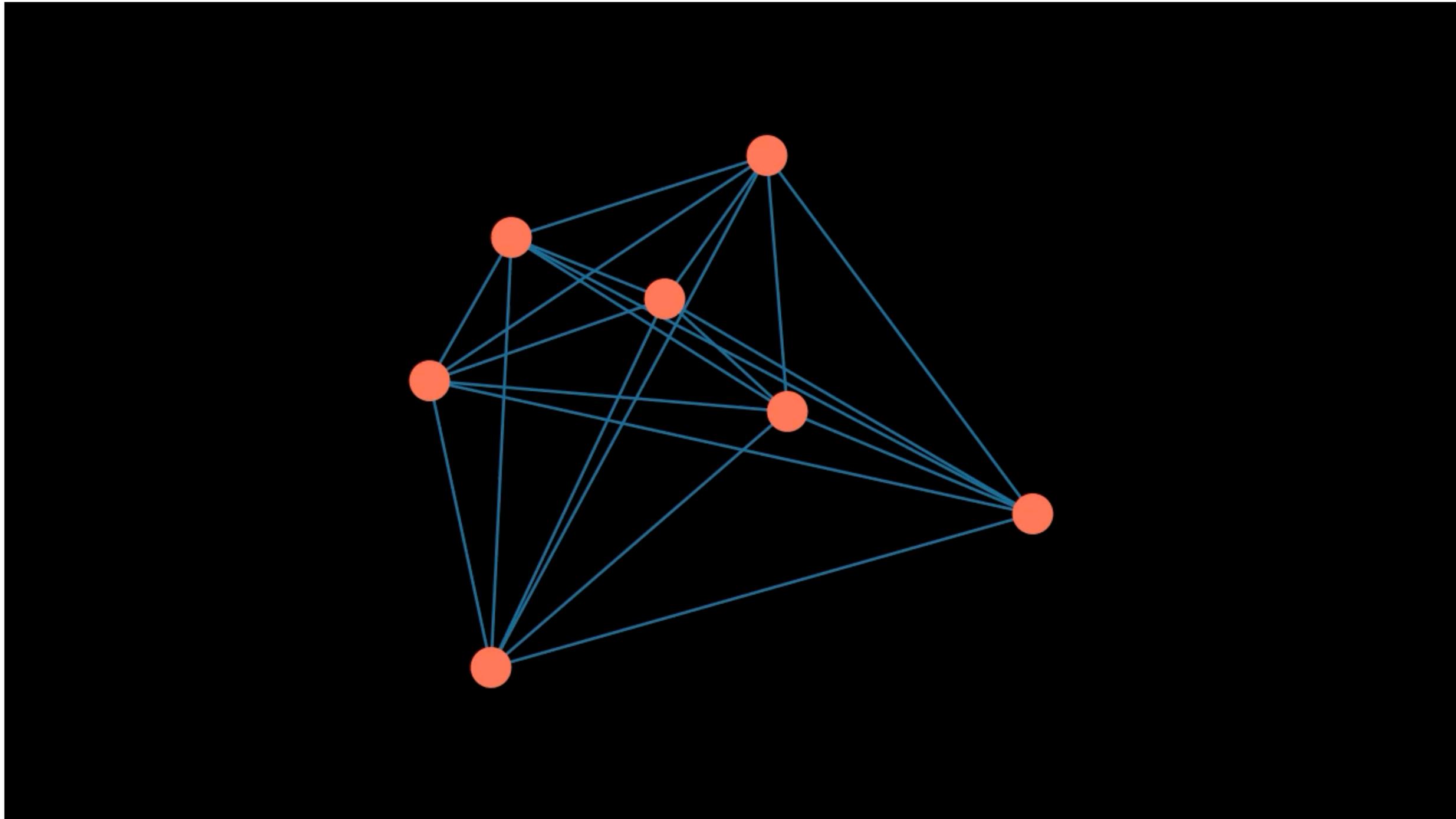
# Stigmergia artificiale

- Una volta che tutte le formiche hanno completato il percorso, viene **aggiorna** il feromone artificiale  $\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^a \Delta_{ij}^k$   
Tasso di evaporazione 
- $\Delta_{ij}^k = \begin{cases} \text{merit}(L_k) & \text{se la formica } k \text{ ha usato l'arco } (i, j) \\ 0 & \text{altrimenti} \end{cases}$   
merit( $L_k$ ) è una misura legata alla qualità della soluzione
- Il metodo è iterativo: a ogni iterazione tutte le formiche partono per la ricerca del cibo
- A un certo punto, il **sistema convergerà**: le formiche hanno determinato un solo percorso

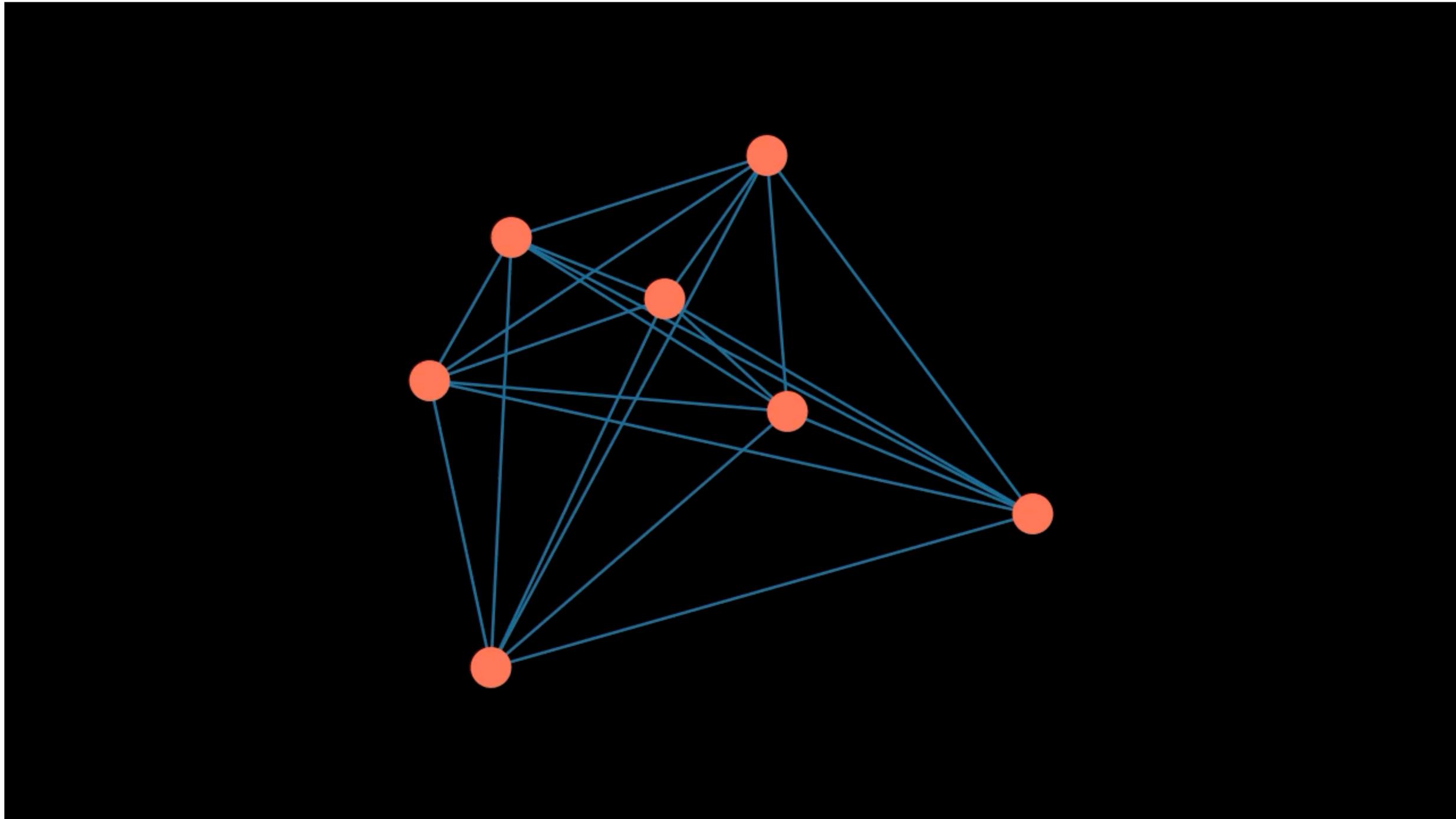
# Stigmergia artificiale

- Una volta che tutte le formiche hanno completato il percorso, viene **aggiorna** il feromone artificiale  $\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^a \Delta_{ij}^k$   
Tasso di evaporazione 
- $\Delta_{ij}^k = \begin{cases} \text{merit}(L_k) & \text{se la formica } k \text{ ha usato l'arco } (i, j) \\ 0 & \text{altrimenti} \end{cases}$   
merit( $L_k$ ) è una misura legata alla qualità della soluzione
- Il metodo è iterativo: a ogni iterazione tutte le formiche partono per la ricerca del cibo
- A un certo punto, il **sistema convergerà**: le formiche hanno determinato un solo percorso
- È una forma di **apprendimento per rinforzo**

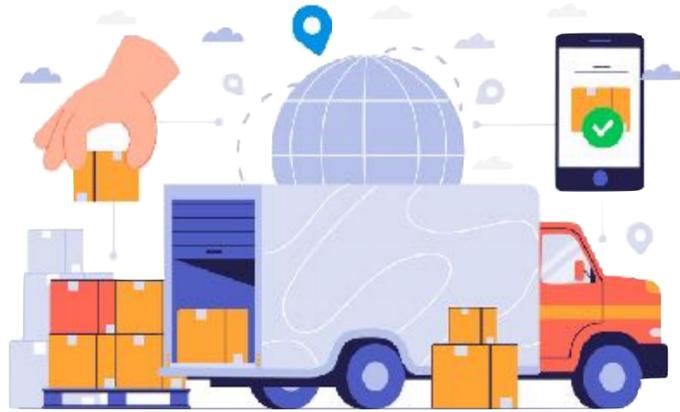
# Stigmergia artificiale



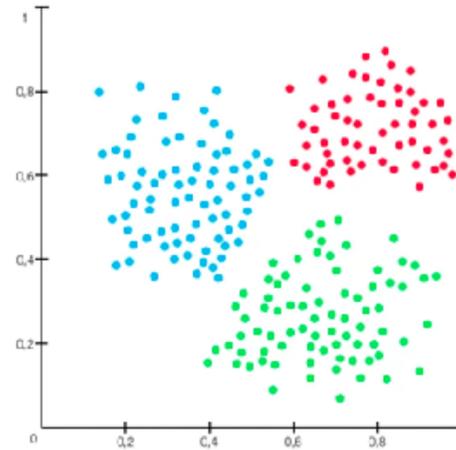
# Stigmergia artificiale



# Applicazioni di ACO



**Instradamento e caricamento**  
di veicoli



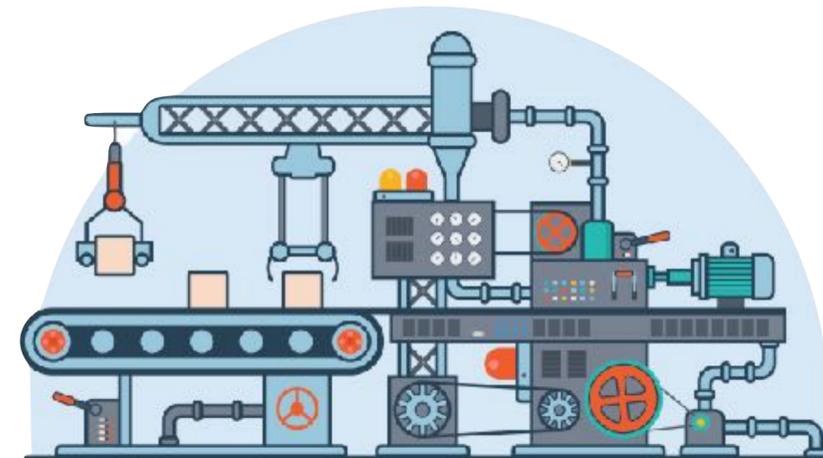
**Clustering** (Data Analytics)



Assegnazione di frequenze in **reti wireless**,  
progettazione di **circuiti integrati**



**Pianificazione** della  
produzione



**Sequenziamento** della  
produzione

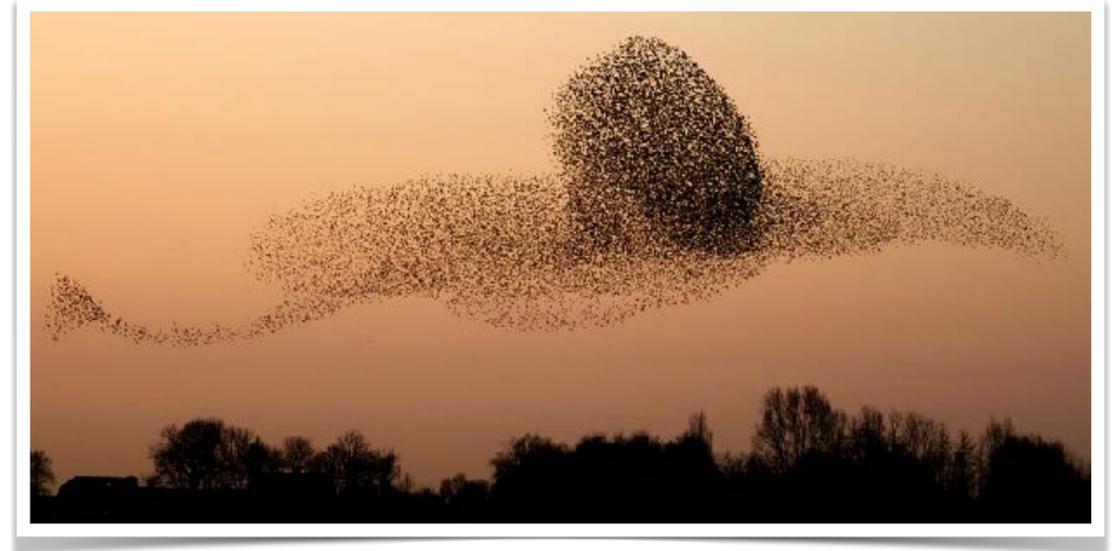
# Swarm intelligence

# Swarm intelligence



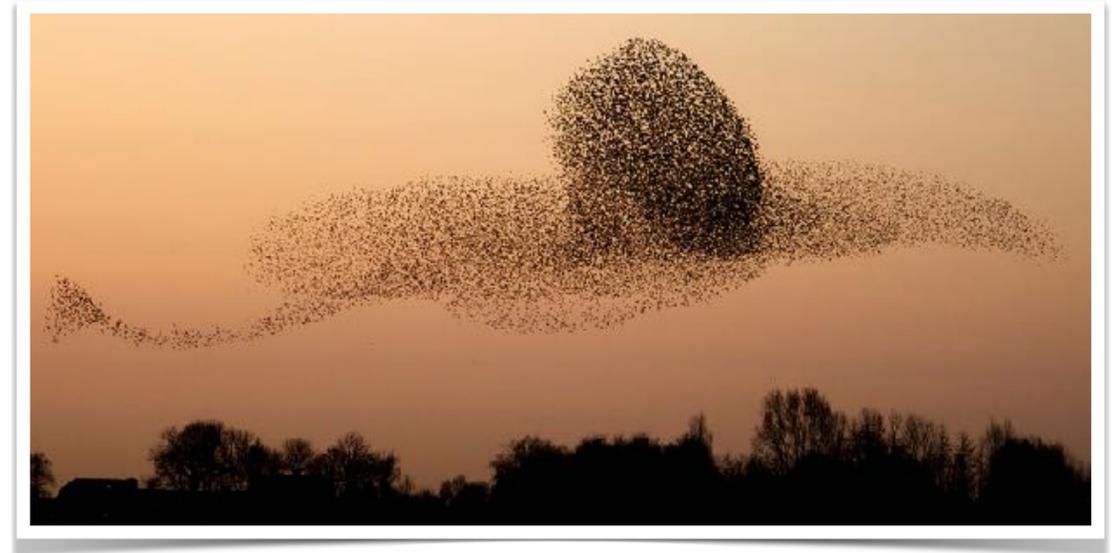
- L'**intelligenza emergente** non è propria dei soli insetti
- Il comportamento di stormi di uccelli, banchi di pesci può essere usato come fonte di ispirazione  
(esistono moltissimi nomi collettivi di animali: gregge, muta, sciame, branco, mandria, *flocks*, *swarms*, *schools*, *colonies*, *caravans*, *clowders*, ...)

# Swarm intelligence



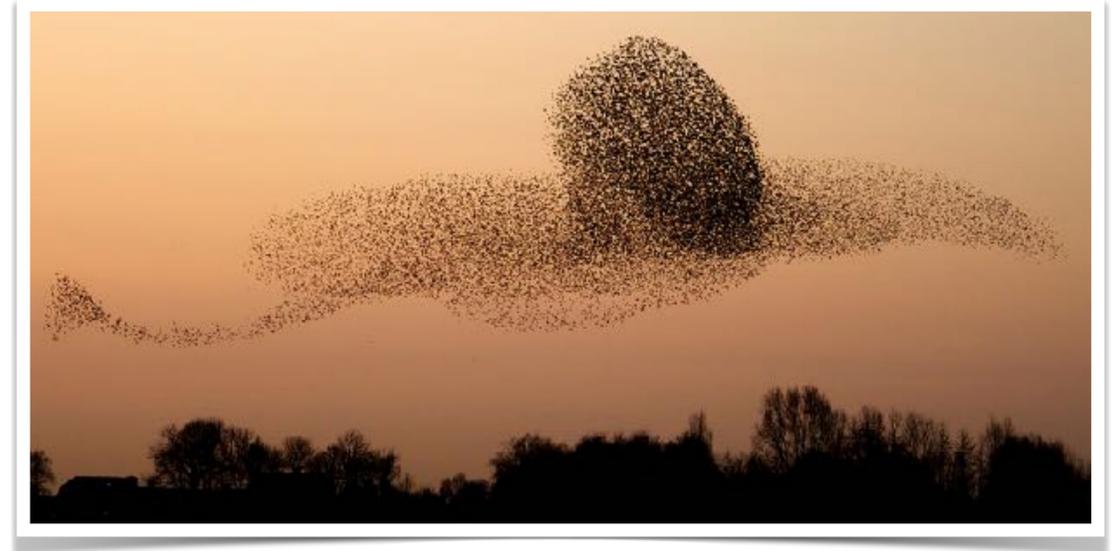
- L'**intelligenza emergente** non è propria dei soli insetti
- Il comportamento di stormi di uccelli, banchi di pesci può essere usato come fonte di ispirazione  
(esistono moltissimi nomi collettivi di animali: gregge, muta, sciame, branco, mandria, *flocks*, *swarms*, *schools*, *colonies*, *caravans*, *clowders*, ...)
- Anche in quest'ambito l'**interazione fra gli individui** gioca un ruolo fondamentale nel descrivere il comportamento dello stormo

# Swarm intelligence

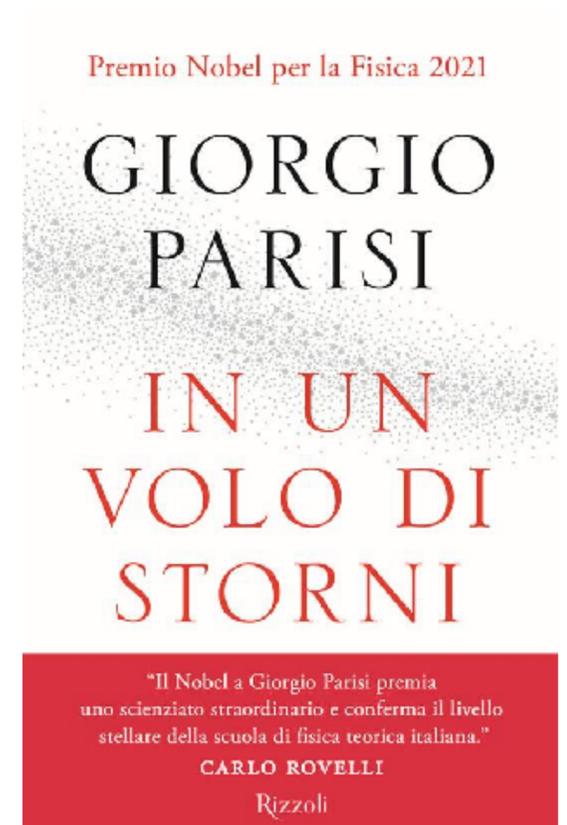


- L'**intelligenza emergente** non è propria dei soli insetti
- Il comportamento di stormi di uccelli, banchi di pesci può essere usato come fonte di ispirazione  
(esistono moltissimi nomi collettivi di animali: gregge, muta, sciame, branco, mandria, *flocks*, *swarms*, *schools*, *colonies*, *caravans*, *clowders*, ...)
- Anche in quest'ambito l'**interazione fra gli individui** gioca un ruolo fondamentale nel descrivere il comportamento dello stormo
- Le traiettorie seguite sono una **combinazione di esperienza** individuale e collettiva

# Swarm intelligence



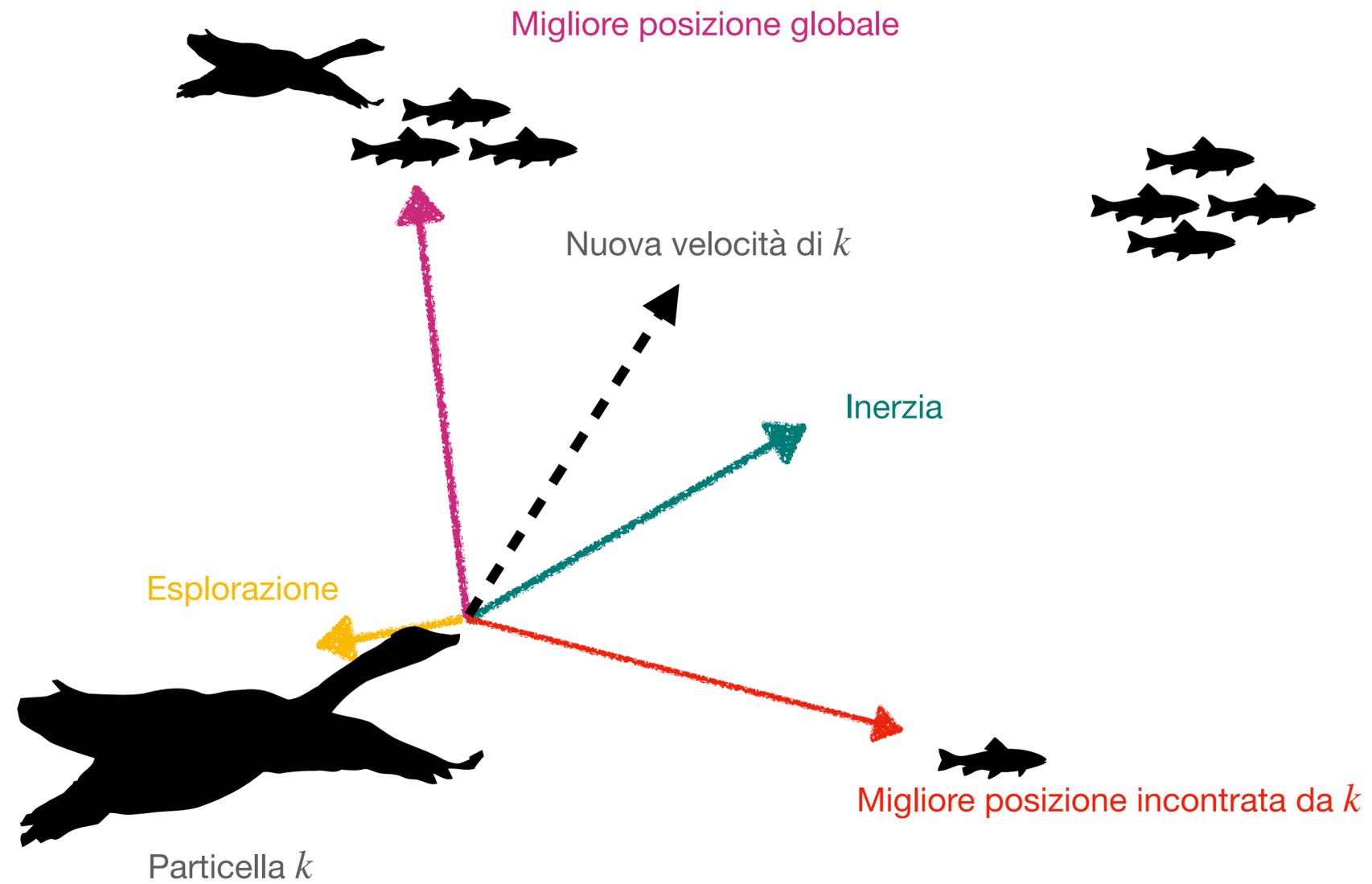
- L'**intelligenza emergente** non è propria dei soli insetti
- Il comportamento di stormi di uccelli, banchi di pesci può essere usato come fonte di ispirazione  
(esistono moltissimi nomi collettivi di animali: gregge, muta, sciame, branco, mandria, *flocks*, *swarms*, *schools*, *colonies*, *caravans*, *clowders*, ...)
- Anche in quest'ambito l'**interazione fra gli individui** gioca un ruolo fondamentale nel descrivere il comportamento dello stormo
  - Le traiettorie seguite sono una **combinazione di esperienza** individuale e collettiva
- Vi è un autoadattamento dello stormo



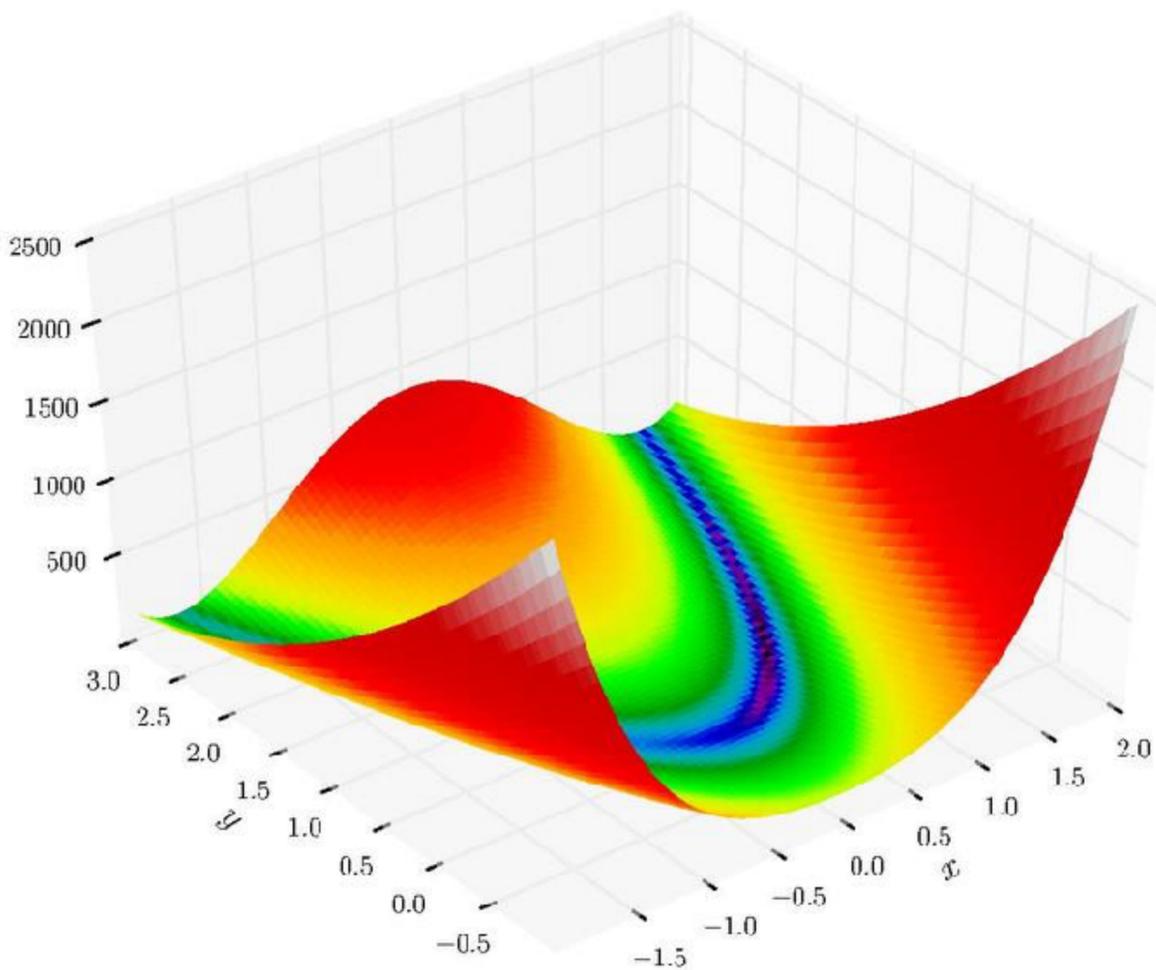
# Particle Swarm Optimization

- Schema algoritmico ispirato al **comportamento di stormi di uccelli** alla ricerca di cibo o di un riparo per la notte
- Un insieme di agenti (particelle) costituisce uno stormo che si può muovere in uno spazio di ricerca (continuo) alla ricerca della soluzione ottima rispetto ad una funzione obiettivo  $F(\bar{x})$
- Ciascun membro dello stormo  $\bar{x}_k$  **aggiusta** il proprio volo **in base** alla propria **esperienza** e quella di altre particelle
  - **Velocità:**  $\bar{v}_k(t + 1) = \bar{v}_k(t) + C_1 r(0,1) \cdot (\bar{x}_k^p(t) - \bar{x}_k(t)) + C_2 \cdot r(0,1)(\bar{x}^g(t) - \bar{x}_k(t))$
  - **Posizione:**  $\bar{x}_k(t + 1) = \bar{x}_k(t) + \bar{v}_k(t + 1)$
  - In base alle **migliori posizioni**  $\bar{x}^g$  e  $\bar{x}_k^p$ , relativamente alla funzione  $F(\cdot)$ , rispettivamente globale e della singola particella  $k$

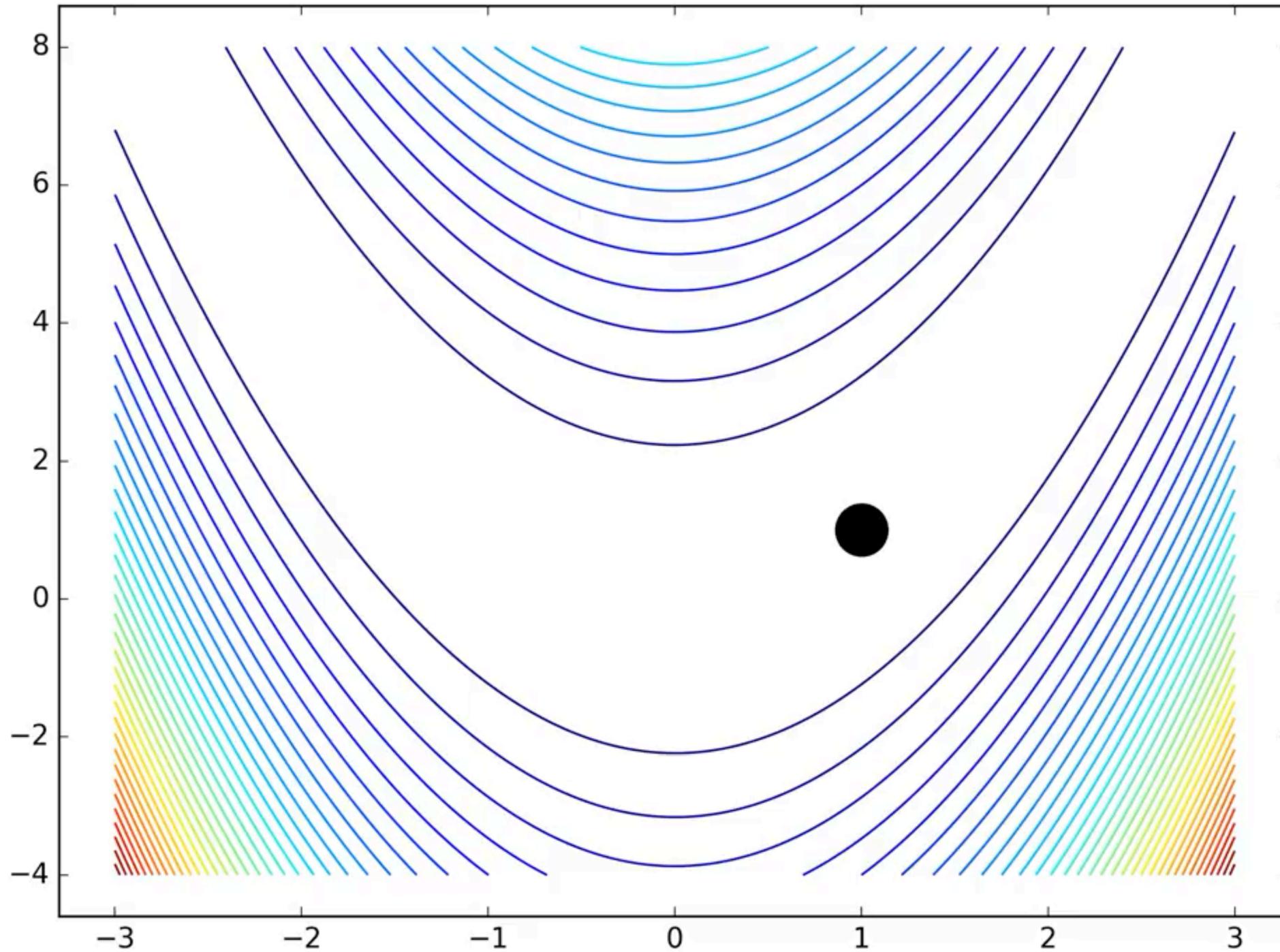
# Dinamica delle particelle



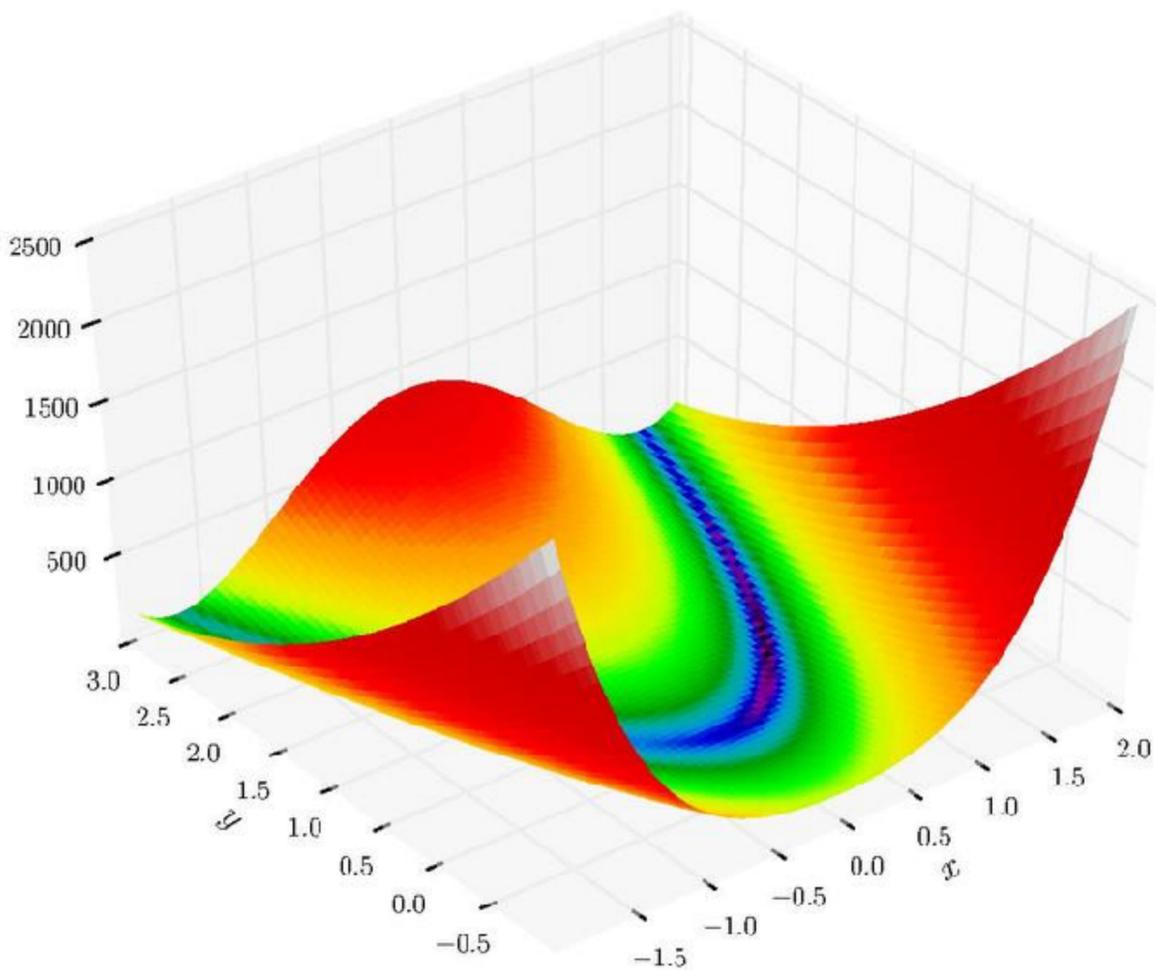
# Esempio PSO



Funzione di Rosenbrock:  
 $f(x, y) = (a - x)^2 + b(y - x^2)^2$

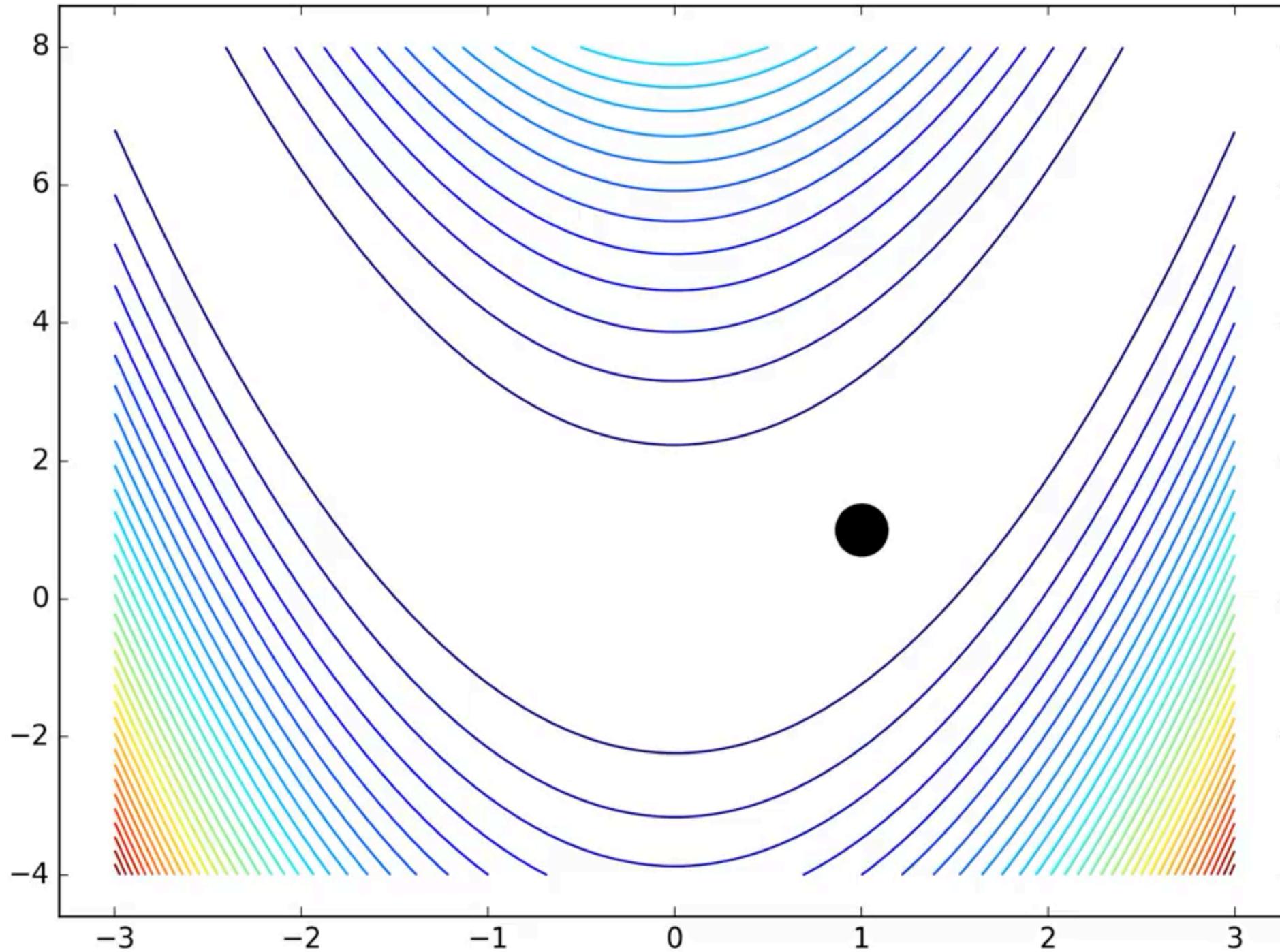


# Esempio PSO



Funzione di Rosenbrock:

$$f(x, y) = (a - x)^2 + b(y - x^2)^2$$



# Applicazioni di PSO



Gestione **finanziaria** e ottimizzazione dei **portafogli di investimenti**



Progettazione di **componenti meccanici**, moderazione di **materiali**, navigazione di **robot**



**Instradamento** di veicoli



Gestione dell'**energia** e ottimizzazione di **reti di distribuzione**

# Computazione evolutiva



- Un ulteriore fonte di ispirazione per gli algoritmi di ottimizzazione è l'evoluzione naturale (Darwiniana)
- Selezione degli **individui più adatti** (misurati attraverso il loro fenotipo), che trasmettono i propri caratteri genetici (genotipo) alle generazioni future
- **Algoritmi Genetici** (Holland 1957) e Memetici
- Particolarmente adatti per l'**ottimizzazione multi-obiettivo**

# Ottimizzazione multi-obiettivo

- Trovare:

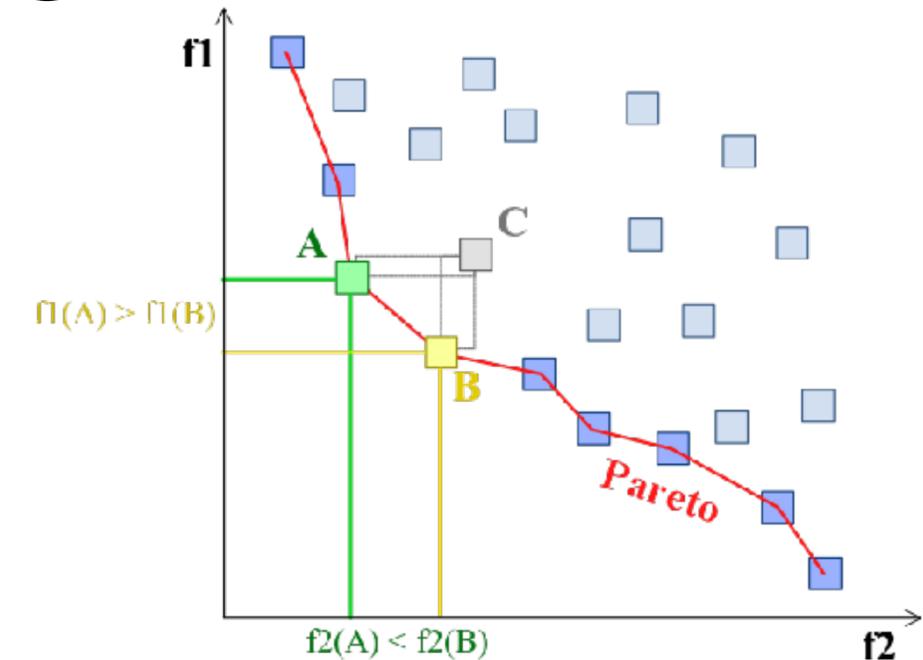
$$\min_{\mathbf{x} \in \mathcal{C}} (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x}))$$
$$f_i : \mathcal{X} \rightarrow \mathbb{R}$$

- In generale, non esiste una soluzione  $\mathbf{x}$  che minimizzi simultaneamente tutte le  $f_i(\mathbf{x})$ , siamo interessati in soluzioni **Pareto ottimali**, ovvero che non siano dominate da altre

- Una soluzione  $\mathbf{x}$  domina un'altra soluzione  $\mathbf{y}$  se:

1.  $f_i(\mathbf{x}) \leq f_i(\mathbf{y}) \forall i \in 1, 2, \dots, k$

2.  $\exists j \in 1, 2, \dots, k$  t.c.  $f_j(\mathbf{x}) < f_j(\mathbf{y})$



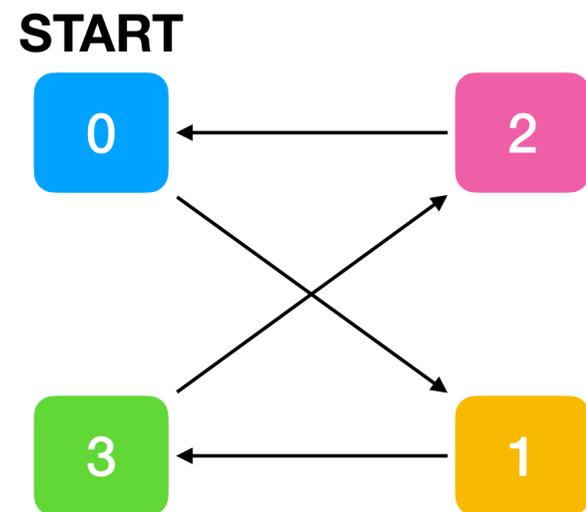
# Algoritmi Genetici

- Prendiamo in considerazione una **popolazione** (insieme di soluzioni) che evolve di **generazione** in **generazione**.
- Le soluzioni sono codificate sottoforma di **cromosomi**
  - Sequenze di **geni** (variabili) a cui viene associato un **allele** (valore della variabile)
  - Debbono essere decodificati nel loro fenotipo (ossia la soluzione al problema in esame)

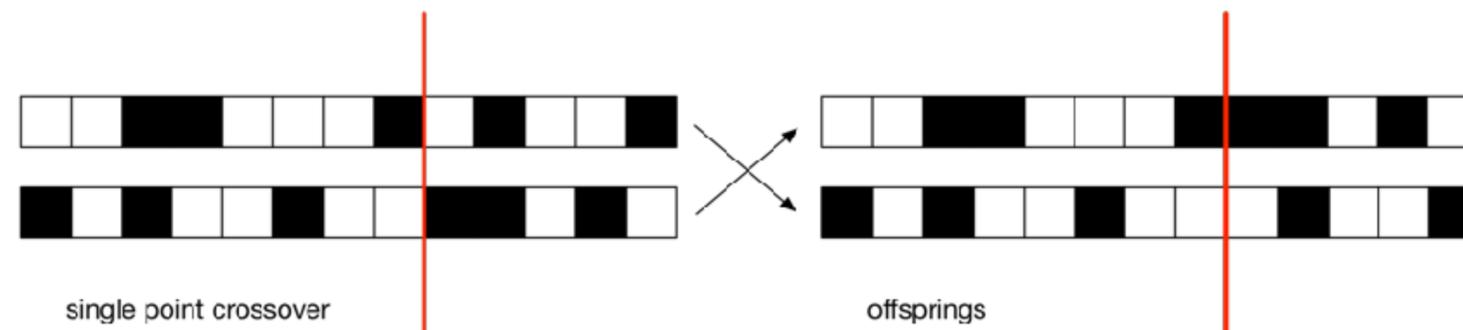
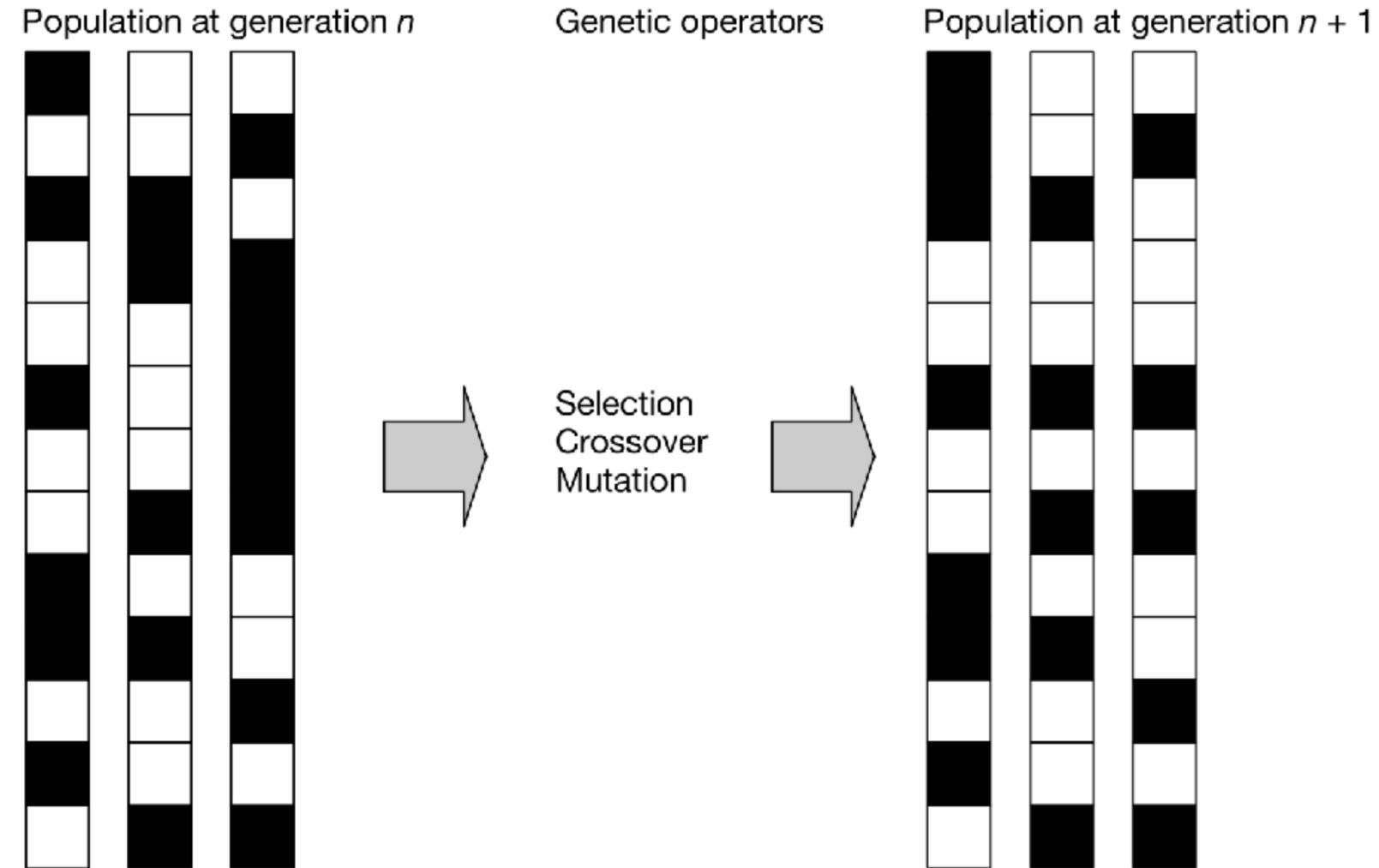
# Algoritmi Genetici

## Esempio di cromosoma

- Ad esempio nel caso del **TSP** con  $n$  nodi, i cromosomi possono essere delle sequenze senza ripetizioni degli interi  $\{0, \dots, n - 1\}$   
(decodificabili come permutazioni,  $x_k = i$  se il nodo  $i$  è visitato al passo  $k$ , oppure come successore,  $x_i = j$  se visito il nodo  $j$  dopo il nodo  $i$ )



# Algoritmi Genetici: operatori



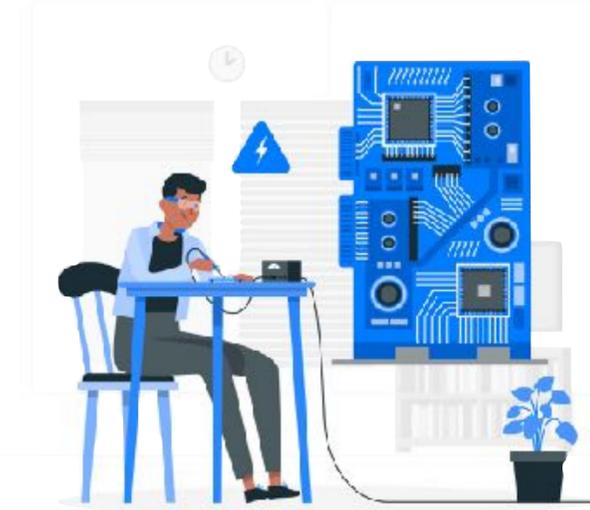




# Applicazioni degli Algoritmi Genetici



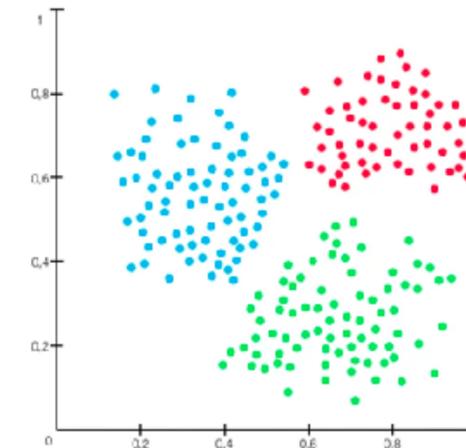
Ambito biologico, es. ottimizzazione della **ricostruzione 3D delle proteine**



Problemi di **progettazione di circuiti integrati**



**Pianificazione e sequenziamento** della produzione

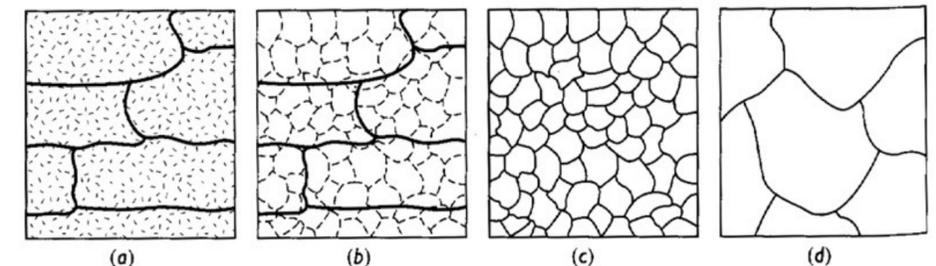


**Clustering** (Data Analytics)

In generale, **problemi multi-obiettivo**

# Metafore fisiche: tempra simulata

- Non solamente le metafore biologiche sono fonte di ispirazione per gli algoritmi di ottimizzazione
- Anche processi fisici sono alla base di algoritmi efficaci
- Ad esempio, il processo di tempra dei metalli (legato ai vetri di spin)



(a)

(b)

(c)

(d)

# Simulated Annealing

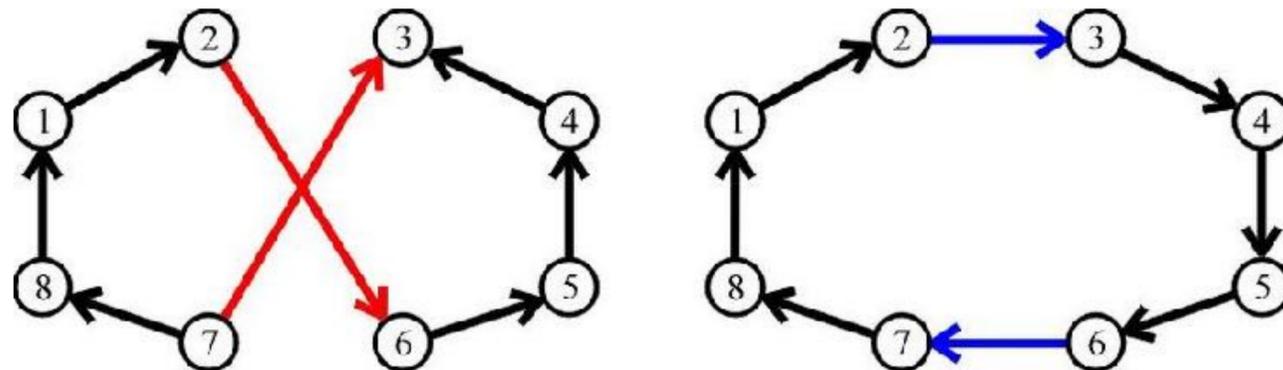
- Simulazione dei livelli di energia di un sistema a seguito del processo di raffreddamento (Metropolis, 1953)
- Data una particella, alla temperatura  $T$ , la probabilità di incremento di energia  $\Delta E$  data dal cambiamento di stato del sistema è data da  $p(\Delta E) = e^{\frac{-\Delta E}{kT}}$  con  $k$  costante di Boltzmann
- A sistema freddo, il sistema si trova in uno stato di equilibrio (energia minima)

# Simulated Annealing

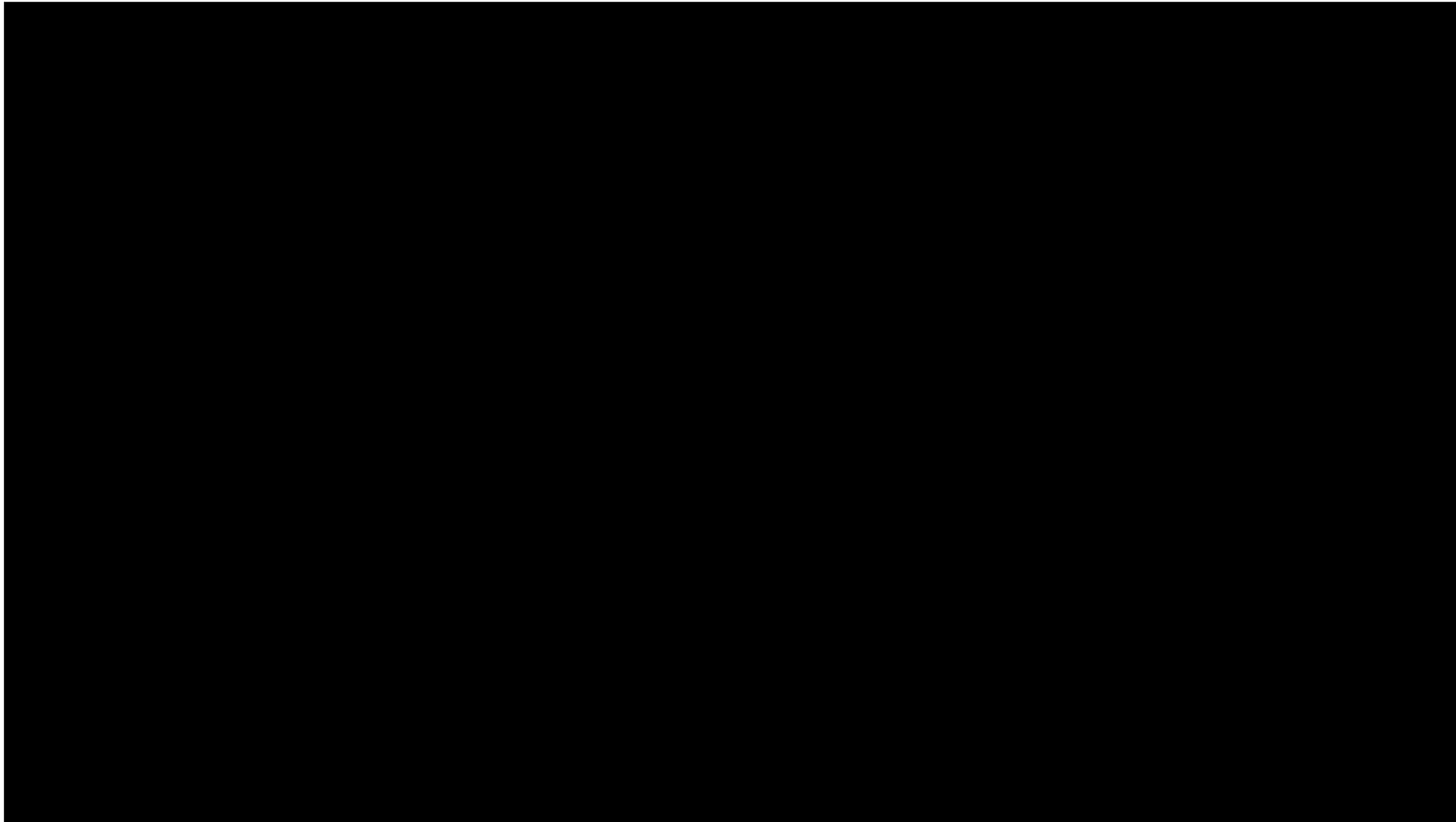
- Lo schema di Metropolis può essere utilizzato per esplorare uno spazio di ricerca:
  - La funzione obiettivo è la funzione di energia
  - Il cambio di stato del sistema consiste in una piccola modifica delle variabili decisionali
  - La temperatura funge da parametro di controllo e viene decrementata con il tempo
  - Il processo termina quando si giunge in uno stato *freddo*

# Simulated Annealing per il TSP

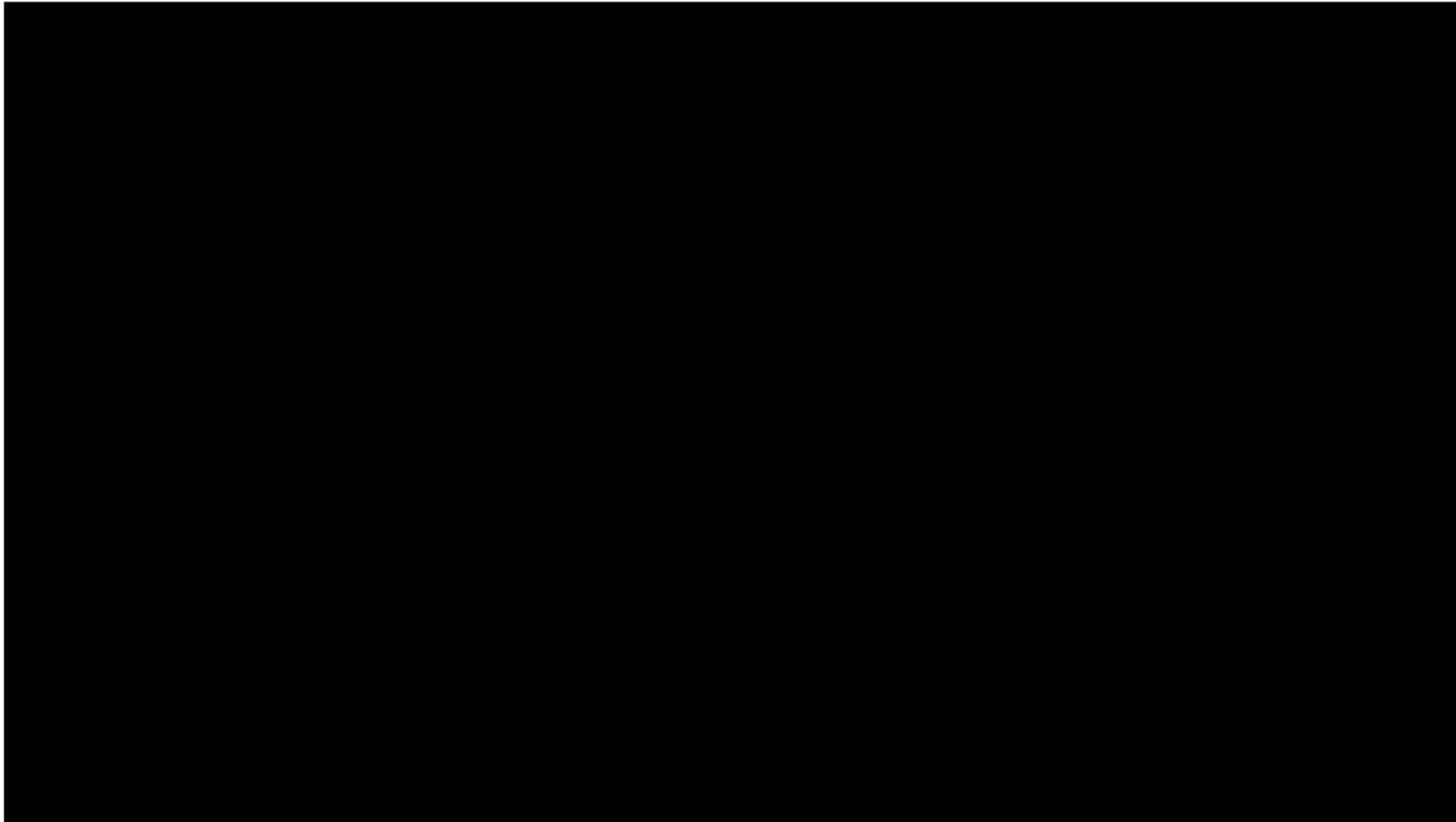
- Spazio di ricerca: tutte le  $\frac{(n-1)!}{2}$  permutazioni dei nodi nel grafo
- Perturbazioni per il cambio di stato: 2-opt
- Una mossa che peggiora l'obiettivo di un valore  $\Delta F$  è accettata con probabilità  $p(\Delta F) = e^{\frac{-\Delta F}{kT}}$ , via via decrescente



# Esempio SA



# Esempio SA



# Conclusioni:

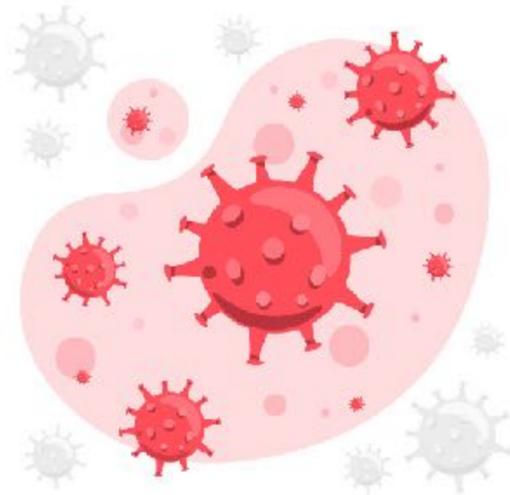
## Algoritmi di ottimizzazione ispirati a metafore naturali

- Ne esistono diversi **livelli di astrazione**/intelligenza:
  - Metafore fisiche
  - Metafore biologiche/evolutive
  - Insetti
- Sono particolarmente **efficaci** su problemi computazionalmente difficili



# Conclusioni:

Esiste un intero **bestiario** di algoritmi nature inspired



Batteri e microbi



Animali domestici (e non)



Gruppi di persone