

# Intelligenza Artificiale II

## Calcolo evolutivo

### Introduzione

Marco Piastra

# Evoluzione (secondo Darwin)

- Individui e popolazione

Un popolazione consiste di un insieme di individui diversificati

La ricombinazione (riproduttiva) di caratteristiche individuali che risultano più adatte all'ambiente tende a diventare prevalente nella popolazione

Gli individui sono le “unità di selezione”

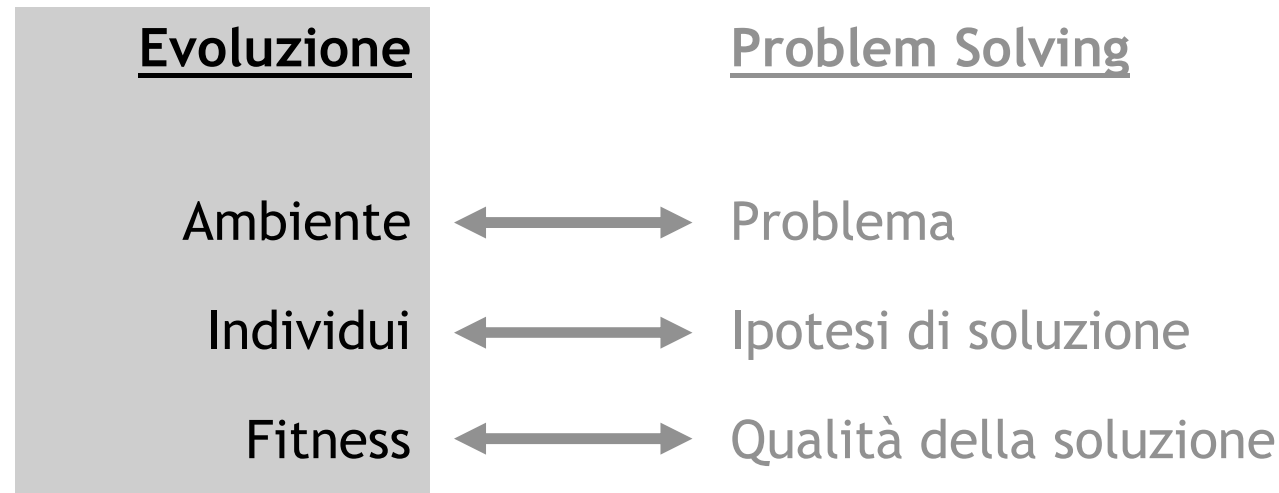
- Ricombinazione e caso

Le variazioni casuali, dovute alle mutazioni spontanee, garantiscono una sorgente costante di diversità e preservano il potenziale evolutivo della popolazione

Le popolazioni sono le “unità di evoluzione”

- Si noti l'assenza di una *supervisione esterna*  
l'evoluzione è una sorta di processo spontaneo

# Evoluzione biologica come algoritmo



- **Analogie**

  - Fitness → possibilità di sopravvivenza e riproduzione

  - Qualità → bontà della soluzione

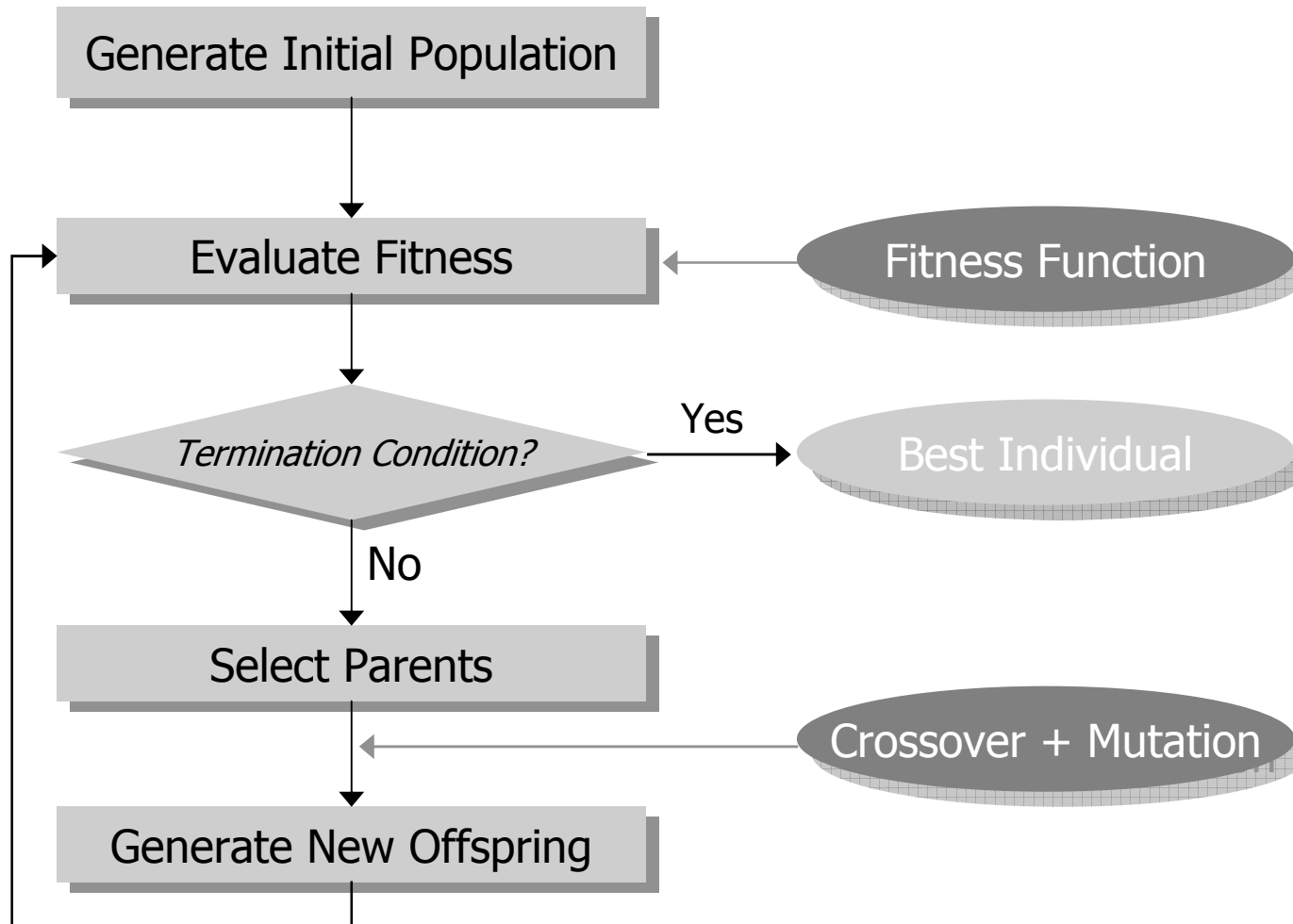
- **Differenze**

  - Popolazione (di *individui*) → insieme di ipotesi di soluzione

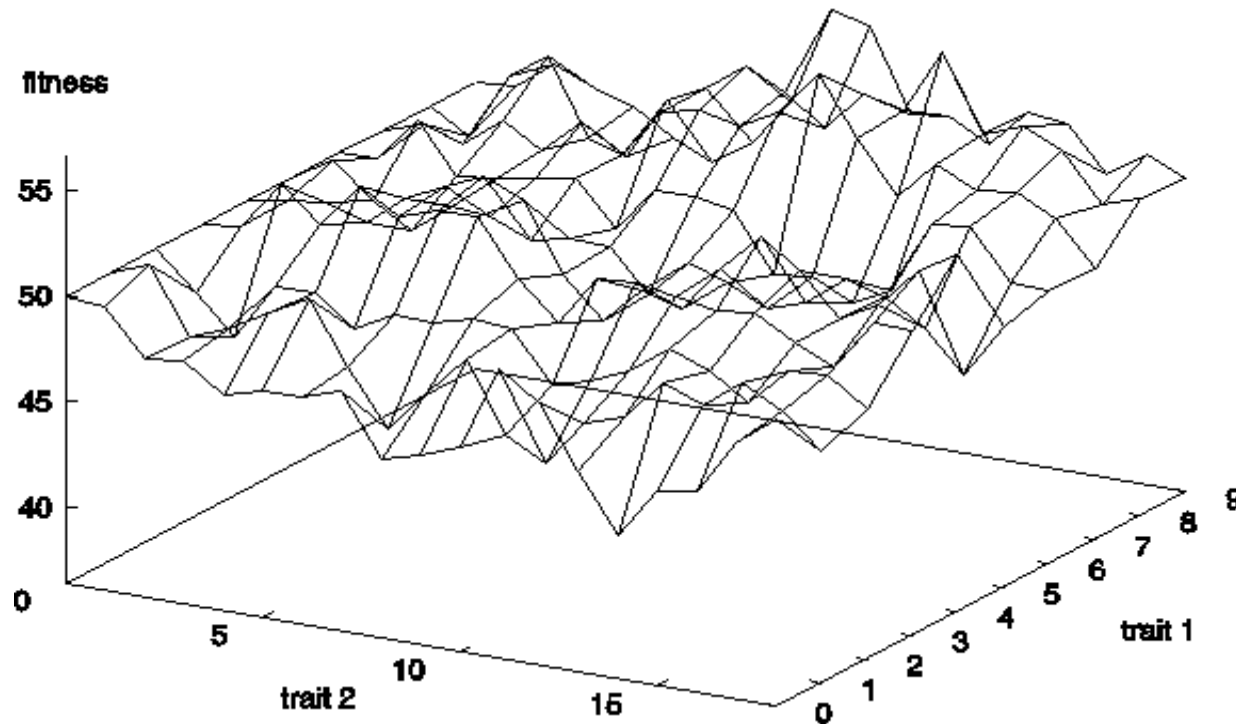
  - Processo evolutivo (della *popolazione*)

    - migrazione progressiva della *popolazione* verso più alti livelli di *fitness*

# Struttura generale del processo



## EC - Fitness landscape

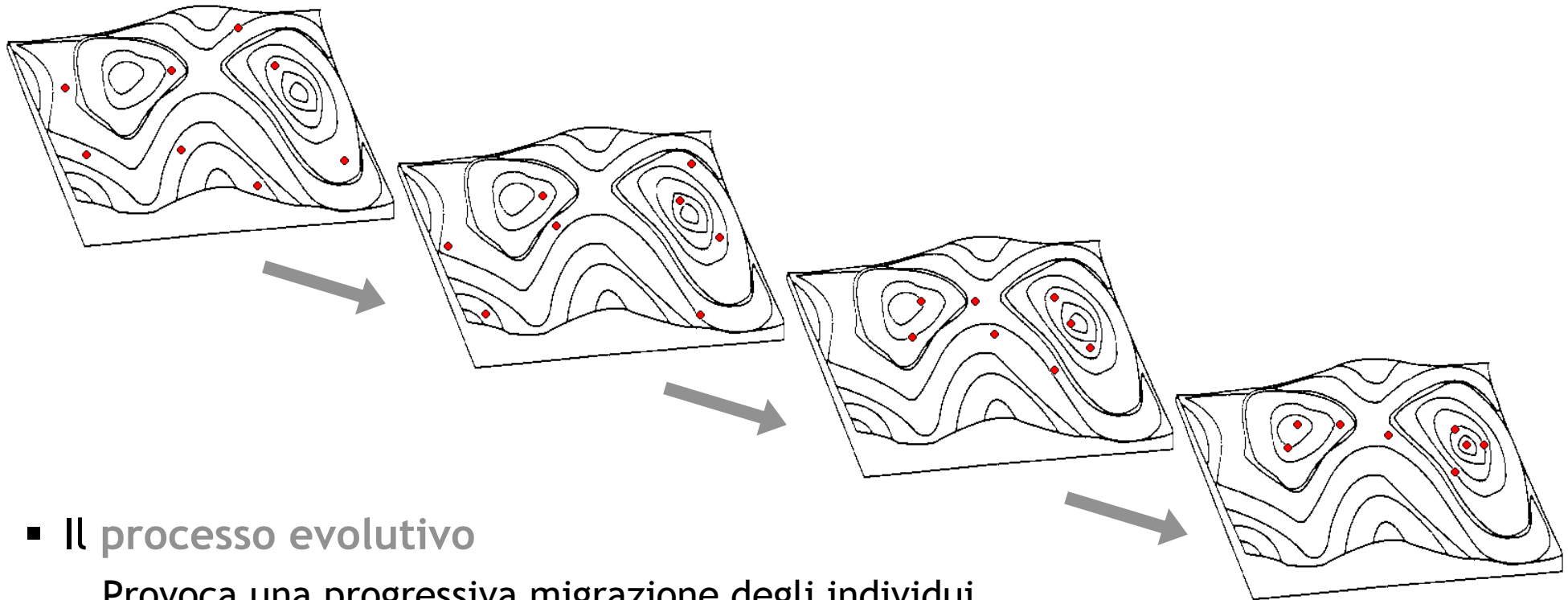


- Ambiente come ‘fitness landscape’

Descritto dai valori della funzione di fitness in funzione delle caratteristiche delle possibili soluzioni (individui)

A ciascun individuo corrisponde un punto nel ‘landscape’

# Evoluzione come migrazione



- **Il processo evolutivo**

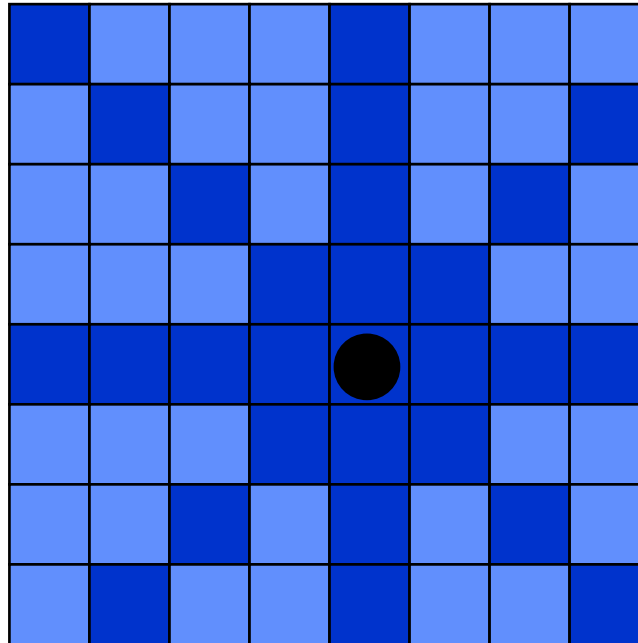
- Provoca una progressiva migrazione degli individui verso zone a fitness più elevata

- **Attenzione:** è la popolazione che migra, non gli individui  
EC vs. Learning

## Cenni storici

- 1948, Turing:  
propone la “*genetical or evolutionary search*”
- 1962, Bremermann  
ottimizzazione tramite *evoluzione e ricombinazione*
- 1964, Rechenberg  
introduce le *evolution strategies*
- 1965, L. Fogel, Owens and Walsh  
introduce lo *evolutionary programming*
- 1975, Holland  
introduce i *genetic algorithms*
- 1992, Koza  
introduce il *genetic programming*

## Esempio: le 8 regine



- Problema

Piazzare 8 regine su una scacchiera 8 x 8  
in modo che nessuna possa attaccare l'altra



# Genotipo e fenotipo

## ▪ Fenotipo

L'individuo completo che interagisce con l'ambiente

Di cui si può valutare la fitness

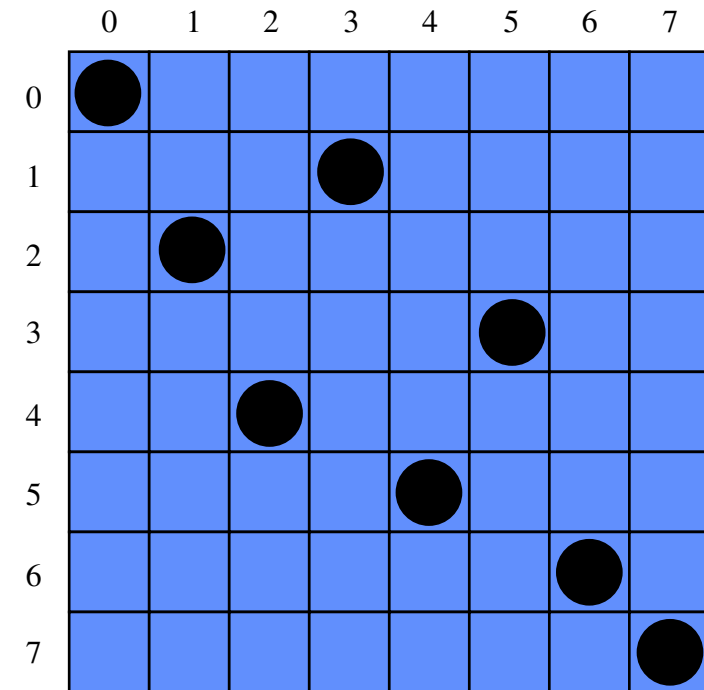
In questo caso,  
una disposizione delle 8 regine  
sulla scacchiera

## ▪ Genotipo

Il 'progetto genetico' dell'individuo

In questo caso, una sequenza di  
otto cifre, da 0 a 7

Su cui agiscono gli operatori genetici



↑ **Espressione**

Genotipo

0	2	4	1	5	3	6	7
---	---	---	---	---	---	---	---

(Rappresentazione binaria)

000	010	100	001	101	011	110	111
-----	-----	-----	-----	-----	-----	-----	-----

# Fitness

## ■ Problema

Prevenire i possibili attacchi

Penalità singola (di un'ipotesi di soluzione)

Un possibile attacco di due regine

Penalità complessiva

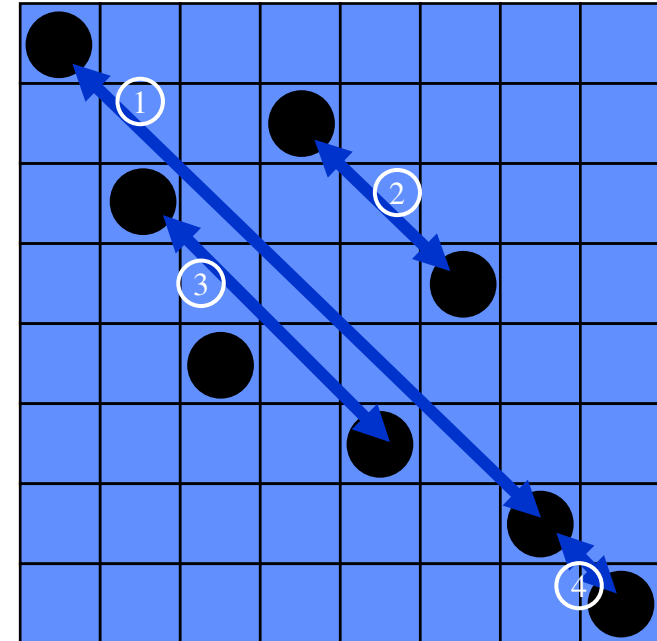
Somma di tutte le penalità singole

## ■ Fitness

Di una possibile soluzione (individuo):

$f = 8$  - la penalità complessiva

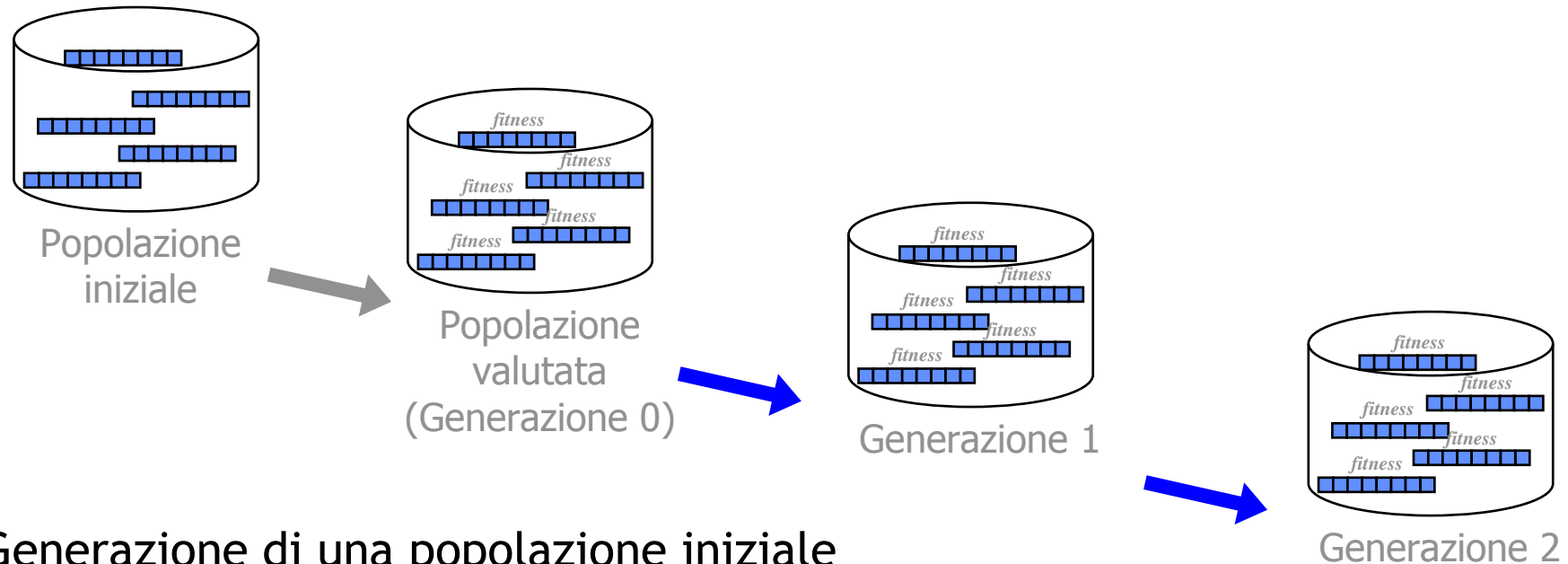
Nel caso in figura, la *fitness*  
del genotipo è 4



$$\text{fitness: } 8 - 4 = 4$$

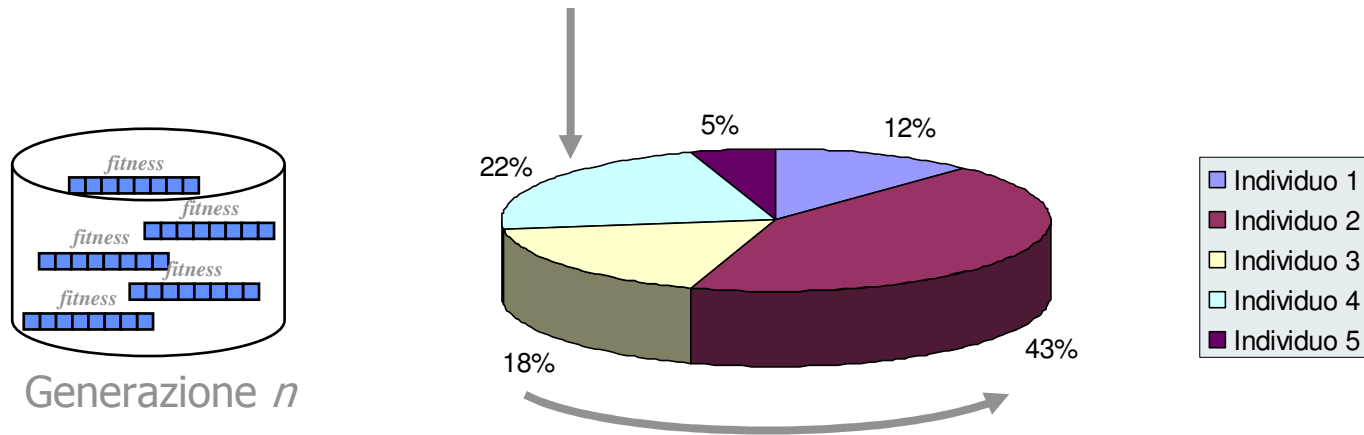
0	2	4	1	5	3	6	7
---	---	---	---	---	---	---	---

# Popolazioni e generazioni



- Generazione di una popolazione iniziale
  - M permutazioni generate a caso (anche con duplicazioni)
- Valutazione della *fitness* di ciascun individuo
  - Si ottiene la generazione 0
- Attivazione del processo evolutivo
  - Produzione iterativa di ulteriori generazioni

# Selezione



- Ruota della roulette (*Roulette wheel*)

A ciascun individuo si assegna un settore della roulette

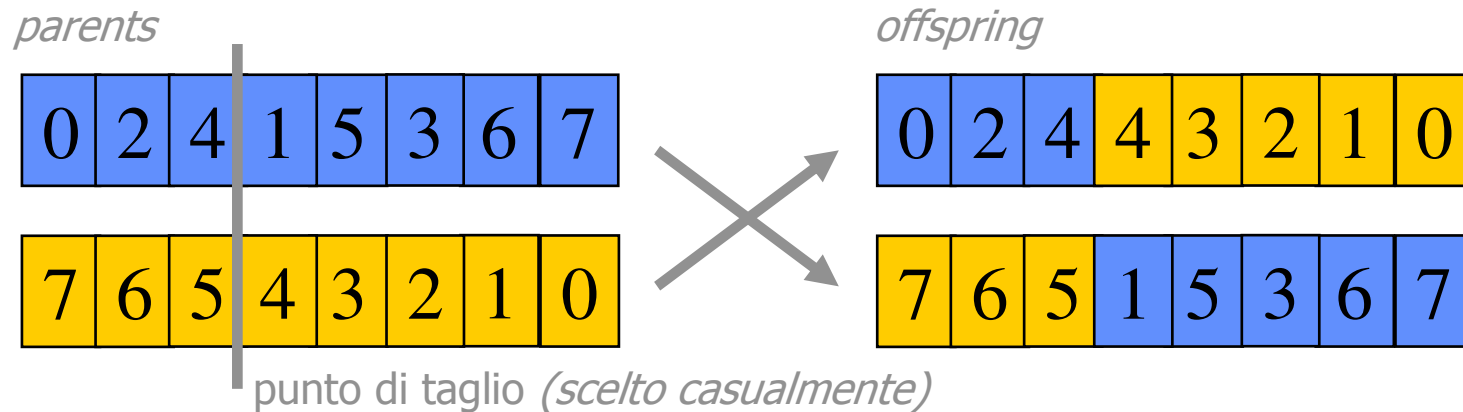
L'ampiezza del settore è proporzionale alla *fitness*

Tipicamente, ampiezza  $\sim f / f_{\text{avg}}$

Migliore è la *fitness*, più largo il settore

La probabilità di selezione è quindi più alta quanto migliore è la *fitness*

## Ricombinazione (Crossover)

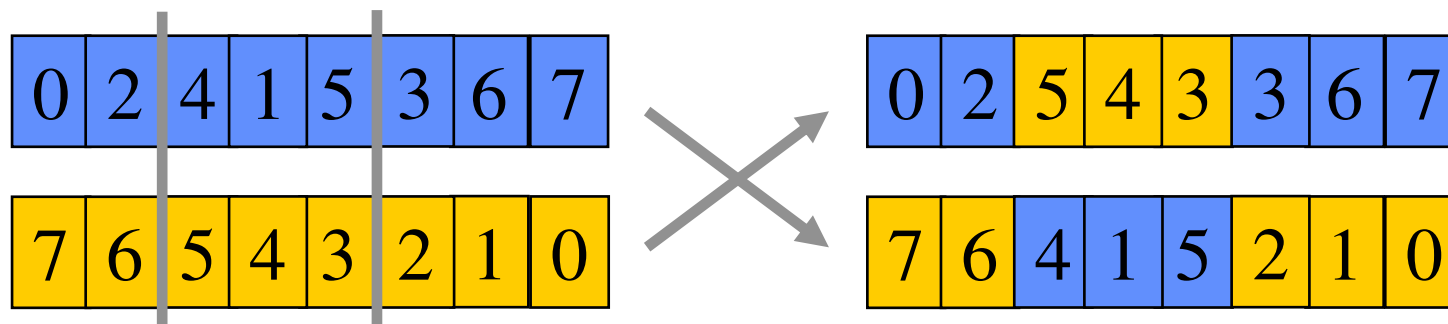


- Generazione di due nuovi individui a partire da due individui esistenti

Si seleziona un punto di *crossover*

Si incrociano le parti dei genotipi

Alternativa: due punti di taglio



# Mutazione

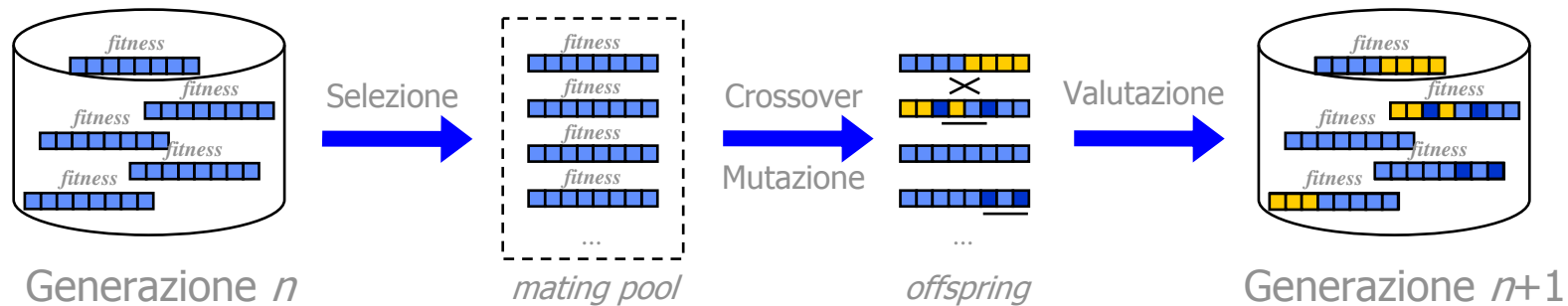


- Variazione casuale di un singolo individuo  
alterazione di una cifra  
Alternativa: inversione di due cifre



Altra alternativa: ciascuna cifra viene mutata con probabilità  $p_m$

# Mating pool



- **Pseudo-algoritmo:**

INPUT: Generazione  $n$  (di  $M$  individui)

Selezione di  $M$  individui (*roulette wheel*)

Per ciascuna coppia di individui (*parents*)

    applicazione del *crossover* con probabilità  $p_C$

    applicazione della *mutazione* (all'*offspring*) con probabilità  $p_M$

    inserimento dei risultati nella Generazione  $n+1$

Valutazione della Generazione  $n+1$  (*fitness*)

# Processo evolutivo

- Un processo stocastico

  - Selezione casuale degli individui come *parents*

  - Identificazione casuale del punto di *crossover*

  - Applicazione casuale della mutazione all'*offspring*

- Cui si aggiunge un effetto 'deriva' causato dalla *fitness*

  - L'unico elemento condizionante è la **selezione**

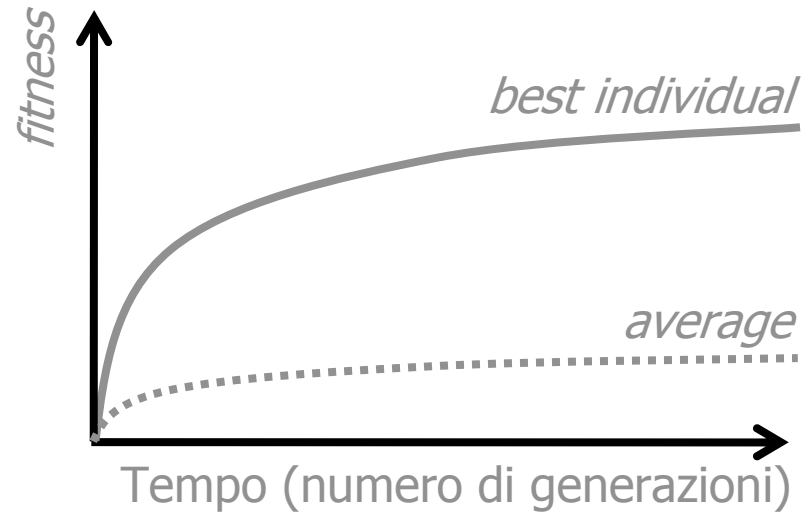
    - Nella selezione si 'bilancia' il caso (*roulette*) ed il determinismo (*fitness*)

    - Si determina così una tendenza migratoria della popolazione verso i picchi del *fitness landscape*



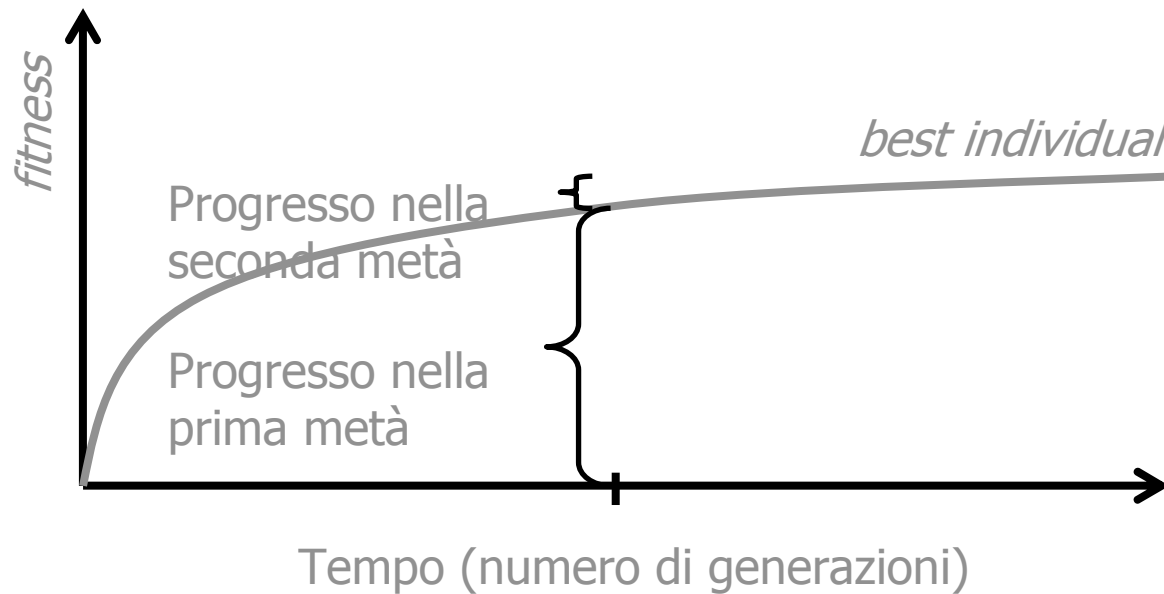


## Tipico andamento del processo evolutivo



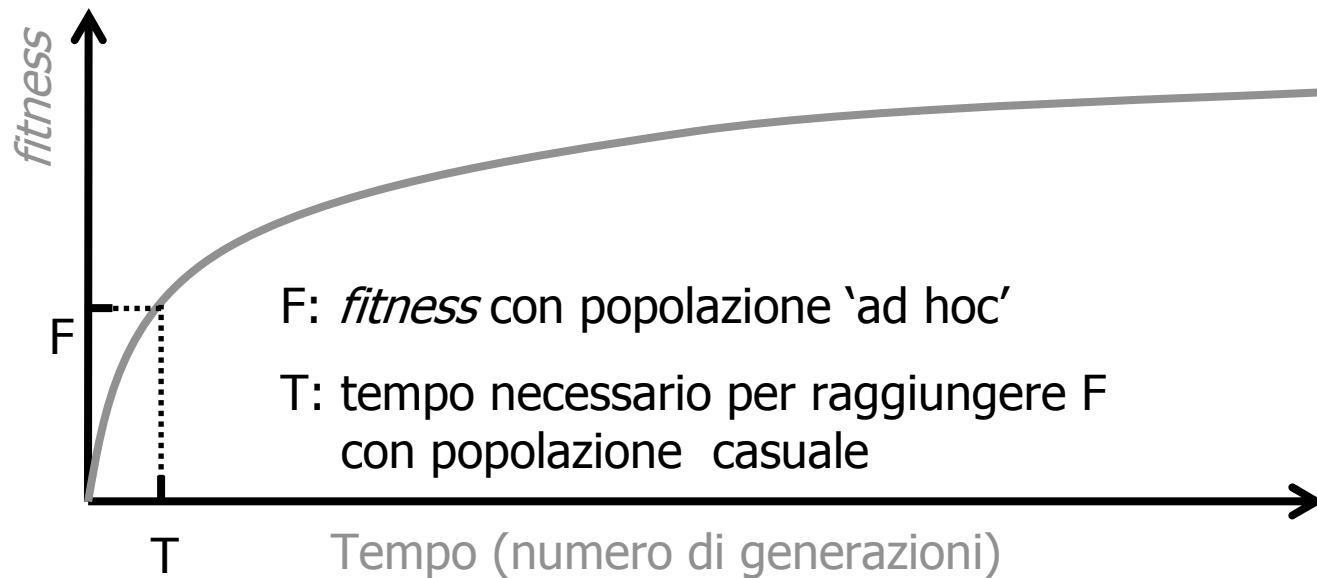
- L'effetto 'deriva' produce un miglioramento progressivo  
*best fitness*, cioè del miglior individuo della generazione  
migliora rapidamente  
fitness media della generazione  
migliora molto più lentamente
- Supervisione: fitness + criterio di terminazione  
raramente, nella pratica, si ottiene un individuo ottimo

## Condizioni di terminazione



- Quando non si trova un individuo ottimo  
Attendere ulteriormente (aumento del numero di generazioni)  
non sempre porta miglioramenti significativi dei risultati

## Popolazione iniziale



- In generale, la generazione *casuale* della popolazione iniziale è una buona scelta

L'iniziale miglioramento della fitness è rapido

Spesso lo sforzo di creare popolazioni iniziali più specifiche non è compensato da risultati apprezzabili

# Varianti del calcolo evolutivo

- Rappresentazione degli individui
  - vettori di bit, di interi di numeri reali
  - grafi
  - alberi
  - dimensione fissa, limitata o variabile
- Operatori genetici
  - mutazione, crossover
  - operatori speciali (dipendenti dalla rappresentazione)
- Metodi di selezione
  - Roulette (fitness proporzionate)
  - Tournament
- Processo evolutivo
  - Generation-based
  - Steady state
- Fitness
  - scalare
  - multi-valore (multi-obiettivo)