**Algoritmi Genetici**

**­**Gli algoritmi genetici si basano sull’analogia tra la soluzione di problemi di ottimizzazione combinatoria. L’idea principale è quella di considerare una popolazione di soluzioni che evolve in accordo con un meccanismo di selezione per produrre soluzioni con buoni valori di funzione obiettivo.

Le caratteristiche principali degli algoritmi genetici sono descritte dal processo di evoluzione naturale nel senso che:

I **Geni** descrivono le caratteristiche di un organismo, ed essi sono presenti nei **Cromosomi**

Ciascun Gene può assumere **Alleli** diversi che producono differenze associate a quel Gene.

L’insieme dei geni è detto **Genotipo** e con il termine **Fitness** si indica la capacità dell’individuo ad adattarsi con l’ambiente esterno.

In particolare l’obiettivo fondamentale di questi algoritmi è valutare il processo di evoluzione selezionando in maniera progressiva individui ad elevata fitness.

Questo è legata al processo di riproduzione e per questo motivo individui con maggiore fitness hanno più probabilità di generare individui con maggiore fitness e hanno maggiore probabilità di sopravvivere.

Definito ciò è importante definire ora le corrispondenze tra l’evoluzione genetica e algoritmo.



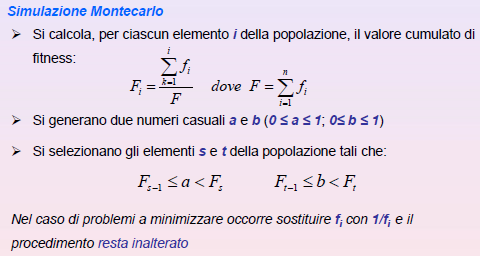
**Fasi di un Algoritmo Genetico**

Per modellare un algoritmo Genetico, bisogna seguire i seguenti passi:

* Codifica del Problema : Individuare una soluzione casuale del problema in termini di stringa di variabili decisionali, indicando i possibili valori assunti.
* Inizializzazione e Valutazione della Fitness(Funzione Obiettivo) : Si genera casualmente un insieme di possibili soluzioni e si associa a ciascuna di esse un valore di funzione obiettivo.
* Selezione : si selezionano coppie di soluzioni della popolazione alle quali applicare gli operatori genetici (crossover, mutazione,…)
* Generazione di nuove Soluzioni : si applicano gli operatori genetici a queste coppie di soluzioni selezionate al fine di produrre nuove soluzioni
* Sostituzione di Elementi della Popolazione : si sostituiscono soluzioni esistenti nella popolazione con le soluzioni prodotte nella fase di generazione

Applicando il problema iterativamente, si ricava un possibile collegamento tra le varie soluzioni. Per questo motivo, un algoritmo genetico può essere applicato al caso del commesso viaggiatore (TSP) dove ad esempio si hanno n città e una soluzione può essere rappresentata da una stringa di n elementi indicante l’ordine secondo il quale vanno visitate le città.

Ovviamente la selezione non viene fatta a caso ma in base alla probabilità di avere maggiore fitness. Ad esempio sfruttando la simulazione montecarlo.



Per la generazione delle nuove soluzioni, bisogna valutare come l’operatore di mutazione cambia i valori delle variabili. In particolare esso li modifica casualmente andando a trasformare nel caso binario gli zeri in uno e nel caso di rappresentazioni diverse sceglie casualmente un nuovo valore di allele tra i possibili valori. Per implementare questo operatore di mutazione bisogna fissare una certa probabilità di mutazione, applicare la simulazione montecarlo per verificare se ciascun gene debba essere modificato o meno e applicare infine le modifiche. In genere i valori di probabilità di mutazione sono molto bassi al fine di non perdere il patrimonio genetico della generazione precedente.

Ritornando al discorso della selezione, nel caso di stringhe di bit viene scelto una partizione (limite inf) contenente lo stesso numero di elementi per ogni stringa e successivamente vengono scambiate le partizioni tra loro. Esempio :

S1 : 0000111 -> Partizione composta dai primi 4 bit (0000) -> S1’ : 0000010  
S2 : 0101010 -> Partizione composta dai primi 4 bit (0101) -> S2’ : 0101111

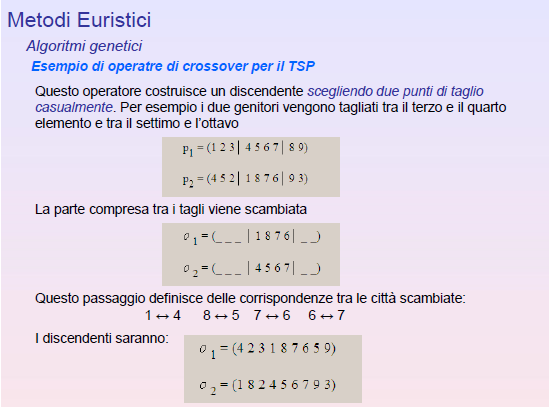
Si è scambiati gli ultimi tre bit dopo il limite inf tra le due stringhe (**SINGLE CROSSOVER)**

Nel **DOUBLE CROSSOVER** si scelgono due partizioni (limite inf e sup) e poi si scambiano i bit compresi tra questi due limiti tra le varie stringhe.

Nella Mutazione viene scelto un bit a caso delle stringhe e mutato da 0 a 1 o viceversa.

Ovviamente queste operazioni possono rendere delle soluzioni ammissibili dove prima erano inammissibili, e questa operazione prende il nome di **FILTRO** e si rende necessario da applicare quando durante l’operazione di crossover o mutazione si genera una soluzione inammissibile.

**ESEMPIO NEL TSP DI OPERAZIONE DI CROSSOVER**



Esistono anche altri operatori genetici, come l’operatore di inversione che data una stringa e scelti in maniera casuale due crosspoint (punti di Crossover) consiste nell’invertire l’ordine degli alleli presenti nella sottostringa individuata in modo tale che se si generano stringhe inammissibili allora o si utilizza un filtro o si accettano queste soluzioni a patto che venga immessa una penalità su di essi (nel senso che si può sottrarre un termine costante nel loro valore di funzione obiettivo).

Per dunque infine sostituire i nuovi elementi trovati, si possono eliminare individui con fitness più bassa con maggiore probabilità applicando il metodo montecarlo.

Per dare una convergenza all’algoritmo genetico, si può verificare che tutti gli elementi della popolazione siano più o meno simili e quindi viene interrotta la generazione di nuove soluzioni e viene prodotto un percorso di esplorazione limitato della regione ammissibile.