

## OTTIMIZZAZIONE NON LINEARE

In molti casi pratici non esistono algoritmi specifici per la soluzione del problema.

Si utilizzano quindi algoritmi basati su approssimazioni locali della funzione o algoritmi **euristici** (basati su regole “furbe” che conducono ad un “buon” subottimo).

Spesso gli algoritmi euristici sono basati su analogie con fenomeni naturali: il raffreddamento dei corpi (*simulated annealing*), i percorsi delle formiche (*ant colony*), l'evoluzione delle specie (*genetic algorithm - GA*). Tutti richiedono elevate potenze di calcolo (preferibilmente parallelo).

Vedremo i seguenti esempi di algoritmi:

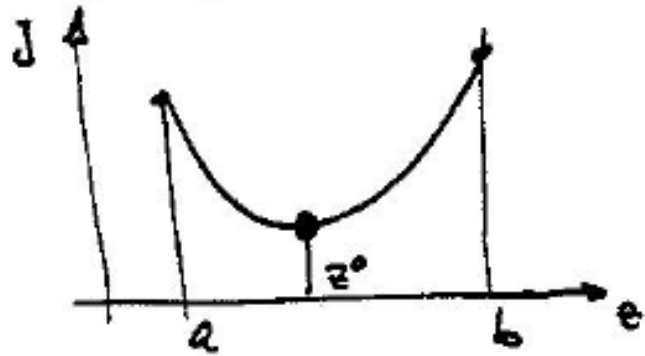
- Per problemi con una sola variabile di decisione (griglia, sezione aurea)
- Per problemi con  $n$  variabili e senza vincoli (griglia, gradiente, Newton)
- Per problemi con  $n$  variabili e vincoli (penalità, genetici)

## Ottimizzazione monodimensionale

Molto spesso nel risolvere problemi di pianificazione o di progetto ci si trova nella situazione seguente:

$$\min_z J(z)$$

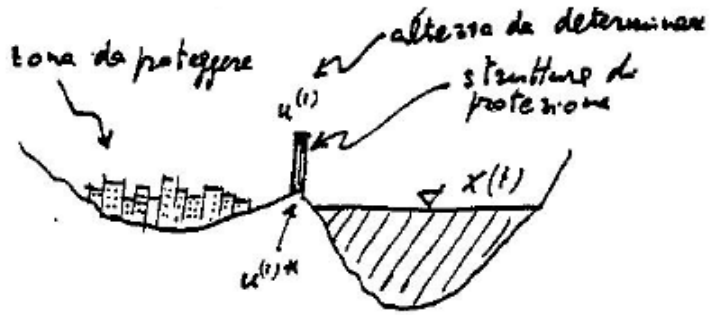
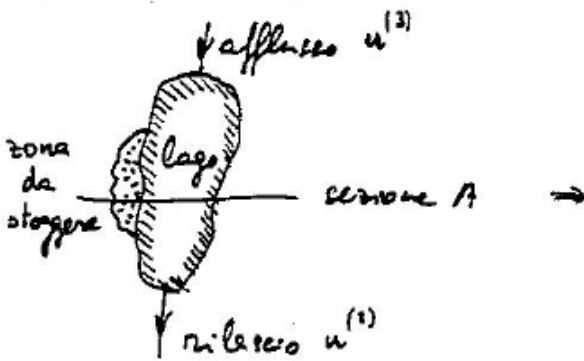
$$z \in Z$$



Con  $z$  scalare (cioè  $a \leq z \leq b$ ) ma con  $J$  valutabile solo per mezzo di un algoritmo molto complesso che richiede l'uso del calcolatore. In questi casi è quindi opportuno cercare di determinare  $z^\circ$  senza valutare la funzione in troppi punti  $z_i$ .

Vediamo dapprima un esempio in cui ci si trova in situazioni di questo tipo.

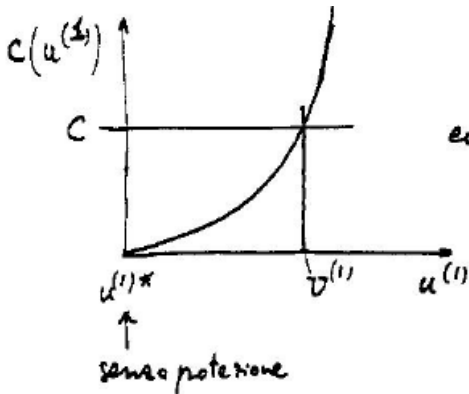
Esempio: altezza di una diga di protezione



$$\dot{x}(t) = \frac{1}{S} [u^{(3)}(t) - u^{(2)}(x, t)]$$

*area lago*

$\bar{u}^{(2)}(x, t) =$  regola operativa plausibilmente usata dal gestore della diga di regolazione



*costo della struttura di protezione*

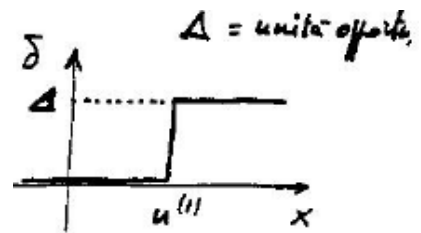
$\bar{u}^{(3)}(t) =$  possibili afflussi futuri (per esempio uguali a quelli passati)

Vincolo sul finanziamento:  $c(u^{(1)}) \leq C \iff u^{(1)*} \leq u^{(1)} \leq U^{(1)}$

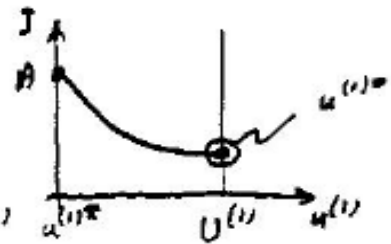
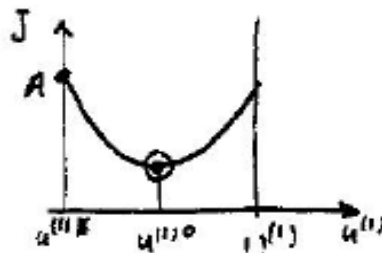
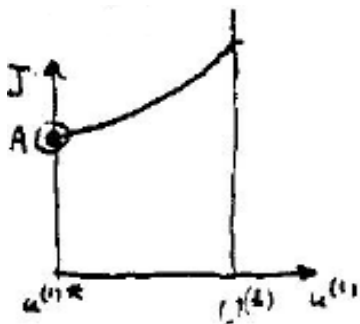
Danni di inondazione = tempo di inondazione

$$\min_{\{u^{(1)}\}} \left[ \int_0^{\infty} e^{-\rho t} \delta(x(t) - u^{(1)}) dt + c(u^{(1)}) \right]$$

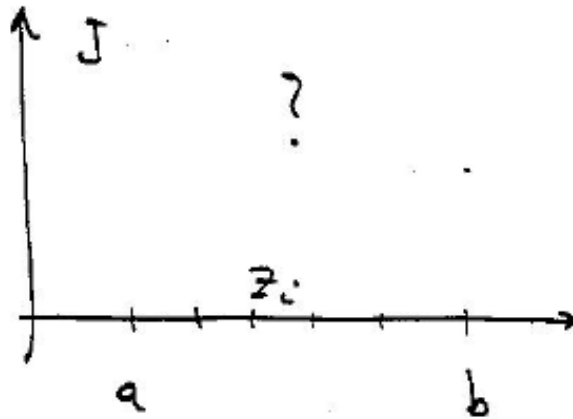
*fattore di costo*



Si hanno tre casi possibili:



## Metodo della Griglia



Si fissano a priori  $n$  punti  $z_i$  in cui valutare  $J$ .

La griglia può essere:   Regolare  
                                  Casuale  
                                  Irregolare

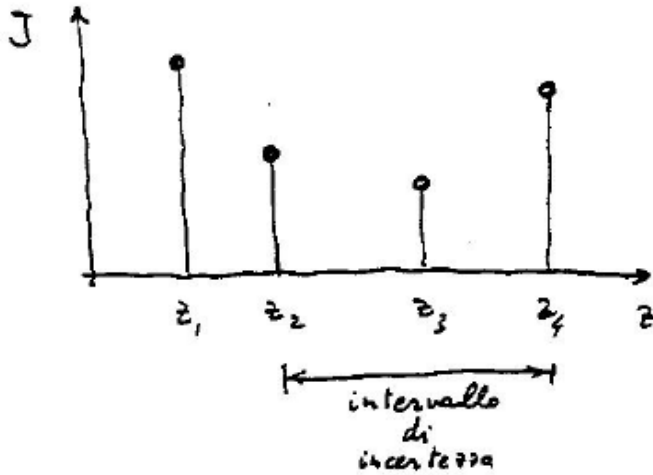
La griglia può essere irregolare per tener conto delle impressioni soggettive dell'operatore che infittirà i punti là dove crede che ci sia il minimo.

Con questo metodo l'intervallo di intermezzo è maggiore o uguale a

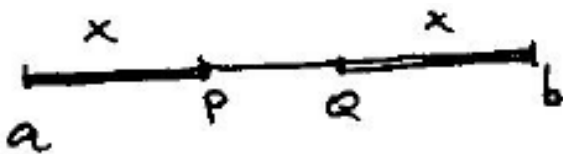
$$I = (b-a) / (n+1).$$

Si può anche fare una ricerca ricorsiva: prima una griglia poco fitta  
→ seconda griglia intorno al punto migliore trovato → ...

## Metodo della sezione aurea



Questo metodo è basato in parte su questa osservazione: se c'è un solo minimo e i valori già provati sono disposti come in figura, il punto di minimo è tra  $z_2$  e  $z_4$  (in altre parole si può eliminare  $[z_1z_2]$ )

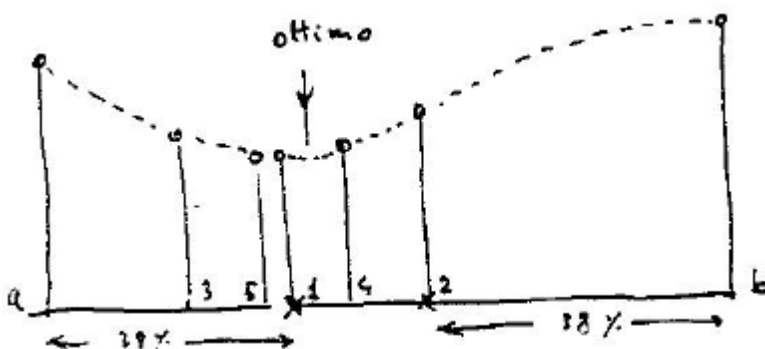


$[a, b] = I$  intervallo di incertezza

Problema:

In quali punti P e Q valutare J in modo che una volta eliminato l'intervallo  $[a, P]$  o  $[Q, b]$  si possa con una sola nuova misura ritrovarsi di nuovo nelle condizioni di figura con un intervallo di incertezza più piccolo?

$$\frac{x}{b-a} = \frac{\overbrace{(b-a) - 2x}^{PQ}}{\underbrace{(b-a) - x}_{Pb}} \rightarrow x = \underbrace{\frac{3-\sqrt{5}}{2}}_{\text{sezione aurea}} (b-a) = 0.38(b-a)$$

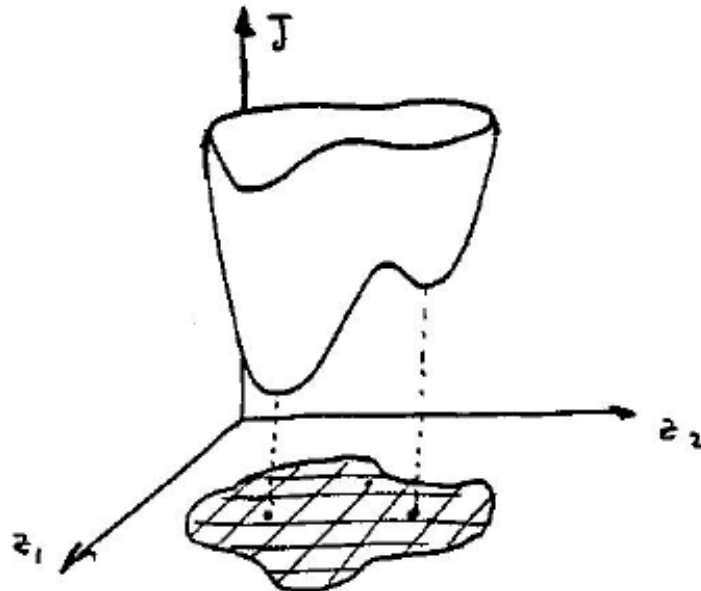


Ad ogni iterazione si riduce l'intervallo di incertezza (o indeterminazione)

$[a, b] \rightarrow [a, 2] \rightarrow [3, 2] \rightarrow [3, 4] \rightarrow [5, 4] \rightarrow \dots$

## Ottimizzazione n-dimensionale non vincolata

### Griglia



La funzione obiettivo  $J$  viene valutata in corrispondenza dei punti di una griglia molto larga (regolare o no a seconda dei casi).

Indi si sceglie il punto  $z^{(i)}$  (tra quelli esaminati) col più piccolo valore di  $J$  e si definisce una nuova griglia (più fitta della precedente) intorno a tale punto.

Così si procede fino a che si ottiene un minimo relativo con l'approssimazione voluta.

Questo metodo funziona solo se  $n$  è basso: se ad esempio si hanno 10 variabili, ognuna delle quali può assumere 10 valori, la griglia è costituita da  $10^{10}$  punti! Se la valutazione della funzione richiede 1 secondo, ciò significa 3 secoli!

Questo metodo tende, più di altri, a fornire il minimo assoluto di  $J$  perché i tentativi non dipendono dalla forma locale della funzione.

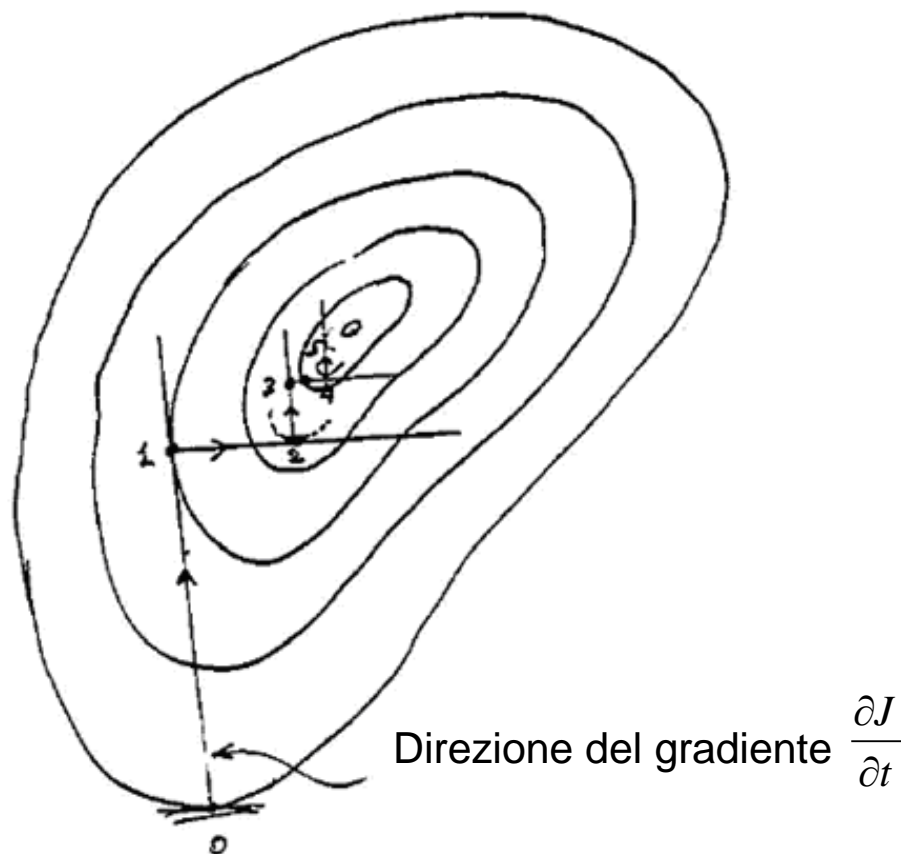
## Gradiente

Questo metodo è definito da

$$z^{t+1} = z^t + \alpha_{t+1} s^{t+1}$$

$$\text{con } s^{t+1} = \left[ \frac{\partial J}{\partial z} \right]$$

e  $\alpha_{t+1}$  ottimizzato con metodo monodimensionale (per esempio, sezione aurea).



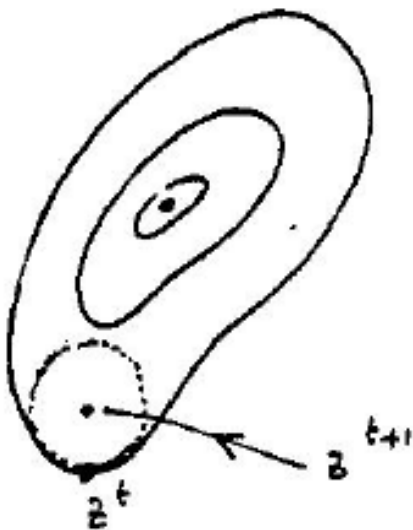
## Newton

Questo metodo usa anche le derivate seconde  $\frac{\partial^2 J}{\partial z_i \partial z_j}$  (cioè

l'Hessiano  $H = \left[ \frac{\partial^2 J}{\partial z_i \partial z_j} \right]$  valutato nel punto  $z^t$ ) per calcolare la variazione

$z^t \rightarrow z^{t+1}$

$$\left. \begin{aligned} z^{t+1} &= z^t + \alpha_{t+1} s^{t+1} \\ \alpha_{t+1} s^{t+1} &= - \left[ \frac{\partial^2 J}{\partial z} \right]_{z^t}^{-1} \left[ \frac{\partial J}{\partial z} \right]^T \end{aligned} \right\} (1)$$



Questo metodo consiste nell'approssimare localmente  $J$  con una funzione quadratica

$$J(z) \cong J(z^t) + \left[ \frac{\partial J}{\partial z} \right] (z - z^t) + \frac{1}{2} (z - z^t) \left[ \frac{\partial^2 J}{\partial z^2} \right]_{z^t} (z - z^t) \quad (*)$$

e nel scegliere come nuovo valore  $z^{t+1}$  il punto di minimo della (\*) che è appunto dato dalla (1)



## Ottimizzazione n-dimensionale vincolata

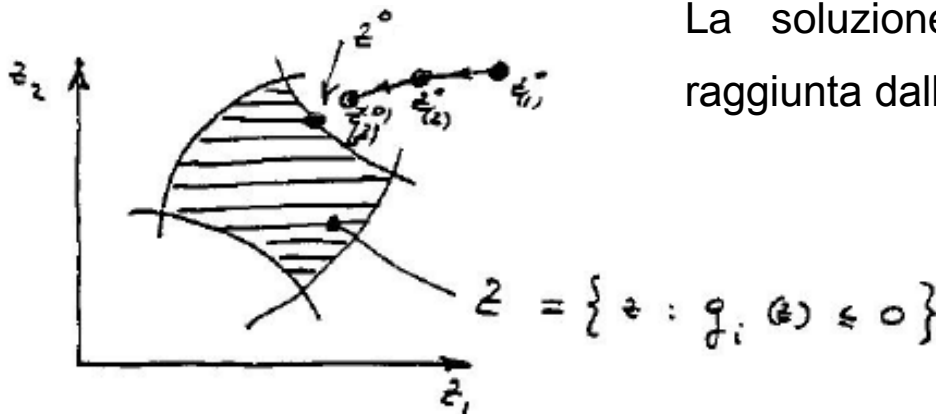
### Metodo di penalità

Questo metodo determina la soluzione ottima  $z^\circ$  del problema:

$$\begin{aligned} \min [J(z)] \\ g_i(z) \leq 0 \quad i = 1, \dots, m \end{aligned}$$

per approssimazioni successive  $z^\circ(1), z^\circ(2), \dots, z^\circ(k), \dots$

Le soluzioni approssimate  $z^\circ(k)$ , in generale, non soddisfano tutti i vincoli  $g_i(z) \leq 0$ .



La soluzione ottima viene raggiunta dall'esterno di  $Z$

La soluzione  $z^\circ(k)$  viene determinata risolvendo il problema non vincolato (\*\*)

$$\min [J(z) + \mu_k P(z)] \quad (**)$$

Dove  $\mu_k > 0$  e  $P(z)$  è la funzione di penalità:

$$P(z) = \begin{cases} 0 & \text{se } z \in Z \\ > 0 & \text{se } z \notin Z \end{cases}$$

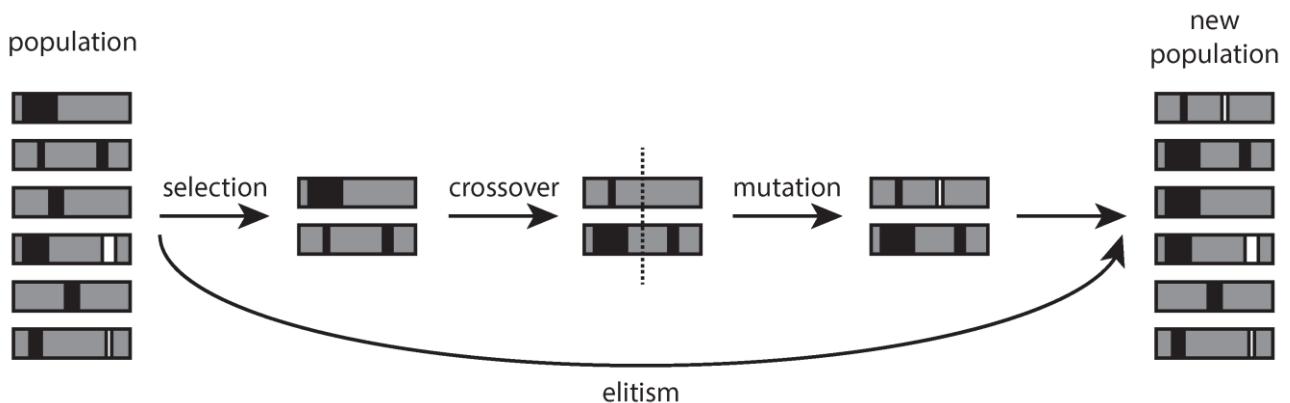
$$p.e. \quad P(z) = \sum_i g_i(z) \quad i \in \text{vincoli non soddisfatti}$$

I pesi  $\mu_k$  sono crescenti con  $k$  e  $\lim_{\mu_k \rightarrow \infty} z^\circ(k) = z^\circ$

## Algoritmi genetici (GA)

Simulano, in modo molto rapido, l'evoluzione di una specie vivente, replicando (approssimativamente) i principali meccanismi che la determinano.

- Si parte con una “popolazione” di soluzioni prese casualmente e si valutano i corrispondenti valori dell'obiettivo
- Si selezionano le soluzioni migliori
- Si “accoppiano” (crossover) le soluzioni prendendo parti di due soluzioni diverse tra quelle selezionate
- Si opera una mutazione, modificando parti (geni) delle soluzioni correnti
- Le soluzioni così ottenute formano una nuova popolazione da cui si riparte.



- Un limitato numero di soluzioni molto buone (élite) viene mantenuto da una generazione alla successiva (cioè non passa attraverso le fasi precedenti).

Esempio: miglior veicolo a 2 ruote ([https://rednuht.org/genetic\\_cars\\_2/](https://rednuht.org/genetic_cars_2/))

