



Francesco Pio Manna  
Claudio Pisa

M63001485  
M63001512

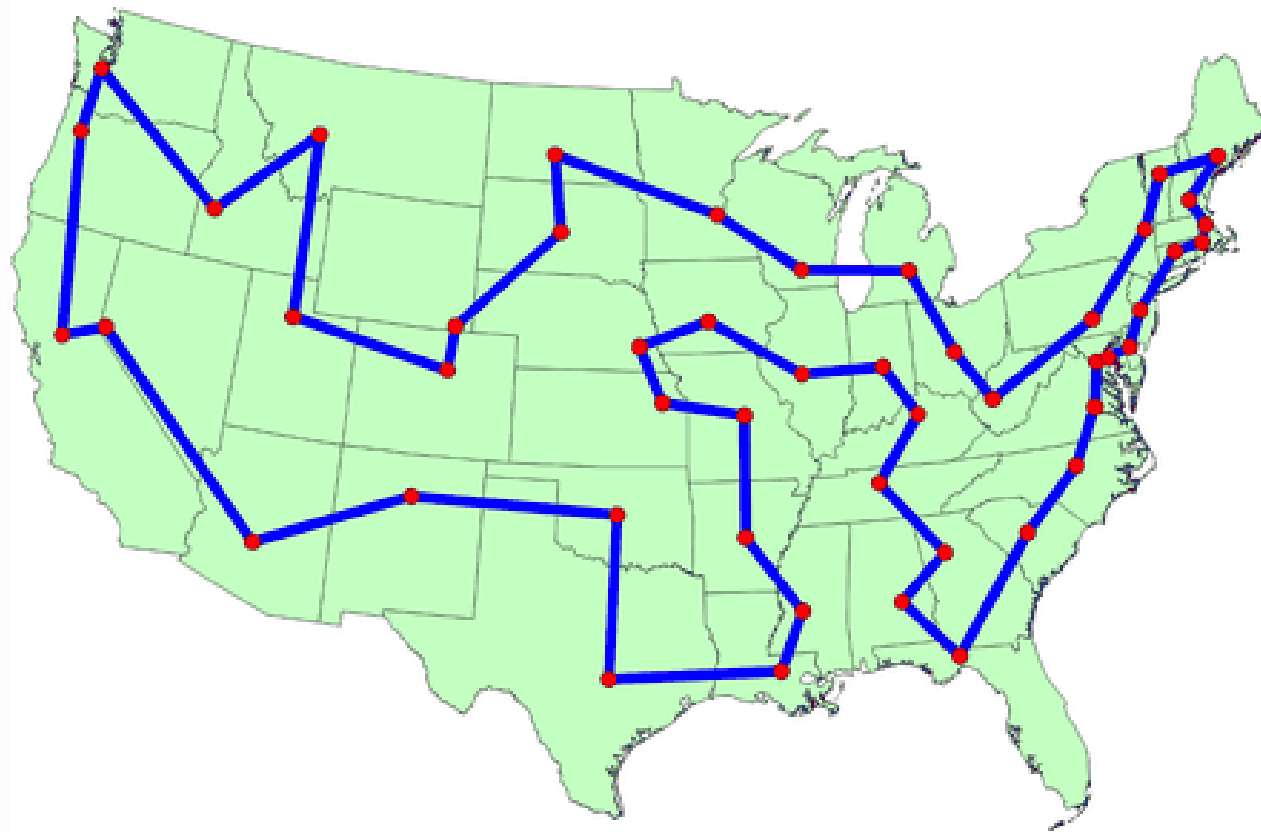
# **Soluzione del TSP mediante le metaeuristiche Simulated Annealing e Tabu Search**

Corso di Algoritmi di Ottimizzazione Combinatoria e su Rete  
A.A. 2022-23



# Introduzione al problema

- Il TSP prevede che un commesso viaggiatore debba visitare un certo numero di città e tornare al punto di partenza, minimizzando la distanza totale percorsa. Ogni città deve essere visitata una sola volta, e il commesso può spostarsi direttamente da una città all'altra senza dover passare per altre città non visitate.



# Formulazione del problema

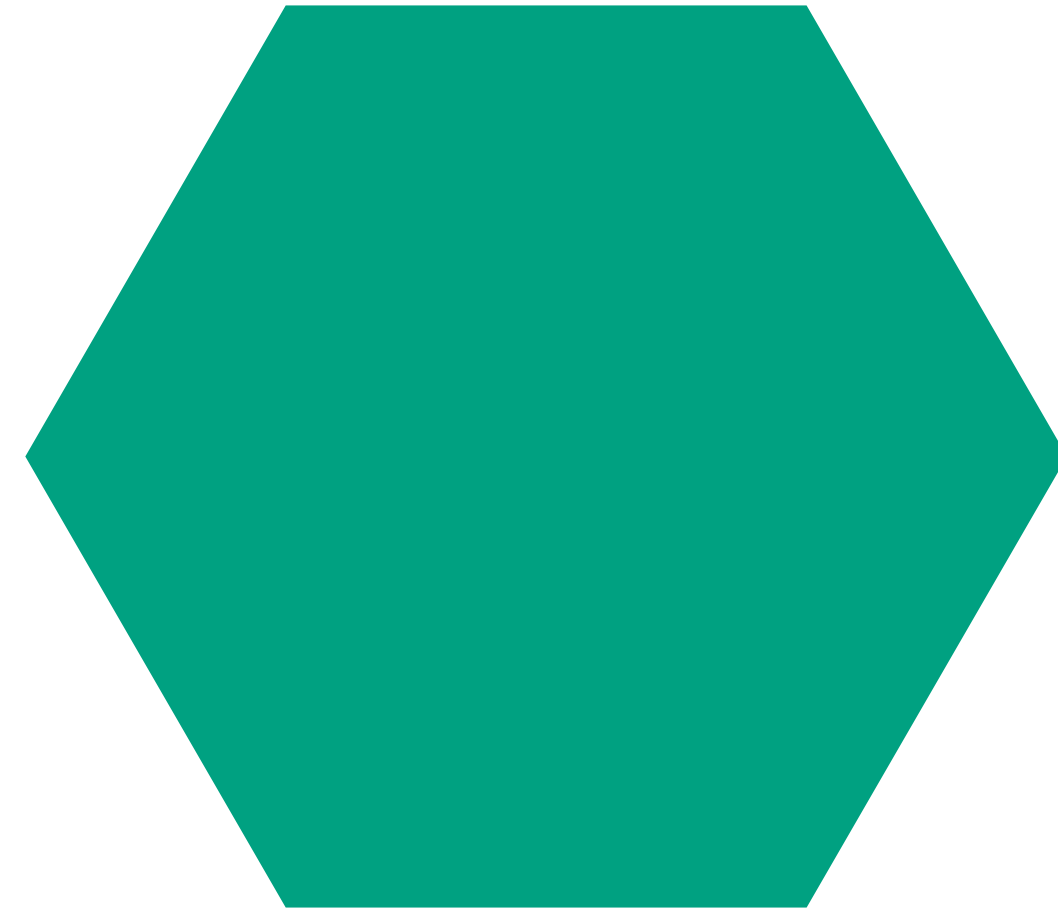
$$\min \sum_{(i,j) \in A} d_{ij} x_{ij}$$

$$\sum_{i:(i,j) \in A} x_{ij} = 1 \quad j \in V$$

$$\sum_{i:(j,i) \in A} x_{ji} = 1 \quad j \in V$$

$$x_{ij} \in \{0,1\} \quad ij \in A$$

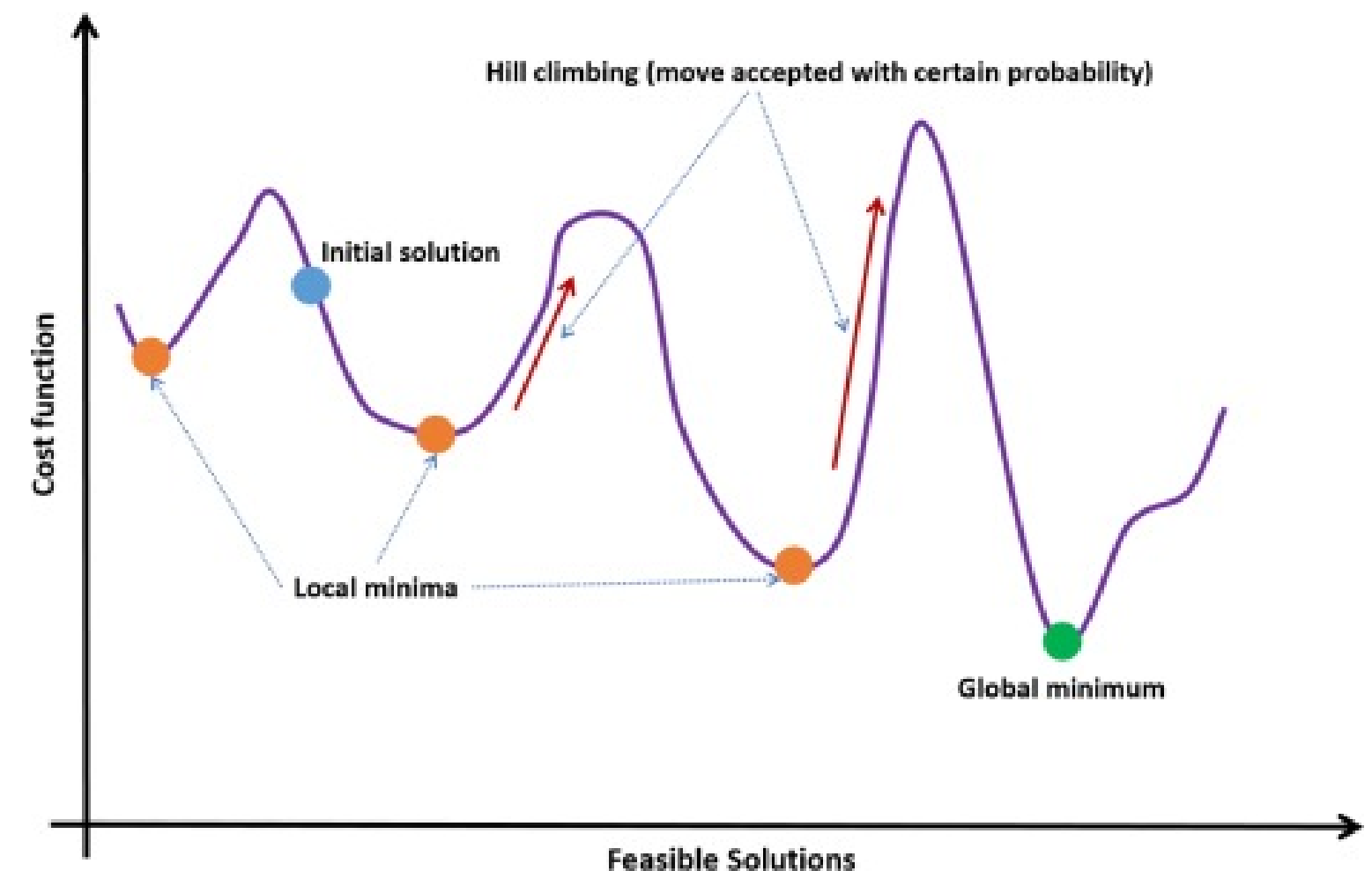
$$\sum_{\substack{i \in S, j \in S: \\ (i,j) \in A}} x_{ij} \leq |S| - 1 \quad \forall S \subset V: 2 \leq |S| \leq |V| - 1$$



# Necessità delle euristiche

- In generale, i problemi di ottimizzazione combinatoria sono NP-Hard, dunque sono richiesti tempi di calcolo esponenziali all'aumentare della dimensione del problema;
- Da questo nasce la necessità delle euristiche, che forniscono una soluzione, non necessariamente ottima, in tempi notevolmente inferiori rispetto ai metodi esatti;
- Nasce la necessità di trovare un compromesso tra la qualità della soluzione trovata e il tempo impiegato dall'algoritmo per trovarla.

- Nel nostro progetto, abbiamo utilizzato il simulated annealing e la tabu search per la risoluzione del problema del TSP



# Simulated Annealing

- La metaeuristica simulated annealing prevede i seguenti passi di risoluzione:

```
Individua soluzione iniziale S
T=T0
Repeat
Trova casualmente una nuova soluzione S' nell'intorno di S
if  $\Delta f \leq 0$  then
    S=S'
else if  $\text{rand}(0,1) < e^{\frac{-\Delta f}{T}}$  then
    S=S'
T =  $\alpha T$ 
Until  $T \leq T_f$ 
return S
```

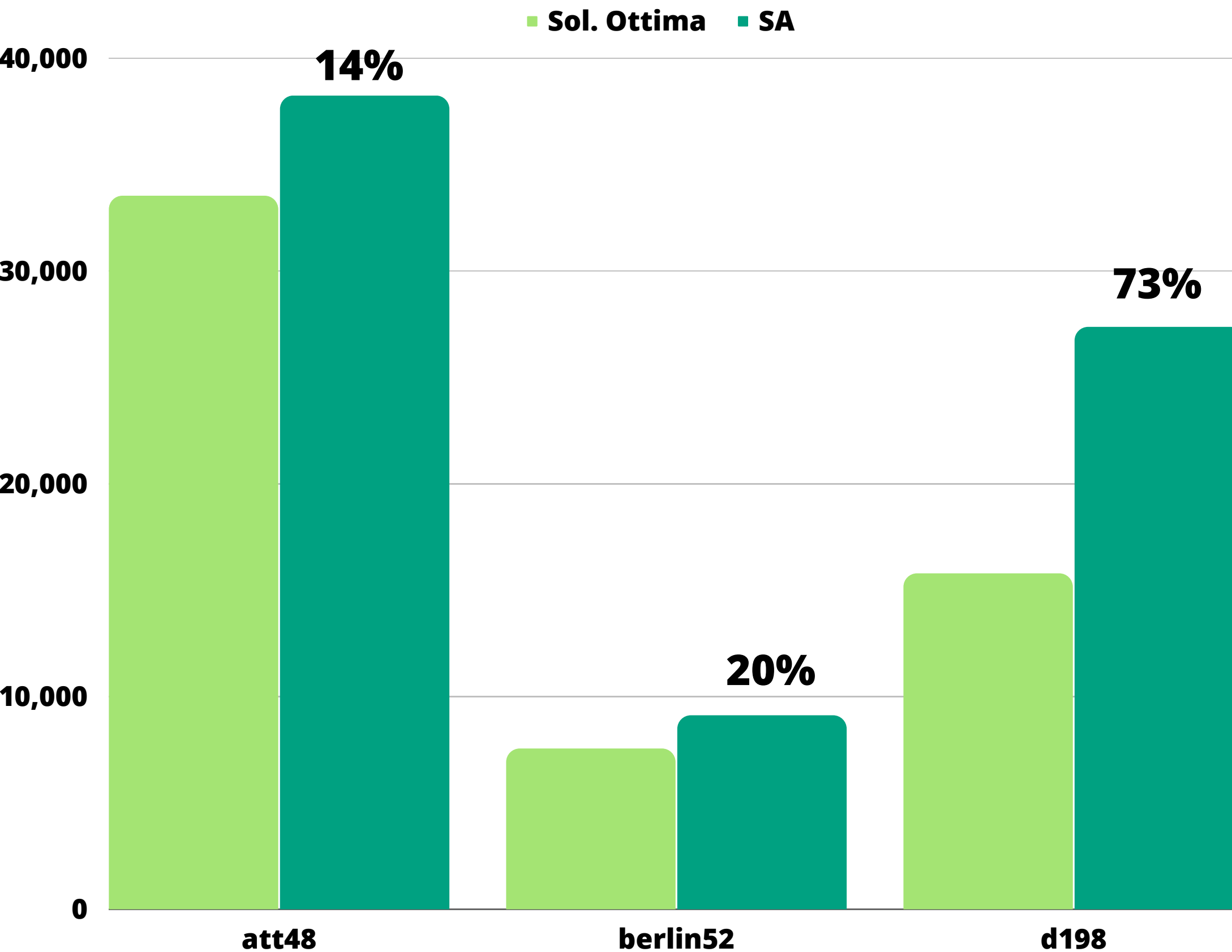
- La soluzione individuata S' è costruita a partire da S scambiando l'ordine di due città casualmente;
- La soluzione iniziale S viene determinata in modo casuale;

# Considerazioni

- Temperatura iniziale proporzionale al valore della soluzione iniziale con un fattore 10.
- Temperatura finale tendente a 0, impostandola come  $10^{-4} \times$  valore della soluzione iniziale.
- Un cooling rate più alto (0.99) ha permesso di ridurre il numero totale di iterazioni dell'algoritmo senza diminuire le prestazioni.



# Risultati Numerici

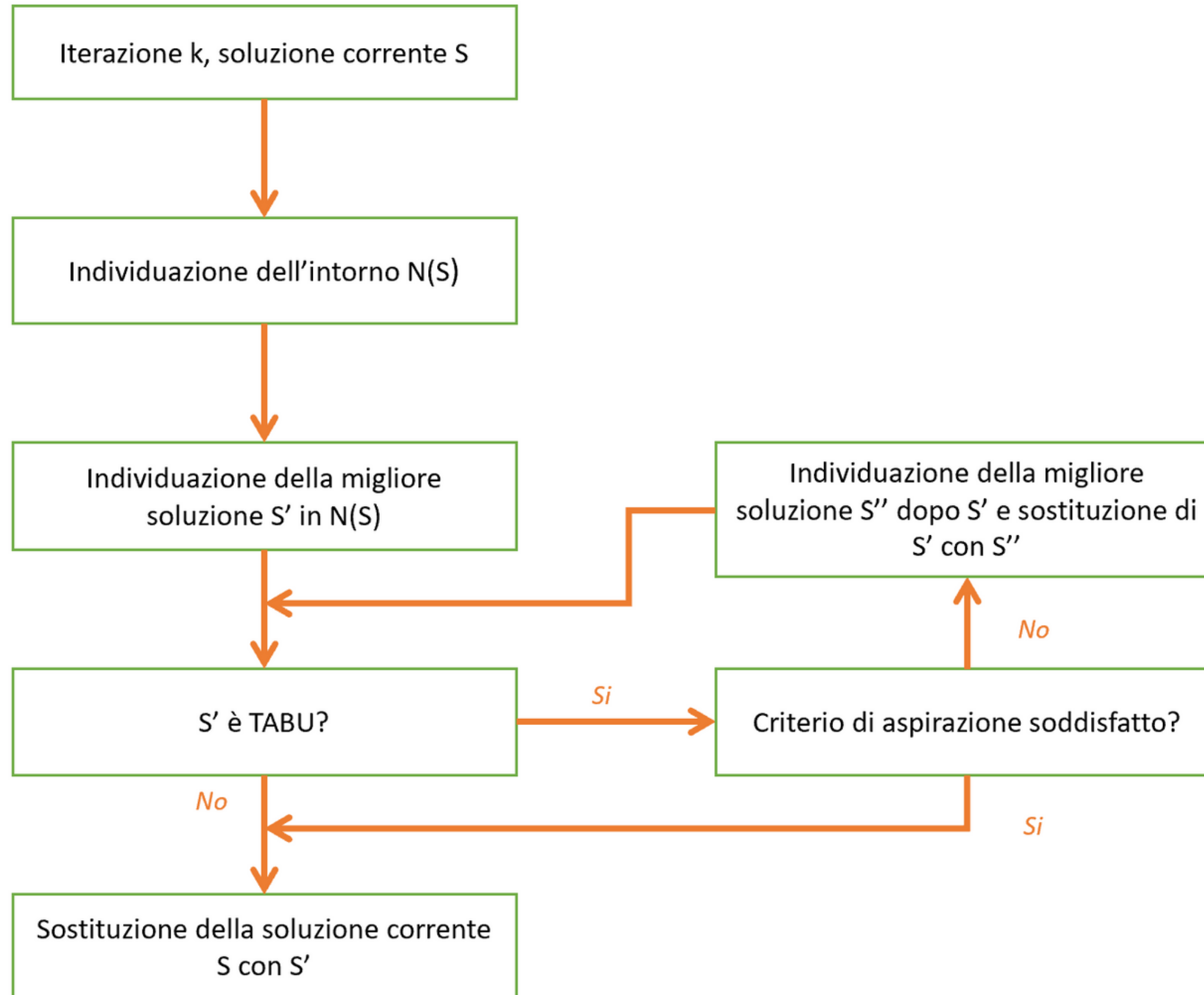


$$gap = \frac{|OPT(I) - EUR(I)|}{|OPT(I)|} \cdot 100$$



# Tabu Search

- La metaeuristica tabu search prevede i seguenti passi principali:



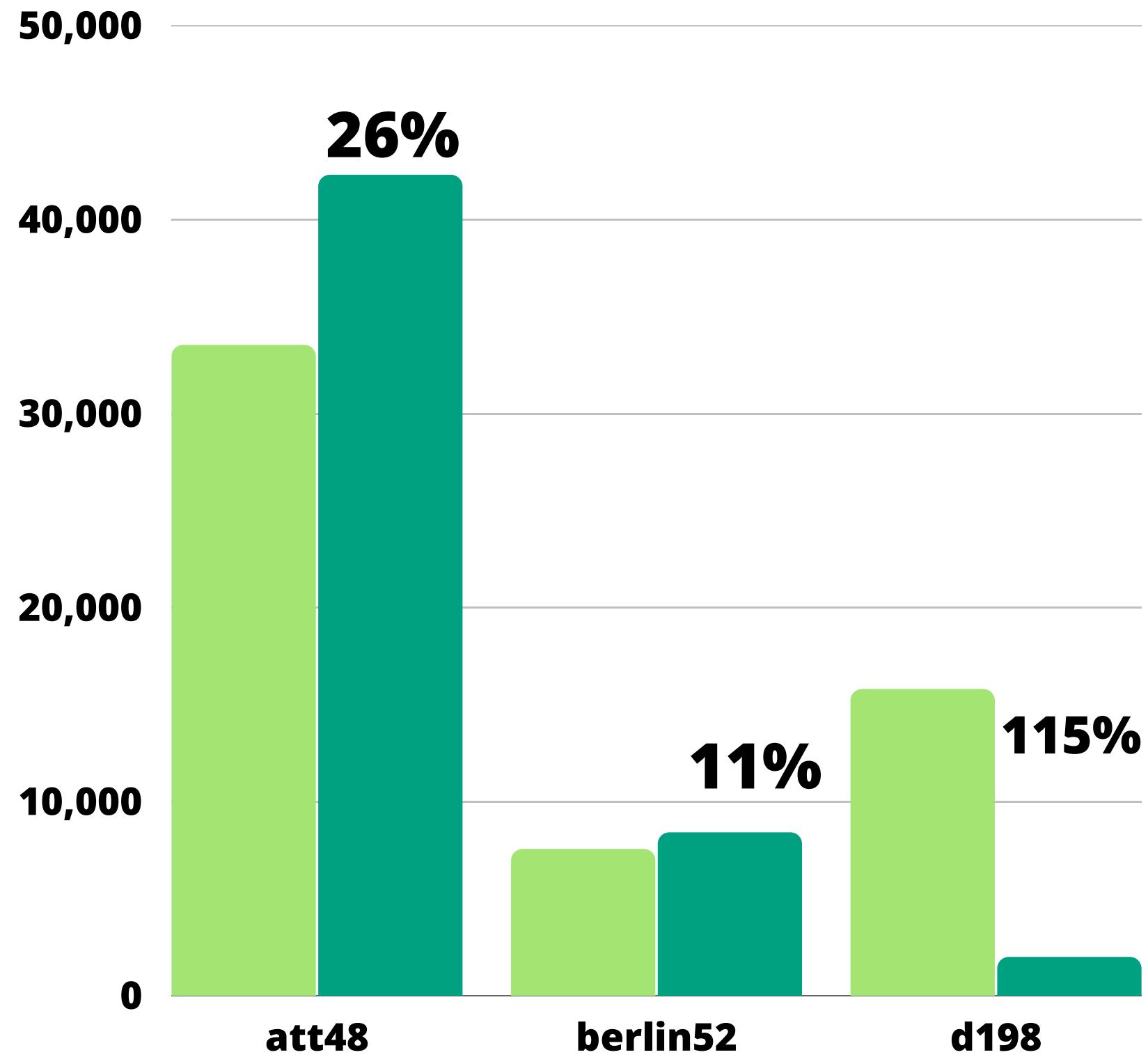


# Considerazioni

- Dimensione lista tabu pari alla radice quadrata del numero di città del problema. Una lista tabu troppo grande mi impedirebbe di analizzare intere zone del dominio delle soluzioni
- Criterio di aspirazione: se una soluzione che è in lista tabu mi porta ad un valore di soluzione migliore della soluzione globale, allora viene eseguita lo stesso.
- L'attributo memorizzato in lista tabu è la mossa.
- Abbiamo utilizzato una strategia best-improvement nell'analisi dell'intorno. Per ridurre i tempi di esecuzione, si potrebbe provare una strategia first-improvement.



# Risultati Numerici



$$gap = \frac{|OPT(I) - EUR(I)|}{|OPT(I)|} \cdot 100$$



# Confronto tempi di esecuzione

- In generale, è difficile confrontare i tempi di esecuzione in quanto dipendono da molti parametri.
- Nel simulated annealing, aumentando il valore iniziale della temperatura e/o modificando il numero di iterazioni da fare ad ogni temperatura  $T_k$ , i tempi aumentano notevolmente (ottenendo però mediamente una soluzione migliore).
- Nella tabu search, i tempi di esecuzione dipendono dal numero di iterazioni che stabiliamo prima dell'esecuzione dell'algoritmo. Una possibile variante potrebbe essere quella di fermare l'algoritmo nel caso in cui ci siano state un certo numero di iterazioni peggiorative.
- Non è sempre immediato individuare un trade-off fra tempo di esecuzione e qualità della soluzione.

