

# RISOLUZIONE DEL PROBLEMA U.F.L. CON TAGLI DI GOMORY

Valutazione dell'efficacia e dei limiti numerici

Sofia Tosti  
0369460  
Algoritmi e metodi di ottimizzazione discreta  
a.a. 2024/2025

# IL PROBLEMA DI LOCALIZZAZIONE DI IMPIANTI NON CAPACITATO

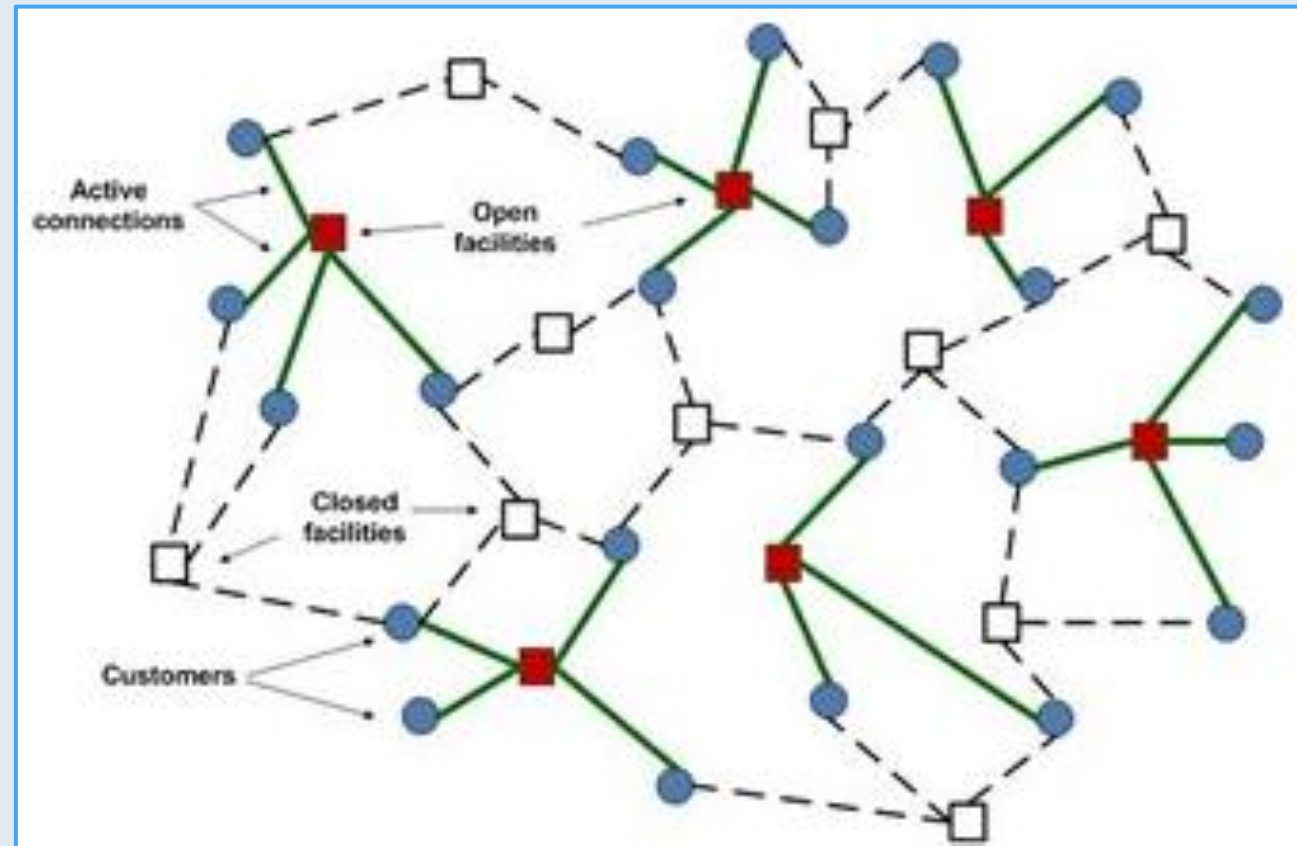
2

Il problema di localizzazione di impianti non capacitato è una delle questioni strategiche più classiche dell'ottimizzazione. Data una serie di possibili sedi per aprire dei magazzini e un gruppo di clienti da servire, quali magazzini dovremmo aprire per soddisfare la domanda di tutti i clienti al costo totale più basso?

I costi da considerare sono due:

- Costo fisso per aprire ogni magazzino
- Costo variabile per servire un cliente da un magazzino aperto

L'obiettivo è trovare il bilanciamento perfetto tra il non aprire troppi magazzini (per risparmiare sui costi fissi) e il non aprirli troppo lontano dai clienti (per risparmiare sui costi di trasporto).



# FORMULAZIONE MATEMATICA LINEARE INTERA (ILP)

3

Prima di tutto, definiamo le nostre decisioni con variabili binarie (0 o 1):

- ◆  $x_i = 1$  se decidiamo di aprire l'impianto  $i$ , 0 altrimenti.
- ◆  $y_{ij} = 1$  se decidiamo di assegnare il cliente  $j$  all'impianto  $i$ , 0 altrimenti.

Il nostro obiettivo è minimizzare il costo totale:

$$\min ( \text{Costo Fisso Totale} + \text{Costo di Assegnazione Totale} ) \longrightarrow \min ( \sum (\text{costo\_apertura\_i} * x_i) + \sum (\text{costo\_servizio\_ij} * y_{ij}) )$$

*Stiamo sommando i costi di tutti gli impianti che decidiamo di aprire ( $x_i=1$ ) e i costi di tutte le assegnazioni che decidiamo di fare ( $y_{ij}=1$ ).*

Dobbiamo rispettare due regole fondamentali (i vincoli):

1. Ogni cliente DEVE essere servito da UN solo impianto.

$$\sum_i y_{ij} = 1 \text{ (per ogni cliente } j)$$

2. Possiamo assegnare un cliente a un impianto SOLO se quell'impianto è aperto.

$$y_{ij} \leq x_i \text{ per ogni coppia cliente } j \text{ e impianto } i \text{ (strong form)} / \sum_j y_{ij} \leq q * x_i \text{ dove } q \text{ è il numero di clienti per ogni } i \text{ (weak form)}$$

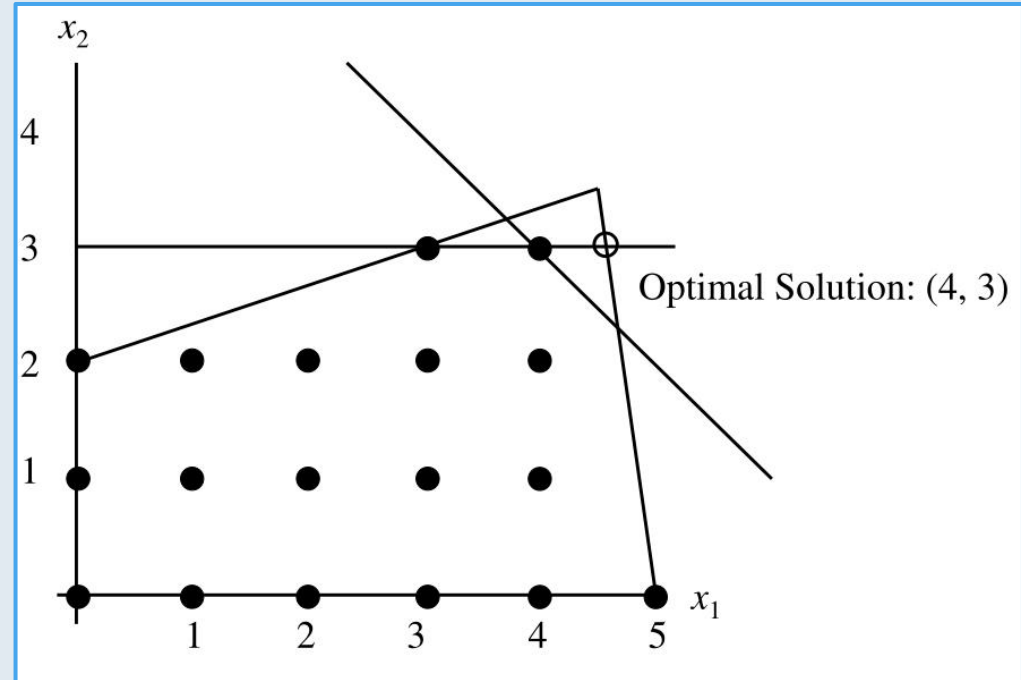
# L'APPROCCIO RISOLUTIVO: PIANI DI TAGLIO

4

- Invece di forzare le variabili a 0 o 1, applichiamo il rilassamento lineare (LP). Otteniamo un limite inferiore (lower bound) al costo minimo.
- L'idea non è testare tutte le combinazioni, ma restringere progressivamente lo spazio delle soluzioni frazionarie.

Il ciclo dei Piani di Taglio:

1. Risolvo LP : ottiene una soluzione frazionaria
2. Identifica la frazionarietà: trova perché la soluzione non è valida (es.  $x_i = 0,7$ )
3. Genera un «Taglio» : aggiungi un nuovo vincolo al modello
4. Ripeti



# LA TECNOLOGIA: I TAGLI DI GOMORY

- Ho implementato due delle più importanti famiglie di tagli di gomory.
- Tagli Frazionari Gomory Fractional Cuts (GFC), derivati direttamente dal tableau del simplesso per ogni variabile base frazionaria, considerato l'approccio «classico» e più stabile.
- Tagli Interi Gomory Mixed-Integer Cuts(GMI), una variante più potente, spesso più efficace in pratica, che considera la natura mista (Intera/continua) delle variabili.
- Ho testato diverse strategie, usare solo GFC, solo GMI e una strategia BEST che genera entrambi i tipi di tagli per ogni istanza e seleziona i più violati ad ogni interazione

# ARCHITETTURA DEL CODICE E DEGLI ESPERIMENTI

- **Linguaggio e Librerie:** Python, CPLEX, Pandas, Matplotlib/Seaborn.
- **Particolarità del Codice:**
  - Generazione di report separati per ogni modalità per un'analisi rigorosa.
  - Generazione di istanze randomica , tramite file «config.ini», (SMALL, MEDIUM,LARGE)
  - Uso delle istanze di or-library<sup>1</sup> per validare l'efficacia del nostro progetto

<sup>1</sup> <https://people.brunel.ac.uk/~mastjjb/jeb/info.html>

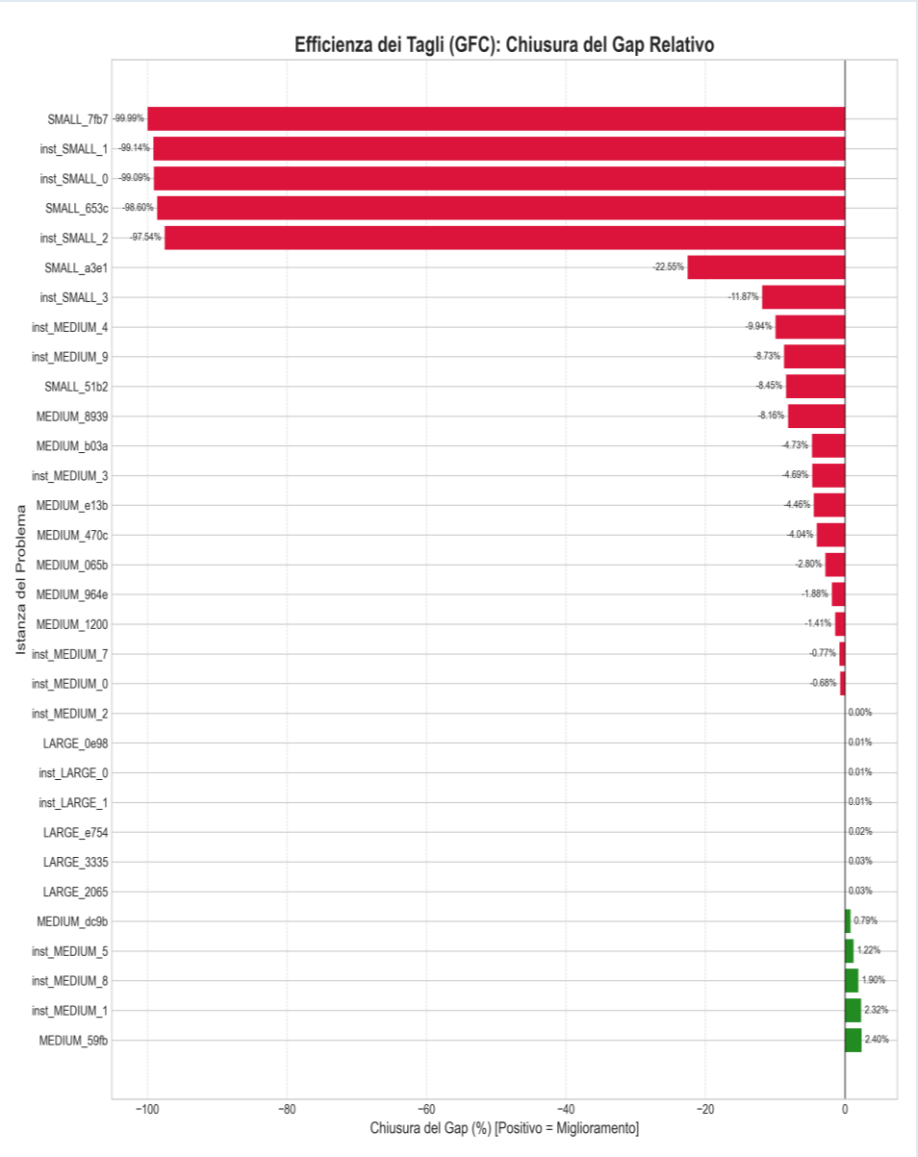


# IL CASO OR-LIBRARY

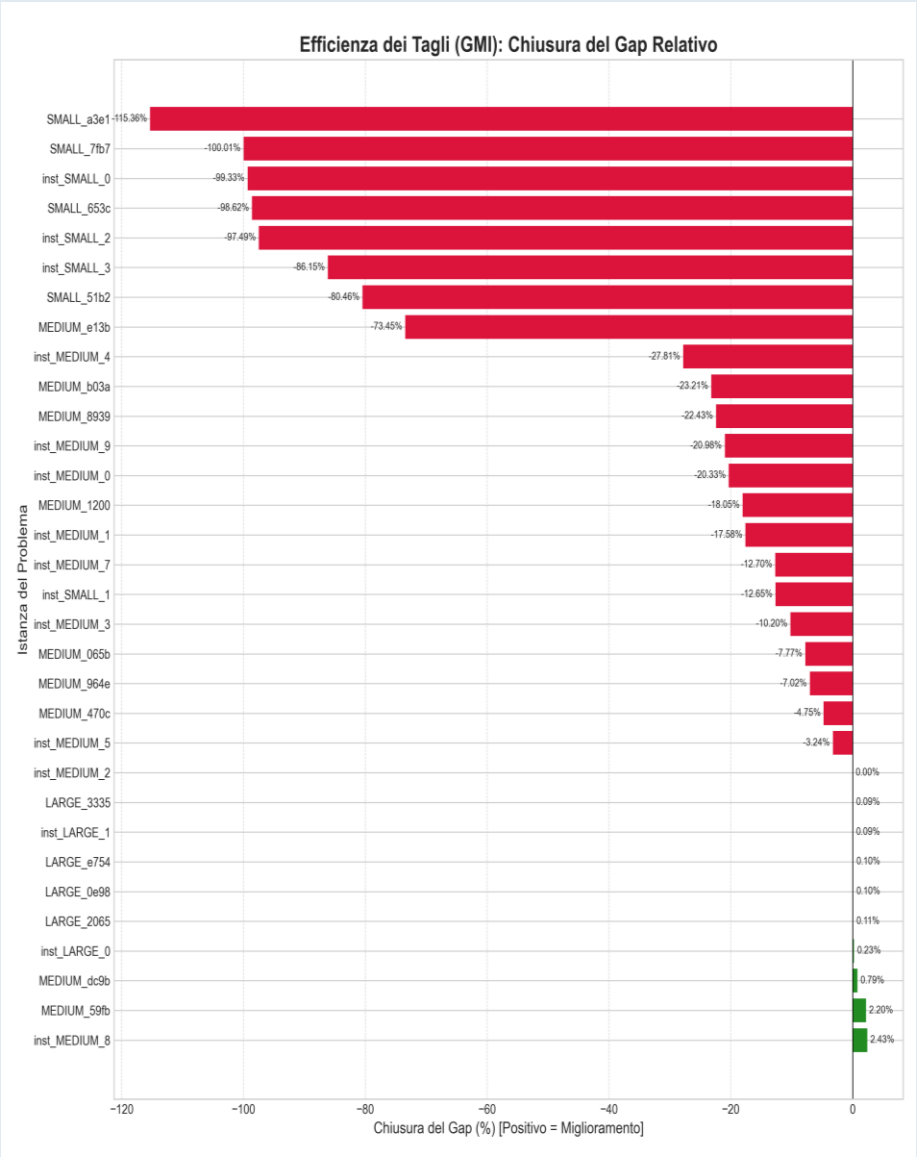
- Infatti come potete vedere dai nostri dati, il rilassamento lineare iniziale ha prodotto direttamente una soluzione ottima **intera**, Il poliedro delle soluzioni ammissibili per queste istanze, combinato con i loro specifici vettori di costo, possiede la cosiddetta **proprietà di integralità**. Questo risultato è cruciale perché **contrasta nettamente** con le istanze che abbiamo generato casualmente (le SMALL, MEDIUM, LARGE). Le istanze generate, essendo più 'disordinate' e prive di una struttura così pulita, hanno quasi sempre un gap di ottimalità e richiedono l'uso dei tagli."

cap101	GMI	0.0000	0.0000	0.0000	0	0	3	optimal	LP Ottimo Intero
cap102	GMI	0.0000	0.0000	0.0000	0	0	3	optimal	LP Ottimo Intero
cap103	GMI	0.0000	0.0000	0.0000	0	0	3	optimal	LP Ottimo Intero
cap104	GMI	0.0000	0.0000	0.0000	0	0	4	optimal	LP Ottimo Intero
cap111	GMI	0.0000	0.0000	0.0000	0	0	25	optimal	LP Ottimo Intero
cap112	GMI	0.0000	0.0000	0.0000	0	0	24	optimal	LP Ottimo Intero
cap113	GMI	0.0000	0.0000	0.0000	0	0	23	optimal	LP Ottimo Intero
cap114	GMI	0.0000	0.0000	0.0000	0	0	25	optimal	LP Ottimo Intero
cap121	GMI	0.0000	0.0000	0.0000	0	0	25	optimal	LP Ottimo Intero
cap122	GMI	0.0000	0.0000	0.0000	0	0	24	optimal	LP Ottimo Intero
cap123	GMI	0.0000	0.0000	0.0000	0	0	25	optimal	LP Ottimo Intero
cap124	GMI	0.0000	0.0000	0.0000	0	0	25	optimal	LP Ottimo Intero
cap131	GMI	0.0000	0.0000	0.0000	0	0	26	optimal	LP Ottimo Intero
cap132	GMI	0.0000	0.0000	0.0000	0	0	24	optimal	LP Ottimo Intero
cap133	GMI	0.0000	0.0000	0.0000	0	0	24	optimal	LP Ottimo Intero
instance_name	cut_mode	initial_gap	final_gap	gap_closure	total_cuts	total_iterations	total_time_ms	final_status	solution_category

# RISULTATI



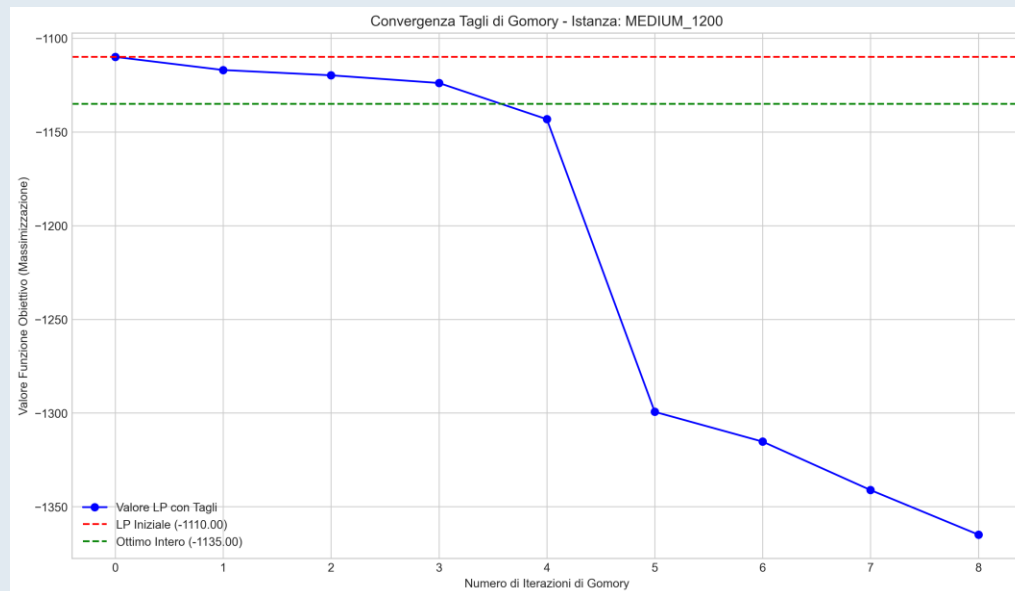
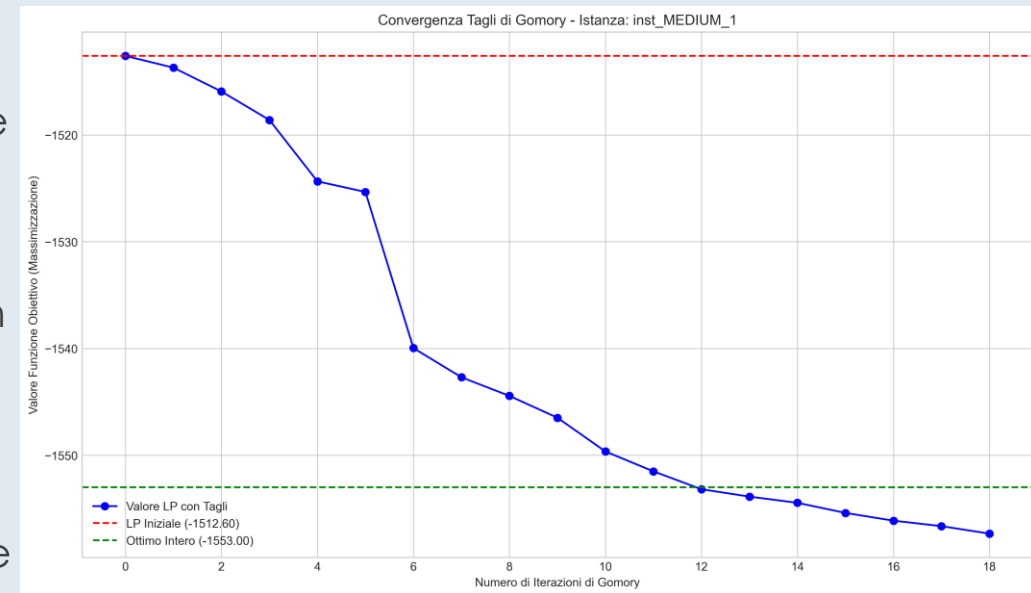
Entrambi i grafici rappresentano l'efficacia dei tagli nei due metodi. In entrambi i casi, ci sono pochissime barre verdi, quindi il gap è stato chiuso per poche limitate istanze ed il miglioramento è spesso piccolo. Per la maggior parte abbiamo un peggioramento con punte del -100%. I tagli GMI sembrano essere più aggressivi, e quindi inclini a instabilità numerica, anche se teoricamente i più forti



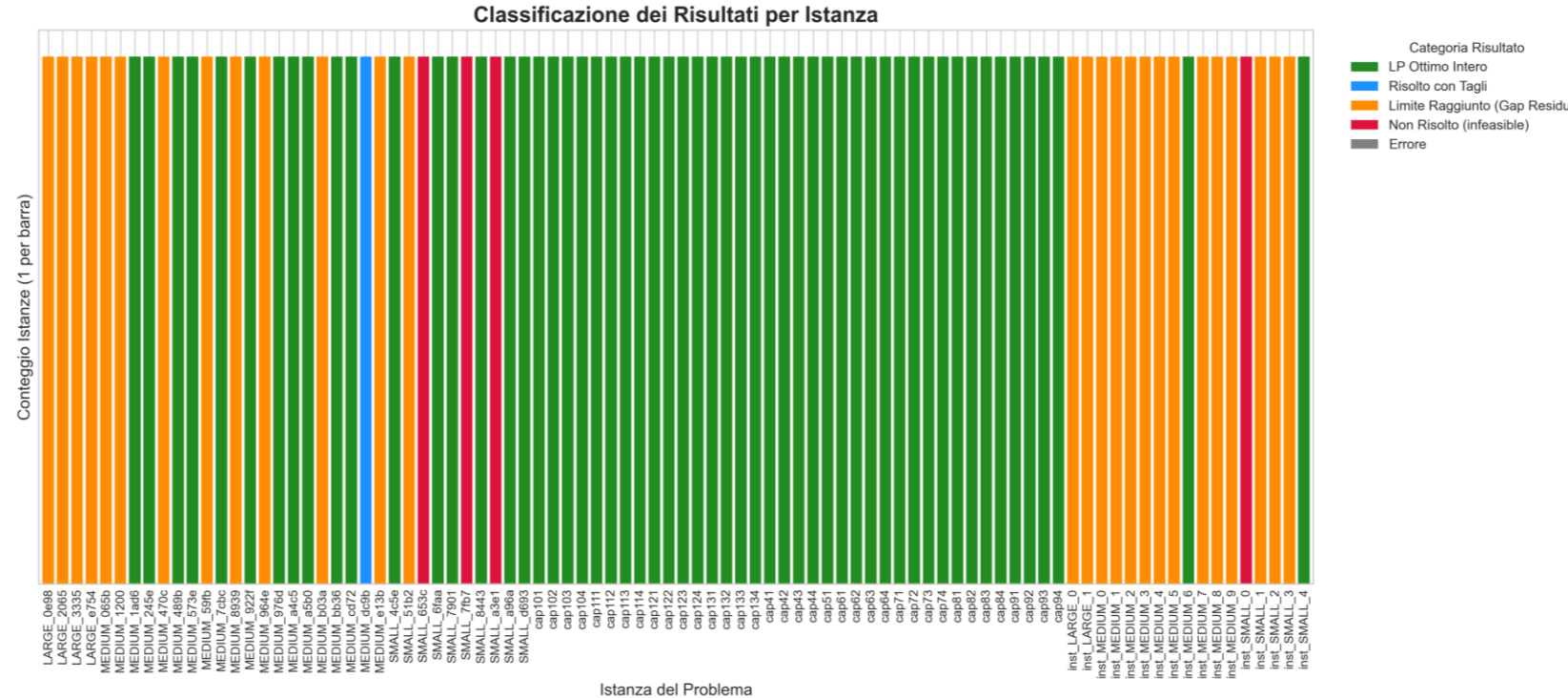


La convergenza dei tagli è stata prodotta per ogni modalità, ho scelto questi tre grafici che rappresentano i casi principali. **Inst\_MEDIUM\_1** ha una convergenza lenta e incompleta, in quanto l'algoritmo fa progressi ma non riesce a chiudere il gap entro i limiti computazionali, un fenomeno noto come «*tailing-off*».

**MEDIUM\_UFL\_instance\_dc9b** è un candidato perfetto sia in GMI che GFC. Una linea che parte dal valore LP iniziale e scende bruscamente fino a toccare il valore ottimo in pochissimi passi. **MEDIUM\_1200** ha una chiusura gap negativa, rappresenta l'instabilità numerica, invece di migliorare peggiora, questo evidenzia il rischio di applicare i tagli in maniera non calibrata.







**NOTA:** in config.py ho inserito i limiti di tempo di interazione, che posso essere cambiati manualmente. I maxcuts invece sono stati impostati nel metodo dell'algoritmo di gomory.

Anche questo grafico è stato prodotto per le tre modalità. Ho selezionato la BEST. Ci da una visione d'insieme del risultato finale per ogni singola istanza.

Verde (LP ottimo intero): come abbiamo discusso sono state risolte immediatamente dal LP. BLU (risolto con tagli) una singola istanza è stata risolta completamente con i tagli.

Arancione (Limite Raggiunto) L'algoritmo si è fermato a causa dei limiti imposti (numero di iterazioni o tempo), lasciando un gap di ottimalità residuo. Rosso (Non risolto) in questi casi l'accumolo di tagli ha reso il problema infattibile per il risolutore

# CONCLUSIONI

- L'analisi da me svolta ha confermato che l'approccio «puro» è afflitto da instabilità numerica e convergenza lenta. Su molte istanze, i tagli hanno peggiorato la soluzione LP invece di migliorarla. Le istanze «accademiche» (OR-Library) spesso non richiedono tagli, poiché il loro rilassamento lineare è già intero. Le istanze casuali, più generiche, si sono dimostrate molto più difficili, evidenziando la debolezza dell'approccio.
- Questo studio dimostra perché i risolutori commerciali moderni (CPLEX, Gurobi) non si affidano a una singola famiglia di tagli. La loro efficacia deriva dall'unire decine di tipi di tagli diversi, applicati tramite euristiche complesse che decidono quali, quanti e quando aggiungere i tagli.

**GRAZIE**

