Set Covering Problem

PIA: Part IV

Operations Research, 032
Team 6: Alan Alejandro Vargas González, 2086183

May, 2025

1 Part IV: Final Results and Presentation

1.1 Final Data Execution

To evaluate the scalability and robustness of the model, I tested with real-world set covering problems:

- `rail2536.txt` (2536 rows, 1,081,841 columns)
- `rail4284.txt` (4284 rows, 1,092,610 columns)
- `rail4872.txt` (4872 rows, 968,672 columns)

Instance	Selected Subsets	Cost	Time (s)
`rail2536`	80	13697.0	30.18
`rail4284`	128	5302.0	32.95
`rail4872`	91	27638.0	33.30

1.2 Result Visualization

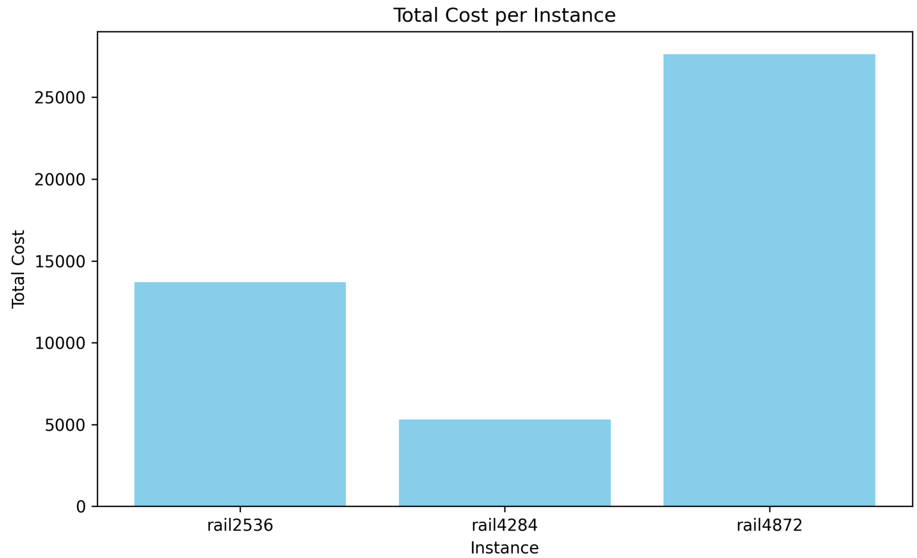

Figure 1: Total cost per instance (railway dataset)

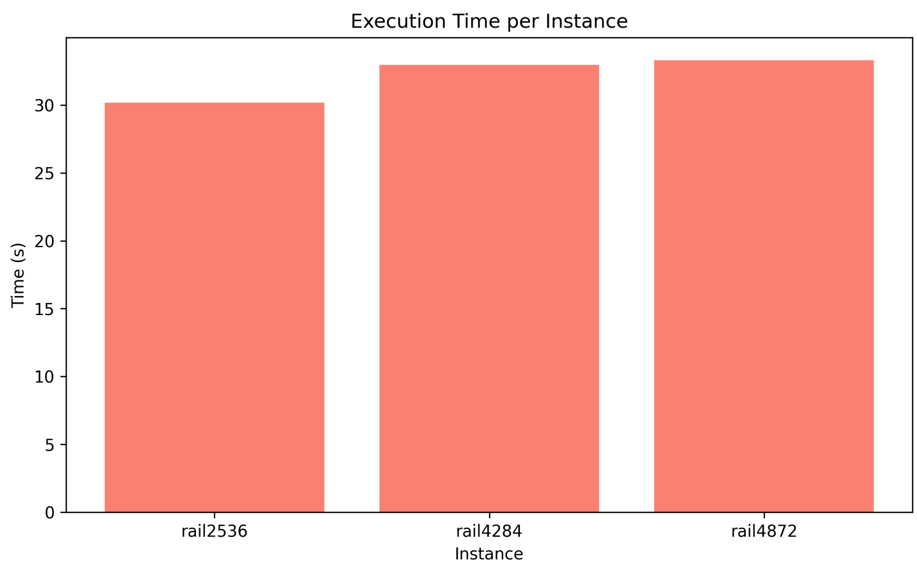

Figure 2: Execution time per instance (railway dataset)

1.3 Result Interpretation

- **Scalability:** The solver remained stable and consistent in datasets with millions of variables.
- **Cost Variation:** Total cost varied significantly across datasets, highlighting structural

differences in coverage density (e.g., some datasets may have cheap subsets capable of covering a broad set of elements).

- **Execution Time:** Execution time remained within 30-33 s, showing good performance given the large size of each dataset.

1.4 Model Limitations and Assumptions

- Assumes deterministic and static input: no dynamic data.
- CBC solver was used; performance may vary with alternative solvers such as AMPL or Gurobi.
- No preprocessing was applied (e.g., eliminating redundant variables or rows), and no heuristic shortcuts were implemented.

1.5 Deliverables

- `resultados.csv` with performance metrics
- `instance_cost.png`, `instance_time.png` with visual results