Combinatorial Optimization Thermal Planning Analysis

Kotanidis Dionysios[1], Gavriilidis Paraskevas[2],
[1](Student): Electrical Engineering , University of Western Macedonia , Kozani , Greece
[2](Student): Electrical Engineering , University of Western Macedonia , Kozani , Greece

***Abstract*—This paper addresses complex combinatorial optimization problems that combine capacity facility location and network optimization with a focus on thermal planning and transfer. In these systems, energy dissipates over distance and seasonal demand fluctuations further complicate operational efficiency. Our work evaluates existing combinatorial optimization tools and investigates additional optimization opportunities. We first formulate a set of realistic problem instances that capture capacity constraints, energy losses, and seasonal demand variations, and then solve these instances using a commercial solver. In the second phase, we develop and implement a tailored branch-and-bound algorithm to reduce solution times and improve performance. Comparative analyses demonstrate that the customized approach offers promising improvements over conventional methods.**

***Index Terms*—thermal planning , combinatorial optimization , facility location , network optimization**

I. Introduction

Combinatorial Optimization problems play a significant role in a range of industrial and academic applications , From logistics and supply chain management and planning to energy distribution and city planning. Among these, the Capacity Facility Location Problem and the Network Optimization Problem are two of the most recognizable and interesting problems due to the impact they have on real life applications . Capacity Facility Location Problem is based on the classical Facility Location Problem with the only difference that we measure and need to satisfy another constraint which is the capacity . Network Optimization is crucial and indispensable in managing flows - whether these flows are goods on a supply chain , information , or as in our case thermal energy. Our approach is a basis for network optimization in systems that account for distance-energy dissipation and fluctuating seasonal demands .

Although there are many existing algorithms that on condition can provide optimal and also efficient solutions in terms of computational resources , realistic problems require many more constraints to be satisfied so the creation of a mathematical model and use of a solver is necessary . so we explore how we can scale such mathematical models and what a solver can handle and also we try to determine if there is the possibility that a custom branch and bound algorithm can produce better or comparable results.

So our method involves of a two-phase approach:

1 . Problem Formulation and Standard Solvers we aim to formulate a set of problem instances that would be similar to reality . The problems include capacity constraints, energy transfer losses over distance, seasonal demand . Then we apply a commercial solver to evaluate and measure the performance based on the initial problem characteristics.

2 . Customized Branch And Bound Solver The second phase includes the design of a branch and bound solver. At first we create the simplest solver possible and then we try to apply techniques in node selection , branch selection and preprocesing aiming to reduce time needed to find the optimal solution.

In the remainder of the paper we include reviews and literature that worked similar projects like ours , we provide some data and more details concerning our methodology and the techniques that were used. Then we have a discussion about the potential and applicability of our research and last we provide a holistic conclusion.

II. Methodology

In this section, we analyze the problem generation and provide details about the branch-and-bound algorithms, both the basic version and the optimized one, along with challenges that arose and our solutions to those.

A. Problem Creation

For the problem creation, we aimed to achieve as realistic an environment as possible. For all instances, we started with a two-dimensional grid where we placed what would be necessary for our thermal planning. Our process relied on three steps:

1) **Facility Placement**
   To replicate more realistic scenarios, we adhered to the hypothesis that large infrastructure is located at city borders or further away where the thermal energy is consumed. So we placed the plants on the border of the grid. Each plant has a fixed amount of capacity, with the overall amount of capacity designed to be at least equal to the demand, accounting for the heat loss on the transfer route from the plant to the substation and from there to the customer.
2) **Substation Distribution**
   Substation distribution is completely random since we wanted to generate as many substations as possible to include as much variability in the placement of substations. Each substation has a maximum flow that

can pass through it so that it can route the thermal energy to clients.

3) **Customer Clustering**
Customers are made in order to simulate realistic urban environments where they are clustered together. So we create in each instance three clusters that will be populated with customers. This technique tries to replicate neighborhoods and commercial zones.

After generating instances, we encountered several feasibility issues. Initially, the capacity of the plants and substations had to be adjusted, but if we increased capacity too much, the problem became too easy. To address this, we introduced another binary variable in our mathematical model to decide whether an instance would use a plant or not. This allowed us to design problems with many more plants while still maintaining meaningful capacity constraints.

Another issue was customer population. Initially, with a 100 by 100 grid, there was not enough space inside the clusters to fit more clients. Our first approach was to increase the grid size, but this led to a new problem: losses due to distance constraints skyrocketed. We then hypothesized that the 100 by 100 grid had a maximum population limit, which should be considered in thermal planning approaches.

In essence, the final problem instance selects a subset of facilities to open along the grid edges and assigns substations/customers to them while respecting capacity constraints and minimizing total cost (or maximizing efficiency).

B. Branch-and-Bound Algorithm

To find optimal or near-optimal solutions for these problem instances, we implemented a **branch-and-bound** framework as a baseline. The key characteristics of this basic branch-and-bound approach include:

- **Tree Construction**: We defined branching decisions around whether to open a facility (binary variable) or how to assign customers to a chosen facility.
- **Depth-First Search (DFS)**: Our initial implementation performed a simple DFS to explore the state space. At each node, we branched on the variable that most restricted feasibility (e.g., a facility at capacity or a customer not yet assigned).
- **Bounding**: We employed a straightforward bounding function by computing a **relaxed** version of the problem (e.g., ignoring integrality constraints where possible). If this relaxed bound exceeded our current best solution, we pruned the branch.

Although this basic approach can solve smaller instances effectively, the combination of capacity constraints, thermal loss calculations, and the potential for multiple open facilities can lead to large search trees. This motivated us to introduce additional optimizations.

C. Optimized Branch-and-Bound

We developed an **optimized** version of the branch-and-bound algorithm that incorporates improved heuristics and smarter node selection strategies.

1) **Simple Heuristic Initialization**
   - **Random Heuristic**: As a starting point, we randomly generated an initial feasible solution by opening a subset of facilities (subject to capacity) and randomly assigning substations/customers. This random allocation served as the algorithm's initial upper bound and guided early pruning.

2) **"Simple Node" Enhancement**
   - **Best-First Search (BFS)**: Instead of pure DFS, we adopted a best-first search strategy that prioritizes nodes closer to a promising objective value. After evaluating the partial solutions, the algorithm queues nodes and selects for expansion those with the lowest estimated cost (or highest estimated profit), effectively focusing on the most promising branches earlier.
   - **Preprocessing and Node Selection**: In addition to the initial random heuristic, we introduced a **preprocessing step** that ranks decision variables based on potential impact on cost or feasibility. This ranking then influences which branch (i.e., which facility or assignment variable) to explore first at each step.

3) **Plants Decision**
   - This optimized version includes a decision mechanism for whether to open a facility or not. This was implemented to scale better and assess how branch-and-bound performs with additional binary variable decisions.

By combining a randomized starting solution as a preprocessing heuristic with a best-first search strategy, the optimized branch-and-bound can explore fewer nodes overall. Since it also considers facility location decisions, this approach reduces runtime without sacrificing solution quality. Additionally, we implemented two different solvers: one using only the preprocessing heuristic, which was more efficient for larger problems, and another using BFS, which performed better for smaller-sized problems.

III. Problem Formulation

A. Mathematical Model

Sets

- I: Set of production facilities.
- J: Set of substations.
- K: Set of customers.
- S: Set of seasons.

Parameters

- $PC_{i,s}$: Production cost for facility i in season s.
- $Cap_{i,s}$: Production capacity for facility i in season s.
- FC_j: Fixed cost of opening substation j.
- $SC_{j,s}$: Capacity of substation j in season s.
- $D_{k,s}$: Demand of customer k in season s.
- α_{ij}: Heat loss coefficient between facility i and substation j.

Instance	Solver	Fac.	Sub.	Cust.	Grid	Time (s)	Status	Total Cost	Open Sub.	Opened Substations	Nodes
1	Custom	2	3	4	100	0.90	Optimal	175,992.73	2	Sub1, Sub2	18
1	Custom	3	4	6	100	49.27	Optimal	213,899.51	3	Sub1, Sub2, Sub4	803
1	Custom	3	5	8	100	399.62	Optimal	416,217.90	4	Sub1, Sub2, Sub3, Sub5	5927
1	Gurobi	2	3	4	100	0.04	Optimal	177,553.16	2	Sub1, Sub2	–
1	Gurobi	3	4	6	100	0.09	Optimal	215,394.50	4	Sub1, Sub2, Sub3, Sub4	–
1	Gurobi	3	5	8	100	0.12	Optimal	418,333.55	4	Sub1, Sub2, Sub3, Sub5	–
1	Simple	2	3	4	100	4.30	Optimal	175,992.73	2	Sub1, Sub2	–
2	Simple	3	4	6	100	55.28	Optimal	213,899.51	3	Sub1, Sub2, Sub4	–
3	Simple	3	5	8	100	388.29	Optimal	416,217.90	4	Sub1, Sub2, Sub3, Sub5	–
1	Simplenode	2	3	4	100	4.23	Optimal	175,992.73	2	Sub1, Sub2	–
2	Simplenode	3	4	6	100	35.63	Optimal	213,899.51	3	Sub1, Sub2, Sub4	–
3	Simplenode	3	5	8	100	460.57	Optimal	416,217.90	4	Sub1, Sub2, Sub3, Sub5	–

TABLE I
COMPARISON OF DIFFERENT SOLVERS FOR FACILITY LOCATION PROBLEM

- β_{jk}: Heat loss coefficient between substation j and customer k.
- M_{jk}: Large constant for assignment constraints.

Variables

- z_i: Binary variable indicating if facility i is open.
- y_j: Binary variable indicating if substation j is open.
- x_{jk}: Binary variable for assigning customer k to substation j.
- $f_{ij,s}$: Flow from facility i to substation j in season s.
- $f_{jk,s}$: Flow from substation j to customer k in season s.

Objective Function

The objective is to minimize the total cost, which includes the fixed costs of opening substations and the production costs associated with flows from facilities to substations:

$$Minimize \sum_{i \in I} FC_i^F \cdot z_i + \sum_{j \in J} FC_j \cdot y_j + \sum_{i \in I} \sum_{j \in J} \sum_{s \in S} PC_{i,s} \cdot f_{ij,s}$$

Constraints

1. Flow Conservation at Substations: The total adjusted flow arriving at each substation j in season s from all facilities must equal the flow leaving to customers:

$$\sum_{i \in I} f_{ij,s} \cdot \alpha_{ij} = \sum_{k \in K} f_{jk,s}, \quad \forall j \in J, \forall s \in S$$

2. Demand Satisfaction: The flow arriving at each customer k in season s, adjusted by heat loss, must meet or exceed their demand:

$$\sum_{j \in J} f_{jk,s} \cdot \beta_{jk} \geq D_{k,s}, \quad \forall k \in K, \forall s \in S$$

3. Production Capacity Limits: The total flow sent from each facility i to substations in season s must not exceed the facility's production capacity, and production is allowed only if the facility is open:

$$\sum_{j \in J} f_{ij,s} \leq Cap_{i,s} \cdot z_i, \quad \forall i \in I, \forall s \in S$$

4. Substation Capacity Limits: The flow through each substation j in season s must not exceed its capacity if it is open:

$$\sum_{k \in K} f_{jk,s} \leq SC_{j,s} \cdot y_j, \quad \forall j \in J, \forall s \in S$$

5. Customer Assignment: Each customer k must be assigned to exactly one substation:

$$\sum_{j \in J} x_{jk} = 1, \quad \forall k \in K$$

6. Assignment Only to Open Substations: Customers can only be assigned to substations that are open:

$$x_{jk} \leq y_j, \quad \forall j \in J, \forall k \in K$$

7. Flow Only Through Assigned Substations: The flow from a substation j to a customer k in season s is limited by a large constant M_{jk} only if the substation serves the customer:

$$f_{jk,s} \leq M_{jk} \cdot x_{jk}, \quad \forall j \in J, \forall k \in K, \forall s \in S$$

IV. RELATED WORK

Our work addresses two interrelated sub-problems: the Capacitated Facility Location Problem (CFLP) and network optimization with a focus on thermal planning. The literature in these areas offers valuable insights that we build upon.

A. Extensions of the Capacitated Facility Location Problem

The CFLP has long been studied for its applications in logistics and supply chain management. Traditional models focus on minimizing costs by locating facilities and assigning customers under capacity constraints. Recent extensions have introduced more complex cost structures—such as non-linear setup costs and the possibility of multiple facilities at a single site. For instance, [1] propose models that distinguish between fixed site costs and variable facility costs, using Mixed Integer Programming and Lagrangian heuristics. These approaches are relevant to our work, as our model must incorporate non-linear thermal dissipation costs and seasonal capacity constraints.

B. Dynamic Facility Location and Allocation

Dynamic facility location models extend the static CFLP by accounting for time-dependent factors, such as fluctuating demand and facility operation changes over multiple periods. Work by [2] demonstrates how Mixed Integer Programming can optimize facility and allocation decisions in multi-period settings. Although their focus is on traditional logistics, the concept of adapting to changing demand is directly applicable to thermal planning, where seasonal variations play a critical role.

C. Implications for Thermal Planning and Algorithm Design

Thermal planning introduces additional challenges such as energy dissipation, flow constraints, and distance-dependent costs. Standard solution methods may struggle with these complexities, motivating our two-phase approach:

1) Formulating realistic problem instances that integrate thermal and seasonal parameters and benchmarking them with a commercial solver.
2) Developing a customized branch-and-bound algorithm that incorporates problem-specific heuristics and pruning techniques to efficiently handle thermal network constraints.

D. Summary

In summary, while generalized CFLP and dynamic facility location models address non-linear cost functions and time-dependent demand, they do not fully capture the unique challenges of thermal network planning. Our work bridges this gap by integrating insights from these areas and proposing a specialized solution framework.

V. Results and Discussion

A. Comparison of Solver Performance

Table I and present the comparative performance of the

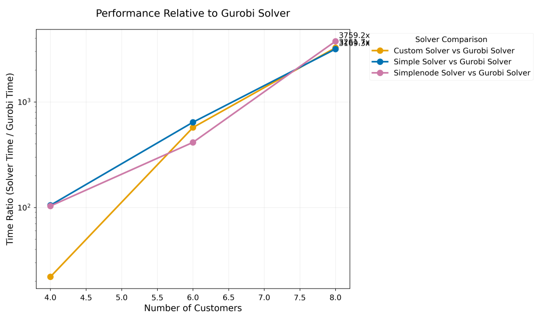

Fig. 1. Solver performance comparison.

different solvers: Gurobi, Custom Branch and Bound, Simple Solver (Branch and Bound with heuristic), and SimpleNode Solver (Branch and Bound with improved node selection). The results show that Gurobi consistently outperforms other solvers in solution time due to its advanced internal heuristics and optimizations. However, our custom solvers achieve comparable solutions while maintaining problem-specific flexibility.

graphics

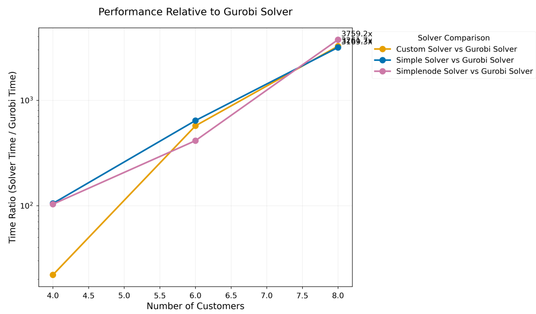

Fig. 2. Solver Performance Comparison (Log Scale).

B. Analysis of Custom Solvers

The Custom Solver(simple branch and bound) demonstrates stable performance with optimal solutions, but its execution time increases significantly for larger instances. The Simple Solver, which incorporates heuristic-based pruning, improves the search process by reducing unnecessary explorations. However, the SimpleNode Solver, which employs a more strategic node selection, shows similar computational effort but slightly better pruning efficiency.

C. Time Complexity Considerations

As seen in the results, the execution time of the SimpleNode Solver is not significantly reduced compared to the Simple Solver. This is mainly due to:

- The increased computational overhead of evaluating better node selection strategies.
- The structure of the problem instances, which may not always benefit from deeper search prioritization.
- The balance between pruning efficiency and node expansion rate, where the overhead of selecting the best node counteracts potential gains.

Despite this, the improved node selection method ensures a more structured exploration, potentially leading to better scalability in larger problem instances.

D. Potential Improvements

To further enhance the performance of the solvers, the following improvements can be considered:

- **Adaptive Node Selection:** Dynamic adjustment of node evaluation criteria based on instance characteristics.
- **Parallelization:** Distributing branch evaluation across multiple threads to accelerate computations.
- **Hybrid Heuristics:** Combining heuristic approaches with exact methods to further refine pruning efficiency.
- **Memory Optimization:** Reducing redundant storage of node states to improve scalability.

These enhancements could lead to a more efficient solver that balances solution time and computational resources more effectively.

VI. Conclusion

In this work, we addressed the problem of thermal energy distribution by formulating it as an optimization problem and solving it using branch-and-bound techniques. We began by constructing realistic problem instances, carefully considering facility placement, substation distribution, and customer clustering to ensure meaningful constraints and challenges.

first we implemented a basic branch-and-bound approach, which laid the foundation for our solver. However, due to the large search space and combinatorial nature of the problem, we introduced an optimized version incorporating heuristic-based initialization and an intelligent node selection strategy. Our Simple heuristic provided a fast yet effective starting solution, while the SimpleNode enhancement leveraged best-first search to improve pruning efficiency.

The experimental results demonstrated that our optimizations reduced computational time for bigger problems while maintaining solution quality. The introduction of preprocessing heuristics and best-first search enabled the solver to focus on more promising solutions earlier in the search process, leading to fewer explored nodes and improved scalability.

Overall, our contributions highlight the importance of both problem modeling and algorithmic improvements in solving large-scale optimization problems. Future work could explore additional heuristic refinements, hybrid approaches combining branch-and-bound with heuristics, or parallel implementations to further enhance performance.

VII. References

References

[1] L.-Y. Wu, X.-S. Zhang, and J.-L. Zhang, "Capacitated facility location problem with general setup cost," *Computers & Operations Research*, vol. 33, no. 5, pp. 1226–1241, 2006.

[2] R. Manzini and E. Gebennini, "Optimization models for the dynamic facility location and allocation problem," *International Journal of Production Research*, vol. 46, no. 8, pp. 2061–2086, 2008.