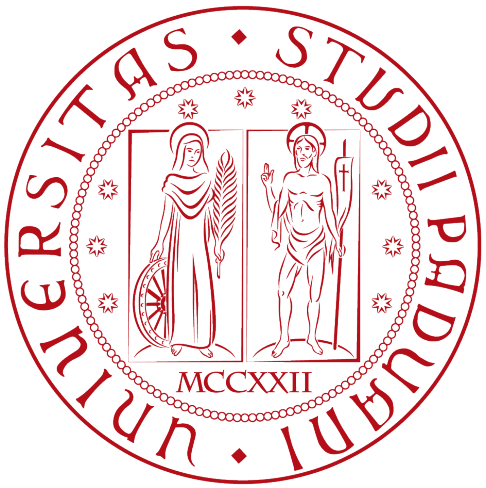

Lab Assignments Report

Methods and Models for Combinatorial Optimization

Student: Gabriel Rovesti - *ID Number* - 2103389

Table of contents

Figures

List of Tables

1. Part I: Exact Method Implementation

1.1. Problem Description and Mathematical Formulation

A company manufactures circuit boards with pre-drilled holes to be used to construct electric panels. The process involves placing boards over a machine where a drill travels across the surface, stops at predetermined positions, and creates holes. Once a board is completely drilled, it is replaced by another one and the process is repeated. The company is interested in finding the optimum process by determining the best sequence of drilling operations for which the total time will be minimal, considering that creating each hole takes the same amount of time.

This optimization challenge can be mathematically represented as a *Traveling Salesman Problem (TSP)* on a weighted complete graph $G = (N, A)$, where:

- N represents the set of nodes corresponding to the positions where holes must be drilled
- A represents the set of arcs $(i, j), \forall i, j \in N$, corresponding to the drill's movement trajectory from hole i to hole j
- Each arc (i, j) has an associated weight c_{ij} representing the time needed for the drill to move from position i to position j

The optimal solution to this problem is the Hamiltonian cycle on G that minimizes the total weight, representing the most efficient drilling sequence.

The model proposed comes from the Gavish-Graves paper[1] in 1978, described generally as follows:

$$\min \sum_{i,j:(i,j)\in A} c_{ij} y_{ij} \tag{1}$$

Subject to:

$$\sum_{i:(i,k)\in A} x_{ik} - \sum_{j:(k,j),j\neq 0} x_{kj} = 1 \quad \forall k \in N \setminus \{0\} \tag{2}$$

$$\sum_{j:(i,j)\in A} y_{ij} = 1 \quad \forall i \in N \tag{3}$$

$$\sum_{i:(i,j)\in A} y_{ij} = 1 \quad \forall j \in N \tag{4}$$

$$x_{ij} \leq (|N| - 1) y_{ij} \quad \forall (i, j) \in A, j \neq 0 \tag{5}$$

$$x_{ij} \in \mathbb{R}^+ \quad \forall (i, j) \in A, j \neq 0 \tag{6}$$

$$y_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \tag{7}$$

Sets:

- $N =$ graph nodes, corresponding to the holes positions
- $A =$ arcs $(i, j), \forall i, j \in N$, serving as the movement from hole i to hole j

Parameters:

- $c_{ij} =$ time to move from i to $j, \forall (i, j) \in A$
- $0 =$ arbitrary starting node, $0 \in N$

Decision variables:

- $x_{ij} =$ amount of flow shipped from i to $j, \forall (i, j) \in A$

[1] https://dspace.mit.edu/handle/1721.1/5363

- $y_{ij} = 1$ if arc (i, j) ships some flow, 0 otherwise, $\forall (i, j) \in A$

The model structure can be explained as follows:

- The objective function (Equation 1) minimizes the total time required for the drill to complete its path by summing the travel times between consecutive holes weighted by the binary decision variables that indicate whether that path segment is used.
- The constraints work together to ensure a feasible drilling sequence:
  - (Equation 2) is the flow conservation constraint that ensures continuity in the drilling path at each node except the origin. For each node k, the difference between incoming and outgoing flow must equal 1, which helps prevent subtours in the solution.
  - (Equation 3) and (Equation 4) are the assignment constraints that guarantee each hole position is visited exactly once. Specifically, (Equation 3) ensures exactly one outgoing arc is selected from each node, while (Equation 4) ensures exactly one incoming arc is selected for each node.
  - (Equation 5) creates the relationship between flow variables and binary path selection variables through a big-M formulation, where $|N| - 1$ serves as the upper bound for any flow. This ensures that flow can only exist on arcs that are part of the selected path.
  - (Equation 6) and (Equation 7) define the variable domains, with x variables being continuous non-negative and y variables being binary. The binary variables effectively determine which arcs form the final drilling sequence.

This formulation, while compact, effectively captures all necessary aspects of the drilling sequence optimization problem. The flow-based approach combined with binary path selection variables provides a robust way to ensure a single, continuous path that visits each hole exactly once while minimizing total travel time.

3.1.2. Runtime Performance

3.1.3. Scalability Trade-offs

3.2. Final Considerations

3.2.1. Practical Implementation Aspects

3.2.2. Recommendations for Usage

3.2.3. Future Improvements

], [B], [A], [B],)

```
// Code example
```