Discrete optimization model for multi-product multi-supplier vehicle routing problem with relaxed time window

Muliawan Firdaus[1], Herman Mawengkang[2], Tulus[2], Sawaluddin[2]
[1]Graduate School of Mathematics, Universitas Sumatera Utara, Medan, Indonesia
[2]Department of Mathematics, Universitas Sumatera Utara, Medan, Indonesia

Article Info

Article history:

Received Jun 8, 2024
Revised Aug 12, 2024
Accepted Aug 20, 2024

Keywords:

Discrete optimization model
Multi-product
Multi-supplier
Time windows
Vehicle routing problem

ABSTRACT

This study examines the complicated logistics optimization issue known as the vehicle routing problem for multi-product and multi-suppliers (VRP-MPMS), which deals with the effective routing of a fleet of vehicles to convey numerous items from multiple suppliers to a set of consumers. In this problem, products from various suppliers need to be delivered to different customers while considering vehicle capacity constraints, time windows, and minimizing transportation costs. We propose a hybrid approach that combines a generalized reduced gradient method to identify feasible regions with a feasible neighborhood search to achieve optimal or near-optimal solutions. The aim of the exact method is to get the region of feasible solution. Then we explore the region using feasible neighborhood search, to get an integer feasible optimal (suboptimal) solution. Computational experiments demonstrate that our model and method effectively reduce transportation costs while satisfying vehicle capacity constraints and relaxed time windows. Our findings provide a viable solution for improving logistics operations in real-world scenarios.

Corresponding Author:

Herman Mawengkang
Department of Mathematics, Universitas Sumatera Utara
Jl. Dr. T. Mansur No. 9, Padang Bulan, Medan Baru, Medan City, North Sumatra 20155, Indonesia
Email: mawengkang@usu.ac.id

1. INTRODUCTION

As a result of its significance to the economy, the field of logistics and supply chain management has been advancing consistently and quickly in recent years. Since the majority of businesses view supply chain management as a function that enhances their market, it plays a crucial part in their strategic decision-making. Due to the constantly increasing demand from customers, businesses need efficient delivery services without sacrificing the standard of their customer care in order to remain profitable. In order to fulfil these increasing expectations, logistics operations have been under continual pressure to become more effective. As a result, for effective management, businesses must design the routes for their vehicles so as to save expenses while still satisfying customer requirements. The vehicle routing problem (VRP) is the name used in operations research to describe this method of route determining [1], [2].

Mathematically, the VRP can be described as following: Let $G = (V, A)$ be a directed graph, where $V = \{0, \ldots, n\}$ is a set of nodes, with vertex 0 representing the depot and the remaining vertices representing customers. $A = \{(i, j): i, j \in V, i \neq j\}$ represents arcs in the graph. The depot was home to a fleet of m identical vehicles, each having a Q capacity. The fleet size, m is a decision variable that needs to be determined. Each customer i has a request q_i, which is a non-negative quantity representing their demand. Additionally, there is a cost matrix c_{ij} defined for each arc (i, j) in A, which represents the cost (or distance) associated with traveling from node i to node j. It is important to note that in this formulation, assume that distances, travel costs, and times are measured equivalent.

The VRP's goal is to create m vehicle routes that adhere to the following restrictions: The depot (node 0) is where every route must begin and terminate. Each customer's demand must be satisfied by having exactly one vehicle visit them. Each route's overall demand cannot be more than the vehicle's capacity. Each route's total length (or price) must not go beyond a certain threshold, L. The objective is to minimize the overall transportation cost or route length while fulfilling the aforementioned limitations by optimizing the assignment of clients to vehicles and the order in which they are visited. In the symmetric case, that is, when $c_{ij} = c_{ji}$ for all $(i, j) \in A$, the solution search is usually done using the set of edges, $E = \{(i, j): i, j \in V, I < j\}$.

The VRP was first developed by Dantzig and Ramser [3] to address the problem of efficiently routing a fleet of fuel delivery trucks from a bulk terminal to numerous service outlets supplied by the terminal. This marked the origin of VRP as a mathematical optimization problem. Over the years, VRP has gained significant attention in the research community, and various extensions and variations of the problem have been explored. Further literature on the VRP and its different attributes can be referred to the following sources for a comprehensive survey and overview [4]–[6]. These references provide valuable insights into the VRP and its evolution, including different problem variants and solution approaches that have been developed over time.

Research on the VRP continues to be a dynamic field due to both unresolved theoretical challenges and the ongoing influx of practical logistics data from supply chain operations. One of the most extensively studied variants of the VRP is with time windows (VRPTW), which was first introduced by Schrage [7]. In the VRPTW, specific time constraints are placed on customer visits, known as time windows. Time window constraints in VRPTW can be driven by various factors, such as product constraints (*e.g.*, product usable dates), production limits, or requirements imposed by customers based on their inventory policies. In addition to these time windows for customer visits, there are also travel times between all customers and between customers and the depot. The main objectives in VRPTW are to plan vehicle routes that satisfy the following criteria: Each vehicle must serve all assigned customers within their respective time windows, Vehicles are permitted to arrive at the location of a client before the time window begins, but they must wait if they do so before the consumer is prepared to be served. Vehicles are not permitted to arrive late or after the time window ends for any customer. The primary goal is to minimize the total transportation cost, taking into account both travel distances and time-related penalties for violating the time windows. VRPTW is particularly relevant in situations where time-sensitive deliveries are crucial, and it poses additional challenges compared to the classic VRP. Due to its practical importance and complexity, VRPTW has been the subject of extensive research and has led to the development of various solution methods and algorithms to find optimal or near-optimal solutions in real-world logistics and transportation applications [8]–[10].

The VRPTW is indeed a challenging problem due to its combinatorial nature, which makes it difficult to find optimal solutions, especially for larger instance [11]. Kohl [12] established that VRPTW is an NP-hard problem, indicating that solving it to optimality becomes computationally infeasible as the problem size increases. As a result, researchers have turned to heuristic and metaheuristic approaches to find good-quality solutions within reasonable computational time. Various metaheuristics and heuristics have been proposed to tackle VRPTW, aiming to strike a balance between solution quality and computational efficiency. Some of these approaches include those developed by researchers such as [13]–[16]. Efficient metaheuristics, in particular, often rely on local search-based refinement procedures and focus much of their computational effort on exploring neighborhoods of solutions. This approach can significantly improve the quality of solutions found, but it also places an emphasis on evaluating the impact of potential solution changes efficiently. However, it's worth noting that even finding a feasible solution for VRPTW, without necessarily aiming for optimality, remains computationally challenging. As mentioned, Savelsbergh [17] demonstrated that determining a feasible solution for VRPTW is also an NP-hard problem. This highlights the inherent complexity of the problem and underscores the need for sophisticated optimization techniques to address it effectively, especially when dealing with real-world instances with practical constraints and larger numbers of customers or subscribers [18].

In order to develop early solutions for the VRPTW, it is frequently used intermediate solutions with flexible time frame limitations. However, as mentioned, it may not always be feasible or optimal for guaranteeing availability of multiple initial solutions. Several relaxation schemes have been explored in previous research to handle time window constraints more effectively: *Penalties for Late Arrivals*: One common relaxation method involves assigning penalties for late arrivals at customer locations. This allows for some flexibility in meeting time windows while penalizing deviations from the desired schedule. This approach was discussed by Sun *et al.* [19]. *Early and Late Arrivals*: Another relaxation scheme considers both early and late arrivals at customer sites. It allows vehicles to arrive before or after the time window but incurs penalties accordingly. This approach was studied by Ibaraki *et al.* [20]. *Refund Penalties on Time*: [21]

proposed a refund penalty approach, where vehicles could earn refunds for arriving earlier than required while being penalized for arriving late. This approach encourages early arrivals but still respects time window constraints.

While these relaxation schemes offer some flexibility in finding initial solutions, they can also introduce complexity in evaluating potential solution changes and may lead to larger search spaces. Therefore, your research aims to establish a new relaxation scheme that balances feasibility and solution quality more effectively. Developing innovative relaxation strategies that strike the right balance between feasibility and solution quality is essential in the context of VRPTW optimization, as it can lead to improved initial solutions and ultimately enhance the performance of heuristic and metaheuristic algorithms for solving this challenging problem.

The influence of real-world traffic conditions on vehicle routing is indeed a significant challenge in practical supply chain operations. Traditional VRP models typically assume constant travel times, which can lead to suboptimal or infeasible solutions when faced with the uncertainties and variability caused by traffic congestion. To address this issue, researchers have introduced more advanced variants of the problem that consider time-dependent travel times. One such variant is the time-dependent vehicle routing problem with time windows (TDVRPTW), which was introduced by Kumar and Panneerselvam [22]. In the TDVRPTW, the modeling explicitly takes into account the variability in travel times due to traffic conditions. This is achieved by considering time-dependent travel times for each arc or route segment, which can change based on factors like traffic congestion, time of day, and road conditions. To solve the TDVRPTW, various approaches have been explored [23]–[25], including mixed integer programming (MIP) formulations [26], which provide a mathematical representation of the problem with time-dependent constraints. Additionally, metaheuristic algorithms like genetic algorithms have been applied to find good-quality solutions within reasonable computational timeframes. By considering the impact of traffic conditions, the TDVRPTW provides a more realistic representation of routing challenges faced by vehicles in supply chain operations. It allows for the optimization of routes that are better adapted to real-world situations, ultimately improving the efficiency and reliability of logistics and transportation processes.

Engaging with multiple suppliers in a TDVRPTW introduces significant complexity. Coordinating visits to suppliers with diverse time windows, ensuring timely pickups, and adhering to delivery periods add challenges. Additionally, accounting for real-time traffic conditions is crucial. To address these complexities, optimization approaches involve dynamic scheduling, advanced mathematical models, heuristic algorithms, and real-time data integration to find efficient solutions that consider the dynamic nature of supply chain operations and traffic conditions, enhancing the reliability and efficiency of multi-supplier logistics and transportation processes.

Experiments involving 56 instances from the Solomon benchmark [27], each comprising 100 customer nodes, 25 vehicles with a 200-unit capacity each, have yielded results only comparable to existing algorithms. Consequently, this study aims to propose an innovative discrete optimization model as an alternative approach for addressing the TDVRPTW in multi-supplier settings. Additionally, the research will develop a metaheuristic algorithm, with initial solutions generated through time window relaxation, to tackle this complex problem effectively.

2. MATERIAL AND METHODS

2.1. Formation of vehicle routes

A route $r \in R$ will still be feasible when the route starts at a different time instant. Thus, for each route r, it is noticed that there are multiple routes r_t, one for each instant t of possible departures. The duration of route r, σ_r, will be different for different instances of departure, due to the waiting time for serving different customers.

Suppose $(i_1, \dots, i_{|N_r|})$ is the order of clients visited on route $r \in R$. The initial possible instant of time to end route r is $T_r'^{-} = \theta^r_{i_{|N_r|}} + s_{i_{|N_r|}} + s_{i_{|N_r|}0}$, where $\theta^r_{i_{|N_r|}}$ is the first instance to begin provision on the latter client $i_{|N_r|}$ in route r. Then calculate $T_r'^{-}$, taking into account that $\theta^r_{i_h} = \max\{\theta^r_{i_{h-1}} + s_{i_{h-1}} + t_{i_{h-1}i_h}, a_{i_h}\}$ for $h \in \{1, \dots, |N_r|\}$ where $\theta^r_{i_0} = a_0$. This means that starting route r at any instance $t^*_r \le T^-_r = \theta^r_{i_1} - t_{0i_1}$ causing it to stop instantly $T_r'^{-}$. Thus, such a route is subject by routes r starting at the T^-_r instance, so they do not need to be concerned.

In the same way, last ending time instantaneous on route r is $T_r'^{+} = \phi^r_{i_{|N_r|}} + s_{i_{|N_r|}} + t_{i_{|N_r|}0}$, where $\phi^r_{i_{|N_r|}}$ is the instance to begin provision on the client $i_{|N_r|}|$ in course r and $\phi^r_{i_h} = \min\{\phi^r_{i_{h-1}} + s_{i_{h-1}} + t_{i_{h-1}i_h}, b_{i_h}\}$, for $h \in \{1, \dots, |N_r|\}$with $\phi^r_{i_0} = b_0$. It concludes that starting route r in time after $T^+_r = \phi^r_{i_1} - t_{0i_1}$ results in that route being infeasible, because it ignores at minimum of one client time windows.

It should be noted that path r starts in time $[T_r^-, T_r^+]$, the waiting time is reduced hence it would have minimized duration. As an example, each $r \in R$, the time $[T_r^-, T_r^+]$, is calculated. Thus, feasible routes quantity will equal $\sum_{r \in R} \left\lceil \frac{T_r^+ - T_r^- + 1}{U} \right\rceil$, where U is the unit of time assumed to equal 1.

2.2. Discrete model development

Incorporating multi-supplier constraints adds significant complexity to the model. In this research, a directed acyclic graph $\Pi = (\Delta, \psi)$ is used to represent each working day. The vertices, $\Delta = \{0,1, \dots, W\}$, signifies distinct time of 0 to the day's length, while $\Psi = \{(u,v)^\wedge r: 0 \leq u < v \leq W, u \in [T\_r^\wedge -, T\_r^\wedge +],$ $v = u + \sigma\_r, r \in R\} \cup \{(u,v)^\wedge 0: 0 \leq u < v \leq W, v = u + 1\}$ represents arcs. These arcs (unit arcs) refer to possible routes or waiting times. The vehicle's time at the depot inside a working day is shown by the waiting time arc. It is essential to note that in the model under development, the start time of each route $r \in R$ is adjusted to the previous time instant $\beta \sum_{i \in N_r} s_i$ to account for the vehicle's loading time.

Considering the factors mentioned above, the model being developed will have constraints that grow polynomially with the size of W, the number of variables that also grow polynomial with W size, and possible routes will be bounded by a constant determined by the parameter t_{max}. As a result, a collection of pseudo-polynomial variables and constraints will be present in the final model.

The λ_{uv}^r variables will serve as a representation of flow in arc $(u,v)^r$, indicating the number of vehicles traveling along route r, departing from stations at time instant u, arries at time v within workday. The z variable will represent the graph total flow and interpreted as the node W return flow back to node 0. Additionally, d_r will be utilized to denote the cost associated with route r, calculated as total distance travelled along that route. The models are defined as (1):

Minimize,

$$\sum_{(u,v)^r \in \Psi} \left(d_r - \alpha \sum_{i \in N_r} g_i\right) \lambda_{uv}^r \tag{1}$$

with constraints

$$\sum_{(u,v)^r \in \Psi | i \in N_r} \lambda_{uv}^r \leq 1, \;\; \forall i \in N \tag{2}$$

$$-\sum_{(u,v)^r \in \Psi} \lambda_{uv}^r + \sum_{(u,v)^s \in \Psi} \lambda_{uv}^s = \begin{cases} z & if \quad v = 0 \\ 0 & \text{if} \quad v = 1, \dots, W-1 \\ -z & \text{if} \quad v = W \end{cases} \tag{3}$$

$$z \leq K \tag{4}$$

$$\lambda_{uv}^r > 0 \text{ and integers}, \;\; \forall (u,v)^r \in \Psi \tag{5}$$

$$z \geq 0 \text{ and integers} \tag{6}$$

In (1) outlines the model's objective, which is to reduce the covered distance of total vehicles within a single working day. Constraint (4) is in place to acknowledge that visiting all customers may not be feasible because of the available vehicle's limitation, expressed within the inequality constraints (2). Nevertheless, increasing the number of customers is the objective, which is a favorable outcome. Constraint (3) represents a fundamental flow conservation constraint within the network, ensuring flow entering a node is at equilibrium with the flow exiting that node. These constraints collectively define the optimization problem and guide the decision-making process for efficient vehicle routing.

2.3. Time window relaxation scheme development

In (1)-(6) model nodes represent instants of time. Distance and time are usually not in integer form, and thus there are two alternatives: using a smooth discretization (each unit of time will be 0.01), using rounding off procedure to utilize time units. The first option would provide a network flow model with a large number of restrictions and variables, making it impractical for an immediate solution. Therefore, in this study, the second alternative is used. Since the solution approach is intended to obtain an exact solution, algorithms are developed that iteratively refine the discretization.

2.3.1. Initial rounding strategy

Notice that the arc $(u,v)^r$ in model (1)-(6) corresponds to a route r that starts at time u and ends at time v. Knowing that vertex of graph Π is demarcated the set of values $\Delta = \{0, \dots, W\}$, it is essential for arc

$(u, v)^r \in \Psi$, rounding u and v to values that are in the set Δ. As previously mentioned, first consider the units of time equal to 1, namely $\Delta = \{0, 1, 2, 3, \ldots, W-1, W\}$. Some possible rounding procedures are as:

- $u = \lfloor u \rfloor$ and $v = \lceil v \rceil$. In this model, routes are constructed to start slightly before and end slightly after their actual occurrence. This implies that some potential solutions may not be considered, but any feasible solution obtained by relaxing the constraints (1)-(6) in this manner will also be viable for original problem.
- $u = \lceil u \rceil$ and $v = \lfloor v \rfloor$. In the current scenario, relaxation is for the model, which means the solutions initiate may not always be reasonable. Though, this relaxation serves the purpose of establishing an effective lower bound for the problem.
- $u = \lceil u \rceil$ and $v = \lceil v \rceil$ or $u = \lfloor u \rfloor$ and $v = \lfloor v \rfloor$. In the current case, a lower bound will also be found.

In this study, the proposed algorithm uses the second rounding procedure, namely by considering $u = \lceil u \rceil$ and $v = \lfloor v \rfloor$. The reasons for choosing this procedure were: i) relaxation is expected to be tight and ii) the inappropriateness is local and can be corrected.

Considering paths in the flow model that involve two successive routes, designated as r and r', with a gap less units of time waiting among them, results in infeasible solutions, it is crucial to highlight. To address this issue, one initial approach is rectifying solution either moving route r rearward or route r' onward to eliminate conflict. If this adjustment results in a feasible solution, it not only resolves the infeasibility but also proves optimality, as feasible solutions share same routes and have lower bound equivalent cost. In essence, after obtaining the x^* result, efforts are made to construct on same routs favorable solution identified in the x^* solution.

The proposed algorithms operate in the following manner: For each working day, it attempts to create a new route while preserving the existing routes sequence in the solution. Suppose $(r_1, \ldots, r_p)$ is the line of routes in weekdays, and T_{r_i} is the starting route and T'_{r_i} is the ending route r_i, $\forall i \in \{1, \ldots, p\}$. Set $T_{r_i} = \max(T^-_{r_i}, T'_{r_{i-1}})$. If $T_{r_i} \le T^+_{r_i}$, $\forall i \in 1, \ldots, p$, is viable solution. otherwise, then feasibility is not proven, and additional algorithm must be cast-off.

2.3.2. Perfecting iterative discretization

The solution approach used involves an iterative correction of infeasibilities resulting from discretization problems. For algorithm each step, instances of infeasibility are identified. For each of these instances, the discretization is locally adjusted by addition fractional values required to complete relaxation initially imposed by original discretization. Several time instants can be combined into a single integer during the first relaxation. The existing graph would be divided into several nodes using this refinement method. conflicting arcs set $(u, v)r$ and u_0, fractional values for v and u_0 are considered to enhance the solution.

3. RESULTS AND DISCUSSION

3.1. Problem description

3.1.1. Problem definition and notation

In MVRPTW, there is a single depot represented as o and it serves as both start and end point for all vehicle routes. The fleet consists of homogeneous vehicles, meaning that all vehicles are identical in terms of capacity and other attributes. Each vehicle in the fleet has a Q capacity units. further expected are a total of K vehicles accessible in this fleet for carrying out routing tasks.

In MVRPTW: The set of customers is denoted as $N = \{1, \ldots, n\}$. Each pair of locations, including customers and the depot, has an associated distance $d_{ij}$and travel time t_{ij}. Each client i has a specific request or request q_i, a service time s_i, a revenue g_i, and a time window $[a_i, b_i]$. The time window specifies the a_i earliest time and the b_i latest time at which provision can begin at client i. The windows must be open for vehicle arrives earlier than a_i. It is assumed that, by default, the vehicle starts serving a customer as soon as it arrives. The $s_o = 0$, indicating that there is no service time required at the depot. The time window signifies the total time W of a workday, which sets the time constraints for the entire routing problem. It is assumed that $b_i + s_i + d_{io} \le b_o$, $\forall i \in N$.

Throughout the working day, each vehicle can go on a number of routes. Until the end of the workday, this entails being able to complete one route, reload at the depot, and head out for the subsequent route. Route r is defined by the order of visits to a subset of customers $N_r \subseteq N$. It is practical if the total number of requests from every client in N_r does not exceed the vehicle's capacity and if the order of the visits allows for the visitation of every customer within a predetermined window of time. In this model it is also considered that the service of all customers on the route cannot be started longer than t_{max} the maximum time unit after the route is started. The collection of all possible routes is denoted by R. There is also a setup time to take into account for each route. Prior to departing the depot to travel route r, the vehicle $\beta \sum_{i\in N_r} s_i$ time units to load, with $\beta \in \mathbb{R}^+$. Due to the limited number of available vehicles, it might not be possible to visit every client. Nevertheless, it is usually preferable to visit as many clients as possible.

3.1.2. Mathematical formulation for MVRPTW

The description expressed in a graph $G = (V, A)$, with $V = N \cup \{o\}$ a set of vertices and $A = \{(i,j) : i,j \in V\}$ a set of arcs. In this description, there is a binary variable representing the subscriber to the route and determines the sequential pair of routes. The binary variables x_{ij}^r and y_i^r respectively define, if arc (i,j) and client i associated to route r, while binary variable z_{rs} determines if any vehicle traveling route r is followed by route s within weekdays. The notation $r < s$ signifies identical vehicle is allocated to do route s afterward doing route r. The variable t_i^r represents the start time on client i, if associated with path r, and t_o^r and $t_o'^r$ characterize the start and end duration of route r. Suppose M is a large enough number. The concise formulation for MVRPTW is stated as (7)-(26).

Minimize,

$$\sum_{r\in R}\sum_{(i,j)\in A} d_{ij}x_{ij}^r - \alpha \sum_{r\in R}\sum_{i\in N} g_i y_i^r \tag{7}$$

With constraints

$$\sum_{j\in V} x_{ij}^r = y_i^r,\ \forall i \in N, \forall r \in R \tag{8}$$

$$\sum_{r\in R} y_i^r \le 1,\ \forall i \in N \tag{9}$$

$$\sum_{i\in V} x_{ih}^r - \sum_{j\in V} x_{hj}^r = 0,\ \forall h \in N, \forall r \in R \tag{10}$$

$$\sum_{i\in V} x_{oi}^r = 1,\ \forall r \in R \tag{11}$$

$$\sum_{i\in V} x_{io}^r = 1,\ \forall r \in R \tag{12}$$

$$\sum_{j\in N} x_{ij} = 1,\ i \in N, i \ne 0, i \ne j \tag{13}$$

$$\sum_{i\in N} x_{ij} = 1,\ j \in N, j \ne 0, i \ne j \tag{14}$$

$$\sum_{i\in N} q_i y_i^r \le Q, \quad \forall r \in R \tag{15}$$

$$q y_i^r \le \sum_{i\in N} q_i^r x_{ij}^r,\ r \in R \tag{16}$$

$$t_i^r + s_i + t_{ij} - M\left(1 - x_{ij}^r\right) \le t_j^r, \quad \forall (i,j) \in A, i \ne j, \quad \forall r \in R \tag{17}$$

$$a_i y_i^r \le t_i^r \le b_i y_i^r, \quad \forall i \in N, \forall r \in R \tag{18}$$

$$t_o^r \ge \beta \sum_{i\in N} s_i y_i^r, \quad \forall r \in R \tag{19}$$

$$t_i^r \le t_o^r + t_{max}, \quad \forall i \in N, \quad \forall r \in R \tag{20}$$

$$t_o^s + M(1 - z_{rs}) \ge t_o'^r + \beta \sum_{i\in N} s_i y_i^s, \quad \forall r,s \in R, \quad r < s \tag{21}$$

$$\sum_{r\in R}\sum_{s\in R|r<s} z_{rs} \ge |R| - K \tag{22}$$

$$x_{ij}^r \in \{0,1\}, \quad \forall (i,j) \in A, \ \forall r \in R \tag{23}$$

$$y_i^r \in \{0,1\}, \quad \forall i \in N, \ \forall r \in R \tag{24}$$

$$z_{rs} \in \{0,1\}, \quad \forall r,s \in R, \quad r < s \tag{25}$$

$$z_{rs} \in \{0,1\}, \quad \forall r,s \in R, \quad r < s \tag{26}$$

It is always preferable to visit as many consumers as possible, according to objective function (7). Be mindful that the constant α must be set to a value that supports the model in order for it to be considered valid. The constraints (13) and (19) determine the fleet size and the vehicle capacity, respectively. The restrictions (10)–(12) are flow conservation limitations. Visits to clients must adhere to their time window, as indicated in (15). Every two clients who make consecutive trips on the same route must have a matching visit time (14), and the same is true for trips taken by the same vehicle on two separate occasions (18). Finally, each route's setup time must always be taken into account (16), (18).

3.2. Methods for optimization based-on active constraints

This study looked at a set of techniques where the search direction of the active constraint coat is set to fall between an orthogonal Z matrix and a conventional constraint matrix. As a result, if $\hat{A}x = \hat{b}$ is the latest active constraints $n - s$, Z is a $n \times s$ matrix that looks like this:

$$\hat{A}Z = 0 \tag{27}$$

The key tasks that need to be accomplished in each iteration (by generating an appropriate descent direction, p) are as:
a. Calculate the reduced gradient $g_A = Z^T g$.
b. Develop approximations for the reduction of Hessian $G_A = Z^T GZ$.
c. Acquire approximations for systems of equations:

$$\begin{aligned} &Z^T GZp_A = -Z^T g \\ &G_A p_A = -g_A \end{aligned} \tag{28}$$

d. Identify the direction to get $p = Zp_A$.
e. Find the closest approximation to α^* using a line search where:

$$f(x + \alpha^* p) = \min_{\substack{\alpha \\ \{x+\alpha p \text{ feasible}\}}} f(x + \alpha p)$$

Along with having full column ranks, Z is only (algebraically) constrained by (27) in the example above, therefore Z can take on a few forms. In specifically, the Z parallel to the method itself has the following form:

$$Z = \begin{bmatrix} -W \\ I \\ 0 \end{bmatrix} = \begin{bmatrix} -b^{-1}S \\ I \\ 0 \end{bmatrix} \begin{matrix} \}m \\ \}s \\ \}n-m-s \end{matrix} \tag{29}$$

This is a straightforward explanation that will be utilized for exposition in the following section, however it should be noted that it is computationally limited to the factorizations of B that are triangular (LU) and S. There is undoubtedly some incompleteness in the Z matrix calculation.

There is a good reason why Z, whose column is orthonormal ($Z^T Z = I$), is suggested. The Z transformation is key benefit is that it doesn't introduce redundant conditions into the problem reduction (see the aforementioned steps a to d, in particular (28)). In programs where Z is accumulated as a dense matrix, this technique has been applied. The matrix $[B \quad S]$ can be expanded to the expansively distributed/sparse linear constraints using the LDV factorization:

$$[B \quad S] = [L \quad O]DV$$

where L is a triangle, D is a diagonal, and $D^{1/2}V$ is normal, and L and V are accumulated as products. Despite this, this factorization will always be significantly denser than B's LU factorization if S has a large number of columns. As a result, it is dependent on how Z is used in (29). Be advised that B must be treated with the utmost care because to B^{-1} unwanted appearance.

3.3. Summary of the procedure

Building upon the previous discussions regarding the optimization challenges associated with the vehicle routing problem for multi-product and multi-supplier (VRP-MPMS), we introduce an effective algorithm designed to address these complexities. This algorithm integrates advanced mathematical modelling techniques with computational methods to optimize vehicle routing while adhering to critical constraints such as capacity limits and time windows. By establishing a clear framework that defines essential components, including decision variables, objective functions, and constraint, this approach facilitates a systematic solution process. With the assumption of:

a. $[B \quad S \quad N]x = b,\ l \leq x \leq u$ is content x.
b. The function $f(x)$ and vector $g(x) = [g_B \quad g_S \quad g_N]^T$.
c. The number of super basis variables, $s(0 \leq s \leq n - m)$.
d. Factorization, LU, on the base matrix $B\ m \times m$.
e. The quasi-Newton method to the $s \times s$ matrix is $Z^T GZ$.
f. The vector gradient $h = g_S - S^T \pi$.
g. A rr vector meets $B^T \pi = g_B$.
h. The positive convergence tolerances for TOLDJ and TOLRG are both modest.

The model is solved via the generalized reduced gradient method, starting with the Lagrange function and proceeding according to the procedure. After that, the algorithm will work as follows:

Step 1. If $\|h\| >$ TOLRG, step 3.
Step 2. ("PRICE", i.e., add one superbase and Lagrange multiplier calculation).
  a. Govern $\lambda = g_N - N^T \pi$.
  b. Choose $\lambda_{q_1} < -\text{TOLDJ}\left(\lambda_{q_2} > +\text{TOLDJ}\right)$, the λ's is the greatest element whose higher bounds correspond to the variables. If not, STOP; the essential conditions for an ideal solution have been satisfied according to Kuhn-Tucker. If this is not the case;
    – Addition a_q as the new S column.
    – Choice $q = q_1$ or q_2 on the basis of $\left|\lambda_{q_1}\right| = \max\left(\left|\lambda_{q_1}\right|, \left|\lambda_{q_2}\right|\right)$.
    – Add a new, pertinent column to R.
    – Insert λ_1 as a new h element.
  c. S is multiplied by 1.
Step 3. (Direction of search, $p = Zp_s$).
  a. Finish $R^T R p_S = -h$.
  b. Finish LU $p_B = -Sp_S$.
  c. Make $p = \begin{bmatrix} p_B \\ p_S \\ 0 \end{bmatrix}$.
Step 4. (Test Ratio, "CHUZR").
  a. If $\alpha_{\max} = 0$, go to step 7.
  b. If $\alpha_{\max} \geq 0$, maximise α value of $x + \alpha p$ is viable.
Step 5. (Line search).
  a. Find α, an α^* for which $F(x + \alpha^* p) = \min_{0<\theta\leq\alpha_{\max}} f(x + \theta p)$
  b. Convert x to $x + \alpha p$ and f and g to their new x values.
Step 6. (Reduced slope calculation, $\bar{h} = Z^T g$).
  a. Process $U^T L^T \pi = g_B$.
  b. New slope determination, $\bar{h} = g_S - S^T \pi$.
  c. Utilizing α, p_S and $R^T R$, adjust R for gradient $\bar{h} - h$.
  d. Set $\bar{h} - h$.
  e. If $\alpha_{\max} = 0$, proceed to step 7.
Step 7. Here, $\alpha < \alpha_{\max}$ reaches limits and for $p(0 < p \leq m + s)$, the p column variable of $[B \quad S]$ also reaches limits.
  a. If limit is reached by base variable $(0 < p \leq m)$,
    – the p-th column replaced with q-th column of $\begin{bmatrix} B \\ X_B^{\mathrm{T}} \end{bmatrix}$ and $\begin{bmatrix} S \\ X_S^T \end{bmatrix}$
    – $U^T L^T \pi_p = e_p$
    – Changes to L, U, R and π also variation in B
    – determine gradient $h = g_S - S^T \pi$;
    – Go to (c).
  b. Otherwise superbase limit is reached $(m < p \leq m + s)$. Determine $q = p - m$.
  c. Create the q-th variable in nonbasis S at the appropriate limit as follows:
    – Eliminate qth column $\begin{bmatrix} S \\ X_S^T \end{bmatrix}$ and $\begin{bmatrix} R \\ h^T \end{bmatrix}$;
    – Add R to the triangular matrix.
  Subtract s by one and return to step 1.

3.4. Simulation

In this section, we present the simulation results obtained from our proposed hybrid approach for solving the vehicle routing problem for multi-product and multi-supplier (VRP-MPMS) with relaxed time windows. The simulations were conducted using Fortran language in mathematical programming system (MPS) format, which allows for efficient modelling and solving of optimization problems. The aim is to evaluate the effectiveness of our model by comparing the outcomes with established benchmarks. Through a series of iterative tests, we assess various performance metrics, including transportation costs, vehicle utilization, and adherence to capacity constraints. The results obtained from these simulations will provide insights into the practical applicability of our proposed solution in real world scenarios.

```
EXIT -- OPTIMAL SOLUTION FOUND.
NO. OF ITERATIONS              238     OBJECTIVE VALUE      5.1600000000000E+02
NORM OF X               2.395E+03     NORM OF PI                 4.081E+01
PROBLEM NAME   VRP              OBJECTIVE VALUE    5.1600000000E+02
STATUS        OPTIMAL SOLN         ITERATION  238
```

Table 1 illustrates the route for vehicle 1, which departs from the depot and sequentially visits client 1, client 2, client 3, and client 4, continuing in this manner. Similarly, for route 1, vehicle 2 departs from the depot and follows a path to client 3, then from client 1 to client 2, and so forth. Table 2 details the travel routes for vehicles using route 1, starting from the depot to customer 3. From customer 1, the vehicle returns to the depot via route 2. Using route 3, the vehicle travels from customer 4 to customer 7. Finally, route 4 depicts the vehicle's journey from customer 6 to customer 5. Table 3 presents the starting times for each node (customer).

Table 1. Result of binary variables x

		Customer								
Vehicle	Customer	0	1	2	3	4	5	6	7	8
1	0		1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	1	0.00000		1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000		1.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000		1.00000	0.00000	0.00000	0.00000	0.00000
	4	1.00000	0.00000	0.00000	0.00000		0.00000	0.00000	0.00000	0.00000
	5	0.00000	0.00000	0.00000	0.00000	0.00000		1.00000	0.00000	0.00000
	6	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		1.00000	0.00000
	7	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		1.00000
	8	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000	
2	0		0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	1	0.00000		1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	2	1.00000	0.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000		1.00000	0.00000	1.00000	0.00000	0.00000
	4	0.00000	1.00000	0.00000	0.00000		0.00000	0.00000	0.00000	0.00000
	5	0.00000	0.00000	0.00000	0.00000	0.00000		0.00000	1.00000	0.00000
	6	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000		0.00000	0.00000
	7	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		1.00000
	8	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000	
3	0		0.00000	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000
	1	0.00000		1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000		0.00000	0.00000	0.00000	0.00000	1.00000
	4	0.00000	0.00000	0.00000	0.00000		0.00000	0.00000	1.00000	0.00000
	5	0.00000	0.00000	0.00000	1.00000	0.00000		0.00000	0.00000	0.00000
	6	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000		0.00000	0.00000
	7	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		0.00000
	8	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	
4	0		1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	1	0.00000		1.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000		0.00000	0.00000	0.00000	1.00000	0.00000
	4	0.00000	0.00000	0.00000	1.00000		0.00000	0.00000	0.00000	0.00000
	5	0.00000	0.00000	0.00000	0.00000	0.00000		0.00000	0.00000	1.00000
	6	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000		0.00000	0.00000
	7	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		0.00000
	8	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000	

Our computational experiments demonstrate that the proposed hybrid approach effectively reduces transportation costs while satisfying vehicle capacity constraints and relaxed time windows. Specifically, our results show that this method outperforms traditional routing approaches by achieving a more significant reduction in overall transportation expenses. Moreover, the hybrid model not only ensures adherence to vehicle capacity limits but also allows for flexibility in scheduling, making it a viable solution for solving the VRP-MPMS with relaxed time windows. These findings underscore the advantages of our approach in enhancing logistics efficiency and improving service delivery in complex supply chain environments.

Table 2. Result of binary variables z

		Route								
Vehicle	Route	0	1	2	3	4	5	6	7	8
1	0		0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	1	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000		0.00000	0.00000	1.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000
	4	0.00000	1.00000	0.00000	0.00000		0.00000	0.00000	0.00000	0.00000
	5	0.00000	0.00000	1.00000	0.00000	0.00000		0.00000	0.00000	0.00000
	6	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		1.00000	0.00000
	7	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		1.00000
	8	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000	
2	0		0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	1	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	2	1.00000	0.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000		0.00000	1.00000	1.00000	0.00000	0.00000
	4	0.00000	1.00000	0.00000	0.00000		0.00000	0.00000	0.00000	0.00000
	5	0.00000	0.00000	0.00000	1.00000	0.00000		0.00000	0.00000	0.00000
	6	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000		0.00000	0.00000
	7	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		1.00000
	8	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000	
3	0		0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	1	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000		0.00000	0.00000	0.00000	0.00000	1.00000
	4	0.00000	0.00000	0.00000	0.00000		0.00000	0.00000	1.00000	0.00000
	5	0.00000	0.00000	0.00000	1.00000	0.00000		0.00000	0.00000	0.00000
	6	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000		0.00000	0.00000
	7	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		0.00000
	8	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	
4	0		1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	1	0.00000		1.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	1.00000		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000		0.00000	0.00000	0.00000	1.00000	0.00000
	4	0.00000	0.00000	0.00000	1.00000		0.00000	0.00000	0.00000	0.00000
	5	0.00000	0.00000	0.00000	0.00000	0.00000		0.00000	0.00000	1.00000
	6	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000		0.00000	0.00000
	7	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		0.00000
	8	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000	

Table 3. Start time from each node variables t

	Customer							
Vehicle	1	2	3	4	5	6	7	8
1	39.74026	69.48052	40.00000	20.64935	20.00000	10.25974	10.00000	19.87013
2	40.00000	40.00000	60.00000	20.00000	10.00000	30.00000	10.00000	20.00000
3	0.00000	0.00000	30.00000	0.00000	0.00000	59.87013	110.00000	100.12987
4	60.25974	10.51948	0.00000	99.35065	110.00000	19.87013	0.00000	0.00000

4. CONCLUSION

This paper considers a company which operates a fleet of vehicles so as to deliver multiple products from various suppliers to a set of customers with no strict time to be fulfilled in deliveries. The objective is to optimize the routing of these vehicles to minimize the total transportation cost, which includes travel distance, vehicle utilization, and delivery time deviations, while ensuring that customer demand is met and relaxed time windows are respected. The model of the problem was formulated as a combinatorial problem. A hybridization approach was proposed for the exact part, a generalized reduced gradient method was developed in a way to get “near” integer feasible solution. Then a feasible neighborhood search was proposed, based on minimizing the deterioration of the objective function.

REFERENCES

[1] B. Golden, S. Raghavan, and E. Wasil, Eds., *The Vehicle Routing Problem: Latest Advances and New Challenges*, vol. 43. Boston, MA: Springer US, 2008.

[2] W. Cao and W. Yang, "A survey of vehicle routing problem," *Matec Web of Conferences*, vol. 100, 2017, doi: 10.1051/matecconf/201710001006.

[3] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Management Science*, vol. 6, no. 1, pp. 80–91, Oct. 1959, doi: 10.1287/mnsc.6.1.80.

[4] T. Vidal, G. Laporte, and P. Matl, "A concise guide to existing and emerging vehicle routing problem variants," *European Journal of Operational Research*, vol. 286, no. 2, pp. 401–416, 2020, doi: 10.1016/j.ejor.2019.10.010.

[5] O. Bräysy and M. Gendreau, "Vehicle routing problem with time windows, Part I: Route construction and local search algorithms," *Transportation Science*, vol. 39, no. 1, pp. 104–118, 2005, doi: 10.1287/trsc.1030.0056.

[6] S. N. Kumar and R. Panneerselvam, "A survey on the vehicle routing problem and its variants," *Intelligent Information Management*, vol. 04, no. 03, pp. 66–74, 2012, doi: 10.4236/iim.2012.43010.

[7] L. Schrage, "Formulation and structure of more complex/realistic routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 229–232, 1981, doi: 10.1002/net.3230110212.

[8] H. Zhang, H. Ge, J. Yang, and Y. Tong, "Review of vehicle routing problems: models, classification and solving algorithms," *Archives of Computational Methods in Engineering*, vol. 29, no. 1, pp. 195–221, 2022, doi: 10.1007/s11831-021-09574-x.

[9] N. A. El-Sherbeny, "Vehicle routing with time windows: an overview of exact, heuristic and metaheuristic methods," *Journal of King Saud University - Science*, vol. 22, no. 3, pp. 123–131, 2010, doi: 10.1016/j.jksus.2010.03.002.

[10] A. Gutiérrez-Sánchez and L. B. Rocha-Medina, "VRP variants applicable to collecting donations and similar problems: a taxonomic review," *Computers and Industrial Engineering*, vol. 164, 2022, doi: 10.1016/j.cie.2021.107887.

[11] A. Maroof, B. Ayvaz, and K. Naeem, "Logistics optimization using hybrid genetic algorithm (HGA): a solution to the vehicle routing problem with time windows (VRPTW)," *IEEE Access*, vol. 12, pp. 36974–36989, 2024, doi: 10.1109/ACCESS.2024.3373699.

[12] N. Kohl, "Exact methods for time constrained routing and related scheduling problems," Ph.D. thesis, Technical University of Denmark, 1995.

[13] O. Bräysy and M. Gendreau, "Vehicle routing problem with time windows, Part II: metaheuristics," *Transportation Science*, vol. 39, no. 1, pp. 119–139, 2005.

[14] M. Gendreau and C. D. Tarantilis, *Solving large-scale vehicle routing problems with time windows: The state-of-the-art*. Cirrelt Montreal, 2010.

[15] T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins, "Heuristics for multi-attribute vehicle routing problems: a survey and synthesis," *European Journal of Operational Research*, vol. 231, no. 1, pp. 1–21, 2013, doi: 10.1016/j.ejor.2013.02.053.

[16] A. Le Bouthillier and T. G. Crainic, "A cooperative parallel meta-heuristic for the vehicle routing problem with time windows," *Computers and Operations Research*, vol. 32, no. 7, pp. 1685–1708, 2005, doi: 10.1016/j.cor.2003.11.023.

[17] M. W. P. Savelsbergh, "Local search in routing problems with time windows," *Annals of Operations research*, vol. 4, no. 1, pp. 285–305, 1985.

[18] J. Ruiz-Meza, I. Montes, A. Pérez, and M. Ramos-Márquez, "VRP model with time window, multiproduct and multidepot," *Journal of Applied Science and Engineering*, vol. 23, no. 2, pp. 239–247, 2020.

[19] Q. Sun, S. Chien, D. Hu, and X. Chen, "ptimizing customized transit service considering stochastic bus arrival time," *Journal of Advanced Transportation*, vol. 2021, pp. 1–19, Dec. 2021, doi: 10.1155/2021/3207025.

[20] T. Ibaraki, S. Imahori, M. Kubo, T. Masuda, T. Uno, and M. Yagiura, "Effective local search algorithms for routing and scheduling problems with general time-window constraints," *Transportation Science*, vol. 39, no. 2, pp. 206–232, 2005, doi: 10.1287/trsc.1030.0085.

[21] Y. Nagata, O. Bräysy, and W. Dullaert, "A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows," *Computers and Operations Research*, vol. 37, no. 4, pp. 724–737, 2010, doi: 10.1016/j.cor.2009.06.022.

[22] S. N. Kumar and R. Panneerselvam, "A time-dependent vehicle routing problem with time windows for e-commerce supplier site pickups using genetic algorithm," *Intelligent Information Management*, vol. 07, no. 04, 2015, doi: 10.4236/iim.2015.7474015.

[23] E. Babaee Tirkolaee, P. Abbasian, M. Soltani, and S. A. Ghaffarian, "Developing an applied algorithm for multi-trip vehicle routing problem with time windows in urban waste collection: A case study," *Waste Management & Research: The Journal for a Sustainable Circular Economy*, vol. 37, no. 1_suppl, pp. 4–13, Jan. 2019, doi: 10.1177/0734242X18807001.

[24] M. Altinoz and O. T. Altinoz, "Multiobjective problem modeling of the capacitated vehicle routing problem with urgency in a pandemic period," *Neural Computing and Applications*, vol. 35, no. 5, pp. 3865–3882, Feb. 2023, doi: 10.1007/s00521-022-07921-y.

[25] Y. Wu, H. Du, and H. Song, "An iterated local search heuristic for the multi-trip vehicle routing problem with multiple time windows," *Mathematics*, vol. 12, no. 11, May 2024, doi: 10.3390/math12111712.

[26] J. Lu, J. Su, R. Zhao, F. Chen, Q. Wang, and W. Yan, "A fast and accurate self-healing scheme for intelligent distribution networks using mixed integer linear programming," *IEEE Access*, vol. 12, pp. 21586–21595, 2024, doi: 10.1109/ACCESS.2024.3362798.

[27] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research*, vol. 35, no. 2, pp. 254–265, 1987, doi: 10.1287/opre.35.2.254.

BIOGRAPHIES OF AUTHORS

Muliawan Firdaus is currently pursuing his Ph.D. at the Faculty of Mathematics and Natural Sciences, Universitas Sumatera Utara. He holds a bachelor's degree in mathematics education from Universitas Muhammadiyah Sumatera Utara, obtained in 2001, and a master's degree in science in mathematics from Universitas Sumatera Utara, completed in 2004. As a lecturer at the Department of Mathematics, Universitas Negeri Medan, he is actively involved in the professional teacher education program. His academic journey reflects a strong commitment to advancing his knowledge and expertise in the fields of education and science, particularly in computer programming and mathematics. His research interests include optimization algorithms, quadratic assignment problems, discrete optimization, vehicle routing problems, and relaxed time windows in logistics. He can be contacted at email: muliawanfirdaus@unimed.ac.id or muliawanfirdaus@gmail.com.

Herman Mawengkang is a distinguished professor at the Department of Mathematics, Universitas Sumatera Utara in Medan, Indonesia. He completed his bachelor's degree in mathematics at Universitas Sumatera Utara in 1974 and earned his Ph.D. in operations research from the University of New South Wales, Sydney, Australia. Professor Mawengkang has extensive expertise in operations research and optimization, contributing significantly to these fields through his research and teaching. His scholarly work is widely recognized, as evidenced by his h-index of 11 and his role as a reviewer for several prestigious academic journals. He is the author or co-author of over 200 research publications in computer science and applied mathematics. He can be reached at email: mawengkang@usu.ac.id or hmawengkang@yahoo.com.

Tulus is a professor at the Department of Mathematics, Universitas Sumatera Utara. He earned his bachelor's degree in mathematics from Universitas Sumatera Utara (1987), a diploma from Technische Universität Dortmund, Germany (1992), a master's degree in mathematics from Institut Teknologi Bandung (1996), and a Ph.D. in mechanical and material engineering from Universiti Kebangsaan Malaysia (2008). His expertise includes neural networks, decision-making, information systems, numerical analysis, computational fluid dynamics, the finite element method, and fuzzy logic. He can be contacted at email: tulus@usu.ac.id.

Sawaluddin is an associate professor in the mathematics program at Universitas Sumatera Utara, Medan, Indonesia. He completed his bachelor's degree in mathematics from Universitas Sumatera Utara in 1987, obtained his master's degree in computer science from Universiti Kebangsaan Malaysia in 2004, and earned his doctorate in mathematics from Universitas Sumatera Utara in 2014. With expertise in eigenvalues, primitive matrices, linear algebra, operations research, and information science, Sawaluddin has made significant contributions to the field of mathematics. His dedication to advancing mathematical knowledge is evident in his academic journey and extensive research endeavors, with over 100 research publications to his name. He can be contacted via email at sawal@usu.ac.id.