# A Summary and Discussion on the Current State of CVRP Research

Yizhan Bao<sup>1,a,\*</sup>

<sup>1</sup>School of Southeast University, Nanjing, Jiangsu Province, 210000, China a. 15088645573@163.com \*corresponding author

*Abstract:* The Vehicle Routing Problem (VRP)is a fundamental combinatorial optimization problem in the field of transportation. Among its important variants, the Capacitated Vehicle Routing Problem (CVRP) focuses on optimizing delivery routes under vehicle capacity constraints to minimize transportation costs and enhance service quality. This paper systematically reviews the current state of CVRP research, encompassing the applications of exact algorithms and approximate algorithms, including genetic algorithms, ant colony optimization, and simulated annealing. It also explores improvements to the parent genetic algorithm (PGA) in CVRP, such as polymorphic mutation strategies and hybrid jump strategies. Finally, the paper addresses the challenges faced by CVRP in practical applications and proposes future research directions, including algorithm adaptability, exploration of hybrid algorithms, and efficient processing in big data environments. This study aims to provide a comprehensive reference framework for researchers in the CVRP field, promoting the further development and application of related technologies.

*Keywords:* Vehicle Routing Problem, Capacitated Vehicle Routing Problem, Exact Algorithms, Approximate Algorithms.

## 1. Introduction

The Vehicle Routing Problem (VRP)[1], initially introduced by Dantzig and Ramser in 1959, addresses a classical optimization issue in transportation and delivery logistics. VRP's main goal is to devise optimal delivery routes for a set of vehicles with the objective of minimizing transportation costs, travel distances, or time while satisfying customer demands and various vehicle constraints[2].

Among VRP's various extensions, the Capacitated Vehicle Routing Problem (CVRP)[3] is particularly noteworthy. CVRP primarily focuses on effectively planning delivery routes under vehicle capacity constraints to achieve optimal resource utilization. Specifically, CVRP requires allocating a route for each vehicle within a given transportation network, ensuring that all customer demands are met without exceeding each vehicle's capacity limit. This problem is especially relevant for industries like food delivery, logistics management, and medical supply transportation, making its research significant both academically and practically.

With the development of e-commerce and the diversification of customer demands, the complexity of CVRP has increased. Traditional delivery models have become inadequate for meeting modern society's demands for rapid and efficient logistics services. Consequently, researchers are now developing advanced algorithms and models to tackle CVRP's complexities, including constraints

 $<sup>\</sup>bigcirc$  2025 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

like time windows, limited vehicle fleets, and customer prioritization.In recent years, significant evolution has occurred in the methods for solving CVRP, driven by advancements in computational capabilities and the continuous development of intelligent algorithms. The transition from initial exact algorithms to subsequent heuristic and meta-heuristic algorithms reflects researchers' ongoing pursuit of effective methods to obtain high-quality solutions within reasonable timeframes. Research in this field has not only advanced algorithmic theory but also provided robust support for practical applications.

In summary, CVRP, as an optimization problem of substantial real-world significance, has attracted considerable attention from researchers. This paper examines CVRP's main progress, application scenarios, and challenges facing, contributing insights for further studies.

# 2. Summary and Analysis of Existing Algorithms

Over the years, diverse algorithms and approaches have emerged for CVRP, evolving from simple methods to more sophisticated ones, and deepening the field's understanding.

# 2.1. Exact Algorithms

Exact algorithms for the CVRP aim to provide optimal solutions for given delivery problems. These algorithms are typically suited for smaller-scale problems due to their high computational complexity. Exact algorithms ensure the optimal solution through mathematical modeling and optimization techniques. Common exact algorithms for CVRP include Integer Linear Programming (ILP), Branch and Bound, and Dynamic Programming.

# 2.1.1. Integer Linear Programming (ILP)

Integer linear programming is an optimization technique that seeks to minimize total transportation costs or travel distances by constructing a linear objective function and a set of linear constraints to find the integer solution that achieves the optimal goal. While broadly applicable, ILP becomes computationally infeasible for complex environments, limiting its use to relatively simple problems.

## 2.1.2. Branch and Bound

This method divides the problem into subproblems, systematically exploring solution space and discarding infeasible paths. Although it guarantees an optimal solution, its computational demands make it challenging for large-scale problems.

Branch and Bound is a method for systematically searching the solution space by decomposing the problem and using bounds to eliminate infeasible solutions, thereby reducing the computational burden.

Implementation Steps:

Branching: Decompose the problem into subproblems, generating a tree structure of solutions.

Bounding: Calculate the lower bound of the current node; if it exceeds the known optimal solution, prune and abandon that subproblem.

Backtracking: Gradually backtrack to higher levels to continue searching for other possible solutions.

This algorithm guarantees a global optimum and is flexible enough to combine with other algorithms. However, it has a long computation time; for large-scale problems, the computation time can increase rapidly. Additionally, its implementation is complex, requiring a deep understanding of the principles of branching and bounding.

# 2.1.3. Dynamic Programming

DP addresses CVRP by breaking it down into manageable subproblems. This efficient approach quickly yields optimal solutions for small-scale issues but is computationally demanding as the number of customers grows.

Dynamic programming solves problems by breaking them down into smaller subproblems and using the solutions of those subproblems to construct solutions to larger problems. For CVRP, dynamic programming can effectively handle smaller-scale issues.

Implementation Steps: Define State: States can represent the current vehicle's path, the set of served customers, and the current load.

Transition Equation: Use a state transition equation to compute the optimal solution from one state to the next.

Final Solution: Construct the complete path through backtracking.

This algorithm is efficient and can quickly find optimal solutions for small-scale problems. Its structure is clear, making it easy to implement and understand. However, as the number of customers increases, the state space can expand rapidly, leading to excessive computation times. Thus, it is typically suited only for small-scale problems and is challenging to extend to larger instances.

#### 2.2. Approximate Algorithms

#### 2.2.1. Nearest Neighbor Algorithm

The Nearest Neighbor Algorithm is a simple and intuitive heuristic method that constructs a solution by starting from the current customer and always selecting the nearest unserved customer.

This algorithm is easy to implement, has clear logic, and has relatively low time complexity, making it suitable for quickly obtaining initial solutions. However, it is prone to local optima and cannot guarantee a global optimum. Moreover, the quality of the solution can be unstable, as different starting points may yield varying solution qualities.

## 2.3. Meta-Heuristic Algorithms

#### 2.3.1. Genetic Algorithm (GA)

GA emulates biological evolution, optimizing solutions through selection, crossover, and mutation. GA is adaptable and flexible, but it may converge slowly and risk local optima.

The Genetic Algorithm simulates the process of biological evolution, generating a population of solutions through selection, crossover, and mutation operations.

The Genetic Algorithm possesses global search capabilities, enabling it to seek better solutions across a larger solution space. It is also highly flexible, allowing for combinations with other algorithms to form hybrid approaches. However, the algorithm tends to have slow convergence rates, and in some cases, convergence may not meet expectations.

Premature convergence may occur, leading the algorithm to fall into local optima.[4]

## 2.3.2. Ant Colony Optimization (ACO)

ACO simulates ants' pheromone-guided foraging behavior to find optimal paths. It adapts well to dynamic changes but requires careful parameter tuning and can be computationally intensive.

Ant Colony Optimization mimics the foraging behavior of ants, guiding the search process through pheromone concentration.

Implementation Steps:

Initialize Pheromones: Set initial pheromone concentrations on all paths.

Path Selection: Ants choose paths based on pheromone concentration and heuristic information.

Update Pheromones: Update the pheromone concentration based on the quality of the paths, with better paths leaving more pheromones.

Iterations involve repeated path selections and pheromone updates to optimize the solution.

This algorithm exhibits good dynamic adaptability and quality of solutions. However, it has longer computation times and is sensitive to parameter settings.

# 2.3.3. Simulated Annealing (SA)

SA uses probabilistic acceptance of less favorable solutions to escape local optima. It is versatile and can be applied to various CVRP scenarios, though it may converge slowly and require complex parameter tuning.

Simulated Annealing is inspired by the physical annealing process, utilizing random search and a probability acceptance mechanism to avoid local optima.

Implementation Steps:

Initialize Solution: Select an initial solution and temperature parameters.

Neighborhood Search: Randomly select a neighboring solution.

Acceptance Criteria: Decide whether to accept the neighboring solution based on its quality and the current temperature.

Cooling: Gradually lower the temperature, reducing the probability of accepting worse solutions, leading to convergence.

Simulated Annealing can avoid local optima by allowing the acceptance of worse solutions, thereby exploring a broader solution space. It is also adaptable to various problem types. However, it may have slow convergence rates and require significant time to find satisfactory solutions in some cases.

Parameter tuning can be complex, as the algorithm is sensitive to the choice of initial parameters.

## 3. Algorithms Combining Data Mining and Machine Learning

## 3.1. Polymorphic Mutation-based Single-Parent Genetic Algorithm (PM-PGA)

PM-PGA, designed to tackle CVRP's capacity constraints, integrates polymorphic mutation, hybrid jump, and adaptive penalty functions to improve convergence speed and diversity, although it raises computational costs and implementation complexity.

PM-PGA is designed to solve CVRP with capacity constraints. This algorithm primarily addresses the shortcomings of traditional genetic algorithms regarding convergence speed and premature convergence by introducing polymorphic mutation operations, hybrid jump strategies, and adaptive penalty functions.

## **3.1.1. Algorithm Introduction**

Polymorphic Mutation Operations: Classify various mutation operators into local search, global search, and random movements to enhance population diversity and optimization capability.

Hybrid Jump Strategy: Combine individual concentration control with the Metropolis criterion to avoid local optima and improve solution quality.

Adaptive Penalty Function: Design a penalty function based on iteration count and vehicle overload to strengthen penalties for infeasible solutions, enhancing the accuracy of fitness calculations.

Fitness Calculation and Selection Mechanism: Use tournament selection to retain the fittest individuals for further operations.

# 3.1.2. Advantages

Improved Convergence Speed: The algorithm demonstrates significant improvements in convergence speed through polymorphic mutation and hybrid jump strategies.

Enhanced Solution Diversity: The combination of various mutation strategies effectively increases population diversity, reducing premature convergence.

Good Adaptability: The adaptive penalty function allows the algorithm to better handle constraints, improving solution quality.

Broad Application Potential: This algorithm can be applied to various path optimization problems across multiple domains, demonstrating considerable practical significance.

#### 3.1.3. Disadvantages

Higher Complexity: The introduction of multiple strategies may increase the complexity of algorithm implementation and parameter tuning.

Computational Cost: In some cases, hybrid strategies may increase computation time, especially when addressing large-scale instances.

Parameter Dependency: The algorithm's performance may be sensitive to parameter settings, requiring meticulous tuning.

## 3.2. Modified Hybrid Genetic Search (MDM-HGS)

This method enhances CVRP solutions by combining frequent pattern data mining with the Clarke-Wright heuristic, boosting convergence and adaptability. However, its added computational burden and implementation complexity require careful parameter adjustments.

MDM-HGS is an improved hybrid genetic search designed to tackle CVRP with capacity constraints. This algorithm integrates data mining techniques with traditional genetic algorithms to enhance solution quality and convergence speed.

## **3.2.1. Algorithm Introduction**

Basic Framework: MDM-HGS is an enhanced version of HGS-CVRP, employing various optimization strategies.

New Solution Generation Method: This method combines frequent patterns extracted from good solutions with a randomized version of the Clarke-Wright savings heuristic.

Initialization Process: During the initial stage of the algorithm, multiple solutions are generated and improved through local search. If no improvement occurs within a certain number of iterations, new initial solutions are regenerated using data mining methods (MDM).

Data Mining: Utilize frequent pattern extraction methods to mine useful patterns from a set of excellent solutions to guide subsequent search processes.

Performance Evaluation: The algorithm's performance in terms of solution quality and convergence speed is evaluated through comparisons with HGS-CVRP and other algorithms.

Advantages and Disadvantages

## 3.2.2. Advantages

Improved Solution Quality: Experimental results indicate that MDM-HGS significantly outperforms the original algorithm regarding the final gap to the best-known solutions and original integral (PI).

Rapid Convergence: The newly introduced pattern mining technique enables the algorithm to quickly identify high-quality solutions during the early phases of the search.

Flexibility: The algorithm can adjust parameters based on different instances and problem scales, enhancing adaptability.

## 3.2.3. Disadvantages

Computational Complexity: The introduction of the data mining process may add computational burdens, particularly in handling large-scale problems.

Parameter Tuning: The algorithm's performance is reliant on parameter settings, necessitating careful tuning to adapt to different instances.

Implementation Complexity: Compared to traditional genetic algorithms, MDM-HGS is more complex to implement, potentially requiring additional development and testing time.[5]

# **3.3. HGS-CVRP**

## **3.3.1. Algorithm Introduction**

Basic Structure: The algorithm first initializes a population of solutions, then iteratively generates new solutions through parent selection, crossover to produce offspring, local search for improvement, and insertion of results back into the population, repeating this process until termination conditions are met (such as consecutive iterations without improvement or reaching a time limit).

## 3.3.2. Key Components

Crossover Operation: Employs ordered crossover (OX) to generate new solutions, utilizing the dynamic programming algorithm (Split) to optimize solution paths.

Neighborhood Search: Introduces the SWAP\* neighborhood, optimizing solutions by exchanging customers between different routes without inserting them in place. This method effectively explores more potential improvements.

Diversity Management: Maintains population diversity through selection mechanisms for parents and survivors.

Open Source Implementation: The C++ implementation of the algorithm has been open-sourced on GitHub, facilitating replication and use by other researchers.

#### 3.3.3. Advantages

Efficiency: HGS-CVRP excels in solving classical CVRP instances, achieving solution quality comparable to the original HGS in a shorter computational time.

Simplicity: The algorithm's design emphasizes maintaining conceptual simplicity, avoiding complex designs that make it easier to understand and apply.

## 3.3.4. Disadvantages

Computational Complexity: Although SWAP\* can explore neighborhoods in sub-quadratic time, it may still encounter computational bottlenecks when handling large-scale instances.

Local Optima: Like other genetic algorithms, HGS-CVRP may fall into local optima, especially in scenarios with high search space complexity.

Parameter Sensitivity: Algorithm performance may be sensitive to parameter settings (e.g., population size, generations), requiring careful adjustment for optimal results.[6]

## 4. **Optimization Directions for Existing Algorithms**

#### 4.1. Enhanced Neighborhood Search

Diversification of Neighborhood Structures: Introduce more neighborhood structures, such as SWAP\*, to explore different types of solution spaces. Researching other innovative neighborhood operations (e.g., dynamic neighborhoods) could further enhance algorithm performance.

Adaptive Neighborhood Search: Dynamically adjust the selection and search strategies of neighborhoods based on the properties of current solutions to more effectively discover improvement opportunities.

#### 4.2. Improved Diversity Management

Diversity Preservation Mechanisms: Design more efficient diversity preservation strategies to prevent the population from converging to local optima during the search process and to ensure population diversity.

Adaptive Selection Operations: Dynamically adjust selection mechanisms based on population diversity and fitness to encourage the retention of superior individuals and promote diversity.

## 4.3. Integration of Learning Techniques

Ensemble Methods: Combine multiple algorithms (e.g., meta-heuristic algorithms, local searches, exact algorithms) to form an ensemble framework that leverages the advantages of various algorithms to improve overall performance.

Reinforcement Learning: Utilize reinforcement learning techniques to learn and optimize parameter settings and search strategies, allowing the algorithm to adaptively adjust to different types of problem instances.

## 4.4. Parallel and Distributed Computing

Multi-threading and Parallel Processing: Leverage modern computational resources to accelerate algorithm execution through parallel processing and multi-threading, enhancing solution search speed.

Distributed Algorithms: Develop algorithms suitable for distributed computing environments to handle larger-scale instances, enhancing algorithm scalability.

Algorithm Parameter Optimization

Adaptive Parameter Adjustment: Research adaptive mechanisms to dynamically adjust algorithm parameters (e.g., population size, crossover rates) based on the current search state to optimize algorithm performance.

Automation of Parameter Settings: Employ machine learning or other heuristic methods to automate the parameter setting process, reducing manual intervention and increasing efficiency.

#### 4.5. Real-time Dynamic Problem Handling

Dynamic Adjustments: Design algorithms capable of quickly responding to real-time changes in demand and constraints to adapt to dynamic environments in vehicle routing optimization.

Online Algorithms: Investigate online algorithms that can adjust routes instantaneously upon receiving customer requests, enhancing flexibility and responsiveness in practical applications.

Hybridization of Heuristic and Meta-Heuristic Approaches

Hybrid Algorithms: Combine exact algorithms and heuristic algorithms to utilize the efficiency of exact algorithms for small-scale instances and the adaptability of heuristic algorithms for large-scale problems, achieving optimal results.

## 5. Conclusion

This paper primarily introduces three representative and innovative studies in the CVRP domain. The first paper is in Chinese and provides a concise overview of the algorithms currently employed in the CVRP field, comparing their benefits, making it an excellent paper for scholars new to CVRP. The subsequent two papers are derived from renowned international journals, each presenting unique insights into CVRP algorithms through different algorithmic combinations. The second paper discusses the MDM-HGS algorithm, which has seen successful applications in other fields and has achieved remarkable results when applied to CVRP. The third paper explores the HGS algorithm, which also optimizes problem-solving to a certain extent.

This review covers three innovative approaches within CVRP research. First, the PM-PGA approach improves the convergence and adaptability of genetic algorithms for CVRP, albeit with increased complexity. Second, MDM-HGS integrates data mining to boost solution quality and convergence speed, though it requires careful tuning and resources. Finally, HGS-CVRP maintains efficiency and simplicity, but faces challenges with larger instances.

These studies suggest that future CVRP research should focus on enhancing adaptability, integrating cross-domain algorithmic approaches, and optimizing performance for large-scale, dynamic environments. Future developments in this field should focus on analyzing and observing existing excellent algorithms, considering their applications across different domains, and combining algorithmic approaches, which represents a significant trend in current research directions.

#### References

- [1] Li, Xiangyong. (2007). Research on Vehicle Routing Problem Models and Algorithms (Doctoral Dissertation, Shanghai Jiao Tong University). Doctoral. https://kns.cnki.net/kcms2/article/abstract?v= Nyg97wmOeE7E44V1mIGKtu6Jl1mkWy-UpBA--CW3HUOz4x7u6nrMM\_y8xgTmwvrykreMC4B8i-VH296TSkSBQ\_6ThfSndO\_mRVWVBcPxznWf0cdMsPpmkSU855ZJd7fry15WA-80jzJZBgvfaB5hhOerYtXE9CgXKt8IHesXMz9w Gig6cNqpU8Y8wYzStB&uniplatform=NZKPT&language=CHS
- [2] Fu, Zhuo. (2003). Research on Open Vehicle Routing Problem and Its Applications (Doctoral Dissertation, Central South University). Doctoral. https://kns.cnki.net/kcms2/article/abstract?v=Nyg97wmOeE7lynMK7E9Af-pNNevOo2vTFt9O2PSw0ddlPMSY1r61R4QG4PSg783yxAATe3BpA0HVBcyJdnQOORE5ZGveKs-wzaNwyBDAAEAIL4hnE5kVyn9roRDeEaIr3FJFBaw-atYMFytgB30qPd-eMkCHRl5prtM\_EiVhb0FZEyMARWdMB-kxkJB4MU42&uniplatform=NZKPT&language=CHS
- [3] Li, Yang, Fan, Houming, Zhang, Xiaonan & Yang, Xiang. (2017). Stochastic Demand Vehicle Routing Problem and Solution via Hybrid Variable Neighborhood Search Algorithm. Control Theory and Applications (12), 1594-1604.
- [4] Chen, Xiaoli & Tan, Dailun. (2024). Solving the CVRP Problem Based on Multi-Modal Mutation of Single-Parent Genetic Algorithm. Journal of Luoyang Normal University (08), 13-17+26. doi:10.16594/j.cnki.41-1302/g4.2024. 08.010.
- [5] Maia, M. R. D. H., Plastino, A., & Uéverton dos Santos Souza. An improved hybrid genetic search with data mining for the cvrp. Networks, 83, 692.
- [6] Vidal, T. (2022). Hybrid genetic search for the CVRP: Open-source implementation and swap neighborhood. Computers & Operations Research, 140, 105-129. https://doi.org/10.1016/j.cor.2022.105-129