An Adaptive Ant Colony System for Skip Loader Operations in Roll-on/Roll-off Logistics

Luis Deutsch[1], Matthias König[2], and Jan Rexilius[1]

[1] Bielefeld University of Applied Sciences and Arts, 32427 Minden, Germany,
luis.deutsch@hsbi.de
[2] University of Southern Denmark, 6400 Sønderborg, Denmark

Abstract. Skip loader operations in Roll-on/Roll-off logistics involve alternately transporting empty and full waste containers under strict constraints related to vehicle capacity, container compatibility, stackability, and service time windows. These operations require careful coordination of skip handling and route planning to ensure feasibility and efficiency. This paper presents an adaptive Ant Colony Optimization (ACO) system tailored to the unique demands of skip loader logistics. Each service request is modeled as a sequence of up to five atomic operations: loading or unloading empty or full skip, and executing skip swaps. These modular atomic operations enable flexible modeling of diverse operational scenarios, including multi skip stacking and varying disposal or delivery points. Ants construct routes incrementally while maintaining an internal state that tracks vehicle load configurations and skip statuses. This state-aware routing mechanism ensures that all transitions are feasible, avoiding deadlocks and invalid loading states. Routing decisions are guided by a composite heuristic that integrates travel cost, service duration, stacking feasibility, and time window alignment. Pheromone trails are reinforced both on spatial paths and action transitions, guiding the search toward combinations of actions that have previously led to feasible and efficient solutions. The system was evaluated on extensive operational data from municipal waste collection fleets, featuring multiple skip loader types, heterogeneous skip inventories, and realistic time-window constraints. The results show that the approach consistently produces feasible high-quality routes, maintains valid load states, utilizes the vehicle capacity effectively, and converges rapidly. The proposed adaptive ACO framework offers a robust and extensible foundation for solving Roll-on/Roll-off routing challenges and supports modular service planning and the integration of operational constraints in complex skip handling environments.

Keywords: Ant Colony Optimization, Vehicle Routing Problem, Roll-on/Roll-off, Skip Loader, Stackability

1 Introduction

Waste management logistics plays a pivotal role in contemporary urban planning, focusing on the systematic and efficient collection, transportation, and disposal of waste materials. Effective waste management not only contributes to maintaining public health standards and cleanliness but also substantially reduces operational expenditures and environmental burdens such as emissions and noise pollution [10]. Within the scope of logistical optimization, the vehicle routing problem (VRP) emerges as a foundational challenge, centered on determining the most efficient routes for fleets of vehicles to service various geographically dispersed customer locations [8].

Traditionally, waste collection problems within the VRP framework are classified based on their operational characteristics and customer profiles, typically segmented into residential, commercial, and industrial waste collection categories [13]. Each of these categories present distinct operational challenges and constraints, ranging from varied waste types and container sizes to specific time windows for service. A specialized variant, known as the Roll-On-Roll-Off Vehicle Routing Problem (RR-VRP) [4], specifically addresses the logistics associated with the management of large waste containers commonly used at construction sites, industrial facilities, or other bulk waste generation points. The RR-VRP differs significantly from conventional waste collection due to the specific operational procedures involved, such as container delivery, retrieval, exchange, or onsite emptying.

Within the domain of RR-VRP, skip loader vehicles present a particularly complex operational scenario. Unlike traditional waste collection trucks, skip loaders transport large, removable skips that must be strategically managed across multiple locations. The operational complexity increases substantially due to several unique constraints: handling different skip types, managing skip stacking configurations during transportation and storage, and executing diverse service requests including skip delivery, pick-up, exchange, and emptying operations.

These complexities necessitate the extension of classical RR-VRP methodologies, leading to the development of tailored optimization frameworks capable of effectively handling multifaceted logistical constraints. To address these specific challenges, this paper introduces an adaptive Ant Colony Optimization (ACO) framework tailored for skip loader operations in RR logistics. Unlike static or narrowly focused approaches, the proposed system incorporates state-aware decision-making mechanisms that allow individual ants to respond to vehicle states, service requirements, and operational constraints. This adaptivity enables the algorithm to navigate a highly dynamic environment characterized by heterogeneous skip types, time-sensitive requests and stackability restrictions, while still generating high-quality routing solutions.

In addition, this research contributes to the literature by bridging practical skip loader operations with an adaptive metaheuristic framework capable of handling complex real-world logistics scenarios. A detailed case study based on authentic operational data from a municipal waste management provider demonstrates the effectiveness and robustness of the proposed system in improving responsiveness, route efficiency, and operational feasibility. The results highlight the potential of adaptive ant-based optimization to support agile and sustainable decision making in Roll-on/Roll-off waste logistics.

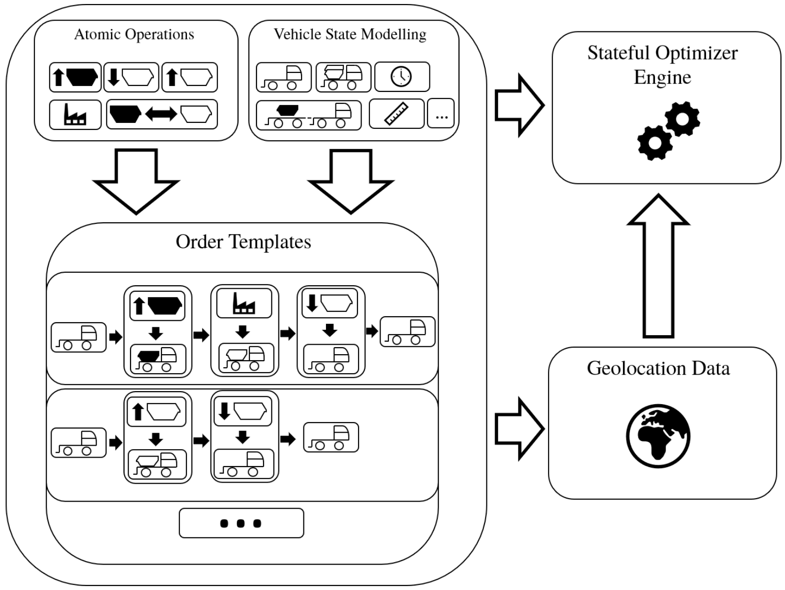

Fig. 1: Proposed system architecture with individual components. Left: Order templates are formed based on a combination of atomic operation and vehicle state modeling. Right. Computational engine that determines an optimized solution based on vehicle and order parameters as well as geolocation data.

2 Related Work

The Roll-on/Roll-off Vehicle Routing Problem (RR-VRP), as a central focus of this paper, represents a particularly challenging and practically relevant domain within waste management and industrial logistics. It involves the efficient routing of specialized vehicles designed to transport large, single-unit containers, commonly known as skips. Unlike classical vehicle routing problems (VRPs), the RR-VRP is defined by unique operational constraints, such as the use of roll-on/roll-off systems that typically limit vehicles to handling a single container at a time. These characteristics demand careful planning of container-specific operations, including delivery, retrieval, exchange, and disposal.
Originally introduced by Bodin et al. (2000) in the context of full-truckload movements between customers and disposal facilities, the RR-VRP has since evolved into a broader class of problems with multiple container types, heterogeneous fleets, time windows, and diverse service types. As noted in the comprehensive survey by Hess et al. (2024), roll-on/roll-off systems are a prominent subcategory in waste collection routing, particularly relevant for the industrial and construction sectors. Their review emphasizes the increasing complexity and variety of RR-VRP formulations, highlighting its divergence from standard VRPs through aspects such as single-container handling, optional repositioning, and multi-facility operations, thereby reinforcing its need for specialized solution

approaches [4][9]. In their original formulation, Bodin et al. define four fundamental trip types that capture the core operational logic of RR-VRP systems: (1) transporting a full customer-owned container to a disposal site and returning it; (2) exchanging a full container for an empty company-owned one; (3) delivering an empty container to a customer; and (4) retrieving a full container without replacement. The authors model the problem as an asymmetric, time-constrained VRP and propose an integer programming formulation supported by several heuristics, demonstrating strong performance across 20 benchmark instances [4]. Building on this structure, Wy and Kim (2013) propose a hybrid metaheuristic for a similar RR-VRP setting involving container exchanges, deliveries, and retrievals. They employ a population-based large neighborhood search combined with problem-specific operators (e.g., route reduction, 2-opt), aiming to minimize fleet size and deadhead time. Their algorithm achieves 17 new best-known solutions on the same benchmark set used by Bodin et al., highlighting its effectiveness in optimizing fleet utilization [15]. Wy et al. (2013b) extend this work by introducing a version with time windows (RR-VRPTW), motivated by real-world industrial waste collection at construction sites and shopping districts. The problem includes seven distinct service types, multiple disposal facilities and container yards, heterogeneous container types and sizes, as well as time windows and driver lunch breaks. To solve it, the authors propose a large neighborhood search (LNS)–based iterative heuristic that integrates route construction, inter- and intra-route improvements, service-type transformations, and vehicle reassignment. Benchmark results, including real-world data, demonstrate significant improvements in both the number of required vehicles and total route time compared to company-generated solutions [16].

A related variant is explored by Archetti and Speranza (2004), who study the 1-skip collection problem where each vehicle carries only one skip at a time and must transport full skips to various disposal plants. Their SMART-COLL heuristic constructs routes via a priority-based nearest-neighbor approach, followed by refinement through insertion and removal heuristics. Tested on data from the city of Brescia, their method significantly outperforms manual planning [1].

Expanding the classical RR-VRP framework, Baldacci et al. (2006) model a more complex scenario involving multiple disposal and inventory depots. Their M-RRVRP is formulated as a set partitioning problem solved via column generation and multilevel Lagrangian relaxation. The algorithm produces optimal or near-optimal solutions in large, realistic instances, demonstrating scalability and generalizability [3].

Raucq et al. (2019) present a real-life RR-VRP that introduces multiple service types (e.g., exchange, round-trip, end-of-contract), heterogeneous vehicle capacities, time windows, and depot constraints. They solve the problem using a column generation approach supported by a constructive heuristic based on Solomon insertion. Applied to large Belgian instances, their method achieves cost savings exceeding 10% compared to commercial solvers [12].

A different extension is presented by Aringhieri et al. (2017), who address roll-on/roll-off routing for bulky recyclable waste. Their model accounts for multiple material types, limited container inventories, and strict route duration limits. They solve it using a hierarchical neighborhood search that alternates between minimizing fleet size and route duration. The algorithm is tested on real-world instances, showing high performance in dynamic and large-scale environments [2].

De Meulemeester et al. (1997) examined a simplified version of the problem involving empty skip deliveries and full skip retrievals under a route-length constraint. They propose savings-based heuristics and an exact branch-and-bound method that leverages route recombination and transportation relaxations. Despite the model's simplicity, it performs well on both synthetic and real data from Belgium [5].

Lu et al. (2017) take a broader urban perspective on roll-on/roll-off waste collection in chinese cities, emphasizing two-stage routing and differentiated collection. Their heuristic, MCMC-RORO, incorporates tabu search and evaluates environmental and operational trade-offs, especially under constraints such as time windows and shift schedules. The study also highlights the role of technologies such as GPS and RFID in enabling smart waste collection [11].

While the studies above provide valuable formulations and algorithmic strategies, they generally focus on fixed-order types and do not consider skip stacking as part of vehicle state modeling. In this paper, we introduce a specialized Ant Colony Optimization algorithm designed to address the specific requirements of RR-VRP with skip loader vehicles. A core innovation is the algorithm's ability to stack empty containers within vehicles a practical constraint with significant implications for routing efficiency that has not been adequately addressed in prior work.

In addition, our method models service requests through atomic operations, allowing for flexible and precise definitions of order types such as retrieval, placement, and exchange. This modular structure enhances adaptability to real-world operational rules. The effectiveness of our system has been validated using authentic operational data from a major waste logistics provider, confirming its applicability in realistic contexts.

By combining adaptive metaheuristics with advanced container handling and flexible order modeling, our approach provides a novel and practically relevant extension to existing RR-VRP literature, suggesting promising opportunities for improved efficiency in complex skip logistics.

3 Concept

3.1 Architecture

The architecture proposed in this paper is composed of several tightly coupled components (cf. Fig. 1). Its foundation rests on two pillars: atomic operations and vehicle-state modeling. An atomic operation is the smallest action a skip loader can perform at a location that still warrants optimization. Vehicle-state modeling simulates every relevant parameter and load configuration of the skip loader. Together, these elements yield order templates-directed graphs whose nodes are atomic operations and whose edges represent feasible transitions. Each template enumerates all valid sequences of atomic operations needed to complete a given order type. A state-aware optimization engine, implemented in this work as an ant colony algorithm, then fuses the order templates with geolocation data, assembles candidate tours, and checks each one against the vehicle-state model to ensure feasibility.

3.2 Atomic Operations

In this section, we define five different atomic operations which most commonly occur when dealing with skip loader operations. Those operations are:

- Pick up empty skip
- Pick up full skip
- Place empty skip
- Empty skip
- Swap skip

The Pick up empty skip operation involves loading an empty skip onto the skip loader, typically carried out at a container depot. The full pick-up, on the other hand, refers to loading a skip that has usually been filled with waste, most often at a customer site, such as a construction area. The Place empty skip operation entails dropping off an empty skip currently carried by the skip loader. This can occur either at a customer location or at a container depot. The Empty skip operation describes the act of unloading a full skip at a disposal facility, effectively converting it into an empty skip. Finally, the Swap skip operation combines two actions: unloading the empty skip from the skip loader and loading a full skip in its place, typically at a customer site.

These atomic operations enable the definition of precise tasks at specific locations, allowing the skip loader to determine which actions are feasible at each stop. Moreover, atomic operations play a crucial role in maintaining a consistent and valid vehicle state by triggering defined state transitions during the execution of a route.

3.3 Vehicle State Modeling

Each vehicle is modeled as a dynamic entity that maintains operational metadata relevant to routing and constraint validation. This includes the current simulation time, cumulative distance traveled, working hours, and the cargo configuration. Additionally, each vehicle retains a record of visited nodes to enable route reconstruction and analysis.

The key parameters defining the vehicle state and operational context are:

- **Trailer Configuration**: Specifies the number of skip slots available on the trailer.
- **Cargo Configuration**: Details which skips are currently loaded onto the vehicle and their respective slots.

- **Vehicle Measurements**: Includes dimensions of the vehicle.
- **Work Time**: Represents both the accumulated working time of the vehicle and the current time-of-day within the planning context (e.g., 14:37). This dual role allows the system to track legal or contractual working hour limits while simultaneously validating customer-specific opening hours, which are often strictly time-bound.
- **Skip Details**: Comprehensive information for each skip, including its size, whether it is lidded, and its current status (full/empty).
- **Vehicle Max Speed**: The maximum operational speed of the vehicle.
- **Current Location**: The vehicle's current geographical position.
- **Cost per KM**: The operational cost incurred per kilometer traveled.
- **Cost per Hour**: The operational cost incurred per hour of work.
- **Service Time per Operation**: The time required to complete a specific atomic operation (e.g., loading, unloading).

By default, vehicles feature a single cargo slot designed to transport skips. In many configurations, multiple skips can be stacked within this slot, subject to well-defined operational and legal constraints:

- **Empty-only stacking:** Only skips that are empty may be stacked. This ensures safe transport and avoids overloading issues.
- **Dimensional compatibility:** Stacking is only permitted when the upper skip is strictly smaller than the skip below it.
- **Lidded Skip Constraint:** Skips equipped with lids (Fig. 2b) must always be placed at the top of any stack and cannot support other skips above them. Additionally, a lidded skip must be the smallest skip within the cargo slot in which it is placed (either on the truck or trailer). This ensures proper fit, stacking stability, and avoids mechanical interference with the lid.
- **Height Limit:** In accordance with § 32 of the German Road Traffic Licensing Regulations (StVZO) [14], the total height of the vehicle, including any stacked skips, must not exceed 4 meters. This constraint applies independently of skip type or lid status and is evaluated per vehicle configuration.

Skips vary in size (Fig. 2a), which affects their stackability. The skip geometry and lid status are modeled explicitly to allow accurate feasibility checks for stacking and routing decisions.

Vehicles may also be equipped with trailers (Fig. 2d), each offering an additional, independent cargo slot. This configuration increases transport capacity and enables more complex tour compositions. Each cargo slot on the truck and trailer is subject to the same stacking and compatibility constraints, but can be loaded and operated independently.

Importantly, the primary rules give rise to additional, higher-order feasibility constraints. For instance, certain operations, such as exchanging an onboard empty skip with a full skip at a customer location, may be disallowed if the resulting configuration would violate stacking rules (e.g., placing a larger full skip beneath a smaller lidded skip). These derived constraints are automatically enforced during planning to ensure that all route configurations remain physically and legally feasible.

(a) Skip size

(b) Skip with lid

(c) Skip stacking

(d) Trailer usage

Fig. 2: Challenges in Skip Handling. This is an illustrative representation of common challenges in skip handling, highlighting various aspects.

3.4 Order Template

Based on the general definition of operations and vehicle parameters in the previous sections, we now propose order templates. Each order template has the structure of a directed graph. The nodes in this graph encapsulate atomic operations along with relevant customer details, skip measurement data, and locations. The child nodes of each node can only be accessed after the atomic operation represented by the current node has been completed. This structure allows every operation type to be precisely modeled, enabling algorithmic processing. Certain atomic operations may be conditionally skipped. For example, picking up a skip can be omitted if the vehicle already has one loaded, or dropping off a skip may be unnecessary if the subsequent customer can use the existing one. To accommodate this, nodes within the order graph can be designated as optional, meaning they are not mandatory for completing the order.
For the specific purpose of using those order types in the evaluation, we define three distinct order types in the form of order templates:

- **Retrieval:** This order type consists of the atomic operations Pick up full skip, Empty skip, and Place empty skip.
- **Placement:** This order type involves combining the operations Pick up empty skip and Place empty skip.

– **Exchange:** This order type is more complex, as it can be executed in two distinct ways:
  1. Pick up full skip, Empty skip, followed by Place empty skip. This sequence resembles the Retrieval type, differing primarily in that the placement occurs at the customer location rather than the container depot.
  2. Pick up empty skip, Swap skip, Empty skip, and Place empty skip. Here, the vehicle exchanges its onboard empty skip with the customer's full skip, empties the full skip, and returns it to the container depot.

Figure 3 illustrates the order graph of an Exchange operation. It shows two possible paths: one where the vehicle empties the customer's skip and another where the full skip is exchanged with an empty one. The optional nodes, indicated by the dotted arrows, primarily facilitate flexible skip handling.

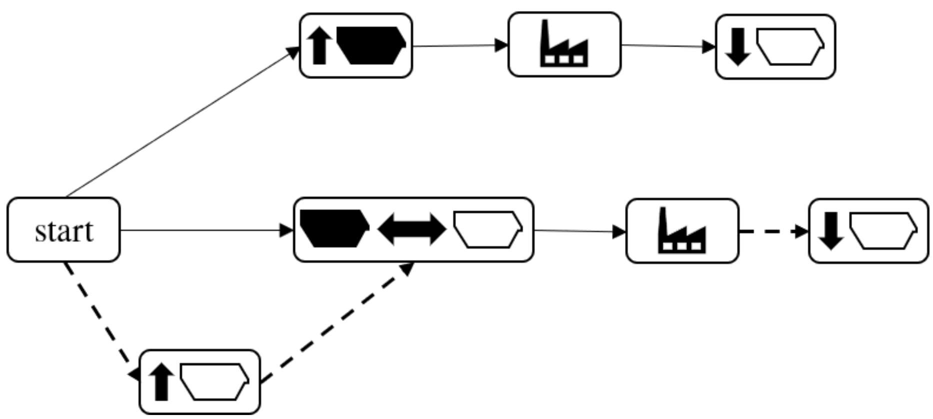

Fig. 3: Representation of an Exchange operation, depicting two possible paths for handling a skip exchange. The diagram shows the sequence of steps, starting from start and leading through a customer interaction. It includes the options of pick up full and drop empty (top path), or an alternative path involving swap and empty operations with a disposal plant or container depot. Optional nodes are indicated by dotted arrows, highlighting the flexibility in the skip handling process.

4 Implementation

4.1 Order Selection

The order selection is the process of retrieving possible nodes from the various order structure graphs. Orders can be strategically combined to achieve more efficient route planning. Using the stacking capability of skips, multiple orders categorized as Placement can be fulfilled sequentially by loading all required skips simultaneously at the depot. Similarly, provided that the skips are compatible, a series of Exchange orders can be executed consecutively without requiring additional visits to the container depot. To effectively optimize routing, an efficient algorithm must identify and exploit these opportunities for concatenation. The process of identifying reachable nodes involves three key steps:

1. Retrieve all nodes from the instantiated order templates.
2. Aggregate all the reachable nodes in this set.
3. Filter out nodes incompatible with the current state of the vehicle.

The resulting set of nodes constitutes the theoretically reachable options. The hard constraints, such as working hours and location-specific opening times, are enforced during the third step.

4.2 Stateful Optimization Engine

The optimization engine is designed to construct high-quality vehicle routes under the constraints of the RR-VRP with skip loaders. At its core, the optimizer takes as input:

- A set of skip loaders, each with detailed configuration (e.g., cargo slots, stacking rules, working time limits).
- A collection of instantiated order templates, modeled as directed graphs of atomic operations.
- Geospatial data, particularly distance and time matrices between all relevant locations, retrieved via an external mapping service.

These inputs define the search space and feasibility constraints of the problem. The optimization process itself is handled by a metaheuristic engine designed to efficiently explore this space and identify feasible, low-cost tours.
To this end, we employ a state-aware Ant Colony Optimization (ACO) algorithm. ACO is particularly well-suited for the RR-VRP context, as its decentralized agents (ants) construct solutions incrementally using only local information. In traditional ACO [7], the transition probability $p_{ij}(t)$ of an ant moving from node i to node j at time t is given by:

$$p_{ij}(t) = \frac{[\tau_{ij}(t)]^{\alpha} \times [\eta_{ij}]^{\beta}}{\sum_{k \in \mathcal{N}_i} [\tau_{ik}(t)]^{\alpha} \times [\eta_{ik}]^{\beta}}, \tag{1}$$

where $\tau_{ij}(t)$ is the pheromone level on edge (i, j), η_{ij} is the heuristic desirability, α and β are influence parameters, and $\mathcal{N}_i$ is the set of feasible neighbors from node i.
While effective in simpler contexts, this approach falls short in capturing the dynamic nature of skip loader operations, where routing decisions are highly dependent on the vehicle's internal state. To address this, we introduce a *stateful pheromone model* in which the pheromone value is defined over tuples of the

form (i, s, j, a), where s represents the vehicle state before the transition, and a denotes the atomic operation executed at location j:

$$p_{(i,s)\to(j,a)}(t) = \frac{[\tau_{(i,s)(j,a)}(t)]^{\alpha} \times [\eta_{(i,s)(j,a)}]^{\beta}}{\sum_{(k,a')\in\mathcal{N}_{(i,s)}}[\tau_{(i,s)(k,a')}(t)]^{\alpha} \times [\eta_{(i,s)(k,a')}]^{\beta}}. \quad (2)$$

Crucially, transitions are only permitted if the resulting vehicle state, after performing atomic operation a at location j, remains valid (e.g., adhering to stacking height limits, skip compatibility, and time window compliance). The concept of this stateful pheromone relation is visually represented in Fig. 4 using an example with two locations. As illustrated in Fig. 4c, considering both the vehicle state and atomic operations for a distinct state yields three unique states. This distinctiveness arises because even with duplicate vehicle states or atomic operations, their combination creates entirely unique configurations. In contrast, Fig. 4b shows that if only the vehicle state is considered, only two distinct states result, as the atomic operation cannot differentiate between occurrences. Furthermore, in Fig. 4a, when neither vehicle state nor atomic operation is considered, only a single state remains, as the entire contextual information of the arrival is absent.

Upon constructing a complete tour, pheromone updates are applied based on the tour's quality. For a solution π, pheromone reinforcement is calculated as follows:

$$\tau_{(i,s)(j,a)}(t+1) = (1-\rho) \times \tau_{(i,s)(j,a)}(t) + \Delta\tau_{(i,s)(j,a)}, \quad (3)$$

$$\Delta\tau_{(i,s)(j,a)} = \begin{cases} \frac{Q}{C(\pi)} & \text{if } (i,s) \to (j,a) \in \pi, \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where $\rho \in (0,1)$ is the evaporation rate, Q is a constant, and $C(\pi)$ represents the cost of solution π. Transitions that lead to constraint violations are penalized by reducing or nullifying the corresponding pheromone deposit.

This enhanced model empowers ants to learn not only where to go, but also how, based on internal states. Consequently, the ACO system converges towards feasible, high-quality skip routing strategies that strictly adhere to operational constraints, as exemplified in Fig. 5. The figure illustrates an optimized tour as a sequence of service locations, each annotated with the executed atomic operation and the vehicle's corresponding configuration. Throughout the tour, the diagram shows how the vehicle's internal state evolves in response to operational decisions, depicting changes in skip arrangements. By tracing these annotated transitions, the figure demonstrates how the final solution reflects a sequence of feasible decisions that respect the evolving vehicle state and the operational context at each location.

4.3 Implementation Details

The system was implemented in Python 3.13 using standard language features and built-in data structures. No external optimization frameworks or numeri-

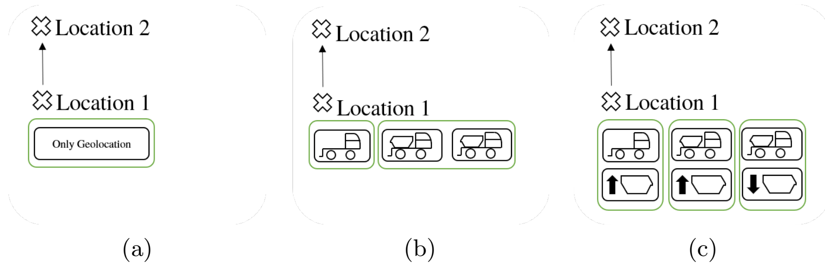

Fig. 4: Conceptual diagram illustrating the different pheromone dimensionality when (not) using a stateful approach. 4a Using no additional information results in 1 distinct state. 4b Using only the vehicle state results in 2 distinct states. 4c Using both vehicle state and atomic operation results in three distinct states

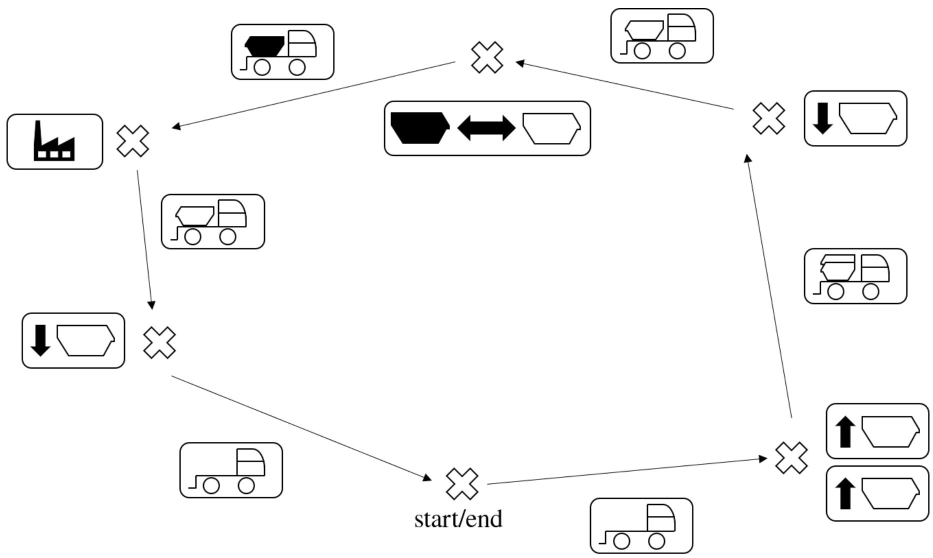

Fig. 5: Representation of an optimized tour where the simplified vehicle state and atomic operations are visualized.

cal libraries were used. Geospatial distance and travel time data were retrieved via the HERE Routing API, enabling realistic estimations based on road network conditions. Pheromone levels were stored in a Python dictionary, with keys constructed as string representations of tuples comprising the origin node, vehicle state, destination node, and atomic operation. This allowed pheromone values to be managed efficiently without additional dependencies. The Ant Colony Optimization algorithm was configured with 50 ants and 100 iterations. The pheromone evaporation rate was set to $\rho = 0.2$, with the pheromone influence parameter $\alpha = 1.5$ and the heuristic influence parameter $\beta = 0.75$. The pheromone reinforcement constant was defined as $Q = 1$. The core entities such as vehicles, skips, and orders were implemented as Python classes to encapsulate their attributes and state transitions.

5 Results

The proposed optimization framework was evaluated using real-world operational data provided by Schwarz IT and PreZero Deutschland. Results across different tour sizes are summarized in Table 1. Tours were grouped based on the number of orders they contained.
The dataset comprises 271 tours, exhibiting a varied distribution with a notable concentration in the 5 to 7 order range, i.e. 37 tours with 5 orders, 59 with 6 orders, and 60 with 7 orders. All tours in the dataset contain between 1 and 10 orders. Regarding travel metrics, the shortest observed distance for a tour was 2.23 KM, while the longest stretched to 408.96 KM, with a mean distance across all tours of 165.13 KM. Similarly, the shortest tour time recorded was 0.41 H, the longest was 9.41 H, and the average tour duration was 5.37 H.
In terms of operational orders, the dataset comprised a total of 290 'Placement' orders, 294 'Retrieval' orders, and 1040 'Exchange' orders.

Table 1: Evaluation results based on real-world data (ACO SkipPlanner vs historical data).

Orders per tour	**Mean cost improvement (%)**	**Mean distance improvement (%)**	**Mean time improvement (%)**	**Mean calculation time per tour (sec)**
1–5 (98 tours)	11.86	11.89	8.36	6.369
6–10 (190 tours)	11.81	11.85	7.31	14.354
All (271 tours)	11.83	11.87	7.62	11.525

As presented in Table 1, the ACO SkipPlanner consistently outperforms the historical baselines across all segments. On average, the optimizer achieves 11.83% cost savings, 11.87% distance reduction, and 7.62% faster execution times. These improvements are particularly notable considering that the historical tours were conducted with only a single vehicle and no trailers, and the optimizer was constrained to the same configuration for direct comparability.
The performance trend remains stable across the two primary tour size segments (1–5 and 6–10 orders).
Figure 6 illustrates the frequency of different operation types for the Historical Data versus the ACO SkipPlanner. The impact of vehicle state modeling, a

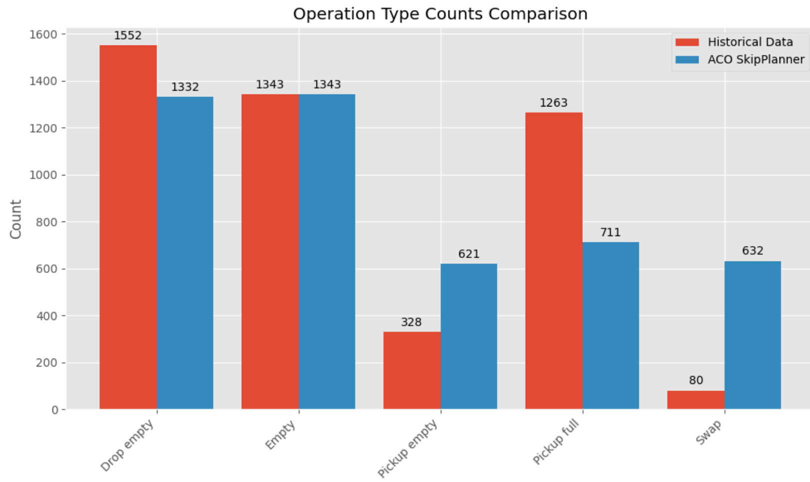

Fig. 6: Comparison of operation type counts. This bar chart illustrates the frequency of different operation types for Historical Data (red/orange) versus the ACO SkipPlanner (blue). It compares the counts for operations such as drop empty, empty, pickup empty, pickup full, and swap. The ACO SkipPlanner consistently shows higher counts for pickup empty and swap operations compared to historical data, suggesting differences in operational strategy or efficiency.

crucial element of the proposed method, is demonstrated in Figure 7. The optimized solution includes a slightly higher average number of operations per tour compared to the historical planning data:

– Average operations per tour (Historical Data): 18.73
– Average operations per tour (ACO SkipPlanner): 18.99

All optimization runs were conducted with 50 ants and 100 iterations, a configuration chosen to balance runtime and solution quality effectively.

6 Discussion

The consistent improvements observed in cost, distance, and time across the majority of tour segments underscore the effectiveness of the ACO SkipPlanner. It is plausible that higher complexity scenarios, such as multi-vehicle routing or the explicit incorporation of trailer support (if consistently applied to both planning and execution), could yield even greater optimization benefits, given the current constraints during evaluation.

The evaluation was performed within a static simulation framework, utilizing fixed service time heuristics that do not account for variations based on operation types or specific customer conditions. In real-world scenarios, service durations can vary significantly due to factors such as accessibility, skip size,

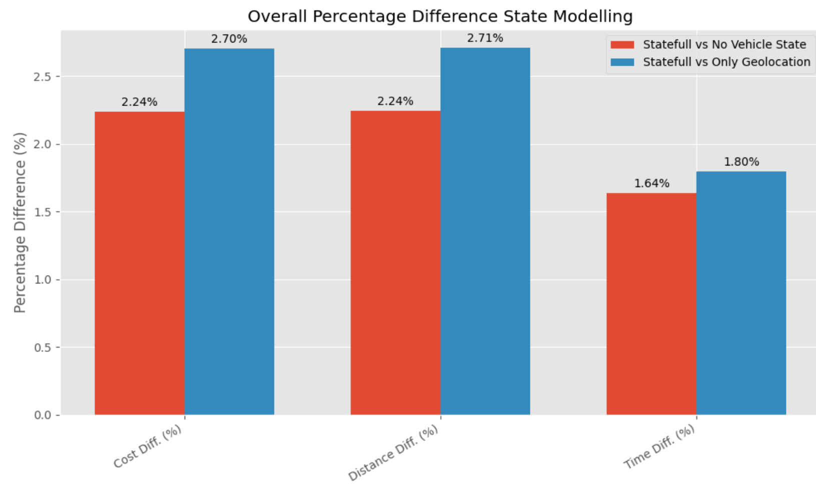

Fig. 7: Comparative analysis of stateful modeling approaches. This bar chart displays the percentage difference in Cost Diff. (%), Distance Diff. (%), and Time Diff. (%) across two stateful modeling approaches. It compares Stateful vs No Vehicle State (red/orange) with Stateful vs Only Geolocation (blue), highlighting the impact of including or limiting vehicle state information on various performance metrics.

and location characteristics. Consequently, the time-related metrics, in particular, are sensitive to these modeling assumptions. The slightly higher average number of operations per tour in the ACO SkipPlanner solution (18.99 vs. 18.73 for historical data) implies that the quality of the optimized solution is more susceptible to variations in these time constants.

As depicted in Figure 6, the ACO SkipPlanner solution strategically utilizes a significantly higher number of swap operations compared to the historical dataset. This operational shift is likely a key factor contributing to the observed improvements in cost efficiency. An important element of the proposed method is the use of vehicle state modeling and atomic operations, whose critical effects are clearly illustrated in Figure 7. The figure presents a comparative analysis of three algorithm variants:

- The full stateful model maintains a detailed internal representation of the vehicle's cargo slots, including loaded skips, their types, stackability, and positions, and integrates atomic operations for state transitions.
- The no vehicle state variant entirely omits cargo slot tracking. While infeasible transitions are still filtered, the absence of state representation causes distinct vehicle configurations to be erroneously merged into the same node representation. For instance, whether the vehicle carries one or three empty skips becomes indistinguishable to the optimizer, as both represent the empty

state. Consequently, the search process loses its ability to differentiate and optimize routes based on specific vehicle state contexts.
- The geolocation-only variant further simplifies the model by removing both cargo and operation awareness. It plans routes based solely on visiting the correct locations, without considering the specific actions or changes occurring at those points.

Both simplified variants consistently demonstrate a marked drop in solution quality across all performance metrics. This conclusively confirms that detailed, state-aware modeling is critical for the optimizer's ability to reason effectively about feasible transitions and make high-quality routing decisions in the complex domain of skip logistics. That said, the reported improvements should be interpreted with appropriate caution. The static assumptions, the lack of variability in service durations, and the limited data available for larger tours mean that the true, real-world effectiveness of the system can only be fully confirmed through comprehensive validation in a live production environment.

7 Conclusion

This paper introduced a comprehensive modeling and optimization framework tailored to the specific challenges of the Roll-On/Roll-Off Vehicle Routing Problem (RR-VRP) in skip loader operations. By modeling skip-handling processes as modular atomic operations such as picking up, placing, emptying, and exchanging skips, the framework provides a flexible yet precise representation of complex service requests. These atomic operations are organized into order templates, which define all feasible sequences of actions to fulfill a given order type, while respecting operational constraints. We applied an Ant Colony Optimization (ACO) algorithm within our framework. This algorithm leverages vehicle state information to make locally informed, context-sensitive routing decisions. The resulting optimization process dynamically assembles and evaluates candidate tours by combining multiple order templates and ensuring consistency with vehicle configurations, stacking constraints, and time windows.
The proposed framework was evaluated using real-world operational data data. Our results demonstrate notable improvements in route quality, particularly in time efficiency, travel distance, and overall operational cost. These findings highlight the real-world applicability and potential of combining atomic operations and vehicle state modeling within a stateful, metaheuristic-based routing framework. Future work will explore the integration of live operational data to enhance responsiveness, as well as adaptations of the method for broader use in dynamic logistics and waste collection environments.

Acknowledgements

We would like to express our sincere gratitude to Schwarz IT KG for their constructive collaboration and their support in the form of financial funding, expertise and data, which significantly contributed to the success of this research.

References

1. Archetti, C., Speranza, M.G.: Vehicle routing in the 1-skip collection problem. J. Oper. Res. Soc. 55(7), 717–727 (2004). doi:10.1057/palgrave.jors.2601743
2. Aringhieri, R., Bruglieri, M., Malucelli, F., Nonato, M.: A Special Vehicle Routing Problem Arising in the Optimization of Waste Disposal: A Real Case. Transp. Sci. 52(2), 277–299 (2017). doi:10.1287/trsc.2016.0731
3. Baldacci, R., Bodin, L., Mingozzi, A.: The multiple disposal facilities and multiple inventory locations rollon–rolloff vehicle routing problem. Comput. Oper. Res. 33(9), 2667–2702 (2006). doi:10.1016/j.cor.2005.02.023
4. Bodin, L., Mingozzi, A., Baldacci, R., Ball, M.: The Rollon–Rolloff Vehicle Routing Problem. Transp. Sci. 34(3), 271–288 (2000). doi:10.1287/trsc.34.3.271.12301
5. De Meulemeester, L., Laporte, G., Louveaux, F.V., Semet, F.: Optimal Sequencing of Skip Collections and Deliveries. J. Oper. Res. Soc. 48(1), 57–64 (1997). doi:10.2307/3009944
6. Dorigo, M., Blum, C.: Ant colony optimization theory: A survey. Theor. Comput. Sci. 344(2), 243–278 (2005). doi:10.1016/j.tcs.2005.05.020
7. Dorigo, M., Stützle, T.: Ant Colony Optimization: Overview and Recent Advances. In: Gendreau, M., Potvin, J.-Y. (eds.) Handbook of Metaheuristics, pp. 311–351. Springer, Cham (2019). doi:10.1007/978-3-319-91086-4_10
8. Eksioglu, B., Vural, A.V., Reisman, A.: The vehicle routing problem: A taxonomic review. Comput. Ind. Eng. 57(4), 1472–1483 (2009). doi:10.1016/j.cie.2009.05.009
9. Hess, C., Dragomir, A.G., Doerner, K.F., Vigo, D.: Waste collection routing: a survey on problems and methods. Cent. Eur. J. Oper. Res. 32, 399–434 (2024). doi:10.1007/s10100-023-00892-y
10. Li, W., Ng, T.F., Ibrahim, H., Wang, S.L.: A literature review of the state of the art of sustainable waste collection and vehicle routing problem. J. Air Waste Manag. Assoc. 75(1), 3–26 (2025). doi:10.1080/10962247.2024.2415298
11. Lu, J.-W., Chang, N.-B., Liao, L., Liao, M.-Y.: Smart and Green Urban Solid Waste Collection Systems: Advances, Challenges, and Perspectives. IEEE Syst. J. 11(4), 2804–2817 (2017). doi:10.1109/JSYST.2015.2469544
12. Raucq, J., Sörensen, K., Cattrysse, D.: Solving a real-life roll-on–roll-off waste collection problem with column generation. J. Vehicle Routing Algorithms 2(1), 41–54 (2019). doi:10.1007/s41604-019-00013-6
13. Rodrigues, S., Martinho, G., Pires, A.: Waste collection systems. Part A: a taxonomy. J. Clean. Prod. 113, 374–387 (2016). doi:10.1016/j.jclepro.2015.09.143
14. Bundesministerium für Digitales und Verkehr: § 32 StVZO – Abmessungen von Fahrzeugen und Fahrzeugkombinationen (2012). https://www.gesetze-im-internet.de/stvzo_2012/__32.html. Accessed: 2025-05-25
15. Wy, J., Kim, B.-I.: A hybrid metaheuristic approach for the rollon–rolloff vehicle routing problem. Comput. Oper. Res. 40(8), 1947–1952 (2013). doi:10.1016/j.cor.2013.03.006
16. Wy, J., Kim, B.-I., Kim, S.: The rollon–rolloff waste collection vehicle routing problem with time windows. Eur. J. Oper. Res. 224(3), 466–476 (2013). doi:10.1016/j.ejor.2012.09.001