

AI-Based Smart Cargo Booking and Empty Return Load Matching System: A Review of Rule-Based, Optimization, and Hybrid Matching Approaches

Yevle Pruthviraj Govardhan, Prof. Abhishek Popat Bangar, Prof. S. B. Bhosale, Prof. Karad V. A.

Department of Computer Engineering
Jaihind College of Engineering, Kuran, Maharashtra, India

Abstract- Booking cargo takes far more than listing loads. Shippers, carriers, vehicles, routes and time windows all have to fit together under firm rules and cost goals, while many trucks still return empty after delivery. This paper looks at five publications relevant to the problem: a review of the causes of empty miles in freight trucking, a two-part survey of pickup and delivery problems (including vehicle routing with backhauls), a study of dynamic pickup and delivery problems, and a seaport truck-sharing model that matches loads to empty truck slots. Between them they draw on search-and-matching economics, vehicle routing with backhauls, dynamic re-planning and computer-based load matching. The trucking review reports an average empty-mile share of 29% across 54 collected estimates, with matching technology lowering empty-trip shares by 4 to 5 percentage points in some studies. Overall, the papers suggest that no single method suits every platform. A practical smart booking system should pair checked input data and strict hard-constraint enforcement with a feasibility-first, improvement-second matching strategy, optional AI-based prediction, open evaluation and human review. Key open problems are scalability, benchmark coverage, fair evaluation, real-time re-matching, and the risk of accepting a generated match without checking it.

Keywords— smart cargo booking, empty return load matching, backhaul, empty miles, vehicle routing with backhauls, pickup and delivery, freight matching, AI

I. INTRODUCTION

Load matching means placing a given set of shipments onto vehicles in space and time so that a collection of goals is met as well as possible, while respecting a set of constraints. In cargo transport, the resources are vehicles, drivers and routes, and what gets placed are shipments, booked loads and return loads. Done by phone or by hand, the job is slow and tends to produce empty return trips, double bookings, poorly filled vehicles and missed pickup windows. Routing problems with pickups, deliveries and time windows are computationally hard, so working systems usually settle for a feasible, good answer instead of a provably optimal one [2], [3].

The cost of getting this wrong is large. A review of the empirical literature collected 54 estimates covering 40 years and 27 countries and found an average empty-mile share of 29%, with high-income countries tending to have lower shares than low- and

middle-income ones [1]. The same review identifies three mechanisms behind empty trips: geographic imbalance in freight demand, search and matching frictions, and regulatory barriers [1]. Only the second is directly addressed by software.

A dependable system therefore has to deal with several levels of requirements instead of leaning on one rule. Constraint validation enforces hard rules such as vehicle capacity, body type and pickup windows. Rule-based matching pairs return loads with returning trucks by route and date. Optimization models weigh costs, detours and soft preferences. AI-based components can predict where return loads will appear. To be useful in practice, the system also needs a web application around it, with data entry, role-based views, booking confirmation and export.

This review brings together five publications and compares their methods, reported results, limitations and what they mean for a smart cargo

booking and return-load matching system. It stays within these sources: it is a focused narrative review, not a fresh systematic search or an independent experiment.

Review approach and scope. We summarise the five publications in narrative form and do not claim a new systematic search. The sources play different roles: one broad review gives the overall picture of why empty trips occur, a two-part survey describes the routing models for backhauls and pickup-and-delivery, one paper covers the dynamic (changing-demand) case, and one presents a deployable truck-sharing matching model. Findings are given as the source papers state them, and numbers from different studies are not treated as directly comparable. For some of the works we relied on their published abstracts and indexing records rather than the full text, and we did not repeat any experiment or check any original dataset. Commercial load boards and freight platforms also exist, but their material is promotional and not peer-reviewed, so it is not used as evidence here. The architecture and recommendations that follow are a synthesis and not a newly tested implementation.

Table 1. Matching layers and their complementary roles

Layer	Typical mechanism	Strength / limitation
Booking validation	Check capacity, vehicle type, route and time rules before each assignment	Conflict-free output; needs precise data
Rule-based matching	Pair return loads with returning trucks by route and date	Simple and fast; ignores cost trade-offs [5]
Optimization models	Routing with backhauls, pickup-and-delivery models	Handles cost and time windows; models grow large [2], [3]
Dynamic re-planning	Update matches as requests arrive over time	Needed for cancellations and delays; harder to evaluate [4]

Layer	Typical mechanism	Strength / limitation
Heuristic / metaheuristic	Local search, annealing, evolutionary search	Scales to many loads; no optimality guarantee
AI / ML prediction	Forecast return-load demand, delays and prices	Proposed extension; not evaluated in the reviewed sources

II. REVIEWED STUDIES AND APPROACHES

1. Causes of Empty Miles in Freight Trucking

Yang reviews the empirical evidence on empty trips, empty miles and backhaul probabilities from the economics and transportation literature [1]. A meta-analysis of 54 estimates gives an average empty-mile share of 29%, with a 42% backhaul probability reported alongside it, and with significant variation across settings. The review examines three mechanisms: geographic imbalance in freight demand, where carriers on the busier direction cannot find return loads; search and matching frictions, where carriers cannot find work and may travel empty to another market; and regulatory barriers such as restrictions on foreign carriers, with cabotage the most commonly restricted. Empirical research points to cases where improved matching technology lowered empty-trip shares by 4 to 5 percentage points, while cautioning that the effect of matching may not be monotonic [1]. A stylised model is used to evaluate policies.

This paper supplies the broad background for our review: a booking system can reduce only the matching-friction part of empty running, and cannot remove demand imbalance or regulation.

2. Vehicle Routing with Backhauls and Pickup-and-Delivery Models

Parragh, Doerner and Hartl survey pickup and delivery problems in two parts [2], [3]. Part I covers transport between customers and a depot and defines the Vehicle Routing Problem with Backhauls (VRPB), whose subtypes include clustered backhauls

(all linehauls before backhauls), mixed linehauls and backhauls, divisible delivery and pickup, and simultaneous delivery and pickup [2]. Part II covers transport between pickup and delivery locations: the pickup and delivery vehicle routing problem (unpaired points), the classical pickup and delivery problem (paired points) and the dial-a-ride problem [3].

For our topic, the clustered-backhaul variant is the closest fit to the idea of delivering first and then carrying a return load, while the paired pickup-and-delivery models fit loads with fixed origins and destinations. The surveys classify models and methods rather than prescribe one for freight booking, and we did not extract benchmark figures from them.

3. Dynamic Pickup and Delivery Problems

Berbeglia, Cordeau and Laporte review pickup and delivery problems in which requests become known over time instead of being fixed in advance [4]. This is the situation a live booking platform faces: new shipments, cancellations and delays arrive while trucks are already on the road. The implication is that a matching engine should be able to re-plan incrementally rather than only produce one plan per day. We summarise the focus of this work from its title and subject and draw no numerical result from it.

4. Truck-Sharing and Computer-Based Load Matching

Islam, Olsen and Ahmed study the reengineering of the seaport container truck hauling process to reduce empty-slot trips [5]. They explore a dynamic truck-sharing facility, with a computer-based matching system that assigns probable export containers to available empty slots of container trucks, built on the truck appointment system and evaluated through a case study. They argue that the empty-trucks problem is rarely treated from a decentralised view in which all operators have an equal chance to contribute, in contrast with optimisation inside one company [5]. Results are qualitative, tied to the seaport setting, and should not be generalised to road freight in general.

5. Commercial Platforms and Load Boards

Digital load boards and freight marketplaces already offer automated load matching, rate comparison and tracking. Their documentation is promotional and not peer-reviewed, and it gives no reproducible evaluation, so we treat it only as evidence that such products exist, not as evidence of how well any method performs.

III. COMPARATIVE DISCUSSION

The five publications show complementary layers of a matching system. The trucking review [1] explains why empty trips occur and where software can help; the routing surveys [2], [3] supply the formal models for return loads; the dynamic study [4] addresses changing demand; and the truck-sharing model [5] shows how a computer-based matching facility can be packaged around an existing booking process.

Constraint validation with simple matching remains a quick way to get a conflict-free baseline, but it does not optimise cost or detours. Optimisation-based methods can handle time windows, capacity and weighted penalties, whereas heuristic and metaheuristic methods generally scale better to very large load pools. Because demand changes continuously, the dynamic view [4] argues for re-planning rather than one-shot matching. Across all of these, the reviewed literature supports a layered design more than reliance on one universal algorithm.

The reported numbers cannot be compared directly, because the studies differ in task, data, setting and metrics. The 29% figure in [1] is a meta-analytic average across many countries, [5] is a qualitative seaport case study, and [2]–[4] are surveys without a common benchmark. For that reason, this review does not rank the approaches by placing their scores side by side as though they measured the same thing.

The practical lesson is that feasibility and optimisation should be handled separately: a first stage finds a match that meets every hard constraint, and a second stage lowers empty distance and cost without losing feasibility. A single quality score can

hide whether a match is poor because of a long detour, a late pickup or a low fill rate, so reports should show what each component contributes.

Table 2. Comparison of the five reviewed sources

Study / method	Evidence reported	Implication
Yang [1] — review and meta-analysis	54 estimates, 40 years, 27 countries; mean empty-mile share 29%; matching technology lowered empty-trip shares by 4–5 points in some studies	Software addresses matching friction only; imbalance and regulation remain
Parragh et al. [2] — VRP with backhauls	Classification of backhaul variants and methods	Clustered-backhaul models fit return loads
Parragh et al. [3] — pickup and delivery	Classification of paired and unpaired models	Fits loads with fixed origin and destination
Berbeglia et al. [4] — dynamic pickup and delivery	Review of the dynamic case	Plan for continuous re-matching
Islam et al. [5] — truck-sharing matching	Case study of a computer-based matching system for empty truck slots	Matching can be built into booking; evidence is qualitative

IV. Proposed Design for a Smart Cargo Booking and Return-Load Matching System Drawing on the reviewed evidence, a practical system can be organised into the following stages:

- **Data collection and validation:** gather shipments, vehicles, drivers, routes, load types and time windows through a web interface; check for missing values, inconsistent weights or

volumes, and invalid assignments before matching.

- **Preprocessing and reduction:** remove options that can never be feasible (wrong body type, capacity too small, window impossible) and group loads by lane to keep the model small.
- **Priority and special-load placement:** place loads with strict requirements first, such as fixed pickup times, temperature control or restricted goods.
- **Feasible match construction:** assign loads to vehicles while checking hard constraints at every step, and flag any load that cannot be placed; for larger instances, add constraints gradually. Return legs are treated as backhaul candidates [2].
- **Improvement:** reduce empty distance, detours and cost with local search or fix-and-optimize while keeping the match feasible.
- **AI return-load prediction (optional):** forecast where return loads are likely to appear so that matching can start before the truck is empty. This is a proposed extension and is not evaluated in the reviewed sources.
- **Dynamic re-matching:** update matches when bookings, cancellations or delays arrive [4], [5].
- **Report, review and approval:** show conflicts and unplaced loads, let administrators adjust inputs and regenerate, provide role-based views for shippers, carriers and administrators, and export reports.

This is a proposed synthesis of the reviewed literature and not a system that has been built or experimentally validated here.

Evaluation framework. A practical evaluation should report hard-constraint violations (which should be zero), the empty-distance share before and after matching, total cost and its components, the share of loads matched, and the gap to a best-known solution where one exists, together with runtime and memory use. Results should also name the data used, the hardware and the time limit.

Recommended validation protocol. First, test on shared or public instances where possible, keeping hardware and time limits comparable. Second, include small, medium and large instances, as well as

instances with different emphasis (tight time windows, many return lanes, mixed vehicle types). Third, check hard-constraint satisfaction with an independent validator. Fourth, test re-matching after changes such as a cancelled booking or a vehicle breakdown. Finally, have trained dispatchers inspect a sample of generated matches before deployment.

Table 3. Suggested evaluation measures for future systems

Measure	What it tells the evaluator	Caution
Hard-constraint violations	Whether the match is usable at all	Must be zero; report unplaced loads
Empty-distance share	Reduction in empty running	Compare on the same lanes and data
Total cost and components	Quality under business weights	Compare only when definitions match
Match rate and response time	Loads matched and speed	Hardware and volume differ
User satisfaction review	Whether shippers, carriers and dispatchers accept the result	Needs surveys or field trials; rarely reported

V. CHALLENGES AND RESEARCH GAPS

First, scalability remains a central difficulty: routing and matching models grow quickly as loads, vehicles and constraints grow [2], [3]. Second, no single dataset represents every market, and the share of empty miles varies widely between countries and settings [1], which limits generalisation. Third, evaluation should report several measures and comparable resources instead of one score. Fourth, demand imbalance and regulation set limits that no matching software can remove [1], so systems should report infeasibility and unavoidable empty legs clearly. Fifth, a generated match should not be

treated as final, and the effect of matching technology may not be monotonic [1].

Future work should test practical systems on shared benchmarks; study decentralised matching in which many small operators participate fairly [5]; develop efficient re-matching methods for the dynamic case [4]; combine feasibility and improvement stages in one multi-objective formulation; evaluate AI-based return-load prediction against simple rules; and report quality, resource use and user outcomes together.

Operational and governance considerations.

Organisations should decide who may edit constraints, who may approve a match, and how changes are recorded. The system should keep an audit trail of input data, constraint settings, weights and generated versions. A generated match should be presented as a proposal awaiting approval and not as an automatic decision, and shippers and carriers should be able to view their bookings and report problems.

Data and constraint handling. A matching system is only as good as its data. If vehicle capacities, load weights or time windows are wrong, even a strong algorithm will produce a poor result. Data should be stored in a structured form, distances and travel times should be pre-calculated, and constraints should be configurable to local rules.

Interpretable reports. A useful report should identify unplaced loads, show each conflict and its reason, indicate which soft goals were missed, and separate cost, distance, time and fill-rate components, instead of showing a single opaque score.

Human-centred workflow. The system should support dispatcher judgement, not replace it. A dispatcher can adjust inputs, relax constraints, regenerate a match and approve it. Where the consequences are serious, such as hazardous or high-value cargo, a second review is appropriate.

Deployment and maintenance. After deployment, the system should be monitored for changes in

lanes, vehicles, regulations and demand. Changes to constraints, weights or algorithms should be versioned and tested before release, and evaluation should be repeated after major updates, because a result in one setting does not guarantee the same performance elsewhere.

VI. CONCLUSION

The reviewed research shows that smart cargo booking and return-load matching work best as a combination of methods and not as a single algorithm. Constraint validation with rule-based matching offers a simple, usable and conflict-free baseline; backhaul and pickup-and-delivery optimisation models handle capacity, time windows and cost; heuristic methods scale better to large load pools; and dynamic re-planning is needed because demand changes continuously. Matching technology can reduce empty trips, but only the part caused by search and matching frictions. The studies report useful findings, but differences in tasks, data and evaluation conditions limit direct comparison, and the AI prediction layer remains a proposal rather than an evidenced result. A dependable system should combine validated data, complementary matching stages and open evaluation on representative data, and it should leave dispatchers and administrators in charge of consequential decisions.

REFERENCES

1. R. N. Yang, "Geographic imbalance, search frictions, and regulation: Causes of empty miles in freight trucking," Policy Research Working Paper 10775, World Bank, Washington, DC, May 2024.
2. S. N. Parragh, K. F. Doerner, and R. F. Hartl, "A survey on pickup and delivery problems, Part I: Transportation between customers and depot," *Journal für Betriebswirtschaft*, vol. 58, no. 1, pp. 21–51, 2008.
3. S. N. Parragh, K. F. Doerner, and R. F. Hartl, "A survey on pickup and delivery problems, Part II: Transportation between pickup and delivery locations," *Journal für Betriebswirtschaft*, vol. 58, no. 2, pp. 81–117, 2008.
4. G. Berbeglia, J.-F. Cordeau, and G. Laporte, "Dynamic pickup and delivery problems," *European Journal of Operational Research*, vol. 202, no. 1, pp. 8–15, 2010.
5. S. Islam, T. Olsen, and M. D. Ahmed, "Reengineering the seaport container truck hauling process: Reducing empty slot trips for transport capacity improvement," *Business Process Management Journal*, vol. 19, no. 5, 2013.