

GIẢI QUYẾT BÀI TOÁN ĐỊNH TUYẾN XE TRONG MẠNG LƯỚI CHUỖI CUNG ỨNG BẤT ĐỊNH THEO CÁCH TIẾP CẬN ĐA MỤC TIÊU VÀ LẬP KỊCH BẢN

SOLVING THE VEHICLE ROUTING PROBLEM IN UNCERTAIN SUPPLY
CHAIN NETWORKS USING A MULTI-OBJECTIVE AND SCENARIO-BASED
APPROACH

LÊ THẾ MINH^{1*}, PHẠM BÙI KHÁNH LINH¹, TẠ VIỆT HÀ¹, ĐẶNG THU THẢO¹,
PHẠM NGỌC DIỆP¹, NGUYỄN THỊ LÊ HẰNG²

¹Viện Đào tạo Quốc tế, Trường Đại học Hàng hải Việt Nam

²Khoa Kinh tế, Trường Đại học Hàng hải Việt Nam

*Email liên hệ: minh96724@st.vimaru.edu.vn

DOI: <https://doi.org/10.65154/jmst.978>

Tóm tắt

Phần này tóm tắt mục đích của bài viết, giúp độc giả nghiên cứu giải quyết bài toán định tuyến xe trong mạng lưới chuỗi cung ứng dưới bất định theo cách tiếp cận đa mục tiêu và lập kịch bản. Bài toán được ứng dụng mô hình hỗn hợp tuyến tính mờ (FMILP) và giải bằng thuật toán Branch-and-cut và thuật toán Adaptive Large Neighborhood Search (ALNS) để tăng tốc độ xử lý bộ dữ liệu quy mô lớn. Phương pháp lập kịch bản cho phép xây dựng các kịch bản tối ưu với các mục tiêu khác nhau như: (i) tối thiểu chi phí vận hành, (ii) tối thiểu thời gian giao hàng và (iii) tối thiểu khối lượng phát thải CO₂. Kết quả thực nghiệm cho thấy việc thay đổi mức độ ưu tiên giữa các mục tiêu sẽ tạo ra những phương án tối ưu khác nhau, tương ứng với mục tiêu được đặt lên hàng đầu. Nghiên cứu có tính khả thi và phù hợp triển khai cho doanh nghiệp Việt Nam.

Từ khóa: Định tuyến xe, VRP đa mục tiêu, lập kịch bản, FMILP, Branch-and-cut, ALNS.

Abstract

This study addresses the vehicle routing problem (VRP) in uncertain supply chain networks using a multi-objective and scenario-based approach. The problem is formulated as a Fuzzy Mixed-Integer Linear Programming (FMILP) model and solved using a Branch-and-Cut algorithm in combination with Adaptive Large Neighborhood Search (ALNS) to accelerate computation on large-scale datasets. The scenario-based method enables the construction of optimal scenarios under different objectives, including: (i)

minimizing operating cost, (ii) minimizing delivery time, and (iii) minimizing CO₂ emissions. Experimental results indicate that varying the priority levels among objectives leads to different optimal solutions, consistent with the objective emphasized. The proposed approach is feasible and well suited for implementation by Vietnamese enterprises.

Keywords: Vehicle routing, multi-objective VRP, scenario planning, FMILP, branch-and-cut, ALNS.

1. Mở đầu

Giải quyết bài toán định tuyến đang trở thành một năng lực cốt lõi quyết định hiệu quả vận hành và sức cạnh tranh của ngành logistics Việt Nam trong bối cảnh hội nhập quốc tế và chuyển đổi số. Theo Ngân hàng Thế giới, chi phí logistics tại Việt Nam hiện vào khoảng 16,8%, cao hơn đáng kể so với mức trung bình 10-13% của các nước ASEAN [1]. Trong cấu trúc chi phí này, vận tải chiếm gần 60% tổng chi phí logistics và vận tải đường bộ đảm nhận khoảng 77% khối lượng hàng hóa. Vì vậy, các bất hợp lý trong định tuyến không chỉ làm tăng chi phí vận hành ở cấp doanh nghiệp mà còn khuếch đại chi phí toàn hệ thống. Bên cạnh đó, tỷ lệ xe chạy rỗng chiều về có thể đạt 50-70% [2], kéo theo gia tăng quãng đường và thời gian khai thác nhưng không tạo thêm doanh thu.

Các hạn chế về chất lượng phương tiện và tác động môi trường càng làm vấn đề định tuyến trở nên cấp thiết. Cụ thể, hơn 90% đội xe tải có tuổi đời trên 5 năm khiến quãng đường di chuyển thường bị kéo dài, tiêu hao nhiên liệu kém hiệu quả và tiêu hao nhiên liệu kém hiệu quả [3]. Đồng thời, phát thải CO₂ từ vận tải đường bộ ước khoảng 30 triệu tấn/năm, tương ứng

86% tổng phát thải của ngành vận tải [4]. Những đặc điểm này hàm ý rằng cải thiện định tuyến không chỉ mang ý nghĩa giảm chi phí mà còn góp phần nâng cao hiệu quả năng lượng và giảm phát thải.

Tuy nhiên, bối cảnh triển khai tại Việt Nam cho thấy nhiều rào cản về năng lực tổ chức và công nghệ. Hơn 90% doanh nghiệp logistics thuộc nhóm vừa và nhỏ, phần lớn vẫn lập kịch bản thủ công hoặc dựa trên các hệ thống quản lý vận tải (TMS - Transportation Management System) với khả năng tối ưu hóa nâng cao còn hạn chế. Do đó, doanh nghiệp khó thích ứng trước các không xác định trong thực tế vận hành, bao gồm: tắc nghẽn giao thông giờ cao điểm khiến tốc độ di chuyển giảm xuống dưới 16 km/giờ tại các đô thị lớn như Hải Phòng [5]; biến động nhu cầu theo mùa vụ làm chi phí tăng tới 25% do mất cân bằng container [6]; và các thay đổi đột ngột về phí cầu đường hoặc giá nhiên liệu. Hệ quả là hiệu suất dịch vụ, thường được đo bằng chỉ số giao hàng đúng và đủ (OTIF), theo chuẩn quốc tế đạt khoảng 80-90% nhưng có xu hướng thấp hơn trong điều kiện Việt Nam [7].

Trong bối cảnh đó, bài toán định tuyến xe (Vehicle Routing Problem - VRP) đóng vai trò như một công cụ nền tảng nhằm đồng thời giảm chi phí vận tải, hạn chế quãng đường dư thừa, cải thiện mức độ đáp ứng dịch vụ và giảm phát thải. Tuy nhiên, các mô hình VRP truyền thống vẫn chủ yếu dựa trên giả định dữ liệu tĩnh hoặc bán tĩnh, do đó khó phản ánh đầy đủ tính biến động theo thời gian thực của mạng lưới giao thông và nhu cầu [8, 9, 10]. Trên cơ sở này, nghiên cứu phát triển một mô hình VRP đa mục tiêu kết hợp lập kịch bản, cho phép tự động tối ưu theo thời gian thực, đồng thời được thiết kế theo hướng dễ triển khai cho doanh nghiệp vừa và nhỏ. Hiệu quả của mô hình đã được kiểm chứng thông qua thử nghiệm ở một doanh nghiệp Hải Phòng, cho thấy mức cải thiện vượt trội so với cách tiếp cận truyền thống.

2. Tổng quan nghiên cứu

2.1. Cơ sở lý thuyết

Bài toán định tuyến (Vehicle Routing Problem - VRP) là một trong những bài toán tối ưu tổ hợp kinh điển và quan trọng nhất của khoa học quản lý chuỗi cung ứng và logistics. Bài toán được Dantzig và Ramser [4] giới thiệu lần đầu tiên vào năm 1959 nhằm giải quyết vấn đề lập kịch bản phân phối xăng dầu bằng xe tải. Nội dung cốt lõi yêu cầu xác định tập hợp các tuyến đường tối ưu cho một đội phương tiện xuất phát từ kho trung tâm (depot), phục vụ đầy đủ nhu cầu của toàn bộ khách hàng, sau đó quay trở lại kho, sao cho tổng chi phí vận hành (hoặc thời gian, quãng

đường, lượng phát thải CO₂) đạt giá trị nhỏ nhất đồng thời thỏa mãn các ràng buộc thực tiễn như tải trọng phương tiện, khung giờ giao nhận hàng và giới hạn quãng đường mỗi xe trong một ngày làm việc. Solomon (1987) là nghiên cứu kinh điển cho VRP có ràng buộc thời gian, nhấn mạnh vai trò của các cơ chế xây dựng và điều chỉnh tuyến để duy trì tính khả thi với ràng buộc thời gian [11]. Vì kế hoạch tuyến xe phụ thuộc chặt chẽ vào các ràng buộc thời gian, mô hình định tuyến có thể suy giảm hiệu quả khi đưa vào vận hành thực tế nếu dữ liệu đầu vào biến động.

Từ góc độ lý thuyết, một trong những cách tiếp cận phù hợp để mô hình hóa các tham số bất định là lý thuyết tập mờ (fuzzy set theory). Theo Bellman và Zadeh, fuzzy cho phép biểu diễn thông tin không chắc chắn hoặc không hoàn toàn rõ ràng dưới dạng các mức độ thuộc, thay vì buộc mọi đại lượng phải được xác định bằng một giá trị duy nhất và tuyệt đối [3]. Cách tiếp cận này đặc biệt phù hợp với các bài toán ra quyết định trong môi trường thực, nơi người phân tích thường chỉ có thể ước lượng các tham số trong một khoảng hợp lý hoặc dưới các trạng thái lạc quan, trung bình và bi quan. Trên cơ sở đó, Tanaka và Asai (1984) mở rộng tư tưởng này vào quy hoạch tuyến tính mờ, cho thấy các tham số của bài toán tối ưu có thể được mô hình hóa bằng số mờ để phản ánh tốt hơn tính bất định của dữ liệu thực tế [11]. Trong bài toán này, các yếu tố như thời gian vận chuyển, chi phí và phát thải CO₂ thường biến động do giao thông, điều kiện vận hành và tải trọng, nên khó xác định chính xác tuyệt đối. Vì vậy, mô hình fuzzy phù hợp vì nó cho phép phản ánh tính bất định gần với thực tế hơn mô hình tất định, đồng thời vẫn có thể chuyển về dạng tương đương để giải bằng các công cụ tối ưu. Trong nghiên cứu này, fuzzy được áp dụng bằng cách thay thế các tham số đầu vào có tính biến động, như chi phí, thời gian di chuyển và phát thải CO₂ trên mỗi cung (i, j), bằng số mờ tam giác thay vì coi chúng là các giá trị cố định.

Lập kịch bản (scenario planning) cũng được dùng để tránh phụ thuộc vào một dự báo duy nhất bằng cách xây dựng nhiều phương án hợp lý, rồi kiểm tra độ khả thi của phương án trên từng kịch bản [12]. Wack (1985) nhấn mạnh kịch bản không phải dự đoán mà là công cụ nhận thức giúp tổ chức nhận ra các cấu hình tương lai khác nhau để điều chỉnh quyết định hiện tại [12]. Schoemaker (1995) hệ thống hóa quy trình lập kịch bản theo logic: nhận diện không xác định then chốt, phát triển một số kịch bản nhất quán, và chạy thử hệ thống các phương án để tìm lựa chọn ít nhất khi bối cảnh thay đổi [14]. Các nghiên cứu trên cũng chỉ ra

việc lập kịch bản đã phát triển thành nhiều trường phái, nhưng điểm chung là chuyển các yếu tố bất định thành một tập tình huống có cấu trúc để hỗ trợ ra quyết định.

VRP thường được biểu diễn dưới dạng MILP (Mixed-Integer Linear Programming), trong đó các quyết định như chọn cung hoặc gắn xe cho khách được biểu diễn bằng biến nguyên hoặc nhị phân, còn hàm mục tiêu và các ràng buộc đều tuyến tính; MILP là mô hình tối ưu hỗn hợp có biến nguyên, trong đó hàm mục tiêu và các ràng buộc đều tuyến tính theo các biến quyết định, với các tham số trong mô hình được biểu diễn dưới dạng tuyến tính. Trong thực tế, các tham số như thời gian di chuyển, chi phí vận hành và phát thải biến động theo giao thông và điều kiện khai thác, nên lý thuyết tập mờ của Zadeh (1965) và khung ra quyết định trong môi trường mờ của Bellman & Zadeh (1970) cung cấp một cách mô tả phù hợp hơn đối với những ràng buộc không có giá trị đúng sai tuyệt đối mà biến động theo giao thông và điều kiện khai thác [13], [14]. Trên cơ sở đó, mô hình hỗn hợp tuyến tính mờ (FMILP) được xem là một hướng tiếp cận hiệu quả nhằm tích hợp tính bất định vào cấu trúc MILP thông qua việc biểu diễn các tham số dưới dạng mờ để xử lý. Zimmermann (1978) và Tanaka & Asai (1984) cho rằng các tham số và ràng buộc có thể được đưa trực tiếp vào mô hình rồi mờ hóa để chuẩn hoá dữ liệu MILP [11], [15].

Trong khi đó, để ứng dụng MILP vào phương pháp lập kịch bản, mô hình đòi hỏi đưa ra nhiều mức giá định khác nhau liên quan đến cấu hình trọng số chi phí-thời gian-phát thải, nên thuật toán meta-heuristics thường được áp dụng. Meta-heuristics là các thuật toán tìm kiếm thông minh để tạo ra nghiệm tối ưu, phù hợp với những bài toán có nhiều tham số bất định. Trong đó một lựa chọn tiêu biểu của thuật toán meta-heuristics là ALNS, phát triển từ LNS (Large Neighborhood Search) của Shaw (1998) và được Pisinger & Røpke (2006) mở rộng thành khung tổng quát cho nhiều biến thể VRP. [17-19]

Tuy nhiên, hạn chế của ALNS là không đảm bảo nghiệm tối ưu tuyệt đối, vì vậy bài toán thường được giải kết hợp với phương pháp khác, tiêu biểu là các phương pháp exact (các phương pháp tối ưu có thể tìm ra nghiệm tối ưu tuyệt đối nhưng thường tốn thời gian khi bài toán lớn) để tối ưu tuyệt đối các tập nghiệm. Thuật toán Branch-and-Cut là một thuật toán exact được sử dụng rộng rãi, với các công trình kinh điển của Padberg & Rinaldi (1991) và các biến thể cho CVRP (Capacitated Vehicle Routing Problem) như Lysgaard và cộng sự (2004) [9], [16].

Nghiên cứu này áp dụng kết hợp giữa thuật toán

(ALNS) để nhanh chóng tạo và cải thiện nghiệm định tuyến và mô hình FMILP giải bằng thuật toán Branch-and-Cut chính xác để tinh chỉnh và kiểm chứng chất lượng nghiệm. Các phương án định tuyến được đánh giá bổ sung thông qua phân tích kịch bản theo nhiều mức bất định và các ưu tiên khác nhau, nhằm đảm bảo lời giải đề xuất có tính linh hoạt và bền vững khi áp dụng trong điều kiện vận hành thực tế.

2.2. Khoảng trống nghiên cứu

Trên thế giới đã có nhiều công trình nghiên cứu về xử lý tình huống bất định trong bài toán định tuyến xe, song phần lớn vẫn còn những hạn chế nhất định: Nhiều nghiên cứu dùng mô hình hỗn hợp tuyến tính truyền thống (MILP) với các tham số cố định hoặc lập trình ngẫu nhiên [20], [8] chưa giải được chính xác toàn cục [21], [22]. Vì vậy, việc xây dựng và giải chính xác mô hình hỗn hợp tuyến tính mờ (FMILP) cho bài toán VRP đa mục tiêu trong các tình huống bất định trên dữ liệu thực tế vẫn còn rất mới và chưa được thực hiện một cách đồng bộ.

Qua tra cứu kỹ các cơ sở dữ liệu khoa học trong nước và quốc tế (VJOL, Google Scholar, Scopus, Web of Science) đến tháng 11 năm 2025, chưa thấy có công trình nào cùng lúc đáp ứng được bốn yếu tố sau:

- (1) Xây dựng mô hình FMILP cho bài toán VRP đa mục tiêu;
- (2) Tích hợp lập kịch bản hoàn toàn dựa trên lý thuyết mờ và tự động sinh kịch bản từ dữ liệu thực tế;
- (3) Sử dụng dữ liệu thực tế từ doanh nghiệp Việt Nam (đội xe 8 xe, trung bình 120-150 điểm giao hàng mỗi ngày tại khu vực nội và ngoại thành Hải Phòng) và giải bằng phương pháp chính xác Branch-and-Cut.

Các nghiên cứu trong nước hiện nay chủ yếu vẫn dùng dữ liệu giả lập Solomon (bộ dữ liệu chuẩn quốc tế gồm 56 trường hợp thử nghiệm với 100 khách hàng được sinh ngẫu nhiên từ năm 1987, thường dùng để kiểm tra và so sánh các thuật toán VRP có ràng buộc thời gian hoặc dữ liệu tự tạo với quy mô rất nhỏ), chưa có công trình nào triển khai thực tế tại doanh nghiệp và cũng chưa áp dụng phương pháp giải chính xác cho bài toán có yếu tố bất định và quy mô lớn.

Xuất phát từ những khoảng trống trên, nghiên cứu này thử nghiệm một hướng đi mới tại Việt Nam: Kết hợp mô hình FMILP với lập kịch bản để giải quyết bài toán định tuyến vận tải trong bối cảnh bất định, so sánh các phương án theo trọng số chi phí - thời gian - phát thải dựa trên dữ liệu thực tế doanh nghiệp logistics tại Hải Phòng. Nghiên cứu có thể góp phần giải quyết các vấn đề trong bài toán định tuyến xe trong mạng lưới chuỗi cung ứng bất định, đồng thời

mang lại một giải pháp khả thi giúp các doanh nghiệp vừa và nhỏ cải thiện hiệu quả vận hành, giảm chi phí và giảm phát thải môi trường.

3. Mô hình và phương pháp nghiên cứu

3.1. Định nghĩa biến

Chỉ số/ Tập chỉ số

k : Chỉ số của xe;

Tham số dữ liệu (đầu vào mạng lưới - chi phí - phát thải);

c_{ij} : Chi phí vận hành trên tuyến (i,j) (VNĐ);

d_{ij} : Chiều dài đoạn đường (km);

g_{ij} : Phát thải khi đi cung (i,j) (kg CO₂);

F_k : Chi phí cố định khi sử dụng xe k (VNĐ/chuyến).

Tham số nhu cầu & năng lực

q_i : Nhu cầu khách i (đơn vị hàng), $q_0=0$;

Q_k : Tải trọng tối đa của mỗi xe.

Tham số thời gian

$[a_i, b_i]$: Cửa sổ thời gian tại khách i (phút);

T_{ij} : Thời gian di chuyển tuyến (i,j) (phút);

S_i : Thời gian phục vụ tại khách i (phút), $s_0=0$;

S_j : Thời gian phục vụ tại khách j (phút), $s_0=0$.

Biến quyết định

$x_{ijk} \in \{0,1\}$: =1 nếu xe k chạy cung (i,j) ;

$v_{ik} \in \{0,1\}$: =1 nếu xe k ghé điểm i ;

$y_k \in \{0,1\}$: =1 nếu xe k được sử dụng;

t_i : thời điểm bắt đầu phục vụ tại khách i ;

t_j : thời điểm bắt đầu phục vụ tại khách j ;

Hệ số/thiết lập cho hàm mục tiêu đa mục tiêu

$\alpha, \beta, \gamma \in [0;1]$, $\alpha + \beta + \gamma = 1$: Trọng số ưu tiên các mục tiêu;

ρ (VNĐ/phút): Hệ số quy đổi thời gian ra tiền;

π (VNĐ/kgCO₂): Giá Carbon/ hệ số quy đổi phát thải CO₂ ra tiền.

3.2. Thiết lập bài toán VRP và ứng dụng FMILP

Bài toán Vehicle Routing Problem (VRP) được thiết lập nhằm xác định tập hợp các tuyến đường tối ưu cho đội xe K tất cả đều xuất phát từ một kho trung tâm (depot). Mỗi xe phải phục vụ một nhóm khách hàng trong đó mỗi khách hàng i có nhu cầu q_i . Các tuyến đường được xây dựng sao cho thỏa mãn đồng thời các ràng buộc về sức chứa, thời gian và các điều kiện vận hành khác. Hàm mục tiêu cổ điển được định nghĩa như sau:

$$\min Z = \sum_k \sum_j \sum_i c_{ij} x_{jik} \quad (1)$$

Với các ràng buộc chính như:

Đảm bảo mỗi khách hàng được phục vụ một lần:

$$\sum_k \sum_i x_{jik} = 1 \quad (2)$$

Không vượt quá tải trọng của từng xe:

$$\sum_{i \in V} q_i v_{ik} \leq Q_k, \quad (3)$$

Đảm bảo thời gian phục vụ nằm trong cửa sổ thời gian:

$$a_i \leq t_i \leq b_i, \quad (4)$$

Thời gian di chuyển hợp lệ:

$$t_j \geq t_i + T_{ij} + s_i - M(1 - x_{ijk}) \quad (5)$$

Trong đó: t_i thời gian đến nút i ; $[a_i, b_i]$: Cửa sổ thời gian, M là một hằng số rất lớn:

$$M = (\max b_i + \max T_{ij} + \max s_i) \quad (6)$$

3.3. Công thức mô hình VRP đa mục tiêu

Mô hình VRP được mở rộng thành dạng đa mục tiêu nhằm đồng thời tối ưu hóa 3 yếu tố chính: Chi phí vận hành, thời gian hoàn thành và phát thải khí nhà kính CO₂, đồng thời xem xét chi phí cố định liên quan đến việc sử dụng phương tiện. Hàm mục tiêu được định nghĩa như sau:

$$\min Z = \alpha (\sum_k \sum_{(ij)} c_{ij} x_{ijk} + F \sum_k y_k) + \beta \rho \sum_{(i,j)} T_{ij} x_{ijk} + \gamma \pi \sum_k \sum_{(ij)} g_{ij} x_{ijk} \quad (7)$$

Trong đó:

$\sum_k \sum_{(ij)} c_{ij} x_{ijk} + F \sum_k y_k$: Chi phí vận hành + mở xe;

$\sum_{(i,j)} T_{ij} x_{ijk}$: Tổng thời gian di chuyển trên tuyến;

$\sum_k \sum_{(ij)} g_{ij} x_{ijk}$: Chi phí CO₂.

Ràng buộc phát thải:

$$\sum_{k \in K} \sum_{i,j \in A} g_{ij} x_{ijk} \leq G_{max} \quad (8)$$

với G_{MAX} ngưỡng phát thải tối đa.

3.4. Quy trình vận hành chi tiết

Bước 1: Thu thập dữ liệu đầu vào. Trước tiên cần thu đầy đủ các thông tin như: Vị trí của kho trung tâm và vị trí từng điểm giao hàng; nhu cầu, số lượng hoặc khối lượng cần giao tại mỗi điểm; thời gian phục vụ; thông tin đội xe gồm số lượng xe, sức chứa, chi phí khởi động xe. Ngoài ra, cần biết khung giờ hoặc mức độ ưu tiên của từng điểm giao. Tiếp theo, sử dụng bản đồ để xác định thời gian và quãng đường di chuyển giữa từng cặp điểm. Những tuyến đường không khả thi hoặc không đáp ứng yêu cầu thời gian sẽ được loại bỏ $a_i + b_i + t_{ij} \leq b_i$.

Bước 2: Xây dựng kịch bản: Lập kịch bản (ví dụ giao thông/chi phí/phát thải); đồng thời thiết lập các bộ trọng số ưu tiên α, β, γ .

Bước 3: Sinh nghiệm nhanh bằng thuật toán ALNS: Chạy ALNS để tạo tuyến khả thi và cải thiện chất lượng; lưu nghiệm tốt nhất làm nghiệm khởi tạo.

Bước 4: Tinh chỉnh và kiểm chứng bằng FMILP: Dùng nghiệm ALNS để khởi chạy mô hình FMILP (đã chuyển về dạng xác định theo kịch bản) và giải bằng thuật toán (Branch-and-Cut) để cải thiện nghiệm.

Bước 5: Tổng hợp KPI (chi phí - thời gian - phát thải, mức vi phạm ràng buộc thời gian nếu có), lọc nghiệm không bị trội và chọn các tuyến đại diện cho từng kịch bản và ưu tiên để đề xuất.

4. Kết quả thử nghiệm

Trong nghiên cứu này, sau khi các tham số bất định trên từng cung (i, j) , gồm chi phí vận hành, thời gian di chuyển và phát thải CO_2 , được biểu diễn dưới dạng số mờ tam giác và khử mờ để thu được bộ tham số xác định tương đương, mô hình FMILP được chuyển về dạng MILP đa mục tiêu theo từng kịch bản ưu tiên. Trên cơ sở đó, nghiên cứu đề xuất một khung giải hai giai đoạn. Ở giai đoạn thứ nhất, Adaptive Large Neighborhood Search (ALNS) được sử dụng để xây dựng và cải thiện nghiệm khả thi ban đầu. Ở giai đoạn thứ hai, nghiệm tốt nhất thu được từ ALNS được dùng làm nghiệm khởi tạo cho bước tối ưu exact theo cơ chế Branch-and-Cut, dự kiến triển khai bằng Gurobi. Cách tổ chức này phù hợp với xu hướng chung trong các nghiên cứu VRP hiện nay, trong đó metaheuristic được dùng để nhanh chóng tìm nghiệm tốt, còn solver exact được dùng để tăng cường chất lượng nghiệm hoặc kiểm tra mức độ tối ưu trên các instance nhỏ và trung bình.

Quy trình giải có thể được mô tả như sau:

$$x^{ALNS} = ALNS(L)$$

$$x^* = \arg \min \{Z(x) | x \in \mathcal{X}\}, x \leftarrow x^{ALNS}$$

Trong đó J là bộ dữ liệu đầu vào, còn \mathcal{X} là tập nghiệm khả thi của mô hình MILP sau khử mờ. Như vậy, ALNS đóng vai trò tìm nghiệm tốt ban đầu trên không gian nghiệm lớn, còn pha Branch-and-Cut đảm nhiệm việc cải thiện và kiểm tra nghiệm trên mô hình xác định tương đương.

4.1. Thiết kế bộ dữ liệu

Xét tập nút $V = \{1, 2, 3, \dots, 18, 19, 20\}$ giải định, trong đó nút 1, 2 và 3 đóng vai trò điểm xuất phát đồng thời là điểm quay về, và các nút còn lại là tập khách hàng. Mỗi cung (i, j) được đặc trưng bởi bốn tham số: Quãng đường d_{ij} (km), thời gian di chuyển T_{ij} (phút), chi phí vận hành c_{ij} (VND) và chi phí mở xe F_k (VND), và phát thải g_{ij} (kg CO_2).

Một tuyến khứ hồi khả thi Z được hiểu là một chuỗi các cung liên tiếp bắt đầu tại 1, 2 hoặc 3, đi qua 20, và kết thúc tại 1, 2 hoặc 3 tương ứng. Khi đó, các chỉ số tổng hợp theo tuyến được xác định theo nguyên

tắc cộng dồn:

$$D(Z) = \sum_{(i,j) \in R} d_{ij}, \quad C(R) = \sum_{(i,j) \in R} c_{ij} + F_k,$$

$$T(Z) = \sum_{(i,j) \in R} T_{ij}, \quad G(Z) = \sum_{(i,j) \in R} g_{ij}$$

Các tham số bất định được biểu diễn bằng số mờ tam giác:

$$\tilde{c}_{ij} = (c_{ij}^L, c_{ij}^M, c_{ij}^U), \tilde{t}_{ij} = (t_{ij}^L, t_{ij}^M, t_{ij}^U), \tilde{e}_{ij} = (e_{ij}^L, e_{ij}^M, e_{ij}^U)$$

Sau đó, các tham số này được khử mờ theo giá trị trung bình:

$$\tilde{c}_{ij} = (c_{ij}^L, c_{ij}^M, c_{ij}^U) / 3,$$

$$\tilde{t}_{ij} = (t_{ij}^L, t_{ij}^M, t_{ij}^U) / 3,$$

$$\tilde{e}_{ij} = (e_{ij}^L, e_{ij}^M, e_{ij}^U) / 3$$

4.2. Thuật toán khởi tạo và ALNS

Để mô tả quy trình tính toán rõ ràng hơn, nghiên cứu tách riêng bước khởi tạo nghiệm khả thi và bước cải thiện nghiệm bằng ALNS

Input:

- Tập nút $V = \{1, 2, \dots, 20\}$;
- Tập depot ứng viên $D = \{1, 2, 3\}$;
- Nút bắt buộc phải ghé: 20;
- Nhu cầu q_i
- Sức chứa xe $Q = 14$;
- Các tham số khử mờ $c^*_{ij}, t^*_{ij}, e^*_{ij}$;
- Bộ trọng số (α, β, γ) .

Output:

- Depot được chọn d tối ưu;
- Nghiệm ban đầu S^0 .

Bước 1: Với từng depot $h \in \{1, 2, 3\}$, sinh một nghiệm sơ bộ;

Bước 2: Bắt buộc chèn nút 20 vào ít nhất một tuyến;

Bước 3: Phân bổ các khách hàng còn lại sao cho không vi phạm tải trọng;

Bước 4: Tính Z cho từng phương án xuất phát từ depot h ;

Bước 5: Chọn depot d tối ưu có Z nhỏ nhất;

Bước 6: Xuất nghiệm ban đầu S^0 tương ứng với d tối ưu.

Khi áp dụng Thuật toán 1 lên bộ dữ liệu giả định trong Bảng 1, hệ thống lần lượt xây dựng ba nghiệm khởi tạo tương ứng với ba depot ứng viên là nút 1, nút 2 và nút 3. Do cấu trúc dữ liệu đã được thiết kế theo ba vùng ưu tiên khác nhau, kết quả lựa chọn depot cho thấy sự phân hóa rõ rệt giữa các nhóm kịch bản. Cụ

Bảng 1. Hệ thống kịch bản ALNS

Nhóm	Kịch bản	α	β	γ
Chi phí	A1	0.60	0.20	0.20
	A2	0.70	0.15	0.15
	A3	0.55	0.30	0.15
	A4	0.65	0.10	0.25
Thời gian	B1	0.20	0.60	0.20
	B2	0.15	0.70	0.15
	B3	0.25	0.55	0.20
	B4	0.10	0.65	0.25
Phát thải	C1	0.20	0.20	0.60
	C2	0.15	0.15	0.70
	C3	0.25	0.20	0.55
	C4	0.10	0.25	0.65

thể, đối với các kịch bản A1-A4 thuộc nhóm ưu tiên chi phí, depot được lựa chọn là nút 1; đối với các kịch bản B1-B4 thuộc nhóm ưu tiên thời gian, depot được lựa chọn là nút 2; trong khi đó, với các kịch bản C1-C4 thuộc nhóm ưu tiên phát thải, depot được lựa chọn là nút 3. Kết quả này cho thấy ngay từ giai đoạn khởi tạo, mô hình đã phản ánh đúng logic nghiên cứu đặt ra, tức là hệ thống không mặc định trước một điểm xuất phát cố định mà chủ động xác định depot tối ưu dựa trên đặc điểm dữ liệu và định hướng ưu tiên của từng kịch bản.

Input:

- Depot đã chọn d^* ;
- Nghiệm ban đầu S^0 ;
- Bộ trọng số (α, β, γ) ;
- Số vòng lặp N_{iter} .

Output:

- Nghiệm tốt nhất S^{ALNS} ;
- Bước 1: $S_{cur} \leftarrow S^0$;
- Bước 2: $S_{best} \leftarrow S^0$;
- Bước 3: for $iter = 1$ đến N_{iter} do;
- Bước 4: Chọn toán tử phá hủy;
- Bước 5: Loại bỏ một phần khách hàng khỏi S_{cur} ;
- Bước 6: Chọn toán tử tái tạo;
- Bước 7: Chèn lại khách hàng vào các vị trí tốt hơn;
- Bước 8: Bảo đảm các nút được phục vụ;
- Bước 9: Tính $Z(S') = \alpha C(S') + \beta T(S') + \gamma E(S')$;
- Bước 10: Nếu S' tốt hơn thì cập nhật S_{best} ;
- Bước 11: Cập nhật S_{cur} theo tiêu chí chấp nhận;

Bước 12: end for;

Bước 13: Return S^{ALNS} .

4.3. Tối ưu tập nghiệm

Sau khi ALNS tạo ra các nghiệm tốt cho từng kịch bản, bước tiếp theo của nghiên cứu là đưa các nghiệm này vào pha tối ưu exact. trong nghiên cứu này, Gurobi được xem là solver thực hiện pha exact trên mô hình MILP xác định tương đương sau khử mờ. Nói cách khác, ALNS không phải là bước cuối cùng, mà chỉ đóng vai trò tạo nghiệm khả thi ban đầu và sàng lọc các phương án tốt; còn Gurobi mới là công cụ thực hiện tối ưu hóa chặt chẽ trên không gian nghiệm nguyên. Cách tách bạch vai trò giữa metaheuristic và solver exact như vậy cũng phù hợp với cách trình bày solution framework trong các nghiên cứu gần đây, nơi thuật toán heuristic được dùng để tạo nghiệm ban đầu, còn Gurobi được dùng để kiểm chứng hoặc tiếp tục cải thiện nghiệm.

Cụ thể, sau khi lựa chọn được kịch bản đại diện trong từng nhóm mục tiêu, nghiệm tốt nhất từ ALNS được chuyển thành một nghiệm khởi tạo x^0 cho mô hình MILP. Nghiệm này được đưa vào Gurobi dưới dạng MIP start. Ý nghĩa của bước này là Gurobi không cần bắt đầu từ trạng thái rỗng, mà có thể xuất phát từ một lời giải đã khá tốt, từ đó rút ngắn thời gian tìm kiếm và tăng khả năng cải thiện nghiệm cuối cùng.

Tối ưu exact được mô tả như sau:

$$x^{ALNS} = ALNS(\mathcal{J}),$$

$$x^* = \arg \min \{Z(x) \mid x \in \mathcal{X}\}, x \leftarrow x^{ALNS},$$

Trong đó \mathcal{J} là dữ liệu đầu vào của kịch bản, còn \mathcal{X} là tập nghiệm khả thi của mô hình MILP sau khi các tham số mờ đã được khử mờ.

Khi nhận nghiệm khởi tạo từ ALNS, Gurobi sẽ thực hiện tối ưu hóa theo cơ chế Branch-and-Cut. Ở mỗi nút của cây nhánh, solver trước hết giải bài toán thư giãn tuyến tính để xác định cận dưới:

$$LB_r = \min \{Z(x) \mid x \in \mathcal{X}_r^{LP}\}.$$

Nếu tại một nút, Gurobi tìm được nghiệm nguyên khả thi, nghiệm này sẽ được dùng để cập nhật cận trên:

$$UB = \min \{Z(x) \mid x \in \mathcal{X}, x \text{ là nghiệm nguyên khả thi}\}.$$

Một nút nhánh sẽ bị loại khỏi cây tìm kiếm nếu:

$$LB_r \geq UB.$$

Điều này có nghĩa là nút đó không còn khả năng tạo ra nghiệm tốt hơn nghiệm tốt nhất hiện tại.

Song song với việc chia nhánh, Gurobi còn tự động sinh các ràng buộc cắt hợp lệ để loại bỏ nghiệm phân số của bài toán thư giãn nhưng vẫn giữ lại toàn bộ nghiệm nguyên khả thi. Khi đó, miền nghiệm tại

Bảng 2. Kết quả ALNS theo từng kịch bản

Kịch bản	Depot chọn	Quãng đường (km)	Chi phí (nghìn VND)	Thời gian (phút)	Phát thải (kg CO ₂)	Giá trị mục tiêu chuẩn hóa
A1	1	712.45	8554.72	1509.82	194.48	0.7399
A2	1	716.32	8639.07	1552.88	196.27	0.7187
A3	1	691.59	8422.11	1457.10	188.18	0.7365
A4	1	789.17	9521.87	1625.55	211.66	0.7966
B1	2	665.58	9318.37	1075.48	162.47	0.6850
B2	2	690.16	9575.49	1123.72	168.98	0.7073
B3	2	749.09	10169.70	1212.63	186.64	0.7701
B4	2	757.67	10282.27	1228.22	188.44	0.7879
C1	3	791.06	12864.38	1407.45	151.34	0.7878
C2	3	874.40	13859.75	1547.24	170.17	0.8556
C3	3	794.41	12907.74	1399.48	153.23	0.8032
C4	3	810.10	13094.67	1460.44	155.02	0.7951

nút được cập nhật:

$$X_r^{LP} \leftarrow X_r^{LP} \cap C_r,$$

Trong đó C_r là tập các bất đẳng thức cắt được bổ sung tại nút đang xét. Nhờ vậy, quá trình hội tụ được tăng tốc và chất lượng nghiệm cuối cùng được cải thiện.

Input:

- Depot đã chọn d^* ;
- Nghiệm tốt nhất từ ALNS: $S^{\wedge}ALNS$;
- Mô hình MILP xác định tương đương;
- Tập ràng buộc khả thi X .

Output:

- Nghiệm cuối cùng x^* ;

Trên bộ dữ liệu giả định 20 nút của nghiên cứu, pha Gurobi không được áp dụng đồng thời cho toàn bộ 12 kịch bản, mà được sử dụng để tinh chỉnh ba phương án đại diện tốt nhất sau bước ALNS, cụ thể là kịch bản A2 trong nhóm ưu tiên chi phí, kịch bản B1 trong nhóm ưu tiên thời gian và kịch bản C1 trong nhóm ưu tiên phát thải. Chẳng hạn, với kịch bản A2, ALNS trước hết tạo ra một cấu hình tuyến mà trong đó depot được lựa chọn là nút 1. Cấu hình tuyến này sau đó được chuyển thành tập biến quyết định của mô hình MILP, bao gồm các biến nhị phân x_{ij} , biến lựa chọn depot z_h và các biến liên quan khác. Trên cơ sở đó, Gurobi tiếp nhận toàn bộ cấu trúc lời giải này như một nghiệm khởi tạo và tiếp tục kiểm tra khả năng cải thiện thêm về thứ tự phục vụ, phân bổ xe cũng như cấu trúc cung trong mạng lưới. Quy trình tương tự

cũng được thực hiện đối với kịch bản B1, trong đó depot được chọn là nút 2, và kịch bản C1, trong đó depot được chọn là nút 3. Như vậy, trong nghiên cứu này, Gurobi không thay thế ALNS mà đóng vai trò là pha tối ưu exact tiếp nối sau ALNS. Nếu ALNS chịu trách nhiệm tạo ra các nghiệm tốt trong thời gian ngắn trên nhiều kịch bản khác nhau, thì Gurobi đảm nhiệm việc tinh chỉnh và kiểm tra chất lượng tối ưu của các nghiệm đại diện đó. Cách kết hợp này không chỉ làm cho phần phương pháp giải trở nên chặt chẽ hơn, mà còn cho thấy solver được sử dụng như một thành phần thực sự nằm trong logic tối ưu hóa của mô hình, chứ không chỉ dừng ở mức nêu tên mang tính hình thức.

Để người đọc thấy rõ thuật toán hoạt động trên dữ liệu này như thế nào, Bảng 3 trình bày luôn cấu hình tuyến của ba kịch bản đại diện A2, B1 và C1.

Bảng 3. Cấu hình tuyến của ba kịch bản đại diện

Nhóm	Kịch bản	Depot	Cấu hình tuyến
Chi phí	A2	1	1-4-10-8-6-1
Thời gian	B1	2	2-10-11-15-2
Phát thải	C1	3	3-9-7-5-4-3

Từ Bảng 3 có thể thấy rõ hơn vai trò của từng depot. Trong nhóm chi phí, depot 1 giúp gom tốt cụm khách hàng phía tây và trung tây. Trong nhóm thời gian, depot 2 nằm gần vùng trung tâm hơn nên rút ngắn đáng kể thời gian vận hành. Trong nhóm phát thải, depot 3 cho phép khai thác cụm phía đông với hệ

số phát thải thấp hơn, dù phải chấp nhận thêm một xe để giảm tải cho các tuyến dài.

Bước 1: Chuyển các tuyến trong S^{ALNS} thành nghiệm khởi tạo x⁰
 Bước 2: Đưa x⁰ vào Gurobi dưới dạng MIP start
 Bước 3: Gurobi giải bài toán thư giãn tuyến tính tại nút gốc
 Bước 4: Thiết lập cận trên UB từ nghiệm khả thi hiện tại
 Bước 5: while còn nút nhánh chưa xử lý do
 Bước 6: Tính cận dưới LB_r tại nút r
 Bước 7: Nếu LB_r ≥ UB thì loại nút r
 Bước 8: Nếu nghiệm tại nút r là nghiệm nguyên khả thi thì
 Bước 9: | Cập nhật UB
 Bước 10: | else
 Bước 11: | Sinh nhánh trên biến nguyên phân số
 Bước 12: | Bổ sung các bất đẳng thức cắt hợp lệ
 Bước 13: end if
 Bước 14: end while
 Bước 15: Return nghiệm tốt nhất x*

Hình 1. Gurobi-Branch-and-Cut với nghiệm đầu vào từ ALNS

5. Kết luận

Bài báo đã xây dựng một mô hình hỗn hợp tuyến tính mờ (FMILP) tối ưu cho bài toán định tuyến, trong đó tuyến được mô tả thông qua các cung đường với các tham số thời gian, chi phí vận hành và khối lượng phát thải CO₂. Trên cơ sở biến quyết định nhị phân x_{ij} hoặc x_{ijk} trong trường hợp nhiều xe), mô hình đảm bảo hình thành các tuyến hợp lệ theo yêu cầu vận hành (xuất phát-quay về depot, phục vụ đúng khách, tuân thủ ràng buộc trọng tải Q, và cửa sổ thời gian hard nếu áp dụng). Các chỉ số tổng hợp theo tuyến được xác định theo nguyên tắc cộng dồn, cho phép lượng hoá nhất quán quãng đường, thời gian, chi phí vận hành và phát thải CO₂ trên toàn hành trình.

Bên cạnh đó, nghiên cứu đề xuất cơ chế đánh giá đa tiêu chí với ba cụm chính: Chi phí vận hành, tổng thời gian và chi phí/phát thải CO₂, qua đó làm rõ quan hệ giữa các mục tiêu. Việc lập kịch bản thông qua điều chỉnh ưu tiên hoặc các tình huống bất định giúp người ra quyết định lựa chọn phương án phù hợp với mong muốn. Trên cơ sở nghiên cứu này, các nghiên cứu tiếp theo triển khai mô hình này trên hệ thống web: Từ dữ liệu cung đầu vào, hệ thống có thể tính toán tuyến đề xuất, ước tính thời gian, chi phí vận hành và tác động phát thải, và hỗ trợ so sánh các kịch bản một cách minh bạch.

Lời cảm ơn

Nghiên cứu này được tài trợ bởi Trường Đại học Hàng hải Việt Nam trong đề tài mã số: **SV25-26.187**.

TÀI LIỆU THAM KHẢO

- [1] Adapt Ideations. (2022, September 4), *OTIF In The Supply Chain - What is it and how to calculate it*.
- [2] Andersson, J. (2020), *Ghost in a Shell: The Scenario Tool and the World Making of Royal Dutch Shell*, Business History Review, Vol.94(4), pp.729-751.
- [3] Bellman, R. E., & Zadeh, L. A. (1970), *Decision-Making in a Fuzzy Environment*. Management Science, Vol.17(4), B-141-B-164.
- [4] Dantzig, G. B., & Ramser, J. H. (1959), *The Truck Dispatching Problem*, Management Science, Vol.6(1), pp.80-91.
- [5] Erdoğan, S., & Miller-Hooks, E. (2012), *A Green Vehicle Routing Problem*, Transportation Research Part E: Logistics and Transportation Review, Vol.48(1), pp.100-114.
- [6] D+C - Development + Cooperation (2013), *Imminent gridlock*, Dandc.eu.
- [7] Karsten, C. V., Pisinger, D., & Røpke, S. (2013), *An algorithm for solving the time-constrained multicommodity flow problem with applications in liner shipping network design*, 26th European Conference on Operational Research (Abstract), Rome, Italy.
- [8] Laporte, G. (2009), *Fifty Years of Vehicle Routing*, Transportation Science, Vol.43(4), pp.408-416.
- [9] Lysgaard, J., Letchford, A. N., & Eglese, R. W. (2004), *A new branch-and-cut algorithm for the capacitated vehicle routing problem*, Mathematical Programming, Vol.100(2), pp.423-445.
- [10] Mordor Intelligence (2017), *Vietnam Road Freight Transportation Market - Size & Share*, Mordor intelligence.
- [11] Tanaka, H., & Asai, K. (1984), *Fuzzy linear programming problems with fuzzy numbers*, Fuzzy Sets and Systems, Vol.13(1), pp.1-10.
- [12] c, P. (1985), *Scenarios: Uncharted waters ahead*, Harvard Business Review, Vol.63(5), pp.72-89
- [13] Schoemaker, P. J. H. (1995), *Scenario planning: A tool for strategic thinking*, Sloan Management Review, Vol.36(2), pp.25-40.
- [14] Zadeh, L. A. (1965), *Fuzzy sets*, Information and Control, Vol.8(3), pp.338-353.
- [15] Zimmermann, H.-J. (1978), *Fuzzy programming*

- and linear programming with several objective functions, *Fuzzy Sets and Systems*, Vol.1(1), pp.45-55.
- [16] Padberg, M. W., & Rinaldi, G. (1991), *A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems*, *SIAM Review*, Vol.33(1), pp.60-100.
- [17] Shaw, P. (1998), *Using constraint programming and local search methods to solve vehicle routing problems*. In M. Maher & J.-F. Puget (Eds.), *Principles and Practice of Constraint Programming — CP98*, Lecture Notes in Computer Science, Vol.1520, Springer, pp.417-431).
- [18] Røpke, S., & Pisinger, D. (2006), *An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows*, *Transportation Science*, Vol.40(4), pp.455-472.
- [19] Pisinger, D., & Røpke, S. (2007), *A general heuristic for vehicle routing problems*, *Computers & Operations Research*, Vol.34(8), pp.2403-2435.
- [20] Lu, D., & Taghipour, S. (2025), *A scenario-based stochastic VRP framework*, *Transportation Research Part E: Logistics and Transportation Review* (in press).
- [21] Goodarzian, F., Ghasemi, P., Gunasekaran, A., Taleizadeh, A. A., & Abraham, A. (2024), *A multi-objective green vehicle routing problem with scenario planning*, *Expert Systems with Applications*, Vol.238, p. 121943.
- [22] Ivanov, D. (2022), *Ripple effect and scenario planning in supply chains*, *International Journal of Production Research*, Vol.60(1), pp.1-20.

Ngày nhận bài:	17/01/2026
Ngày nhận bản sửa:	12/04/2026
Ngày duyệt đăng:	12/04/2026