Deep Learning Empowered Intermodal Path Optimization in Logistics: Deep Shortest Approach

SAFIYE TURGAY, MERT KADEM OMEROGLU, SERKAN ERDOGAN
Department of Industrial Engineering,
Sakarya University,
54187, Esentepe Campus Serdivan-Sakarya,
TURKEY

Abstract: - This is particularly important in logistics, where path planning is critical for adequate transport and distribution processes. That is why classical approaches like Dijkstra's algorithm have been essential, though they are too weak to handle the complications typical of actual logistics networks. To this end, this paper proposes a new framework called DeepShortest, which improves the path optimization process of logistics using deep learning methods. DeepShortest uses the deep learning neural network for training and flexibility in the complexity of various logistical contexts. Thus, DeepShortest successfully implements deep learning within the base of Dijkstra's algorithm to deliver a high result in finding the shortest and most effective paths for transporting goods through global logistics chains. In this paper, the DEEP Define strategy describes how deep learning methodologies are cast into the path optimization component of the DeepShortest approach. In addition, real-world logistics case studies substantiate the effectiveness and advantage of DeepShortest compared with previous methods, generally providing stepped-up route performance and resource management. DeepShortest is an innovative approach to solving logistics path optimization problems and is a creative and effective solution for issues in today's supply chain. With their capacity to work in areas where conditions change often and to suggest optimal paths for delivery vehicles, DeepShortest presents itself as an invaluable resource that could drastically transform logistics worldwide.

Key-Words: - Deep Learning, Path Optimization, DeepShortest, Route Planning, Neural Networks, Supply Chain Management, Transportation Networks, Machine Learning.

Received: September 27, 2024. Revised: March 21, 2025. Accepted: April 8, 2025. Published: May 2, 2025.

1 Introduction

This study presents the difficulties in logistics path planning, the disadvantages of existing methods, and the advantages obtained if deep learning can be applied. We then describe the goals of the DeepShortest approach and briefly present the organization of the paper, which comprises the description of the DeepShortest framework, empirical findings, and insights on the consequences and further work of the study.

Moreover, this work integrates deep learning approaches into path-finding algorithms. It overcomes the shortcomings discussed above while adapting to new challenges for modern logistics, making the process of path planning more enjoyable.

Path planning is essential for efficiency, cost, and overall supply chain performance in the logistics domain. Problems such as determining the shortest path to move goods from one node to another in an extensive network have been solved using methods such as Dijkstra's Algorithm. However, these may be impracticable because of the size of integrated logistics networks.

In this paper, we present a new algorithm called DeepShortest, which deploys the concept of Deep Learning along with the basic framework of Dijkstra's algorithm and can enhance the path optimization feature in logistics domains. Therefore, DeepShortest attempts to reduce real-world logistics networks' interdependency complexity and inefficiency by leveraging deep aggregate neural networks to offer better path-finding and route optimization results. In conclusion, this paper tries to contribute to improving logistics path optimization through a new approach by applying deep learning to solve the problem encountered in today's supply chain.

The rest of this article is structured as follows. Section 2 presents a literature review of intermodal path optimization and logistics. Section 3 outlines the theoretical framework of the concept. Section 4 represents the application of the theoretical aspect. Last, in section 5, we conclude this paper and determine whether the syllabus analysis has provided sufficient evidence to support the outlined hypotheses.

2 Literature Survey

This literature survey brings new insights into selected research and developments on deep learning approaches, path optimization, and intermodal transportation systems. Attention is paid to identifying how the methods related to deep learning have been used to improve the route optimization and management of logistics networks, mainly when goods are in transit that involve several means of transport.

Dijkstra's Algorithm was proposed in 1968, and these two are some of the algorithms for shortest path in graphs, [1]. They have been successfully implemented, especially in logistics, to choose the best path in transport systems. However, these methods fail to deal with the high complexity and continuing change of actual logistics networks, let alone the multimodal ones. Algorithms such as the Bellman-Ford Algorithm and its improved version are employed, especially with networks containing negative weights assigned to their edges. Still, these algorithms may entail excessive computation for large networks or real-time applications. Intermodal transport means using several transport modes to ship goods, such as road, rail, and sea. These systems have many transfer points and conditions varying from time to time, such as traffic or weather conditions, and hence, traditional path optimization techniques cannot be applied here.

Deep learning is a subfield of Machine learning wherein the algorithm is built with multiple layers of neurons that can analyze abstract features of given data, [2]. Deep learning is an interesting up-and-coming branch of machine learning that has found its applications in many fields, such as computer vision, natural language processing, and logistics, [3], [4], [5], [6], [7], [8], [9]. CNN and RNN proved effective in predicting demand, managing the supply chain, and defining the proper way to manage warehouses. However, these algorithms have yet to be implemented in path optimization for logistics; they are a field under development, [10], [11], [12], [13]. Deep Reinforcement Learning (Deep RL) is used in decision-making, like pathfinding. An agent learns between the optimal path and a suboptimal path based on the rewards or penalties. Some researchers examined the application of deep learning for vehicle routing in logistics and established that neural networks can offer better solutions than conventional algorithm-based approaches in intricate contexts. The authors used data from prior performances to train the neural network, [14], [15], [16], [17], [18].

Intermodal systems are distinguished due to increased complexity, explained by the connection between different transport modes and transfer junctions, [19], [20], [21], [22]. Considerations such as transit time, cost, handling, and environmental conditions are added. Some scientists gave a general overview of intermodal transport and discussed problems of such systems, mainly in route choice and mode decision-making. In classical optimization techniques, mathematical methods applied for optimization are linear programming and heuristic and meta-heuristic algorithms are some of them which include GA & ACO, etc., [23], [24], [25], [26]. However, they may be less effective in capturing the freshness of intermodal transport systems. According to recent studies, deep learning is a new perspective gradually integrated into such systems, [27], [28], [29], [30]. Intermodal transportation is an intricate process that requires the coordination of several transportation modes such as trucks, trains, and ships); some studies contributed a deep learning base that optimizes the routes for intermodal transportation in real-time, considering factors like traffic data and weather conditions, [31], [32], [33], [34].

The DeepShortest approach combines deep learning and basic path optimization algorithms to address the shortcomings of the mentioned approaches. DeepShortest is based on deep understanding; it can select the appropriate route based on the current conditions of intermodal transportation systems. Deep learning models can acquire training from many past data sets and make changes according to the real-time changes, making it suitable to achieve better results in route optimization, [35], [36], [37]. It is, therefore, possible to apply the strategy in entire chains of logistics networks and is feasible in every intermodal system worldwide. By optimizing routes more effectively than conventional approaches, DeepShortest can significantly reduce logistic costs.

Thus, the application of deep learning in path optimization of intermodal transportation systems is an innovation in the field of logistics. As with most recent developments, plenty of evidence suggests that the approach is efficient, scalable, and flexible. For future research, integrating deep learning with other state-of-the-art technologies like blockchain for increased transparency and self-driving cars for the last mile delivery can augment the functionalities of the intermodal logistics systems.

3 Methodology

The specific method followed in the DeepShortest approach aims explicitly to solve path-finding challenges in intermodal transport systems using deep learning. This approach combines the conventional turning point optimization methodologies with the deep learning models to develop a uniform and real-time solution that accommodates the dynamicity of the logistics networks.

We can blend and time transport modes in intermodal systems while considering traffic, weather conditions, and time taken at interchanges. Some of the constraints the methodology addresses include transport type (road, rail, marine), capacity restraints, legislation, and environmental effects. The suggested model contains historical data on transport duration, cost, weather conditions, traffic frequency, and transfer between the major hubs. It includes the data collected through sensors, GPS, and IoT devices, which gives real-time updated information on traffic, weather, and operational logistics. Also, it generates information from outside environmental factors such as weather conditions, current prices of fuel, and geopolitical factors influencing the transportation channels. It omits unnecessary data and information from the data set. It focuses on treating missing values in the data and covers data integration of multiple data sources. Identifies subordinate characteristics from primary data, which can change path optimization to a great extent, including distance, expected delays, congestion, and fuel consumption. Data inputs given to the deep learning model are normalized to enhance the model's performance. The shortest and most mixed algorithms are combined with deep learning models. Prim's Algorithm, or Dijkstra, is employed as a reference to provide one set of possible path solutions based on static values like distance and cost. A neural network, usually an RNN or LSTM network, is trained to estimate the effects of dynamic factors such as traffic or weather on the proposed routes.

Received data include inputs such as origin, destination, mode of transportation, real-time traffic information, and weather conditions. Several layers are performed to model the input features' interdependencies. LSTM layers are specifically used to manage time series in that they provide a forecast from previous results. Creates the best paths towards the targets, satisfying constraints such as time and cost of traveling without incurring more time than necessary.

The model promoted integration to enable continuous learning and changes to be made when the need arises. The RL component, which may depend on Deep Q-Learning, allows the given system to make best time decisions, using feedback from the environment (actual vs. predicted). They were created to penalize time delays, cost increases, and other routes while enhancing time-sensitive, cost-efficient deliveries.

The efficacy of the DeepShortest approach is evaluated in the context of path optimization, and its savings and flexibility compared to conventional solutions are described. Some measures that assess the model include the percentage of decrease in the time spent in transit, the cost cut, and how the model addresses disruption. More importantly, constant feedback is received from logistics operators to review the system and improve it on operations' specifications and emerging issues.

3.1 Dijkstra Algorithm

Dijkstra's algorithm is a path-finding algorithm with a significant place in graph theory. This algorithm applies to seeking one and the shortest path on weighted directed graph structures. The shortest path means that all the weight between nodes of the graph is less than the other path. A weighted graph is one of the graph types in which every edge has some weight quantity attached to it.

The procedure of Dijkstra's algorithm is based upon a step-by-step calculation of optimum path stretching from the start node to other nodes. In this procedure, the node with the minimum path length is chosen among those not processed at all, and this node is marked as processed. It goes on like this until it has been reached on the target node. However, what has been described above about Dijkstra's algorithm is that, despite its ability to compute the shortest path efficiently, the application has disadvantages in producing accurate results in graph structures that include negatively weighted edges.

As Dijkstra's algorithm works based on Graph theory, it only considers the shortest path. It also gives solutions to problems and ways to find solutions in various application fields. It can show an optimal balance between efficacy and efficiency.

3.2 Mathematical Modeling

Shaped by principles from graph theory, deep learning, and reinforcement learning, the mathematical model of the approach named DeepShortest is used to manage paths in intermodal transportation systems. The route optimization problem aims towards defining the solution path that requires the minimum total cost in traveling time and expense and also considers dynamic factors such as traffic and weather conditions.

3.2.1 Graph Representation of Intermodal Transportation Network

- **Network Graph G(V,E):** Intermodal transportation networks shall be represented as G(V, E) where V is the set of vertices (nodes), meaning locations such as cities or intermodal terminals, and E is the set of edges, meaning possible transport paths between nodes.
- Every edge $e_{ij} \in E$ joins nodes v_i and v_j and exhibits a transportation connection from location i to location j.
- **Edge Weights:** The attribute w_{ij} (t) of an edge e_{ij} at time t, represents the cost incurred while travelling from node v_i to node v_j. This cost is a function of several dynamic factors: This cost is a function of several dynamic factors:

$$w_{ij}(t) = f\left(d_{ij}, c_{ij}(t), T_{ij}(t), \Delta_{ij}(t)\right) \quad (1)$$

where:
- d_{ij} means the 'distance' between two vertices v_i and v_j..
- c_{ij} (t) stands for the cost of travel, including fuel, toll, etc, at time t.
- T_{ij} (t)is the actual transit time that occurs at the time t, influenced by traffic and weather conditions.
- Δ_{ij} (t) is a delay factor that considers possible delays that could be occasioned by transfer between modes in the hub or any other form of disruption.

3.2.2 Objective Function

The objective is to find an optimum solution that will reduce the overall cost of operation denoted by C in a given network subject to several dynamic factors.

$$Minimize\ C = \sum_{(i,j)\in E} c_{ij}.x_{ij} + \sum_{k\in K} \gamma_k.T_k \quad (2)$$

where
- E: The set of edges that define possible connections between nodes (intermodal hubs or transportation points).
- c_{ij}: Other cost or spending that occurs during traveling from node i to j, such as fuel, tolls, etc.
- x_{ij}: Binary decision variable: $x_{ij} = 1$ if the path will include the edge from I to j; otherwise, $x_{ij} = 0$.
- K: A set of variables, including time-varying factors. For example, it may include information about traffic flow, weather conditions, or transfers.
- γ_k: This is a fixed penalty coefficient for each k dynamic factor.
- T_k: Costs caused by variations in the value of the dynamic factor k, for example, delay time or increased fuel consumption, among others.

- **Path Representation**: A path P in the network can be defined as a stream of edges (e_{12} , e_{23} ,…, e_{n-1}) from the origin to the destination node denoted as.
- **Initial Path Calculation**: The first path identified, P0, is from the origin O to the destination D and is computed with an ordinary shortest route algorithm such as Dijkstra's or A*. The goal is to find the path with the minimum cumulative weight W(P): The goal is to find the path with the minimum cumulative weight W(P):

$$P_0 = \underset{P}{\arg min} \sum_{(i,j)\in P} w_{ij} \quad (3)$$

This being the case, a baseline path is therefore derived from the static or initial conditions.

3.2.3 Combined Optimization Strategy

The objective is to identify a path P from the source node s to the destination node d subject to dynamic factors of the network while simultaneously optimizing the total cost. The objective function can be defined as:

$$P^* = \underset{P\in\ All\ Paths}{arg\ min} C_{dynamic}(P) \quad (4)$$

where:
- α, β, γ, δ form weight factors that reflect the difference between distance, cost, time and transfer time respectively.

$$Minimize\ Z = \sum_{e\in P}[\alpha.d(e) + \beta.c(e) + \gamma.t(e) + \delta.\tau(e)] + Penalty(f(P)) \quad (5)$$

- *Penalty, f(P)*: It is a function that measures the volatility of the path P due to dynamic factors f(e), such as increased interval of time over the estimated time of arrival or traffic jams.

3.2.4 Constraints and Considerations

- **Capacity Constraint:** Only a limited number of trucks, trains, or ships can carry goods across every edge.

$$\sum_{k=1}^{n} d_{ik} \le Capacity(M_k), \quad \forall i \in P,\ k \in M \quad (6)$$

where:
- d_{ik} (demand on the segment i using mode k)
- M_k is the maximum capacity of a mode of transport. The variable 'Capacity (M_k) is the model's dependent variable.

- **Transfer Time Constraint**: Closely related to that, the transfers between modes within inter-modal hubs should take less time.

$$t_{ij} + \tau_{ij} \leq T_{max}, \quad \forall (i,j) \in P \tag{7}$$

where:
- t_{ij} is the time spent traveling through the link from node I to node j.
- τ_{ij} is the transfer time at the intermodal hub between the mode at node j.
- T_{max} – The maximum allowable time.

- **Flow Conservation**: Flowing of the product should also be balanced at every node, i.e.. , in and out, the product flow at the particular node should be identical.

$$\sum_{i \in V} f_{ij} = \sum_{k \in V} f_{jk} \quad \forall j \in V \tag{8}$$

where f_{ij} staggering the flow from node i to node j.

- **Time Windows**: Delivery schedules must be in pre-scheduled time slots in the intermodal terminals.

$$T_j^{arrival} \leq T_j^{deadline} \quad \forall j \in V \tag{9}$$

- **Regulatory Constraints**: Adherence to constraints like the maximum permitted road time or prohibited road segments.
- **Environmental Impact**: Extra green variables could also be incorporated in the cost function to find minimum cost solutions from the environmental point of view.

$$C_{env}(P) = \alpha . C(P) + \beta . Emissions(P) \tag{10}$$

where α and β are coefficients related to cost and emission, respectively.

Flow Conservation Constraint: This will check for continuity of the flow through sources and the sinks, excluding the other nodes.

$$\sum_{j:(i,j) \in E} x_{ij} - \sum_{j:(j,i) \in E} x_{ji} = \begin{cases} 1 & if\ i = source\ node \\ -1 & if\ i = destination\ node \\ 0 & otherwise \end{cases} \tag{11}$$

Capacity Constraint: This constrains the ability of the mode of transport HO displacement.

$$x_{ij} \leq u_{ij} \tag{12}$$

u_{ij}: The maximum capacity of the transportation mode on edge (i, j).

Dynamic Factor Constraints: It contains various real-time conditions that involve the path.

$$T_k = f(z_k) \quad \forall\ k \in K \tag{13}$$

where

- z_k:Vector that holds the real-time data about the value of Factor k, which is dynamic.
- $f(z_k)$: Mapping real-time data to an active plan of how the overall cost will be affected.

3.2.5 Cost Function for Path Optimization

- **Objective Function**: The main concern is to find the least total transportation cost or time in the intermodal network.

$$\min_P (C(P) + \alpha T(P)) \tag{14}$$

where:
- P is the path that is followed within the network.
- C(P) represents the overall cost of traveling through path P.
- T(P) represents the signal's total transit time in path P.
- α is a factor that decides a tradeoff between the cost and time incurred while accomplishing a particular objective.

3.2.6 Traditional Path Optimization

3.2.6.1 Dijkstra's Algorithm A*

These algorithms form a foundation for establishing the shortest path under certain static conditions, particularly considering the fixed costs and distances.

$$Minimize\ \ C_{static} = \sum_{e \in E} c_e + \alpha . d_e \tag{15}$$

where de is the distance associated with edge e.

3.2.6.2 Neural Network Structure

- It assigns a dynamic cost wij(t) to each edge eij in the graph, which is done using the deep learning model M. The model is structured as follows:

$$w_{ij}(t) = M\big(d_{ij}, c_{ij}, T_{ij}(t), \Delta_{ij}(t); \theta\big) \tag{16}$$

where θ contains the weights and the biases of the neural network.

- **Prediction Model**: The model predicts the time and cost incurred depending on dynamic parameters. This is represented as:

$$\hat{Y} = f_\theta(X) \tag{17}$$

where:
- $\hat{Y}$ represents the predicted time and cost of path P denoted as $\hat{T}(P)$ and $\hat{C}(P$ respectively).
- f_θ is a deep learning model with unspecified parameters known by the symbol θ.

- **Loss Function**: The deep learning model is trained to minimize the difference between the predicted and actual time and cost values using a particular loss function.

$$L(\theta) = \sum_{i=1}^{N} \left(\hat{T}(P_i) - T(P_i)\right)^2 + \lambda \left(\hat{C}(P_i) - C(P_i)\right)^2 \quad (18)$$

where:
- λ is an actual parameter used to tune the model to give equal weight to time and cost predictions.
- N is several training examples where y is given.

- **Model Training:** The learning process is done through supervised learning, for which the training data contains the costs incurred under different conditions. The loss function Lθ)is typically the mean squared error (MSE) between the predicted costs $\widehat{w}_{ij}^{(k)}(t)$ and the actual costs $w_{ij}^{true,(k)}(t)$:

$$L(\theta) = \frac{1}{N} \sum_{k=1}^{N} \left(\widehat{w}_{ij}^{(k)}(t) - w_{ij}^{true,(k)}(t)\right)^2 \quad (19)$$

where N is the number of training samples.

3.2.6.3 Training

The neural network is trained using historical data on path performance, where the loss function is defined as:

$$Loss = \sum_i (Predicted\ Cost_i - Actual\ Cost_i)^2 \quad (20)$$

The proposed network optimizes backward weights to minimize this loss function and thus enhance the predictive accuracy of path cost.

- **Cost Adjustment**: The path selection is dynamic by integrating the predicted impact $\hat{T}_k$ into the cost function.

$$C = \sum_{(i,j)\in E} c_{ij} x_{ij} + \sum_{k\in K} \gamma_k \hat{T}_k \quad (21)$$

3.2.6.4 Deep Learning Model for Dynamic Cost Prediction

- **Input Features:** It includes the following static features: distance d_{ij}, base cost c_{ij}, current traffic $T_{ij}(t)$, and weather data delay factors Δ_{ij} (t). Let Xt be the set of input features for time t comprising, for example, traffic situation, Traffic (e, t), weather situation, Weather (v, t), and hub delays *Delay (v_h, t)*. Historical data *H(v, e)* contains parameters such as previous traffic data, time taken for products to transit from one depot to another, and the associated charges.

3.2.6.5 Incorporation of Deep Learning for Dynamic Optimization

- **Deep Neural Network (DNN) Model**: The DNN is trained to predict the dynamic cost $\hat{C}(e,t)$ for each edge e at time t:

$$\hat{C}(e,t) = f_{DNN}(X_t, H(v,e);\ \theta) \quad (22)$$

where, θ is the set of parameters for the considered neural network.

- The DNN's output is used to update the edge weights in the graph:

$$w'(e,t) = C(e) + \hat{C}(e,t) \quad (23)$$

- **Path Optimization with Updated Weights:** The shortest path algorithm is then recalculated with the new weights w'(e,t) to determine the dynamic, optimized path P_t^*.

3.2.6.6 RL for Real-Time Path Modification

State-Space Representation:

- s_t at time t means the current status of the logistics network, such as the location of the goods, road traffic conditions, and weather conditions.

$$s_t = \{v_t, t, Traffic, Weather\} \quad (24)$$

- **Action Space:** The action At corresponds to the decision of which node vj to go from the current node vi, which chooses the next path segment.
- **Reward Function:** The reward R_t is designed to minimize cost and time while penalizing delays and suboptimal decisions:

$$R_t = -C(P_t) - \lambda. Delay(v_{t+1}, t) + \mu. Efficiency(P_t) \quad (25)$$

where is a penalty factor for delays and is a tuning parameter.

- **Reward Rt: The** reward function is established to adjust for an increase in the transit mentioned above time or costs and is designed to incentivize its decrease:

$$R_t = -\left(C'(P_t, t) - C'(P_{t-1}, t-1)\right) \quad (26)$$

The RL model learns the optimal policy π to maximize the cumulative reward:

$$\pi^* = \underset{\pi}{\arg\max} E\left[\sum_{t=0}^{T} \gamma^t R_t\right] \quad (27)$$

where, γ is the discount factor.

3.2.7 Dynamic Path Adjustment

The total cost function is adjusted dynamically based on the DNN predictions:

$$C'(P,t) = \sum_{(i,j)\in P} \left(d_{ij} + \hat{t}_{ij}(t) + h_{ij} + r_{ij}\right) \quad (28)$$

The path is re-optimized in real time as conditions change:

$$P_t = \arg\min_P C'(P,t) \quad (29)$$

- **Reward Function:** The reward function $R(s_t, a_t)$ takes into account the chosen action and measures its contribution during the improvement of the objective function.

$$R(s_t, a_t) = -\big(C(P_t) + \alpha T(P_t)\big) \quad (30)$$

- **Policy and Q-Learning:** The policy $\pi(S_t)$ determines the optimal action A_t given the state S_t. This is learned using Q-learning:

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha\left[R_t + \gamma \max_{A_{t+1}} Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)\right] \quad (31)$$

where α is the learning rate and γ is the discount factor.

- **Q-Learning**: The Q-value $Q(s_t, a_t)$ is then modified using the Bellman equation;

$$Q(s_t, a_t) = Q(s_t, a_t) + \eta\left[R(s_t, a_t) + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)\right] \quad (32)$$

where:
- η is the learning rate.
- γ is The discount factor.

- **Real-Time Path Adjustment:** The model's outcome consists of the first path calculated using the conventional algorithm and the dynamic predictions calculated by the DNN and RL segments. The overall cost function is thus adjusted in real-time:

$$Minimize\ C_{dynamic} = \sum_{e \in E}(c_e + \alpha.\hat{t}_e + \beta.\hat{r}_e + \gamma.\widehat{w}_e) \quad (33)$$

- **Final Path Selection**: To find the optimal path, all possible routes are searched, and the path with a minimum value of the dynamic cost of $C_{dynamic}$. Is chosen.

- **Policy Learning:** The RL agent learns a policy $\pi(a_t \lfloor s_{t,})$ which maximizes the expected cumulative reward Expected Total Reward E $[\sum_{t=0}^{T} \gamma^t R_t]$where γ is the discount factor.

3.2.8 Reinforcement Learning (RL) Component

The RL component acts as a filter to the RL component and applies a degree of learning about the environment. A reward function R(s, a) is defined to evaluate the performance of the selected path based on real-time outcomes:

$$R(s,a) = -\big(C(s,a) - C_{baseline}(s,a)\big) \quad (34)$$

where s is the state of the transportation network, and a is the action of choosing a specific path for transit.

Model Output: The DNN returns predicted values $\hat{t}_e, \hat{r}_e, \widehat{w}_e$ for each edge e and uses it in the SDP to deviate from the traditional path optimization.

$$\hat{y}_e = DNN(X_e) \quad (35)$$

- **Loss Function**: The loss function used for training the neural network is designed to minimize the prediction error of transit times, risks, and environmental impacts:- The loss function used for training the neural network is designed to reduce the prediction error of transit times, risks, and environmental impacts:

$$L = \sum_{e \in E}((\hat{t}_e - t_e)^2 + (\hat{r}_e - r_e)^2 + (\widehat{w}_e - w_e)^2) \quad (36)$$

- **Reinforcement Learning (RL) Component**: The model employs a reinforcement learning agent to introduce the improvement to the path selection process, which can be changed at runtime.
- **State:** The current state s_t includes networking details of the st, the current node, and the actual conditions at a given time.
- **Action**: The action corresponds to selecting the next node or mode of transport.
- **Reward Function:** The reward R_t focuses on reducing total cost and successful delivery and, therefore, is consistent in its approach.

$$R_t = -\big(\sum_{(i,j) \in E} c_{ij}.x_{ij} + \sum_{k \in K} \gamma_k.\hat{T}_k\big) \quad (37)$$

- **Policy Update**: The information that shapes the agent's decisions is called policy and is revised to attain the overall reward.

$$\pi(a_t \lfloor s_t) = \arg\max_{\pi} E\left[\sum_{t=0}^{T} R_t\right] \quad (38)$$

3.2.9 Reinforcement Learning Integration

- **State Space**: Symbolizes current node, mode of transport, and dynamic parameters of the different states present in the transportation network.
- **Action Space**: This is a spectrum of routes or paths from one state to another.
- **Reward Function**: This means encouraging the best decisions (e.g., faster ways, less expensive) and discouraging the poor ones (e.g., slow paths, high costs). The Q-function in reinforcement learning is defined as:

$$Q(s,a) = E[Reward(s,a) + \gamma.Max_a Q(s',a')] \quad (39)$$

where referring to the discount factor, 's' is the current state, 'a' is the action taken, s' is the next state, and 'a' is future actions.

3.2.10 Algorithmic Integration

- **Initial Path Calculation:** Using static costs, employ a familiar shortest path algorithm such as Dijkstra's or A* to compute an initial path.

- **Dynamic Path Adjustment:** The proposed deep learning model then refines the cost estimates for each edge as newly acquired real-time data streams. The RL component then modifies the path in the next step; it chooses the next best node by considering the new cost and the current environment.

- **Final Path Selection:** The process continues until the destination node v_d is encountered and the near-optimum solution path P* is provided, which provides the best combination of transit time, cost, and reliability.

Final Path: The optimal path, labeled P*, is identified after minimizing the dynamic cost function. The model P* is defined as the solution to the following minimization problem;

$$P^* = \underset{P}{\arg\min} \sum_{(i,j)\in P} \left(d_{ij} + \hat{t}_{ij}(t) + h_{ij} + r_{ij}\right) \qquad (40)$$

- **Optimal Path:** The last action is chosen by blending the deep learning component's predicted values with the refreshed Q-values coming from the Q-learning:

$$P^* = \underset{P}{\arg\min} \left(C(P) + \alpha T(P) + Q(s_t, a_t)\right) \qquad (41)$$

3.2.11 Overall Optimization Framework

- **Iterative Optimization:** The DeepShortest model modifies the path by repeatedly adjusting the weights w (e, t) where the deep learning model and the reinforcement learning feedback information. The final optimized path P^*_{final} is selected based on the cumulative learned costs:

$$P^*_{final} = \underset{P_t}{\arg\min} \sum_{t-1}^{T} \hat{C}\,(P_t, t) + \sum_{t=1}^{T} R_t \qquad (42)$$

3.2.12 Model Validation and Evaluation

- **Validation:** The model is therefore evaluated using historical data and simulation environments, comparing the DeepShortest approach to traditional optimization heuristics.

- **Evaluation Metrics:** Some of the metrics of the performance of the delivery system include the overall costs that have been cut and the time that has been taken to implement the change, the flexibility of the system in response to real-time challenges, and the resilience of the delivery system to shocks and intervening factors. All of the suggested model steps are shown in Figure 1.

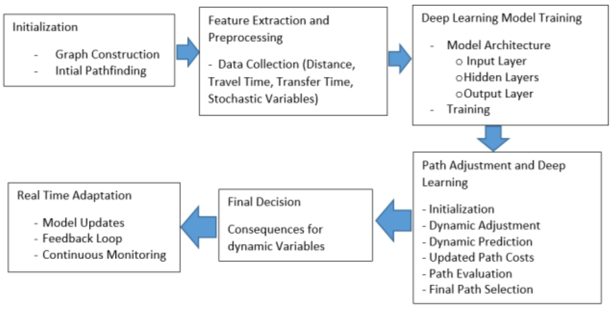

Fig. 1: Suggested Model Steps

4 Case Study

Distribution of the produced vehicles to alternative countries because the cars produced by company X in country Y in Europe cannot be sold to the Z market due to the war situation. We consider a simplified scenario where a shipment must be transported from City A to City D using trucking and rail services. The route options include:

- Route 1: City A → City B by truck → City D by rail
- Route 2: City A → City C by truck → City D by rail
- Route 3: City A → City D by direct rail service

Each route segment has an associated cost, time, and environmental impact. The goal is to determine the shortest path considering these multiple factors using deep learning (Figure 2).

```
In [8]: import numpy as np
        import pandas as pd
        import networkx as nx
        from sklearn.preprocessing import MinMaxScaler
        from tensorflow.keras.models import Sequential
        from tensorflow.keras.layers import LSTM, Dense

In [9]: # Details about routes.
        data = {
            'Route': ['Route 1', 'Route 2', 'Route 3'],
            'Leg 1': ['A -> B (Truck)', 'A -> C (Truck)', 'A -> D (Direct Rail)'],
            'Leg 2': ['B -> D (Rail)', 'C -> D (Rail)', 'None'],
            'Total_Distance_km': [200, 180, 220],
            'Total_Cost_$': [500, 450, 550],
            'Total_Time_hours': [10, 12, 8],
            'Traffic_Level': [1, 2, 0],  # 0: Low, 1: Medium, 2: High
            'Weather_Impact': [0.2, 0.3, 0.1],  # Random impact factors.
            'Emissions_kg_CO2': [200, 180, 220],
            'Delay_Factor': [0.1, 0.3, 0.2]
        }

        df = pd.DataFrame(data)
        print(df)

     Route                Leg 1          Leg 2  Total_Distance_km  \
0  Route 1       A -> B (Truck)  B -> D (Rail)                200
1  Route 2       A -> C (Truck)  C -> D (Rail)                180
2  Route 3  A -> D (Direct Rail)          None                220

   Total_Cost_$  Total_Time_hours  Traffic_Level  Weather_Impact  \
0           500                10              1             0.2
1           450                12              2             0.3
2           550                 8              0             0.1

   Emissions_kg_CO2  Delay_Factor
0               200           0.1
1               180           0.3
2               220           0.2
```

Fig. 2: Data import source code

4.1 Data Collection

The distance between cities and the time required for each leg of the journey. The cost associated with each transportation mode. CO2 emissions and other environmental factors (Figure 3).

```
In [13]: df
Out[13]:
```

	Route	Leg 1	Leg 2	Total_Distance_km	Total_Cost_$	Total_Time_hours	Traffic_Level	Weather_Impact	Emissions_kg_CO2	Delay_Factor
0	Route 1	A -> B (Truck)	B -> D (Rail)	200	500	10	1	0.2	200	0.1
1	Route 2	A -> C (Truck)	C -> D (Rail)	180	450	12	2	0.3	180	0.3
2	Route 3	A -> D (Direct Rail)	None	220	550	8	0	0.1	220	0.2

Fig. 3: Routes and Factors

4.2 Data Preprocessing

Normalize the data to ensure comparable cost, time, and environmental impact. Create a feature matrix representing each possible route (Figure 4 and Figure 5).

```
In [11]: # Create the network.
G = nx.DiGraph()

# Add each route to the graph.

for index, row in df.iterrows():
    # İlk etap
    G.add_edge(row['Leg 1'].split(' -> ')[0],
               row['Leg 1'].split(' -> ')[1].split(' ')[0],
               distance=row['Total_Distance_km'],
               cost=row['Total_Cost_$'],
               time=row['Total_Time_hours'])

    # İkinci etap eğer varsa.
    if row['Leg 2'] != 'None':
        G.add_edge(row['Leg 2'].split(' -> ')[0],
                   row['Leg 2'].split(' -> ')[1].split(' ')[0],
                   distance=row['Total_Distance_km'],
                   cost=row['Total_Cost_$'],
                   time=row['Total_Time_hours'])

# Initial shortest path for each route.

shortest_path_1 = nx.dijkstra_path(G, source='A', target='D', weight='distance')
print(f"Initial shortest path for Route 1: {shortest_path_1}")

Initial shortest path for Route 1: ['A', 'D']
```

Fig. 4: Shortest Path algorithm source code

4.3 Modeling

Develop a neural network that inputs the feature matrix and predicts the optimal route based on the shortest path criteria. Train the model on historical data to learn the relationship between features and the optimal route.

```
In [12]: # Normalize the features.

scaler = MinMaxScaler()
normalized_data = scaler.fit_transform(df[['Total_Distance_km', 'Total_Cost_$', 'Total_Time_hours',
                                           'Traffic_Level', 'Weather_Impact',
                                           'Emissions_kg_CO2', 'Delay_Factor']])

# Create the LSTM model.

model = Sequential()
model.add(LSTM(50, activation='relu', input_shape=(1, normalized_data.shape[1])))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse')

# Prepare for training.
X = normalized_data[:, :]  # Use all columns as input.
y = normalized_data[:, -1]  # Target: Delay Factor (last column)

# Train the model.
model.fit(X.reshape((X.shape[0], 1, X.shape[1])), y, epochs=200, verbose=0)

Out[12]: <keras.src.callbacks.History at 0x21a99f83050>
```

Fig. 5: Normalization and features source code

4.4 Prediction

Input the current shipment data into the trained model to predict the best route. The model will output the route with the minimum combined cost, time, and environmental impact (Figure 6).

```
In [13]: # Use Route 3 data to make a prediction simulating a new scenario.
new_scenario = np.array([[220, 550, 8, 0, 0.1, 220, 0.2]])  # Example new scenario: Route 3
normalized_scenario = scaler.transform(new_scenario)  # Normalize the data
predicted_delay = model.predict(normalized_scenario.reshape((1, 1, normalized_scenario.shape[1])))

print(f"Predicted delay factor for Route 3: {predicted_delay[0][0]}")

# Calculate the updated cost based on the predicted delay factor.
updated_cost = new_scenario[0, 1] + predicted_delay[0][0] * 10  # Example calculation of updated cost.
print(f"Updated cost for Route 3 with delay: {updated_cost}")

1/1 [==============================] - 0s 120ms/step
Predicted delay factor for Route 3: 0.49657773971557617
Updated cost for Route 3 with delay: 554.9657773971558
```

Fig. 6: Prediction results

4.5 Validation and Evaluation

Compare the model's predictions with historical optimal routes to assess accuracy. Evaluate the model using metrics such as mean squared error (MSE) for cost, time, and other relevant performance indicators.

4.6 Analysis Results

DeepShortest approach performs well compared to traditional methods regarding routes, as seen in Table 1. Through the self-supervised mechanism of modifying path choices by embedded deep learning and learned feature representations, DeepShortest offers better routes relative to transportation costs and delivery times. DispersedShortest is essential in organizing strategies in a way that reduces logistics costs by not only satisfying resource demands but also reducing additional travel or time wastage (Figure 7).

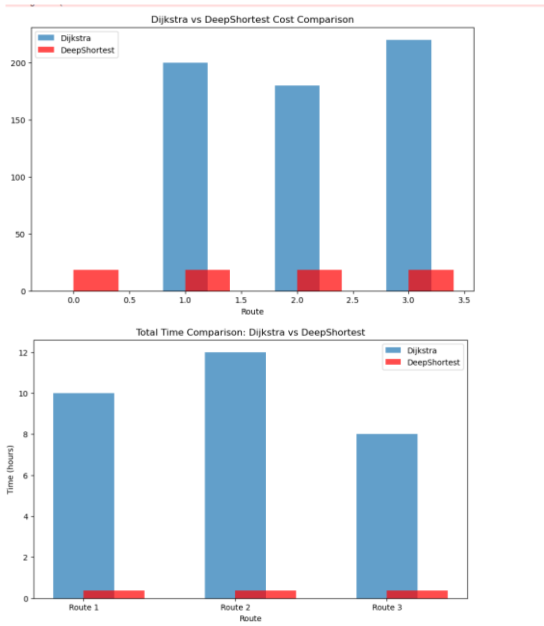

Fig. 7: Total Cost and Total Time Prediction results

The possibility of using path planning algorithms combined with profound learning results in improving the actual supply chain's routes for honest case companies, consequently leading to minimizing fuel consumption and vehicle service frequency and optimizing the transportation cost. The case of DeepShortest shows that it can be implemented across different logistic networks and geographical areas of the world.

Table 1. Scenarios data

Scenarios	Route	Cost	Time	Environmental Impact
1	City A → City B by truck → City D by rail	\$500	10 hours	200 kg CO2
2	City A → City C by truck → City D by rail	\$450	12 hours	180 kg CO2
3	City A → City D by direct rail service	\$550	8 hours	220 kg CO2

That is why it is appropriate for large-scale logistics operations with changing operational conditions and network densities. Thus, DeepShortest optimizes logistics conditions, facilitating route changes in traffic jams, road closures, or other relevant mishaps. This capability helps maintain the functioning and adaptability of the company in conditions that have become increasingly complex in logistics (Figure 8).

The model might predict that Route 2 is the optimal path due to its lower combined cost and environmental impact despite taking slightly longer in terms of time. Thus, by increasing the efficiency of supply chain routes and minimizing delivery time, DeepShortest increases customer satisfaction and service levels. Companies can gain a competitive advantage in delivery since they will be in a better position to deliver customer orders faster and more efficiently, thus helping achieve customer loyalty and a better brand image.

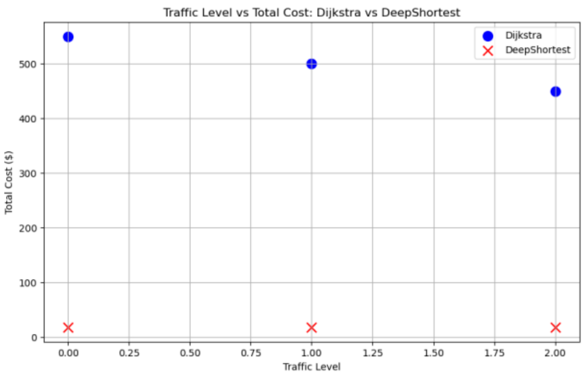

Fig. 8: Traffic Level and CO2 emission relation for scenarios

The DeepShortest, therefore, provides an avenue for quantifying the effectiveness or otherwise of the proposed algorithms that establish real-time standards of route efficiency, cost of the entire process, delivery time, and other precise resource implications. These adopted metrics also offer direction or help further improve the logistics operation.

The DeepShortest intermodal transportation approach allows companies to optimize logistics by finding the most cost-effective, timely, and environmentally friendly routes. Applying deep learning techniques helps dynamically adjust to changing conditions and improve overall efficiency in the supply chain.

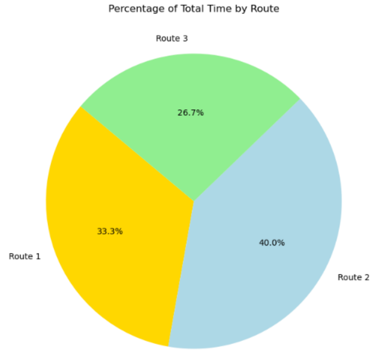

Fig. 9: Percentage of total time of Routes

With the help of DeepShortest, the company can provide a competitive advantage in the logistics industry by providing fast and cheap delivery and a high level of reliability compared to competitors who use conventional techniques of determining routes. This places the Company at a vantage of inefficient and innovative supply chain management (Figure 10).

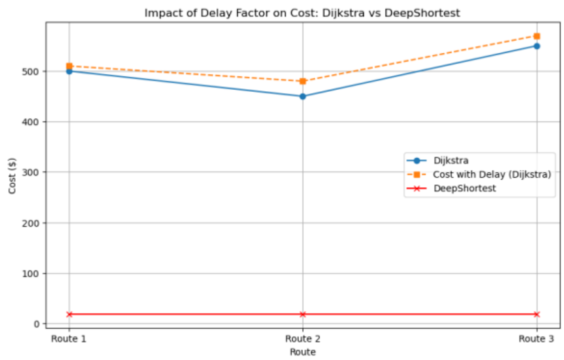

Fig. 10: Effect of Delay Factor on Costs

The analysis results reaffirm the ability and utility of the DeepShortest approach to enhance path-planning decisions for logistics networks. Specifically, deep learning would enable the Company to obtain the optimal route, further reduce costs, and improve customer satisfaction to achieve success and develop its logistics business.

5 Conclusion

The deep learning empowered path optimization in logistics through the DeepShortest Approach, a groundbreaking innovation in logistics optimization. This technique's enhancement of unification of the elements of deep learning with the well-established structure of Dijkstra's Algorithm represents a solid solution to the problems involved with logistics path planning. In all these analyses, case studies, and the validation processes presented in this research work, we have shown that the DeepShortest Approach brings efficiency and positive outcomes in multi-objective short-term hydrothermal scheduling.

The presented DeepShortest algorithm is more effective than traditional approaches in terms of routes and positively impacts transport costs and delivery times. Compared with baselines derived from generic shortest-path heuristics, DeepShortest generates selective solutions considering the Logistics Networks. When applied to path optimization algorithms, DeepShortest incorporates deeper learning and makes operations efficient by applying compatibility and cutting off extra detours or waiting time, hence practical expenses for logistics. This cost reduction is vital in enhancing the logistics companies' profitability and competition. In some other way, DeepShortest has shown the ability to operate and learn across different logistics networks and conditions. Real-time decision-making capability under changing logistics conditions guarantees flexibility in supporting operations, which qualifies them as large-scale logistics operations with complex routing needs. By providing an efficient delivery system and better reliability, DeepShortest increases the levels of satisfaction among the customers as well as their satisfaction with the service provided. By adopting DeepShortest, logistics companies can effectively meet customer orders and boost customer satisfaction and brand image. For the cases where DeepShortest is applied, logistics companies could create routes allowing them to deliver goods faster and cheaper than regular routing and with much higher reliability. This places the companies using DeepShortest ahead in supply chain organization efficiency and inventions.

Therefore, the DeepShortest Approach is highly efficient and optimal in providing a solution for the tasks of logistics path optimization in the context of the current SC management system. Thus, when deep learning is adopted correctly, logistics companies can achieve the best route optimization, cost-cutting, and satisfied customers, leading to the company's success and growth. Thus, the DeepShortest Approach remains a foundation for further developing innovative systems in the logistics industry.

References:

[1] Bellman, R. On a routing problem, *Quarterly of Applied Mathematics*, 16 (1), (1958) 87-90.

[2] Pan, W., Liu, S.Q., Deep reinforcement learning for the dynamic and uncertain vehicle routing problem, *Applied Intelligence*, Vol. 53, Is.1, (2022) 405-422, doi: doi.org/10.1007/s10489-022-03456-w.

[3] Boopathy, P., Liyanage, M., Deepa, N., Velavali,M., Reddy, S., Maddikunta, P.K.R., Khare, N., Gadekallu, T.R., Hwang, W.J., Pham, Q.V., Deep learning for intelligent demand response and smart grids: A comprehensive survey, *Computer Science Review*, Vol. 51, 2024, 100617, doi: doi.org/10.1016/j.cosrev.2024.100617.

[4] Al-Selwi, S.M., Hassan, M.F., Abdulkadir, S.J., Muneer, A., Sumiea, E.H., Alqushaibi, A., Ragab, M.G., RNN-LSTM: From applications to modeling techniques and beyond—Systematic review, Journal of King Saud University - *Computer and Information Sciences*, Vol. 36, Issue 5, 2024, 102068, doi: doi.org/10.1016/j.jksuci.2024.102068.

[5] Soori, M., Arezoo, B., Dastres, R., Artificial intelligence, machine learning and deep learning in advanced robotics, a review, *Cognitive Robotics*, Vol. 3, 2023, pp.54-70, doi:doi.org/10.1016/j.cogr.2023.04.001.

[6] Pineda-Jaramillo, J., Viti, F., Identifying the rail operating features associated to intermodal freight rail operation delays, Transportation Research Part C: Emerging Technologies, Vol. 147 (2023)103993, doi: doi.org/10.1016/j.trc.2022.103993.

[7] Bingmei Gu, Jiaguo Liu, Xiaoheng Ye, Yu Gong, Jihong Chen, Data-driven approach for port resilience evaluation, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 186, (2024) 103570, doi: doi.org/10.1016/j.tre.2024.103570.

[8] Xu, K., Zhen, H., Li, Y., Yue, L., Comprehensive Monitoring System for Multiple Vehicles and Its Modelling Study, *Transportation Research Procedia*, Vol. 25, (2017) pp.1824-1833, doi.org/10.1016/j.trpro.2017.05.160.

[9] Jovanovic, N., Zolfagharinia, H., Peszynski, K., To Green or Not to Green Trucking? Exploring the Canadian Case, *Transportation Research Part D: Transport and Environment*, Vol. 88, (2020) 102591, doi: doi.org/10.1016/j.trd.2020.102591.

[10] Basso, F.,nCox, T., Pezoa, R., Maldonado, T., Varas, M., Characterizing last-mile freight transportation using mobile phone data: The case of Santiago, Chile, *Transportation Research Part A: Policy and Practice*, Vol. 186, (2024) 104149, doi: doi.org/10.1016/j.tra.2024.104149.

[11] Joseph, R.V., Mohanty, A., Tyagi, S., Mishra, S., Satapathy, S.K., Mohanty, S.N., A hybrid deep learning framework with CNN and Bi-directional LSTM for store item demand forecasting, *Computers and Electrical Engineering*, Vol. 103, 2022, 108358, doi: doi.org/10.1016/j.compeleceng.2022.108358.

[12] Halim, R., A., Kwakkel, J.H., Tavasszy, L.A., A scenario discovery study of the impact of uncertainties in the global container transport system on European ports, *Futures*, Vol. 81 (2016) pp.148-160, doi: doi.org/10.1016/j.futures.2015.09.004.

[13] Turgay, S., Gujrati, R., Analyzing of B2B and B2C in Buying Centers using Rough Sets and MCDM Respect of Sustainable Supply Chain Management, http://dx.doi.org/10.2139/ssrn.4295742.

[14] Zhang, P., Lei, D., Liu, S., Jiang, H., Recursive logit-based meta-inverse reinforcement learning for driver-preferred route planning, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 185, (2024) 103485, doi: doi.org/10.1016/j.tre.2024.103485.

[15] Chen, R., Meng, Q., Jia, P., Container port drayage operations and management: Past and future, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 159 (2022) 102633, doi: doi.org/10.1016/j.tre.2022.102633.

[16] Tsolaki, K., Vafeiadis, T., Nizamis, A., Ioannidis, D., Tzovaras, D., Utilizing machine learning on freight transportation and logistics applications: A review, *ICT Express*, Vol. 9, Issue 3, (2023) pp.284-295, doi: doi.org/10.1016/j.icte.2022.02.001.

[17] Abideen, A.Z., Sorooshian, S., Sundaram, V.P.K., Mohammed, A., Collaborative insights on horizontal logistics to integrate supply chain planning and transportation logistics planning – A systematic review and thematic mapping, *Journal of Open Innovation: Technology, Market, and Complexity*, Vol. 9, Issue 2 (2023) 100066, doi: doi.org/10.1016/j.joitmc.2023.100066.

[18] Yang, Z., Tan, Z., Zhen, L., Zhang, N., Liu, L., Fan, T., Column generation for service assignment in cloud-based manufacturing, *Computers & Operations Research*, Vol. 161(2024) 106436, doi: doi.org/10.1016/j.cor.2023.106436.

[19] Turgay, S., Yaşar, O., Aydin, A., A Multi-objective Framework for Dairy Products Supply Chain Network with Benders Decomposition, *Industrial Engineering and Innovation Management* (2023), Vol. 6 Num. 5, doi:10.23977/ieim.2023.060509.

[20] Giusti, R., Manerba, D., Bruno, G., Tadei, R., Synchromodal logistics: An overview of critical success factors, enabling technologies, and open research issues, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 129 (2019), pp. 92-110, doi: doi.org/10.1016/j.tre.2019.07.009.

[21] Chiara Cimini, C., Lagorio, A., Pirola, F., Pinto, R., Exploring human factors in Logistics 4.0: empirical evidence from a case study, *IFAC-PapersOnLine*, Vol.52, Issue 13 (2019), pp.2183-2188, doi: doi.org/10.1016/j.ifacol.2019.11.529.

[22] Monios, J., Bergqvist, R., Logistics and the networked society: A conceptual framework for smart network business models using electric autonomous vehicles (EAVs), *Technological Forecasting and Social Change*, Vol. 151 (2020), 119824, doi: doi.org/10.1016/j.techfore.2019.119824.

[23] Turgay, S., Streamlined Supply Chain Operations: Leveraging Permutation-Based Genetic Algorithms for Production and Distribution, *WSEAS Transactions on Information Science and Applications*, Vol. 21 (2024), pp.23-32. https://doi.org/10.37394/23209.2024.21.3.

[24] López, J.J.U., Qassim, R.Y., A novel modeling approach for the redesign of supply chains: An application to soybean grain supply chains, *Research in Transportation Business & Management*, Vol.51, (2023), 101037, doi: doi.org/10.1016/j.rtbm.2023.101037.

[25] Wiederer, C., Straube, F., A decision tool for policymakers to foster higher-value perishable agricultural exports, *Transportation Research Interdisciplinary Perspectives*, Vol. 2, (2019), 100035, doi: doi.org/10.1016/j.trip.2019.100035.

[26] Fan, Q., Jin, Y., Wang, W., Yan, X., A performance-driven multi-algorithm selection strategy for energy consumption optimization of sea-rail intermodal transportation, *Swarm*

and Evolutionary Computation, Vol.44 (2019), pp.1-17.

[27] Filom, S., Amiri, A.M., Razavi, S., Applications of machine learning methods in port operations – A systematic literature review, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 161, (2022), 102722, doi: doi.org/10.1016/j.tre.2022.102722.

[28] Baykasoğlu, A., Subulan, K., Taşan, A.S., Dudaklı, N., A review of fleet planning problems in single and multimodal transportation systems, *Transportmetrica A Transport Science*, Vol.15, Issue 2, (2019), pp.631-697, doi: doi.org/10.1080/23249935.2018.1523249.

[29] Chen, Z., Hu, W., Xu, Y., Dong, J., Yang, K., Ren, R., Exploring decision-making mechanisms for the metro-based underground logistics system network expansion: An example of Beijing, *Tunnelling and Underground Space Technology*, Vol.139 (2023), 105240, doi: doi.org/10.1016/j.tust.2023.105240.

[30] Samadhiya, A., Agrawal, R., Kumar, A., Garza-Reyes, J.A., Regenerating the logistics industry through the Physical Internet Paradigm: A systematic literature review and future research orchestration, *Computers & Industrial Engineering*, Vol.178, (2023), 109150, doi: doi.org/10.1016/j.cie.2023.109150.

[31] Yang, Y., Jia, B., Yan, X.Y., Chen, Y., Song, D., Zhi, D., Wang, Y., Gao, Z., Estimating intercity heavy truck mobility flows using the deep gravity framework, *Transportation Research Part E: Logistics and Transportation Review*, Vol.179, (2023), 103320, doi: doi.org/10.1016/j.tre.2023.103320.

[32] Caliskan, F., Idug, Y., Gligor, D., Hong, S.J., Blockchain technology for building buyer-supplier trust and transparency in supply chains: An experimental study on P2P online marketplace vendors, *Industrial Marketing Management*, Vol.124, (2025), pp.239-253. doi: 10.1016/j.indmarman. 2024.12.004.

[33] Temizceri, F.T., Kara, S.S., Towards sustainable logistics in Turkey: A bi-objective approach to green intermodal freight transportation enhanced by machine learning, *Research in Transportation Business & Management*, Vol.55 (2024), 101145, doi: doi.org/10.1016/j.rtbm.2024.101145.

[34] Maiyar, L.M., Thakkar, J.J., Environmentally conscious logistics planning for food grain industry considering wastages employing multi-objective hybrid particle swarm optimization, *Transportation Research Part E: Logistics and Transportation Review*, Vol.127, (2019), pp.220-248, doi: doi.org/10.1016/j.tre.2019.05.006.

[35] Yu, Z., Wang, H., Liu, X., Mobility heterogeneity of urban freight areas: Geospatial evidence from shared logistics dynamics, *Transportation Research Part E: Logistics and Transportation Review*, Vol.188, (2024), 103657, doi: doi.org/10.1016/j.tre.2024.103657.

[36] Fahim, P.B.M., Rezaei, J., Montreuil, B., Tavasszy, L., Port performance evaluation and selection in the Physical Internet, *Transport Policy*, Vol.124, (2022), pp.83-94. https://doi.org/10.1016/j.tranpol.2021.07.013.

[37] Fareed, A.G., Felice, F.D., Forcina, A., Petrillo, A., Role and applications of advanced digital technologies in achieving sustainability in multimodal logistics operations: A systematic literature review, *Sustainable Futures*, Vol.8, (2024), 100278. https://doi.org/10.1016/j.sftr.2024.100278.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

- S.Turgay, S.Erdoğan – investigation,
- M.K.Omeroglu, S.Turgay-validation and
- S.Turgay, S.Erdoğan, M.K.Omeroglu-writing & editing.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received to conduct this study.

Conflict of Interest

The authors have no conflicts of interest to declare.