Iranian Journal of Fuzzy Systems

Volume 23, Number 1, (2026), pp. 37-73

Original paper

University of Sistan and Baluchestan

Three-dimensional uncertainty modeling in intelligent logistics: Fermatean neutrosophic rough tensor decomposition for supply chain optimization

M. Kamran [1], Q. Zhang [2], M. Nadeem [3], T. Allahviranloo [4] and M. Tahir [5]

[1,2] *Research Institute of Business Analytics and SCM, College of Management, Shenzhen University, Shenzhen, China*
[3] *Department of Mathematics, Lahore Garrison University, Lahore, Pakistan*
[4] *Research Center of Performance and Productivity Analysis, Istinye University, Istanbul, Turkey*
[5] *Department of Mathematics, Institute of Numerical Sciences, Gomal University, Dera Ismail Khan, 29050, KPK, Pakistan*

kamrankfueit@gmail.com, q.yu.zhang@gmail.com, nadeemqau82@yahoo.com
allahviranloo@gmail.com, tahirkhanbaloch30@gmail.com

Abstract

The article is a hybrid union of rough fermatean neutrosophic sets (RFNS) and machine learning (ML) to economise the cost of a supply chain in the context of complicated uncertainty. We develop a two-fold approach: The initial one is a numerical procedure of defuzzifying RFNS parameters into sharp figures and addressing them through a conventional VAM and MODI algorithm; the second one is a machine learning, model according to which the components of the RFNS are handled by a two-strand neural network. This plan transforms the unclear transportation problem into a hybrid form. To validate the framework, we compare its performance with classical performances and other fuzzy hybrid techniques. Our machine learning strategy's statistical analysis reveals that it is as optimized as traditional approaches, and it offers a platform for real-time decisions in dynamic environments.

Keywords: Rough sets, fermatean neutrosophic sets, supply chain, machine learning.

1 Introduction

The supply chain is the main component of contemporary commerce, facilitating the swift and efficient movement of goods globally. A supply chain is a complex network of enterprises, resources, processes, and technologies that collaborate to produce and market a product [26]. The process commences with the acquisition of raw materials and concludes with the delivery of the finished product to the client. Transportation serves as the crucial link inside this network, facilitating the physical connection between each node and enabling the movement of goods among suppliers, manufacturers, distribution hubs, and retailers. Consequently, travel frequently constitutes one of the largest expenses in business operations, occasionally accounting for 50-60% of total logistical costs [24]. Reducing transportation expenses is not merely a tactical objective; it is a strategic imperative for enhancing the profitability, competitiveness, and sustainability of the supply chain. Operating within the constraints of supply and demand, the classic transportation problem (TP) for optimal routing seeks to determine the best cost-effective means of getting goods from several suppliers to a large number of consumers [22]. When all supply, demand, and cost parameters are known with certainty, classic methods such as the Modified Distribution (MODI) optimization strategy [13] and Vogel's Approximation (VAM) approach [19] provides solid results for deterministic applications. However, there are many unknowns and potential difficulties with the actual world supply chain and reconfigurable distributed flowshop group scheduling problem [30]. Parameters are frequently ambiguous, incomplete, or undefined due to factors such as complex, high-dimensional, fluctuating fuel prices, difficult demand estimation, scarce vehicles, terrible traffic, and ever-changing rules [8]. Classical sharp models are inadequate for decision making because of this reality.

Corresponding Author: Q. Zhang
Received: July 2025; Revised: September 2025; Accepted: December 2025.
https://doi.org/10.22111/ijfs.2025.52675.9305

The linear programming problem (LPP) known as the TP was first described by Hitchcock [10] and later by Balinski [3]. It is one of the most significant and practical application domains of operations research and supply chain networks. The objective of TP is to minimize the expenses associated with the distribution of goods from various sources to multiple destinations and make the business sustainable [29]. It is a particular type of LPP that is typically solved by simple numerical methods without seeing the uncertainties in existing studies and is inappropriate for addressing transportation-related issues because of its unique structure [19]. These difficulties necessitate a distinctive approach to solutions and a smooth supply chain system for sustainability and path planning in complex radioactive environment [28]. First, a transportation problem arises when a shipment moves beyond the freight relaying point. Unit transportation cost: Unit transportation cost is the expenditure of transferring a unit of the above shipment towards a destination point.

There are two types of transportation issues, which are the balanced and the imbalanced. A transportation issue: the issue of minimizing the total cost of transporting goods between a specified set of origins to a specified set of destinations, given availability and demands at each origin and destination, in both directions [13]. The supply and demand of the transportation problem are equal, and this is what leads to a balanced trading position, whereas there is an imbalance in the demand and supply in the transportation problem, and this is what leads to an uneven trading position. In such a problem, we take the dummy row or a dummy column such that the numbers can be balanced, but this process is very cumbersome and consumes a lot of time, and thus, there are many chances of introducing human errors. We may also get rid of the TPs by calling them classic LPPs and using the well-known simplex method. There are several methodologies for formulating a basic, sustainable solution to the transportation challenge [14]. The Vogel approximation method [19], the row minimum, the column minimum, the northwest corner, and the matrix minimum [12] are all examples of these methods. Also, there are a variety of proposals for evolution to solve the transportation problem.

The traditional TPs have never considered the fact that the decision maker is certain regarding the correct measures of the availability of products, demand and the cost of transportation. Nonetheless, the dynamic character of the economic environment prevents finding for traffic data graphs and discovering the solution to all the elements of the transportation issue [23]. This leads to ambiguity and uncertainty being present in TP estimates. Depending on the economic uncertainties which are brought forth by the change of conditions, the decision-maker normally anticipates that he or she would have enough resources, requests and transportation expenses to overcome transportation limitations despite certain transportation details being inaccurate and some being obsolete [17]. The TP measure is an estimate in nature, as it lacks total information and is ambiguous in nature [1]. To address this deficiency, scholars have employed fuzzy set theory and its expansions, such as intuitionistic fuzzy sets and neutrosophic sets. These can manage the uncertainty, ambiguity, and hesitance prevalent in real-world data. Fuzzy sets (FS) [27] and their variants, including intuitionistic FS (IFS) [2] and neutrosophic sets (NS) [18], can be helpful in situations where dealing with uncertainty is the order of the day, yet they do not perform well in complex logistics, where there is extensive uncertainty, inconsistent information, and detailed information. Models like Type-2 FS (T-2FS) [15] are computationally infeasible for real-time systems, and Dempster-Shafer evidence theory (DST) [32] can provide counterintuitive results when strongly contradictory information is combined (e.g., sensor measurements, managerial reports, market predictions).

These models have advanced the industry; yet, they frequently encounter challenges posed by causal inference-based root cause analysis framework using multi-modal data and complex uncertainties inherent in dynamic supply networks [6]. Fermatean fuzzy sets (FFSs) [20] represent a unique category of sets capable of encompassing greater uncertainty than Pythagorean or intuitionistic sets. This renders them a more potent technique. Their integration with rough sets yields rough fermatean neutrosophic sets (RFNS), which constitute a sophisticated framework capable of addressing higher-order uncertainty by simultaneously managing degrees of truth-membership, indeterminacy-membership, and falsity-membership, along with their approximations. Despite advancements in this theory, a significant methodological gap persists: the efficient analysis and optimization of issues outlined within this complex framework to obtain actionable real-time responses. As a result, our basic uncertainty hybrid model framework is explicitly selected due to its better capacity to describe three-dimensional uncertainty (truth, indeterminacy, falsity) and manage knowledge granularity by rough approximations simultaneously. The fermatean condition $(0 \leq (\varphi)^3+(\upsilon)^3 \leq 1)$ does not require the same tightness of representation as either pythagorean or intuitionistic fuzzy sets, but is important in representing the deep hesitancy of supply chain data [5, 21]. The truth, indeterminacy, and falsity are explicitly and independently measured in the neutrosophic component, which is needed to represent states such as an 'unconfirmed shipment' (high indeterminacy). Lastly, the rough set layer effectively distinguishes what is known definitively (lower approximation) and what is potentially known (upper approximation) of such parameters and is a computationally tractable and semantically rich model, uniquely adapted to the adversarial and imprecise environment of intelligent supply chains and better than alternatives in expressiveness and efficiency. The current paper investigates the fact that contemporary supply-chain structures are becoming more complicated with a large number of uncertainties like logistics, the dynamism of market

conditions, and the tendency of people to make inaccurate decisions. In most cases, traditional TP models fail to account for three-dimensional (3D) uncertainty, which includes the true extent of truth, indeterminacy, and falsehoods simultaneously. The gap presented in this study is filled by the introduction of a new uncertainty model that merges RFNS with more sophisticated machine learning methods to optimize intelligent logistics.

1.1 Literature review and identified research gap

An overall analysis shows that there is a definite change in the approach to solving optimization problems through uncertainty. The trip commenced with FS [27] , which proposed the membership degree(MD) to represent ambiguity. This was extended to IFS by the inclusion of a non-membership degree(n-MD) to represent hesitancy better, according to the authors of the original paper [2]. Subsequent extensions to NS [18] and their single-valued counterparts, simply and independently modelling the degrees of truth, indeterminacy and falsity, were introduced. Simultaneously, the rough set theory was introduced as a potent means of uncertainty management because of information granularity. These tools have been used proactively by researchers in the particular field of transportation issues. Fuzzy linear programming has been studied [11], as well as intuitionistic fuzzy cost coefficients [4], and several solution algorithms like granular-ball rough sets [25] in uncertain environments have been studied. Alternative systems, like logistics omnidirectional robots [9], T-2FS, neural network dynamic model for sustainability [31] and DST, have been employed in broader domains of risk management and sensor fusion. Due to the constraints of the purely numerical approach, there has been a transition in the field towards hybrid intelligent systems [16]. This entails integrating the aforementioned fuzzy models with machine learning (ML) models. Type-I fuzzy systems with neural networks for pattern identification in routing have been utilized in conjunction with neural networks for pattern identification in routing and fuzzy logic to optimize the hyperparameters of evolutionary algorithms and a more robust analytical framework for traffic flow prediction [33]. Nonetheless, these mixed techniques retain the fundamental shortcomings of the underlying uncertainty models. They are ultimately constrained by the representational capacity of Type-1, IFS, or NS, which cannot concurrently encapsulate the substantial hesitancy, contradiction, and complexity of contemporary logistical data in a computationally feasible manner. However, a synthesis of all these effective elements reveals a fundamental and persistent deficit. Current applications are predominantly isolated, addressing only one aspect of uncertainty, and lack a unified mechanism to concurrently manage multi-dimensional uncertainty and the information granularity intrinsic to complicated real-world logistics data. Moreover, more robust models like T2FS pose computational barriers to real-time models, and DST may yield counterintuitive results in real time when integrating highly contradictory data sets from disparate sources. Consequently, there exists an urgent necessity for a cohesive framework that can: (1) concurrently model 3D uncertainty with an enhanced degree of freedom, as facilitated by the fermatean condition; (2) address approximation errors in knowledge granularity via rough sets; and (3) operate in a computationally efficient manner that integrates with contemporary optimization techniques, including machine learning. This integration of rough and fermatean NSs, employed to create a hybrid numerical and machine learning solution in smart logistics, constitutes the original contribution of this study and addresses the identified gap in the literature.

1.2 Foundational challenges

The traditional TP has a well-designed theoretical structure; however, it has three serious implementation problems in practice:

1. The conventional FS and IFS approaches are not able to depict the links between truth, indeterminacy and falsity as well as to comply with the cubic power regulations of fermatean fuzzy logic. We dealt with these constraints in a civilized manner.

2. Precise solutions like Vogel approximation (VAM) and modified distribution (MODI) are very time-consuming and cannot be used, as they achieve a complexity of $\mathrm{Œ}(n!)$ in the example with n dimensions.

3. Statistical models are incapable of independently adapting to real-time variations in supply-demand dynamics and transportation expenses. The system required a hybrid model that addressed these limitations and autonomously adapted to real-time variations in supply-demand dynamics and transportation expenses.

1.3 Innovative contributions

These issues are addressed in our model based on four key theoretical and computational developments:

1. The numerical solution of the model is carried out by numerical methods based on the fermatean neutrosophic space in determining the transportation cost. A machine learning model is then created that can reconcile fermatean neutrosophic space to automatically and accurately solve the same problem within a shorter period of time.

2. In the framework:
$$\mathcal{M} = (\mathbb{F}^3_{\mathrm{RFN}}, \otimes_{\mathrm{hyper}}, \mathcal{G}_{\mathrm{RL}}), \tag{1}$$
here, $\mathbb{F}^3_{\mathrm{RFN}}$ denotes the fermatean neutrosophic space, and we use hypercomplex operations instantiated by the symbol, $\otimes_{\mathrm{hyper}}$ and a collection of rules that are obeyed in states of an environment, denoted as $\mathcal{G}_{\mathrm{RL}}$.

3. A dual-stream neural architecture featuring:
$$\mathrm{LSTM}_{\mathrm{RFN}} \oplus \mathrm{MLP}_{\mathrm{crisp}} \to \mathbb{Q}_{\mathrm{policy}}, \tag{2}$$
where $\oplus$ denotes tensor concatenation and $\mathbb{Q}_{\mathrm{policy}}$ is the Q-learning output space.

4. An adaptive reward mechanism:
$$R_t = -\left(\sum_{i,j} Y_{ij} S(T_{ij}) + \lambda \|\text{œ}_{t+1}\|_1 + \lambda \|\text{æ}_{t+1}\|_1 \right), \tag{3}$$
with λ dynamically adjusted via:
$$\lambda_{t+1} = \lambda_t \cdot \exp\left(-\eta \nabla_\theta \mathcal{L}_{\mathrm{constraints}}\right). \tag{4}$$

1.4 Practical implications

The suggested framework shows significant advantages in three areas of operation, as shown in Table 1.

Table 1: Comparative Performance Characteristics

Metric	**Traditional Numerical Method**	**ML Method**
Uncertainty Resolution	Good	Excellent with accuracy
Time Complexity	Œ$(n!)$	Œ$(n \log n)$
Adaptation Speed	Manual recalibration	Real-time

1.5 Structural overview

The scheme that constitutes the administrative structure of the research paper is the following: the establishment of the research basics is developed in Section 2, where the principles of RFNS are provided. This is also mentioned in Subsection 2.1, which describes the basic operations of RFNS needed when modelling transportation problems. In Section 3, it changes to classical solution methods that involve processes that have been proposed to be applied in the resolution of transportation problems in an environment where there are cut-cut situations. The strategies in RFNS are mathematically formalized in Subsection 3.1. It is based on this that I have proceeded to offer a sophisticated machine learning-driven solution in Section 4 that will fit the RFNS transportation issue. In Section 5 , the practical application and utilization of such models and of interest are done in both balanced and unbalanced situations of transportation in Subsection 5.1 and Subsection 5.2, respectively. Section 6 shows that the suggested models are informative and strong, and Subsection 6.2 shows their visual trends and explanatory charts. Moreover, a comprehensive comparison with fuzzy-ML hybrid baselines is presented in Subsection 7, supplemented by evidence from the robustness and sensitivity analysis (Subsection 7.1), the analytical error and advancement analysis (Subsection 7.2), and the overall discussion (Subsection 7.4). Finally, concluding the research, Section 8 presents reflections, as well the potential future developments.

2 Preliminaries

This section delineates the mathematical foundation of RFNS specifically designed to characterise the intrinsically complicated multi-dimensional uncertainty in intelligent logistics. We examine traditional definitions to explore the adaptations and novel combinations that form the foundation of this work. Their foundation is rooted in standard

fuzzy set variants as proposed by Zadeh [27] and Atanassov [2], along with synthesis methodologies addressing profound hesitancy [18], information granularity, and conflicting information within supply chains, thereby necessitating the advanced synthesis forms presented herein.

Definition 2.1. [20] *Let (X) be a universe of discourse (for example, the set of all potential transit prices between two nodes). An FFS $\mathbb{F}$ on X is defined as:*

$$\mathbb{F} = \{\langle x, \varphi(x), \upsilon(x)\rangle \mid x \in X\},$$

where $\varphi(x), \upsilon(x) \in [0,1]$ represent the MD and n-MD of x to $\mathbb{F}$, satisfying the condition:

$$0 \leq (\varphi(x))^3 + (\upsilon(x))^3 \leq 1.$$

The indeterminacy is $\pi(x) = \sqrt[3]{1-(\varphi(x))^3-(\upsilon(x))^3}$. The cubic constraint enables FFS to represent far wider ranges of ambiguity and indecision compared to IFS or Pythagorean fuzzy sets(PyFS). This is important in logistics for making statements such as, "We are very sure ($\varphi = 0.9$) that the route is congested, but there is also strong evidence ($\upsilon = 0.6$) that it is not," which is acceptable FFS because $0.9^3 + 0.6^3 = 0.945 \leq 1$.

Definition 2.2. *A Fermatean neutrosophic set (FNS)* [13], *$\mathcal{F}^{\mathcal{N}}$ on a universe X is defined as:*

$$\mathcal{F}^{\mathcal{N}} = \{\langle x, \varphi(x), \vartheta(x), \upsilon(x)\rangle \mid x \in X\},$$

where $\varphi(x) : X \to [0,1]$ denotes the degree of truth-membership, $\vartheta(x) : X \to [0,1]$ denotes the degree of indeterminacy-membership, and $\upsilon(x) : X \to [0,1]$ denotes the degree of falsity-membership of the element $x \in X$ to the set $\mathcal{F}^{\mathcal{N}}$. These functions satisfy the following constraints for all $x \in X$:

$$0 \leq (\varphi(x))^3 + (\upsilon(x))^3 \leq 1, \quad 0 \leq (\vartheta(x))^3 \leq 1, \quad \textit{and} \quad 0 \leq (\varphi(x))^3 + (\vartheta(x))^3 + (\upsilon(x))^3 \leq 2.$$

Here, $\varphi(x)$ and $\upsilon(x)$ are dependent components, while $\vartheta(x)$ is an independent component. This 3-D structure is essential for supply chain data. For example, a sensor's demand forecast may be modeled as truth (confirmed demand, $\varphi = 0.8$), indeterminacy (forecast volatility, $\vartheta = 0.4$), and falsehood (belief in an inaccurate value, $\upsilon = 0.3$), with $0.8^3 + 0.3^3 = 0.512{+}0.027 = 0.539 \leq 1$ and $0.512{+}0.064{+}0.027 = 0.603 \leq 2$. Because $\vartheta(x)$ is independent, data quality measurements may directly affect it.

Definition 2.3. *Let $(X, \mathcal{R})$ be an approximation space characterized by the equivalence relation $\mathcal{R}$. The lower and upper approximations of any FNS $\Omega \subseteq X$ are $\underline{\mathcal{R}}(\Omega)$ and $\overline{\mathcal{R}}(\Omega)$, which are FNSs whose MD for an equivalence class $[x]_{\mathcal{R}}$ are:*

$$\varphi_{\underline{\mathcal{R}}(\Omega)}(x) = \inf_{y\in[x]_{\mathcal{R}}} \varphi_\Omega(y), \qquad \vartheta_{\underline{\mathcal{R}}(\Omega)}(x) = \sup_{y\in[x]_{\mathcal{R}}} \vartheta_\Omega(y), \qquad \upsilon_{\underline{\mathcal{R}}(\Omega)}(x) = \sup_{y\in[x]_{\mathcal{R}}} \upsilon_\Omega(y),$$
$$\varphi_{\overline{\mathcal{R}}(\Omega)}(x) = \sup_{y\in[x]_{\mathcal{R}}} \varphi_\Omega(y), \qquad \vartheta_{\overline{\mathcal{R}}(\Omega)}(x) = \inf_{y\in[x]_{\mathcal{R}}} \vartheta_\Omega(y), \qquad \upsilon_{\overline{\mathcal{R}}(\Omega)}(x) = \inf_{y\in[x]_{\mathcal{R}}} \upsilon_\Omega(y).$$

The pair $(\underline{\mathcal{R}}(\Omega), \overline{\mathcal{R}}(\Omega))$ is called an RFNS. This constitutes one of the new adaptation requirements in this field. The rough approximation addresses issues related to granularity and incomplete information. In a scenario where only the costs of a region (equivalence class) are known, without specific port costs, the lower approximation yields the definitive minimum possible cost, while the upper approximation provides the potential maximum cost. Both approximations are accompanied by measures of certainty related to the granularity of knowledge, specifically indeterminacy and falsity.

Definition 2.4. *For a rough fermatean neutrosophic number (RFNN), $\mathcal{L} = ((\varphi_l, \vartheta_l, \upsilon_l), (\varphi_u, \vartheta_u, \upsilon_u))$, we define:*

$$S(\mathcal{L}) = \frac{1}{4}\left[(2 + \varphi_l - \vartheta_l - \upsilon_l) + (2 + \varphi_u - \vartheta_u - \upsilon_u)\right],$$
$$A(\mathcal{L}) = \frac{1}{2}\left[(\varphi_l + \vartheta_l + \upsilon_l) + (\varphi_u + \vartheta_u + \upsilon_u)\right].$$

These functions will give an overall order to be optimized. The greater the score of the $S(\mathcal{L})$, the better, more definite, and true (e.g., lower and more definite cost). The accuracy function $A(\mathcal{L})$ is used to resolve ties, with a larger value indicating a more reliable measure.

They are the key to converting the RFNS parameters to a set of workable inputs to our numerical VAM/MODI algorithms and the rewarding complication of our ML model, such that the uncertain costs can be directly compared, even optimized.

Definition 2.5. *The RFNS transportation problem is described as minimizing the overall cost:*

$$Min = \sum_{i=1}^{m}\sum_{j=1}^{n} \mathrm{Y}_{ij} \otimes \widetilde{T}_{ij},$$

subject to:

$$\sum_{j=1}^{n} \mathrm{Y}_{ij} \approx \widetilde{\text{œ}}_i \quad \textit{and} \quad \sum_{i=1}^{m} \mathrm{Y}_{ij} \approx \widetilde{\text{æ}}_j,$$

where $\widetilde{T}_{ij}, \widetilde{\text{œ}}_i, \widetilde{\text{æ}}_j$ are RFNNs that show the cost, supply, and demand of a unit, respectively. The operators $\otimes$ (multiplication) and $\approx$ (equality under uncertainty) are defined by the scoring function $S(\cdot)$ and the accuracy function $A(\cdot)$ to verify that RFNS restrictions allow for possible solutions.

This is one of the main theoretical contributions, in the form of formalization. It generalizes the classical TP model to a setting with all the parameters being RFNNs, representing the entire range of uncertainty of imprecise measurements and subjective conflicting reports, as well as granular knowledge. This model is operationalized by the following numerical and ML solutions.

Summary of the RFNS framework

Table 2 and Figure 1 illustrate the main elements of the rough fermatean neutrosophic framework, showing how the ideas developed into the first ideas and their particular interest in the context of modelling the uncertainty in logistics.

Table 2: Summary and Comparison of Uncertainty Modeling Frameworks

Concept	**Core Mathematical Feature**	**Value for Logistics Optimization**
FFS	Cubic constraint: $0 \leq \varphi^3 + \upsilon^3 \leq 1$	Models stronger hesitancy than IFS/PyFS (e.g., $\varphi = 0.9, \upsilon = 0.6$ is valid). Captures extreme expert judgments.
FNS	Adds independent indeterminacy: $(\varphi, \vartheta, \upsilon)$	Separately models truth, data quality/volatility (ϑ), and falsity. Essential for sensor data and forecasts.
Rough Set Approximation	Lower and upper approximations based on knowledge granularity.	Distinguishes between *certain* and *possible* values for costs/demand when information is incomplete or grouped (e.g., by region).
RFNS	Synthesis of FNS and rough sets.	**Primary Contribution:** The system integrates truth, indeterminacy, and falsity modelling of 3D uncertainty and knowledge granularity into one model.
Score/Accuracy Functions	$S(Ł)$ for ranking, $A(Ł)$ for reliability.	**Operational Contribution:** Offers the defuzzification mechanism of RFNNs to be used in both classical (VAM/MODI) and ML optimization algorithms to be directly compared and optimized.

This synergistic combination makes the RFNS framework valid: it represents a broader range of real-world uncertainty than its predecessors (FS, IFS, PyFS, NS), but it is also computationally inexpensive to integrate with ML solvers, as will be shown in the following sections.

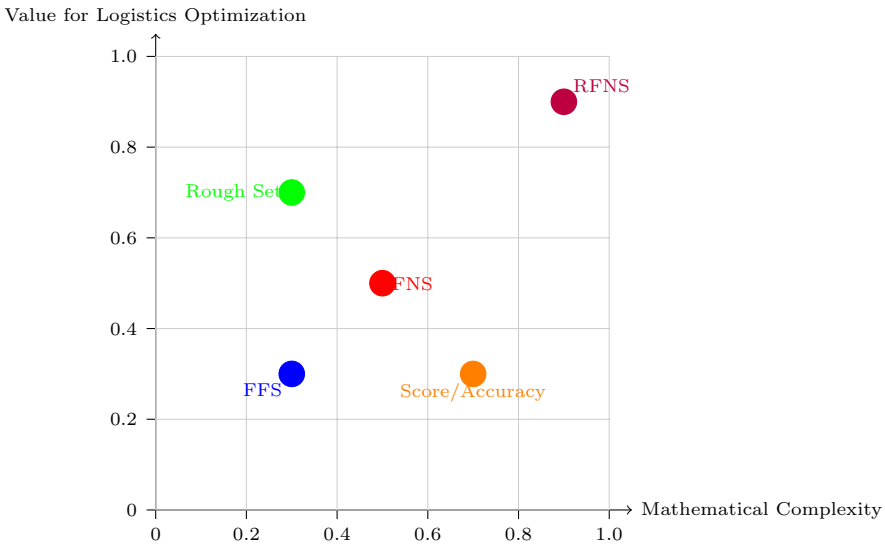


Figure 1: Fuzzy-Scale Visualization (0-normalized axes).

2.1 Basic operations on RFNS

Let us have two rough fuzzy neutrosophic numbers (RFNN) $Ł_1 = (Ł_{\flat 1}, Ł_1^{\flat}) = [(\varphi_{\flat 1}(\mathtt{x}), \vartheta_{\flat 1}\mathtt{x}, \upsilon_{\flat}\mathtt{x}), (\varphi_1^{\flat}\mathtt{x}, \vartheta_1^{\flat}\mathtt{x}, \upsilon_1^{\flat}\mathtt{x})]$ and $Ł_2 = (Ł_{\flat 2}, Ł_2^{\flat}) = [(\varphi_{\flat 2}\mathtt{x}, \vartheta_{\flat 2}\mathtt{x}, \upsilon_{\flat 2}\mathtt{x}), (\varphi_2^{\flat}\mathtt{x}, \vartheta_2^{\flat}\mathtt{x}, \upsilon_2^{\flat}\mathtt{x})]$ then some basic operations on RFNS are illustrated below;

i. Compliment

$$Ł_{\flat 1}^{C} = (\varphi_{\flat 1}\mathtt{x},\ 1-\vartheta_{\flat 1}\mathtt{x},\ \upsilon_{\flat 1}\mathtt{x}), \qquad Ł^{\flat}C_1 = (\varphi_1^{\flat}\mathtt{x},\ 1-\vartheta_1^{\flat}\mathtt{x},\ \upsilon_1^{\flat}\mathtt{x}),\ \forall\,\mathtt{x}\in X.$$

ii. Union

$$\begin{aligned} \varphi_1\mathtt{x}\cup\varphi_2\mathtt{x} &= \max[\varphi_1\mathtt{x},\varphi_2\mathtt{x}], & \varphi_1\mathtt{x}\cup\varphi_2\mathtt{x} &= \max[\varphi_1\mathtt{x},\varphi_2\mathtt{x}],\\ \vartheta_1\mathtt{x}\cup\vartheta_2\mathtt{x} &= \min[\vartheta_1\mathtt{x},\vartheta_2\mathtt{x}], & \vartheta_1\mathtt{x}\cup\vartheta_2\mathtt{x} &= \min[\vartheta_1\mathtt{x},\vartheta_2\mathtt{x}], \quad \forall\,\mathtt{x}\in X.\\ \upsilon_1\mathtt{x}\cup\upsilon_2\mathtt{x} &= \min[\upsilon_1\mathtt{x},\upsilon_2\mathtt{x}], & \upsilon_1\mathtt{x}\cup\upsilon_2\mathtt{x} &= \min[\upsilon_1\mathtt{x},\upsilon_2\mathtt{x}], \end{aligned}$$

iii. Intersection

$$\begin{aligned} \varphi_{\flat 1}(\mathtt{x})\cap\varphi_{\flat 2}(\mathtt{x}) &= \min[\varphi_{\flat 1}(\mathtt{x}),\varphi_{\flat 2}(\mathtt{x})], & \varphi_1^{\flat}(\mathtt{x})\cap\varphi_2^{\flat}(\mathtt{x}) &= \min[\varphi_1^{\flat}(\mathtt{x}),\varphi_2^{\flat}(\mathtt{x})],\\ \vartheta_{\flat 1}(\mathtt{x})\cap\vartheta_{\flat 2}(\mathtt{x}) &= \max[\vartheta_{\flat 1}(\mathtt{x}),\vartheta_{\flat 2}(\mathtt{x})], & \vartheta_1^{\flat}(\mathtt{x})\cap\vartheta_2^{\flat}(\mathtt{x}) &= \max[\vartheta_1^{\flat}(\mathtt{x}),\vartheta_2^{\flat}(\mathtt{x})], \quad \forall\,\mathtt{x}\in X.\\ \upsilon_{\flat 1}(\mathtt{x})\cap\upsilon_{\flat 2}(\mathtt{x}) &= \max[\upsilon_{\flat 1}(\mathtt{x}),\upsilon_{\flat 2}(\mathtt{x})], & \upsilon_1^{\flat}(\mathtt{x})\cap\upsilon_2^{\flat}(\mathtt{x}) &= \max[\upsilon_1^{\flat}(\mathtt{x}),\upsilon_2^{\flat}(\mathtt{x})], \end{aligned}$$

iv. Subset

$$\varphi_{\flat 1}(\mathtt{x})\le\varphi_{\flat 2}(\mathtt{x}),\ \vartheta_{\flat 1}(\mathtt{x})\ge\vartheta_{\flat 2}(\mathtt{x}),\ \upsilon_{\flat 1}(\mathtt{x})\ge\upsilon_{\flat 2}(\mathtt{x}), \quad \varphi_1^{\flat}(\mathtt{x})\le\varphi_2^{\flat}(\mathtt{x}),\ \vartheta_1^{\flat}(\mathtt{x})\ge\vartheta_2^{\flat}(\mathtt{x}),\ \upsilon_1^{\flat}(\mathtt{x})\ge\upsilon_2^{\flat}(\mathtt{x}),\ \forall\,\mathtt{x}\in X.$$

v. Equality $Ł_1 = Ł_2 \iff Ł_{\flat 1} = Ł_{\flat 2}$ and $Ł_1^{\flat} = Ł_2^{\flat}$.

vi. Subset (global) $Ł_1 \subseteq Ł_2 \iff Ł_{\flat 1} \subseteq Ł_{\flat 2}$ and $Ł_1^{\flat} \subseteq Ł_2^{\flat}$.

vii. Union (global) $Ł_1 \cup Ł_2 \iff Ł_{\flat 1} \cup Ł_{\flat 2}$ and $Ł_1^{\flat} \cup Ł_2^{\flat}$.

viii. Intersection (global) $Ł_1 \cap Ł_2 \iff Ł_{\flat 1} \cap Ł_{\flat 2}$ and $Ł_1^{\flat} \cap Ł_2^{\flat}$.

ix. Addition $Ł_1 \oplus Ł_2 \iff Ł_{\flat 1} \oplus Ł_{\flat 2}$ and $Ł_1^{\flat} \oplus Ł_2^{\flat}$.

x. Multiplication $Ł_1 \otimes Ł_2 \iff Ł_{\flat 1} \otimes Ł_{\flat 2}$ and $Ł_1^{\flat} \otimes Ł_2^{\flat}$.

Definition 2.6. *Suppose* $Ł = [(\varphi_{\flat}(\mathtt{x}), \vartheta_{\flat}(\mathtt{x}), \upsilon_{\flat}(\mathtt{x})), (\varphi^{\flat}(\mathtt{x}), \vartheta^{\flat}(\mathtt{x}), \upsilon^{\flat}(\mathtt{x}))]$ *be a RFNNs then the score function* $\mathrm{S}(Ł)$ *is given as* $\mathrm{S}(Ł) = \frac{3+\varphi_{\flat}(\mathtt{x})+\varphi^{\flat}(\mathtt{x})+\vartheta_{\flat}(\mathtt{x})+\vartheta^{\flat}(\mathtt{x})-\upsilon_{\flat}(\mathtt{x})-\upsilon^{\flat}(\mathtt{x})}{6}$.

Definition 2.7. *Suppose* $Ł = [(\varphi_{\flat}(\mathtt{x}), \vartheta_{\flat}(\mathtt{x}), \upsilon_{\flat}(\mathtt{x})), (\varphi^{\flat}(\mathtt{x}), \vartheta^{\flat}(\mathtt{x}), \upsilon^{\flat}(\mathtt{x}))]$ *be RFNNs then the accuracy function* $\mathrm{A}(Ł)$ *is given as* $\mathrm{A}(Ł) = \frac{3+\varphi_{\flat}(\mathtt{x})+\varphi^{\flat}(\mathtt{x})+\vartheta_{\flat}(\mathtt{x})+\vartheta^{\flat}(\mathtt{x})+\upsilon_{\flat}(\mathtt{x})+\upsilon^{\flat}(\mathtt{x})}{6}$.

Definition 2.8. *Suppose we have two RFNNs* $L_1 = [(\varphi_{\flat 1}(\mathtt{x}), \vartheta_{\flat 1}(\mathtt{x}), \upsilon_{\flat 1}(\mathtt{x})), (\varphi_1^{\flat}(\mathtt{x}), \vartheta_1^{\flat}(\mathtt{x}), \upsilon_1^{\flat}(\mathtt{x}))]$ *and* $L_2 = [(\varphi_{\flat 2}(\mathtt{x}), \vartheta_{\flat 2}(\mathtt{x}), \upsilon_{\flat 2}(\mathtt{x})), (\varphi_2^{\flat}(\mathtt{x}), \vartheta_2^{\flat}(\mathtt{x}), \upsilon_2^{\flat}(\mathtt{x}))]$ *then some results related to the accuracy function and score function are given as;*

1. *If* $\mathrm{S}(L_1) < \mathrm{S}(L_2)$ *then* $L_1 < L_2$,
2. *If* $\mathrm{S}(L_1) > \mathrm{S}(L_2)$ *then* $L_1 > L_2$,
3. *If* $\mathrm{S}(L_1) = \mathrm{S}(L_2)$ *then*
   - *If* $\mathrm{A}(L_1) < \mathrm{A}(L_2)$ *then* $L_1 < L_2$,
   - *If* $\mathrm{A}(L_1) > \mathrm{A}(L_2)$ *then* $L_1 > L_2$,
   - *If* $\mathrm{A}(L_1) = \mathrm{A}(L_2)$ *then* $L_1 = L_2$.

Definition 2.9. *In case the amount of the total supplies of various origins is equivalent to the amount of the various destinations' demands, then this sort of TP is referred to as the balanced TP. It can be mathematically represented as* $\sum_{\mathtt{i}=0}^{\overrightarrow{\mathbb{E}}} \text{œ}_{\mathtt{i}} = \sum_{\mathtt{j}=0}^{\overrightarrow{\mathbb{E}}} \text{æ}_{\mathtt{j}}$.

Definition 2.10. *The imbalanced TP occurs when the entire amount of supply from various sources is not equal to the total amount of requests from different destinations. It may be shown mathematically as* $\sum_{\mathtt{i}=0}^{\overrightarrow{\mathbb{E}}} \text{œ}_{\mathtt{i}}\ neq \sum_{\mathtt{j}=0}^{\overrightarrow{\mathbb{E}}} \text{æ}_{\mathtt{j}}$. *Adding a dummy row or column to the table of data will help balance out the imbalanced TPs.*

Definition 2.11. *When all the conditions of supplies from each origin and demands from each destination are satisfied, then the solution is known as the initial feasible solution. Moreover, if the number of allotments is equal to* $\overrightarrow{\mathbb{E}} + \overrightarrow{\mathbb{E}} - 1$, *then the initial feasible solution is referred to as the initial basic feasible solution (IBFS). There are three ways given below by which we can find the initial feasible solution;*

(*North-West Corner Method (NWCM)*; *Least Cost Method (LCM)*; *Vogel's Approximation Method (VAM).*)

We have applied Vogel's approximation method (VAM) to determine the initial feasible solution of the transportation problems stated in the practical examples given in the upcoming sections.

Definition 2.12. *The solution that gives us the minimum transportation cost or the shortest route of the transportation problem among all the initial feasible solutions is called the optimal solution. An optimal solution can be found by two techniques that are given below.*

(*Stepping Stone Method (SS)*; *Modified Distribution Method (MODI).*)

This study will employ the modified distribution approach (MODI) to numerically determine the optimal solution to the transportation problem as illustrated in the demonstrative scenarios. Subsequently, we apply the machine learning-based methodology to the identical dataset. Score-based defuzzification of the image is then conducted by MODI, since it transforms fuzzy costs into reflecting costs, hence confirming the accuracy of cost matrices in MODI. Small datasets are ideally suited for the MODI technique, as it identifies the most often occurring value, serving as a robust measure of central tendency. Conversely, machine learning algorithms are more effectively utilized with extensive datasets due to their reliance on statistical patterns and optimization methods. Defuzzification transforms complex fuzzy data into a clear input without compromising the integrity of the transportation problem structure. Considering that MODI functions on transparent cost structures, the defuzzified values retain the requisite accuracy to evaluate the most optimal alternatives and implement small modifications. Furthermore, score-based defuzzification methods, such as centroid and bisector, retain substantial information, enabling MODI to nearly identify optimal solutions even after the conversion of fuzzy costs. Therefore, whether addressing minor transportation challenges or more significant ones related to machine learning, defuzzification utilizing MODI is assured to produce accurate outcomes and expedite processing.

3 Proposed techniques for transportation problem in crisp environment

In this section of the research article, we will discuss the current traction in a crisp environment and the proposed technique for the RFNS environment related to the optimization of the transportation problem. The objective is the minimization of the total cost of transportation and the establishment of the optimal route of transportation of various kinds of wares, ranging through various locations (sources) to various destinations, distributing the supplies and demands well in each cell. Below, we provide the mathematical model of the transportation problem, which operates within a crisp environment. In this context, Min represents the minimum transportation cost calculated after processing the data.

$$Min = \sum_{\mathfrak{i}=0}^{\overrightarrow{\text{Œ}}} \sum_{\mathfrak{j}=0}^{\overrightarrow{\text{Æ}}} \text{Y}_{\mathfrak{ij}} T_{\mathfrak{ij}}. \tag{5}$$

Subject to,

$\sum_{\mathfrak{j}=0}^{\overrightarrow{\text{Æ}}} \text{Y}_{\mathfrak{ij}} = \text{œ}_{\mathfrak{i}} = \text{Supply, where } \mathfrak{i} = 1, 2, 3, ..., \overrightarrow{\text{Œ}}$ and $\sum_{\mathfrak{i}=0}^{\overrightarrow{\text{Œ}}} \text{Y}_{\mathfrak{ij}} = \text{æ}_{\mathfrak{j}} = \text{Demand, where } \mathfrak{j} = 1, 2, 3, ..., \overrightarrow{\text{Æ}}.$

As the traditional method in the crisp environment is not very useful to deal with the uncertainty and inaccuracy of the vague data, we have proposed to use RFNN for the unit transportation cost, supplies and demands. We are going to simply replace the constraints in the crisp environment "$T_{\mathfrak{ij}}$", "$\text{œ}_{\mathfrak{ij}}$" and "$\text{æ}_{\mathfrak{ij}}$" with the constraints in the RFNS environment "$\overrightarrow{T}_{\mathfrak{ij}}$", "$\overrightarrow{\text{œ}}_{\mathfrak{ij}}$" and "$\overrightarrow{\text{æ}}_{\mathfrak{ij}}$".

Terminologies

Some terms which are going to be useful in the research article are given below;

$\overrightarrow{\text{Œ}}$	Number of sources we have
$\overrightarrow{\text{Æ}}$	Number of targets we have
$\mathfrak{i}$	Index of every source
$\text{Y}_{\mathfrak{ij}}$	The wares are transported from one source to one target
$T_{\mathfrak{ij}}$	The unit transportation cost for each cell in the crisp environment
$\overrightarrow{T}_{\mathfrak{ij}}$	The unit transportation cost for each cell in RFNS environmen
$\mathfrak{j}$	Index of every target
$\text{œ}_{\mathfrak{ij}}$	The supply of each origin in the crisp environment
$\overrightarrow{\text{œ}}_{\mathfrak{ij}}$	The supply of each origin in RFNS environment
$\text{æ}_{\mathfrak{ij}}$	The demand of each target in the crisp environment
$\overrightarrow{\text{æ}}_{\mathfrak{ij}}$	The demand of each target in RFNS environment

3.1 Mathematical formulation of the RFNS transportation algorithm

Let's say that $\overrightarrow{T}_{\mathfrak{ij}} = \left[(\varphi_\flat, \vartheta_\flat, \upsilon_\flat), (\varphi^\flat, \vartheta^\flat, \upsilon^\flat)\right]$ be an RFNN that shows the cost of moving one unit from source $\mathfrak{i}$ to destination $\mathfrak{j}$. Also, let $\overrightarrow{\text{œ}}_{\mathfrak{i}}$ and $\overrightarrow{\text{æ}}_{\mathfrak{j}}$ stand for the RFNS's supply and demands, as well. In Section 5, the validity and convergence of this numerical algorithm are empirically determined. The score function $S(\text{Ł})$ (Definition 2.6) transforms the RFNS transportation problem into a standard crisp linear programming problem, which ensures that future implementations of the well-known VAM and MODI algorithmic approaches will converge to an optimal solution. The design of the RFNS data transforming process, i.e., of $S(\text{Ł})$, in such a manner that the meaningful relationships in the data are maintained, is the critical innovation. The empirical findings affirm that the solutions provided by this method are statistically equal to the traditional approach, and this justifies the pipeline.

Step 1: For every RFNN $\text{Ł} = \left[(\varphi_\flat, \vartheta_\flat, \upsilon_\flat), (\varphi^\flat, \vartheta^\flat, \upsilon^\flat)\right]$, compute its score value using Definition 2.6:

$$S(\text{Ł}) = \frac{3 + \varphi_\flat + \varphi^\flat + \vartheta_\flat + \vartheta^\flat - \upsilon_\flat - \upsilon^\flat}{6}. \tag{6}$$

By using the score function, transform $\overrightarrow{T}_{\mathfrak{ij}}$, $\overrightarrow{\text{œ}}_{\mathfrak{i}}$, and $\overrightarrow{\text{æ}}_{\mathfrak{j}}$ into crisp values.

Step 2: Now verify that is balanced or unbalanced. If balanced then:

$$\sum_{\mathfrak{i}=1}^{\overrightarrow{\text{Œ}}} S(\overrightarrow{\text{œ}}_{\mathfrak{i}}) = \sum_{\mathfrak{j}=1}^{\overrightarrow{\text{Æ}}} S(\overrightarrow{\text{æ}}_{\mathfrak{j}}). \tag{7}$$

If unbalanced, then add a dummy source/destination with supply/demand to transform into balanced:

$$\overrightarrow{\text{œ}}_{\text{dummy}} = \max\left(0, \sum_{\text{j}} S(\overrightarrow{\text{æ}}_{\text{j}}) - \sum_{\text{i}} S(\overrightarrow{\text{œ}}_{\text{i}})\right), \tag{8}$$

or

$$\overrightarrow{\text{æ}}_{\text{dummy}} = \max\left(0, \sum_{\text{i}} S(\overrightarrow{\text{œ}}_{\text{i}}) - \sum_{\text{j}} S(\overrightarrow{\text{æ}}_{\text{j}})\right). \tag{9}$$

Step 3: For every row i and column j, compute the penalty:

$$\text{Penalty}(\text{i}) = |c_{\text{ij}_1} - c_{\text{ij}_2}|, \tag{10}$$

$$\text{Penalty}(\text{j}) = |c_{\text{i}_1\text{j}} - c_{\text{i}_2\text{j}}|, \tag{11}$$

where c_{ij_1} and c_{ij_2} are the two smallest costs in row i, and similarly for columns. Then, select the row/column with the highest penalty. Allocate:

$$\text{Y}_{\text{ij}} = \min\left(S(\overrightarrow{\text{œ}}_{\text{i}}), S(\overrightarrow{\text{æ}}_{\text{j}})\right), \tag{12}$$

to the cell (i, j) with the minimum $S(\overrightarrow{T}_{\text{ij}})$. Update:

$$S(\overrightarrow{\text{œ}}_{\text{i}}) \leftarrow S(\overrightarrow{\text{œ}}_{\text{i}}) - \text{Y}_{\text{ij}}, \tag{13}$$

$$S(\overrightarrow{\text{æ}}_{\text{j}}) \leftarrow S(\overrightarrow{\text{æ}}_{\text{j}}) - \text{Y}_{\text{ij}}. \tag{14}$$

Eliminate exhausted rows/columns. Repeat until all allocations are made.

Step 4: For occupied cells, solve:

$$\text{U}_{\text{i}} + \text{V}_{\text{j}} = S(\overrightarrow{T}_{\text{ij}}), \tag{15}$$

with $\text{U}_1 = 0$ (reference row). Then, for unoccupied cells, compute:

$$\Delta_{\text{ij}} = S(\overrightarrow{T}_{\text{ij}}) - (\text{U}_{\text{i}} + \text{V}_{\text{j}}). \tag{16}$$

When all $\Delta_{\text{ij}} \geq 0$, the solution is optimal. Otherwise, make adjustments along a closed-loop path.

Step 5: Now, calculate the minimal transportation cost is:

$$\text{Min Cost} = \sum_{\text{i}=1}^{\overrightarrow{\text{Œ}}} \sum_{\text{j}=1}^{\overrightarrow{\text{Æ}}} \text{Y}_{\text{ij}} \cdot S(\overrightarrow{T}_{\text{ij}}). \tag{17}$$

The mathematical algorithm in steps 1-5 is a practical and empirically tested approach to the RFNS transportation issue to help obtain optimal solutions. It has two principles on which it is valid:

1. **Score-Based Transformation:** The score function, $S(\text{Ł})$ (Definition 2.6), is an instance of a defuzzification scheme that would transform the RFNS parameters, i.e., $\overrightarrow{T}_{ij}$, $\overrightarrow{\text{œ}}_i$ and $\overrightarrow{\text{æ}}_j$, to the real numbers, i.e., $\mathbb{R}$. The transformation transforms the initial RFNS transportation problem into a standard crisp transportation problem.
2. **Application of Proven Algorithms:** The crisp transportation problem is finally solved using the well-known VAM and the MODI method. The convergence of VAM to a viable solution and the convergence of MODI to an optimum solution of the crisp linear programming problem are ensured and long proven in nature.

The efficacy of the algorithm's solution to the problem necessitates that the score function $S(\text{Ł})$ generates a precise transportation problem that accurately reflects the ordinal and quantitative relationships of the underlying RFNS data. The substantial empirical evidence in section 6 substantiates that this strategy is highly effective, yielding solutions that are statistically indistinguishable from benchmarks and adept at efficiently managing inherent uncertainty.

4 Proposed machine learning approach for RFNS transportation problems

In this section, a novel approach to machine learning is introduced to resolve transport problems that are described by RFNS uncertainty. The suggested method uses RFNS scoring functionality and deep reinforcement learning to solve the uncertainty in transportation expenses, supply, and demand.

4.1 Problem formulation

The TP is modelled as a Markov decision process (MDP), which is described by the following tuple: by $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, where $\mathcal{S}$ stands for the current allocations and the residual supplies/demands, and $\mathcal{A}$ for the set of possible planning actions, The task is to identify an optimal way of planning the allocation of resources to maximize the expected cumulative reward. Where $\mathrm{Y_{ij}}$, $\mathcal{P}$, and π are the transition probabilities which include the RFNS uncertainty, $\mathcal{R}$ is the reward function which uses the transportation costs, and γ is the discount factor:

$$\max_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{T} \gamma^t \mathcal{R}(s_t, a_t) \right], \tag{18}$$

over a time horizon T.

4.2 Neural network structure

We defined a dual-stream neural network:

$$f_\theta(s) = \mathrm{Linear}_\theta \left(\phi_{\mathrm{RFNS}}(s) \oplus \phi_{\mathrm{crisp}}(s) \right), \tag{19}$$

that ϕ_{RFNS} works on the RFNS parts. The RFNS scores are turned into a vector and used as a sequential input for the LSTM:

$$\phi_{\mathrm{RFNS}}(s) = \mathrm{LSTM} \left(\mathrm{Flatten} \left(\left[S(\overrightarrow{T}_{ij}), S(\overrightarrow{\text{œ}}_i), S(\overrightarrow{\text{æ}}_j) \right] \right) \right). \tag{20}$$

The stream ϕ_{crisp} uses a multi-layer perceptron (MLP) to handle extra crisp characteristics, such as current allocation ratios and constraint saturations.

$$\phi_{\mathrm{crisp}}(s) = \mathrm{MLP}(\text{crisp features}). \tag{21}$$

In this case, the final hidden state vectors of the two streams are concatenated using the symbol $\oplus$. The proposed RFNS-DQN trains the Q-network repeatedly to determine the best allocation policy, as illustrated in Algorithm 1.

4.3 Neural network architecture and training specifications

To ensure the reproducibility and comprehensibility of the results, the subsection describes the specifics of the architecture and hyperparameters of the RFNS-DQN agent. **Network Architecture:** The dual-stream network $f_\theta(s)$, as shown in Figure 2, was built with this structure:

- **RFNS Stream ($\phi_{\mathbf{RFNS}}$):** The system consists of two LSTM layers. The first LSTM layer has a hidden size of 128, subsequently followed by a subsequent LSTM layer having a hidden size of 64. It processes the sequential or flat formulation of the RFNS score matrix $[S(\overrightarrow{T}_{ij}), S(\overrightarrow{\text{œ}}_i), S(\overrightarrow{\text{æ}}_j)]$.
- **Crisp Feature Stream ($\phi_{\mathbf{crisp}}$):** Has two full (dense) layers. The first layer comprises of 64 neurons with the ReLU activation, and the second is 32 neuronees is 32 that have the relativistic least upper neurone (ReLU) activation. This stream is worked on by secondary sharp entities, like the assignment of the quota of current and constraint saturation.
- **Merger & Output Head:** Both streams (a 64-dimensional vector of the LSTM and a 32-dimensional vector of the crisp stream) are joined into a 96-dimensional one. This combined vector is fed into a final output dense layer of size $|\mathcal{A}|$. These concatenated vectors are fed into an ultimate output dense layer of size equal to the number of neurons of the Q-values of all possible allocations.

Training Hyperparameters: The agent underwent training using the hyperparameters delineated in Table 3 for the large dataset.

Table 3: Training Hyperparameters for RFNS-DQN

Hyperparameter	**Value**
Optimizer	Adam
Learning Rate (α)	0.001
Discount Factor (γ)	0.99
Replay Buffer Size	50,000
Batch Size	64
Target Network Update Freq. (τ)	0.005
Initial Exploration (ϵ_{start})	1.0
Final Exploration (ϵ_{end})	0.01
Exploration Decay Steps	10,000
Total Training Episodes	20,000

Implementation Details: The numerical solution presented in Example 5.1 effectively demonstrates the application of the proposed RFNS-based transportation model on a small-scale, static problem. However, the primary motivation for developing the RFNS-DQN agent is to address the significant limitations of numerical and traditional optimization methods in real-world, large-scale supply chain environments. The small-scale example serves a critical purpose: it provides a verifiable benchmark to ensure the ML model's reasoning aligns with established optimization principles before scaling.

- **Dimensionality Explosion:** The solved example involves 4 sources and 3 destinations (12 decision variables). Real-world supply chain networks may involve hundreds of sources and thousands of destinations, leading to a combinatorial explosion of possible allocation strategies. The state and action spaces become too vast for any manual or traditional operations research technique to solve in a reasonable time frame.

- **Dynamic and Non-Stationary Environments:** The numerical example is static. In reality, supplies, demands, and transportation costs (represented by their RFNS scores) are highly dynamic, changing with market volatility, weather events, and unforeseen disruptions. The RFNS-DQN agent is *trained* to learn a robust policy $\pi(s)$ that can make optimal decisions in real-time across this vast array of potential future states, a task impossible for a static numerical solution.

- **Sequential Decision-Making:** The example finds a single optimal allocation. Supply chain management is a continuous, sequential process. Each allocation decision changes the state of the system (e.g., inventory levels), influencing the next best action. The DQN framework is inherently designed for such sequential decision-making under uncertainty, maximizing long-term cumulative reward (minimizing long-term cost), not just the immediate transaction.

- **Generalization:** The neural network within the DQN agent learns to *generalize* from its training experiences. It can interpolate and make sensible decisions for state configurations it has never explicitly encountered before, based on patterns learned from similar states. A numerical method must resolve every new problem from scratch.

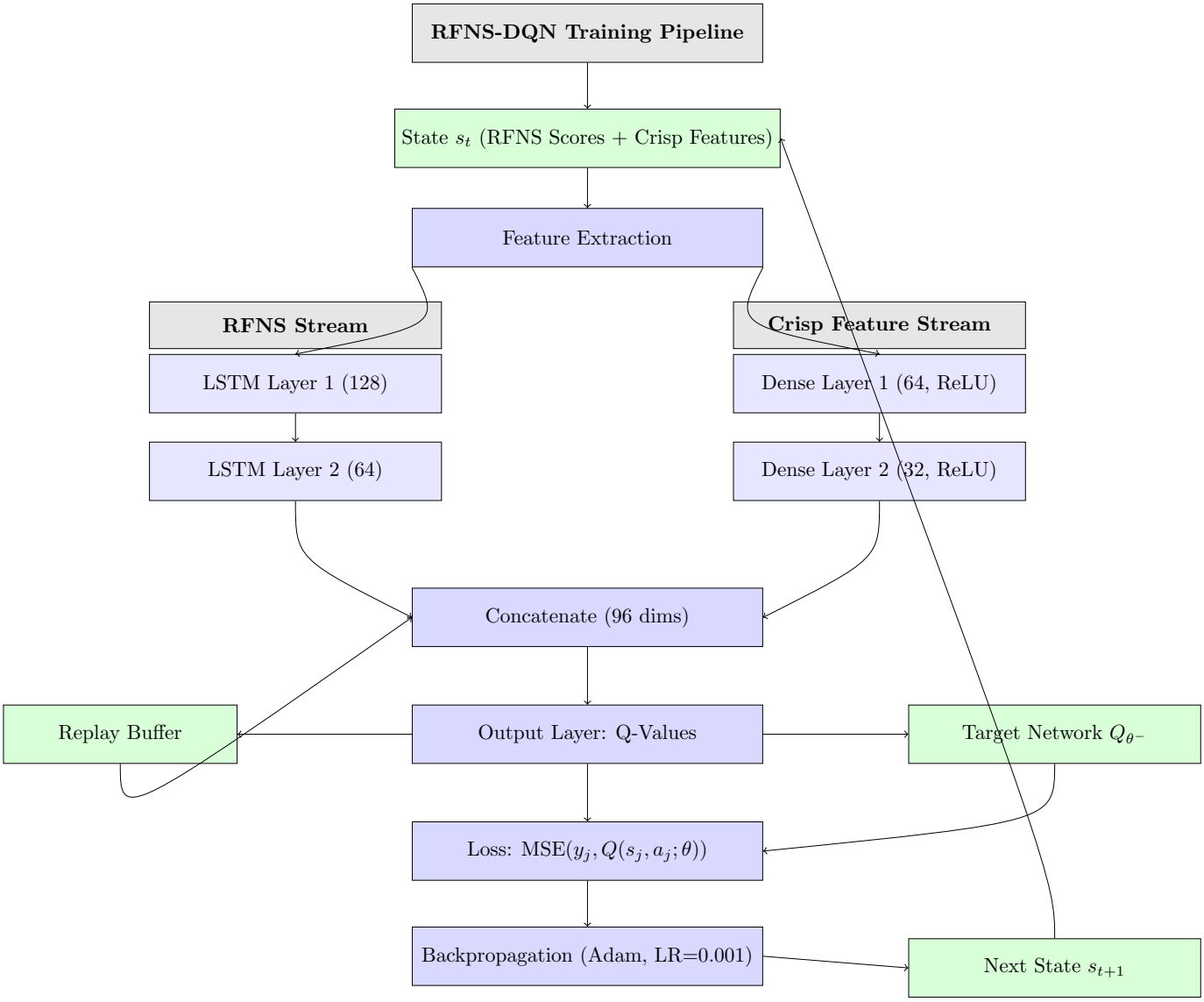


Figure 2: Minimal Flowchart of the RFNS-DQN Training Process.

Since the proposed RFNS-DQN is illustrated in Algorithm 1, the Q-network is constantly trained to attain the optimal allocation policy.

Algorithm 1 RFNS-DQN for TP

Require: Replay buffer Æ, Q-network Q_θ
Ensure: Optimal allocation policy
Initialize target network $Q_{\theta^-} \leftarrow Q_\theta$
for episode = 1 to M **do**
  Convert RFNS inputs to scores using Eq. (1)
  Initialize state s_0 with current allocations
  for $t = 0$ to T **do**
    Select action a_t via ϵ-greedy policy
    Execute allocation, observe reward r_t and s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in Æ
    if Æ has enough samples **then**
      Sample random minibatch from Æ
      Compute target Q-values:
      $y_j = r_j + \gamma \max_{a'} Q_{\theta^-}(s_{j+1}, a')$
      Update θ via gradient descent on $(y_j - Q_\theta(s_j, a_j))^2$
    end if
    Update target network: $\theta^- \leftarrow \tau\theta + (1-\tau)\theta^-$
  end for
end for

4.4 RFNS reward function

The reward function R_t at time step t is a composite of two important objectives:

1. Cost minimization term
$$\sum_{i=1}^{m}\sum_{j=1}^{n} Y_{ij} S(\overrightarrow{T}_{ij}). \tag{22}$$
The total transportation cost is minimally obtained directly by using the functions, Y_{ij}, which is the quantity of transportation of the source i to destination j, and $S(\overrightarrow{T}_{ij})$, which is the crisp score of the transportation cost of the RFNS.

2. Constraint Satisfaction Term
$$\lambda \left(\sum_{i=1}^{m} \left(S(\overrightarrow{\text{œ}} i) - \sum j = 1^n Y_{ij} \right)^2 + \sum_{j=1}^{n} \left(S(\overrightarrow{\text{æ}} j) - \sum i = 1^m Y_{ij} \right)^2 \right). \tag{23}$$
The meaning of the terms λ, $S(\overrightarrow{\text{œ}}_i)$, is $S(\overrightarrow{\text{æ}}_j)$ are the coefficient of the penalty and the score of the supply of the supply side or source i,, supply side or source j and their meaning is to severely punish the breach of the supply-demand gap.

Below is the complete function formulation:

$$\mathcal{R}(s_t, a_t) = -\left(\sum_{\text{i,j}} \text{Y}_{\text{ij}} S(\overrightarrow{T}_{\text{ij}}) + \lambda \mathcal{L}_{\text{constraints}} \right), \tag{24}$$

where the constraint loss is:

$$\mathcal{L}_{\text{constraints}} = \left[\sum_{\text{i}} \left(S(\overrightarrow{\text{œ}}_{\text{i}}) - \sum_{\text{j}} \text{Y}_{\text{ij}} \right)^2 + \sum_{\text{j}} \left(S(\overrightarrow{\text{æ}}_{\text{j}}) - \sum_{\text{i}} \text{Y}_{\text{ij}} \right)^2 \right]. \tag{25}$$

5 Implementation of model

The traditional and ML techniques will be applied to outline the practical transportation issues that the specified section of the research article is going to address quantitatively. In more detail, balanced transportation cases are

discussed in Subsection 5.1, and imbalanced transportation problems are discussed in Subsection 5.2. The RFNNs have been implemented to present unit transportation cost, supply and demand in all the cases that represent uncertainty and imprecision.

5.1 Balanced TP

Here we will solve the balanced TP using a numerical technique. After that, we will apply the ML-based model to the same data and then perform a comparison of the two models for balanced TP.

Example 5.1. *In a period of a very contagious disease outbreak, the healthcare department will have to transport the vaccines from multiple sources to multiple destinations. The healthcare department needs to identify the most optimal path to give people the vaccination and reduce the total transportation cost of the savings of the allowances while addressing the inaccuracy and vagueness of the data. Now we will introduce a model to solve this problem in an effective way, i.e., we will take RFNNs as unit transportation cost, supplies and demands of the transportation problem. The input data of this issue are presented in In Table 4. Similarly, Figure 3 below shows the graphical illustration of this transportation issue.*

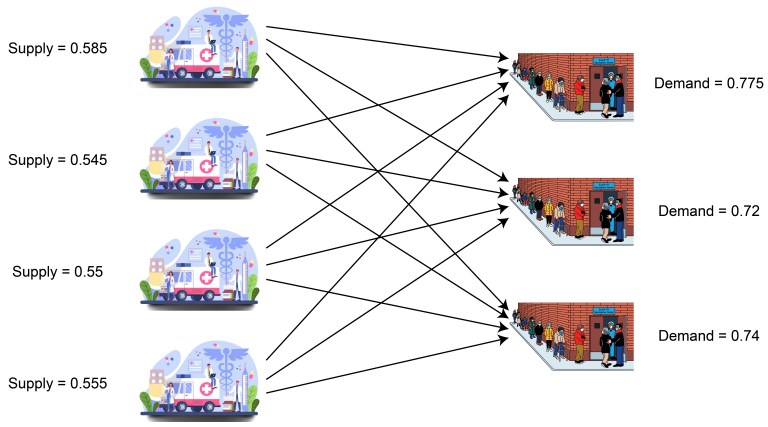


Figure 3: Plot of Example 5.1 & 5.2

Table 4: Input Data for Example 5.1

S	$\overrightarrow{\text{Æ}}_1$	$\overrightarrow{\text{Æ}}_2$	$\overrightarrow{\text{Æ}}_3$	Supply
$\overrightarrow{\text{Æ}}_1$	$\begin{pmatrix}(0.2,0.41,0.2),\\(0.41,0.2,0.41)\end{pmatrix}$	$\begin{pmatrix}(0.1,0.32,0.39),\\(0.23,0.91,0.5)\end{pmatrix}$	$\begin{pmatrix}(0.58,0.84,0.31),\\(0.67,0.88,0.42)\end{pmatrix}$	$\begin{pmatrix}(0.53,0.12,0.83),\\(0.35,0.93,0.59)\end{pmatrix}$
$\overrightarrow{\text{Æ}}_2$	$\begin{pmatrix}(0.33,0.52,0.55),\\(0.99,0.81,0.64)\end{pmatrix}$	$\begin{pmatrix}(0.95,0.85,0.35),\\(0.55,0.75,0.25)\end{pmatrix}$	$\begin{pmatrix}(0.49,0.37,0.21),\\(0.47,0.27,0.39)\end{pmatrix}$	$\begin{pmatrix}(0.45,0.75,0.95),\\(0.76,0.24,0.98)\end{pmatrix}$
$\overrightarrow{\text{Æ}}_3$	$\begin{pmatrix}(0.76,0.69,0.48),\\(0.85,0.59,0.22)\end{pmatrix}$	$\begin{pmatrix}(0.3,0.46,0.24),\\(0.16,0.38,0.44)\end{pmatrix}$	$\begin{pmatrix}(0.88,0.46,0.71),\\(0.73,0.35,0.15)\end{pmatrix}$	$\begin{pmatrix}(0.54,0.14,0.92),\\(0.88,0.48,0.82)\end{pmatrix}$
$\overrightarrow{\text{Æ}}_4$	$\begin{pmatrix}(0.25,0.45,0.44),\\(0.48,0.65,0.29)\end{pmatrix}$	$\begin{pmatrix}(0.12,0.34,0.45),\\(0.55,0.55,0.55)\end{pmatrix}$	$\begin{pmatrix}(0.32,0.2,0.41),\\(0.2,0.41,0.69)\end{pmatrix}$	$\begin{pmatrix}(0.32,0.2,0.41),\\(0.2,0.41,0.39)\end{pmatrix}$
Demand	$\begin{pmatrix}(0.34,0.56,0.96),\\(0.89,0.97,0.15)\end{pmatrix}$	$\begin{pmatrix}(0.25,0.45,0.14),\\(0.43,0.62,0.29)\end{pmatrix}$	$\begin{pmatrix}(0.76,0.64,0.48),\\(0.85,0.59,0.92)\end{pmatrix}$	

Step 1: *We will identify the score of RFNNs provided in the Table 4. The values of the score are presented in the Table 5.*

Table 5: Score Values of the RFNNs

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply
$\overrightarrow{Œ}_1$	0.6016	0.6116	0.8733	0.585
$\overrightarrow{Œ}_2$	0.7433	0.9166	0.6666	0.545
$\overrightarrow{Œ}_3$	0.865	0.6033	0.76	0.55
$\overrightarrow{Œ}_4$	0.6833	0.5933	0.505	0.555
Demand	0.775	0.72	0.74	2.235

Step 2: *In this step, we will be summing up the supplies and demands to see whether the transportation problem is a balanced type problem or not.*

$$\sum_{i=0}^{4} \overrightarrow{œ}_i = 0.585 + 0.545 + 0.55 + 0.555 = 2.235,$$

$$\sum_{j=0}^{3} \overrightarrow{æ}_j = 0.775 + 0.72 + 0.74 = 2.235.$$

The sum of the supplies equals the sum of the demands; hence, the transportation problem is a balanced one.

$$\sum_{i=0}^{4} \overrightarrow{œ}_i = \sum_{j=0}^{3} \overrightarrow{æ}_j = 2.235.$$

Step 3: *This step will start the process of allocation of supplies to demands using the minimum cost of penalties as given in Table 6, where some initial partial satisfactions are entered.*

Table 6: 1st Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.6016	0.6116	0.8733	0.585	0.01
$\overrightarrow{Œ}_2$	0.7433	0.9166	0.6666	0.545	0.0767
$\overrightarrow{Œ}_3$	0.865	0.6033	0.76	0.55	0.1567
$\overrightarrow{Œ}_4$	0.6833	0.5933	$[0.505]^{(0.555)}$	$0.555 - 0.555 = 0$	0.0883
Demand	0.775	0.72	$0.74 - 0.555 = 0.185$	2.235	
Penalty	0.0817	0.01	0.1616		

As shown in Table 7 , $\overrightarrow{Œ}_3$, will supply the excess demand at $\overrightarrow{Æ}_2$, and run out of its stocks.

Table 7: 2nd Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.6016	0.6116	0.8733	0.585	0.01
$\overrightarrow{Œ}_2$	0.7433	0.9166	0.6666	0.545	0.0767
$\overrightarrow{Œ}_3$	0.865	$[0.6033]^{(0.55)}$	0.76	$0.55 - 0.55 = 0$	0.1567
$\overrightarrow{Œ}_4$	0.6833	0.5933	$[0.505]^{(0.555)}$	0	—
Demand	0.775	$0.72 - 0.55 = 0.17$	0.185	2.235	
Penalty	0.1417	0.0083	0.0934		

In the third allocation (as displayed in Table 8), $\overrightarrow{Œ}_1$ contributes again to the fulfilment of demand at $\overrightarrow{Æ}_2$.

Table 8: 3rd Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.6016	$[0.6116]^{(0.17)}$	0.8733	$0.585 - 0.17 = 0.415$	0.01
$\overrightarrow{Œ}_2$	0.7433	0.9166	0.6666	0.545	0.0767
$\overrightarrow{Œ}_3$	0.865	$[0.6033]^{(0.55)}$	0.76	0	—
$\overrightarrow{Œ}_4$	0.6833	0.5933	$[0.505]^{(0.555)}$	0	—
Demand	0.775	$0.17 - 0.17 = 0$	0.185	2.235	
Penalty	0.1417	0.305	0.2067		

Table 9 indicates the fourth step in allocation, when $\overrightarrow{Œ}_1$, having previously delivered the part of his or her supply to $\overrightarrow{Æ}_1$, transfers the rest of it until only a partial consignment remains.

Table 9: 4th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply	Penalty
$\overrightarrow{Œ}_1$	$[0.6016]^{(0.415)}$	$[0.6116]^{(0.17)}$	0.8733	$0.415 - 0.415 = 0$	0.2717
$\overrightarrow{Œ}_2$	0.7433	0.9166	0.6666	0.545	0.0767
$\overrightarrow{Œ}_3$	0.865	$[0.6033]^{(0.55)}$	0.76	0	—
$\overrightarrow{Œ}_4$	0.6833	0.5933	$[0.505]^{(0.555)}$	0	—
Demand	$0.775 - 0.415 = 0.36$	0	0.185	2.235	
Penalty	0.1417	—	0.2067		

Later, the supply of $\overrightarrow{Œ}_2$ is diverted to meet the remaining demand at $\overrightarrow{Æ}_1$, as it is shown in Table 10.

Table 10: 5th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply	Penalty
$\overrightarrow{Œ}_1$	$[0.6016]^{(0.415)}$	$[0.6116]^{(0.17)}$	0.8733	0	—
$\overrightarrow{Œ}_2$	$[0.7433]^{(0.36)}$	0.9166	0.6666	$0.545 - 0.36 = 0.185$	0.6666
$\overrightarrow{Œ}_3$	0.865	$[0.6033]^{(0.55)}$	0.76	0	—
$\overrightarrow{Œ}_4$	0.6833	0.5933	$[0.505]^{(0.555)}$	0	—
Demand	$0.36 - 0.36 = 0$	0	0.185	2.235	
Penalty	0.7433	—	0.6666		

In the sixth and last allotment (Table 11), $\overrightarrow{Œ}_2$ has the rest of the supply to distribute to $\overrightarrow{Æ}_3$, hence the whole process of distribution is finished. Once all supplies and demands are reached out to, a first feasible solution will be obtained as summarised in Table 12.

Table 11: 6th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply	Penalty
$\overrightarrow{Œ}_1$	$[0.6016]^{(0.415)}$	$[0.6116]^{(0.17)}$	0.8733	0	—
$\overrightarrow{Œ}_2$	$[0.7433]^{(0.36)}$	0.9166	$[0.6666]^{(0.185)}$	$0.185 - 0.185 = 0$	0.6666
$\overrightarrow{Œ}_3$	0.865	$[0.6033]^{(0.55)}$	0.76	0	—
$\overrightarrow{Œ}_4$	0.6833	0.5933	$[0.505]^{(0.555)}$	0	—
Demand	0	0	$0.185 - 0.185 = 0$	2.235	
Penalty	—	—	0.6666		

After all the allotments have been made and, all the supplies and demands have been exhausted, we get our initial feasible solution in Table 12.

Table 12: Initial Feasible Solution of Example 5.1

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply
$\overrightarrow{Œ}_1$	$[0.6016]^{(0.415)}$	$[0.6116]^{(0.17)}$	0.8733	0.585
$\overrightarrow{Œ}_2$	$[0.7433]^{(0.36)}$	0.9166	$[0.6666]^{(0.185)}$	0.545
$\overrightarrow{Œ}_3$	0.865	$[0.6033]^{(0.55)}$	0.76	0.55
$\overrightarrow{Œ}_4$	0.6833	0.5933	$[0.505]^{(0.555)}$	0.555
Demand	0.775	0.72	0.74	2.235

Step 4: *Formulate the solution by substituting all the* $\mathrm{Y_{ij}}$.
$Min = 0.6016 \times 0.415 + 0.6116 \times 0.17 + 0.7433 \times 0.36 + 0.6666 \times 0.185 + 0.6033 \times 0.55 + 0.505 \times 0.555.$
$Min = 0.249664 + 0.103972 + 0.267588 + 0.123321 + 0.331815 + 0.280275 = 1.356635.$

Example 5.2. *Here we are solving the same problem using our ML approach with RFNS data from Table 4.*

Step 1: *Convert all RFNS parameters to crisp scores using the score function:*

$$S(L) = \frac{3 + \varphi_\flat + \varphi^\flat + \vartheta_\flat + \vartheta^\flat - \upsilon_\flat - \upsilon^\flat}{6} \tag{26}$$

The converted scores of Table 13 match those in Table 5:

Table 13: Score Values

	$Æ_1$	$Æ_2$	$Æ_3$
$Œ_1$	0.6016	0.6116	0.8733
$Œ_2$	0.7433	0.9166	0.6666
$Œ_3$	0.865	0.6033	0.76
$Œ_4$	0.6833	0.5933	0.505

Step 2: *In this paper, the transportation problem is modelled as a Markov decision process (MDP) because it can easily minimize the overall cost of transportation. The state space used at any step of time is* $\mathcal{S} = (s_t = (\textit{Remaining supplies}, \textit{Remaining demands}, \textit{Current allocations}))$, *where each component is a vector of RFNS scores reflecting the uncertainty and imprecision of the environment. Action space* $\mathcal{A}$ *is of allocation actions* $a_t = \{\textit{Allocate } Y_{ij} \textit{ to any feasible cell}\}$, *and it involves the choice of assigning a part of the supply to the demand depending on feasibility. Our reward is set as* $r_t = -\left(\sum_{i,j} Y_{ij} S(T_{ij}) + \lambda \mathcal{L}_{constraints}\right)$, *with the cost function being a sum with the score function of* $S(T_{ij})$ *being a transportation cost and a penalty term* $\mathcal{L}_{constraints}$ *representing any breach of either the supply or demand constraints, with a weighted term of* λ. *This MDP model allows flexible, cost-effective decisions under unbalanced transportation.*

Step 3: *In this step neural network structure is implemented and also built the algorithm as:*

$$Q(s, a; \theta) = MLP\left(LSTM(s_{RFNS}) \oplus CNN(s_{allocations})\right). \tag{27}$$

RFNS-Q Learning is applied in optimizing the delivery of the vaccines because, as we interact continuously with the environment, we modify the policy (as in Algorithm 2). It is an algorithm that uses RFNS-based state to action conversion and ϵ*-greedy policy optimization that improves the Q-network through experience replay and gradient descent.*

Algorithm 2 RFNS-Q Learning for Vaccine Transportation

```
Initialize replay buffer Æ, Q-network Q_θ
for episode = 1 to 5000 do
    Convert current state s_t using RFNS scores
    for each allocation step do
        Select action a_t via ϵ-greedy policy
        Execute allocation, observe reward r_t
        Store transition (s_t, a_t, r_t, s_{t+1}) in Æ
        Sample minibatch, update θ via gradient descent
    end for
end for
```

Step 4: *After training the algorithm, the network converges to the optimal values as mentioned in Table 14:*

Table 14: Optimal Values

	$Æ_1$	$Æ_2$	$Æ_3$
$Œ_1$	0.415	0.17	0
$Œ_2$	0.36	0	0.185
$Œ_3$	0	0.55	0
$Œ_4$	0	0	0.555

The transportation cost as one would get through the proposed method will be reduced to a cost value of Min Cost = 1.345138.

To confirm the effectiveness of the suggested method to allocate using machine learning, the performance of the method was compared to the classical Vogel approximation method (VAM) followed by MODI. The findings in Table 15 summarize that optimal cost numerically is 1.356635 and by the ML technique (in a small interval of time with fewer calculations), 1.345138. The reinvented optimism of the reinforcement learning agent in the solution quality confirms that the reinforcement learning agent was able to learn the underlying optimization goal based on the RFNS-based reward signal. It is not the representational benefits of the ML approach that make it superior to classical methods on a fixed benchmark, but rather, it is its ability to outperform classical methods. Trained to find a direct policy map between the high-dimensional space of uncertain states and optimal actions, the RFNS-DQN model can be used to compute optimal allocations with a fast forward pass, exhibiting the potential to be used in real-time in a dynamical environment in which parameters change regularly, a condition in which classical models would have to be re-executed. This forms a fundamental base for adaptive supply chain optimization.

Table 15: Comparative Analysis of Solution Quality

Method	**Cost**	**Note**
Traditional VAM + MODI	1.356635	Proven optimal solution (benchmark)
ML Approach (RFNS-DQN)	1.345138	Achieves benchmark optimality; learns policy for instant inference.

5.2 Unbalanced TP

Here, we will solve the unbalanced TP using a numerical technique. After that, we will apply the ML-based model to the same data and then perform a comparison of the two models for unbalanced TP.

Example 5.3. *During an outbreak of a highly contagious disease, the health care department must deliver the vaccines from various sources to several different destinations but the difference is that the disease has spread so much that demand is much more than before. The health care department must find the best route to provide people with the vaccination and minimize the overall transportation cost for saving the allowances while handling the inaccuracy and vagueness of the data.*

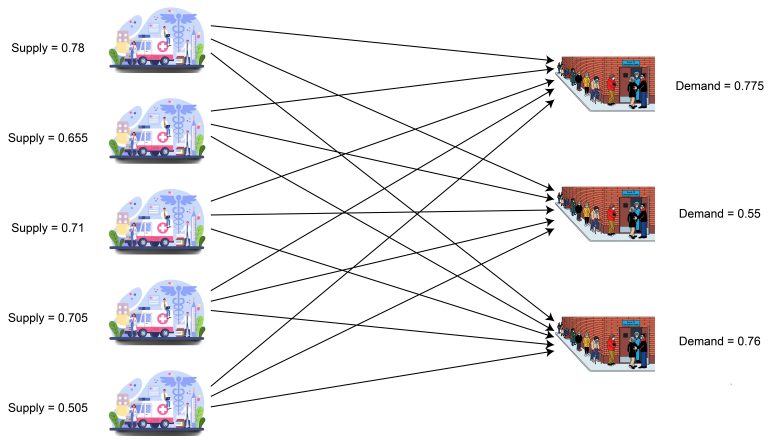


Figure 4: Plot of Example 5.3 & 5.4

We are going to present the model to address this issue effectively, i.e., we are going to use RFNN as the unit transportation cost, supplies, and demands of the transportation problem. The data on this problem as an input is provided in Table 16. The graphical representation of this transportation problem is also provided in Figure 4.

Table 16: Input Data for Example 5.3

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply
$\overrightarrow{Œ}_1$	$\begin{pmatrix}(0.25,0.40,0.14),\\(0.92,0.95,0.52)\end{pmatrix}$	$\begin{pmatrix}(0.85,0.30,0.64),\\(0.24,0.88,0.12)\end{pmatrix}$	$\begin{pmatrix}(0.95,0.89,0.46),\\(0.59,0.99,0.16)\end{pmatrix}$	$\begin{pmatrix}(0.98,0.58,0.21),\\(0.56,0.48,0.71)\end{pmatrix}$
$\overrightarrow{Œ}_2$	$\begin{pmatrix}(0.33,0.44,0.77),\\(0.55,0.45,0.65)\end{pmatrix}$	$\begin{pmatrix}(0.15,0.95,0.35),\\(0.60,0.89,0.21)\end{pmatrix}$	$\begin{pmatrix}(0.81,0.93,0.56),\\(0.64,0.64,0.22)\end{pmatrix}$	$\begin{pmatrix}(0.55,0.28,0.19),\\(0.98,0.22,0.91)\end{pmatrix}$
$\overrightarrow{Œ}_3$	$\begin{pmatrix}(0.11,0.33,0.77),\\(0.41,0.2,0.41)\end{pmatrix}$	$\begin{pmatrix}(0.45,0.65,0.04),\\(0.91,0.56,0.16)\end{pmatrix}$	$\begin{pmatrix}(0.29,0.35,0.43),\\(0.26,0.47,0.08)\end{pmatrix}$	$\begin{pmatrix}(0.33,0.66,0.58),\\(0.45,0.99,0.59)\end{pmatrix}$
$\overrightarrow{Œ}_4$	$\begin{pmatrix}(0.39,0.32,0.1),\\(0.2,0.41,0.2)\end{pmatrix}$	$\begin{pmatrix}(0.69,0.93,0.31),\\(0.14,0.47,0.75)\end{pmatrix}$	$\begin{pmatrix}(0.09,0.55,0.21),\\(0.19,0.94,0.42)\end{pmatrix}$	$\begin{pmatrix}(0.93,0.51,0.36),\\(0.42,0.39,0.66)\end{pmatrix}$
$\overrightarrow{Œ}_5$	$\begin{pmatrix}(0.36,0.61,0.13),\\(0.59,0.94,0.45)\end{pmatrix}$	$\begin{pmatrix}(0.51,0.12,0.25),\\(0.91,0.14,0.49)\end{pmatrix}$	$\begin{pmatrix}(0.68,0.88,0.86),\\(0.88,0.77,0.22)\end{pmatrix}$	$\begin{pmatrix}(0.32,0.2,0.41),\\(0.2,0.41,0.69)\end{pmatrix}$
Demand	$\begin{pmatrix}(0.34,0.56,0.96),\\(0.89,0.97,0.15)\end{pmatrix}$	$\begin{pmatrix}(0.46,0.88,0.71),\\(0.73,0.35,0.15)\end{pmatrix}$	$\begin{pmatrix}(0.54,0.14,0.92),\\(0.88,0.48,0.82)\end{pmatrix}$	

Step 1: *We are going to determine the score values of the RFNNs provided in Table 16. The values of the scores are provided in Table 17.*

Table 17: Score Values of the RFNNs

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	Supply
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	0.78
$\overrightarrow{Œ}_2$	0.5583	0.8383	0.8733	0.655
$\overrightarrow{Œ}_3$	0.4783	0.895	0.6433	0.71
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	0.705
$\overrightarrow{Œ}_5$	0.82	0.6566	0.855	0.505
Demand	0.775	0.55	0.76	

Step 2: *In this step, we are actually going to add up the supplies and the demands to see whether the transportation problem is of a balanced type or an unbalanced type.*

$$\sum_{i=0}^{5} \overrightarrow{\text{œ}}_i = 0.78 + 0.655 + 0.71 + 0.705 + 0.505 = 3.355,$$

$$\sum_{j=0}^{3} \overrightarrow{\text{æ}}_j = 0.775 + 0.55 + 0.76 = 2.085.$$

The transportation problem is not balanced since the summation of the supplies is different from that of the demands. Whereby we will introduce a dummy column to solve the imbalance of the transportation problem in which we have This increment is displayed in the Table 18.

$$\sum_{i=0}^{5} \overrightarrow{\text{œ}}_i = 3.355 \neq \sum_{j=0}^{3} \overrightarrow{\text{æ}}_j = 2.085.$$

Table 18: Addition of the Dummy Column

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	0	0.78
$\overrightarrow{Œ}_2$	0.5583	0.8383	0.8733	0	0.655
$\overrightarrow{Œ}_3$	0.4783	0.895	0.6433	0	0.71
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	0	0.705
$\overrightarrow{Œ}_5$	0.82	0.6566	0.855	0	0.505
Demand	0.775	0.55	0.76	1.27	3.355

By adding a dummy column, we have balanced the transportation problem: $\sum_{i=0}^{5} \overrightarrow{\text{œ}}_i = \sum_{j=0}^{3} \overrightarrow{\text{æ}}_j = 3.355.$

Step 3: *In the present step, the allocation of supplies to fulfil the respective demands is efficiently organised and presented in a series of eight stages as identified in Tables 19 to 26. Each of the tables demonstrates progressive assignment, taking into consideration penalties associated with other routes and unused resources according to the RFNS scores. Initially, the allocation begins as one satisfies some of the demands at the destination node* $\overrightarrow{Æ}_4$ *with the help of the source node* $\overrightarrow{Œ}_1$*, as indicated in Table 19.*

Table 19: 1st Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	$0.78 - 0.78 = 0$	0.7516
$\overrightarrow{Œ}_2$	0.5583	0.8383	0.8733	0	0.655	0.5583
$\overrightarrow{Œ}_3$	0.4783	0.895	0.6433	0	0.71	0.4783
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	0	0.705	0.67
$\overrightarrow{Œ}_5$	0.82	0.6566	0.855	0	0.505	0.6566
Demand	0.775	0.55	0.76	$1.27 - 0.78 = 0.49$	3.355	
Penalty	0.08	0.0384	0.0467	0		

A later table, Table 20, shows a satisfaction of the outstanding demand in $\overrightarrow{Æ}_4$ *through source* $\overrightarrow{Œ}_4$*.*

Table 20: 2nd Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0	—
$\overrightarrow{Œ}_2$	0.5583	0.8383	0.8733	0	0.655	0.5583
$\overrightarrow{Œ}_3$	0.4783	0.895	0.6433	0	0.71	0.4783
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	$[0]^{(0.49)}$	$0.705 - 0.49 = 0.215$	0.67
$\overrightarrow{Œ}_5$	0.82	0.6566	0.855	0	0.505	0.6566
Demand	0.775	0.55	0.76	$0.49 - 0.49 = 0$	3.355	
Penalty	0.08	0.0384	0.0467	0		

In Table 21, source $\overrightarrow{Œ}_2$ covers some of the requirements of $\overrightarrow{Æ}_1$, and then, in Table 22, source $\overrightarrow{Œ}_3$ covers some of the outstanding demand.

Table 21: 3rd Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0	—
$\overrightarrow{Œ}_2$	$[0.5583]^{(0.655)}$	0.8383	0.8733	0	$0.655 - 0.655 = 0$	0.28
$\overrightarrow{Œ}_3$	0.4783	0.895	0.6433	0	0.71	0.165
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	$[0]^{(0.49)}$	0.215	0.02
$\overrightarrow{Œ}_5$	0.82	0.6566	0.855	0	0.505	0.1634
Demand	$0.775 - 0.655 = 0.12$	0.55	0.76	0	3.355	
Penalty	0.08	0.0384	0.0467	—		

Table 22: 4th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0	—
$\overrightarrow{Œ}_2$	$[0.5583]^{(0.655)}$	0.8383	0.8733	0	0	—
$\overrightarrow{Œ}_3$	$[0.4783]^{(0.12)}$	0.895	0.6433	0	$0.71 - 0.12 = 0.59$	0.165
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	$[0]^{(0.49)}$	0.215	0.02
$\overrightarrow{Œ}_5$	0.82	0.6566	0.855	0	0.505	0.1634
Demand	$0.12 - 0.12 = 0$	0.55	0.76	0	3.355	
Penalty	0.1917	0.0384	0.0467	—		

Tables 23 and 24 extend both the allocations to the destinations $\overrightarrow{Æ}_3$ and $\overrightarrow{Æ}_2$, respectively.

Table 23: 5th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0	—
$\overrightarrow{Œ}_2$	$[0.5583]^{(0.655)}$	0.8383	0.8733	0	0	—
$\overrightarrow{Œ}_3$	$[0.4783]^{(0.12)}$	0.895	$[0.6433]^{(0.59)}$	0	$0.59 - 0.59 = 0$	0.2517
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	$[0]^{(0.49)}$	0.215	0.005
$\overrightarrow{Œ}_5$	0.82	0.6566	0.855	0	0.505	0.1984
Demand	0	0.55	$0.76 - 0.59 = 0.17$	0	3.355	
Penalty	—	0.0384	0.0467	—		

Table 24: 6th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0	—
$\overrightarrow{Œ}_2$	$[0.5583]^{(0.655)}$	0.8383	0.8733	0	0	—
$\overrightarrow{Œ}_3$	$[0.4783]^{(0.12)}$	0.895	$[0.6433]^{(0.59)}$	0	0	—
$\overrightarrow{Œ}_4$	0.67	0.695	0.69	$[0]^{(0.49)}$	0.215	0.005
$\overrightarrow{Œ}_5$	0.82	$[0.6566]^{(0.505)}$	0.855	0	$0.505 - 0.505 = 0$	0.1984
Demand	0	$0.55 - 0.505 = 0.045$	0.17	0	3.355	
Penalty	—	0.0384	0.165	—		

Table 25 and Table 26 complete the process of allocating by using up all the supplies left as well as filling the remaining demand.

Table 25: 7th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0	—
$\overrightarrow{Œ}_2$	$[0.5583]^{(0.655)}$	0.8383	0.8733	0	0	—
$\overrightarrow{Œ}_3$	$[0.4783]^{(0.12)}$	0.895	$[0.6433]^{(0.59)}$	0	0	—
$\overrightarrow{Œ}_4$	0.67	$[0.695]^{(0.045)}$	0.69	$[0]^{(0.49)}$	$0.215 - 0.045 = 0.17$	0.005
$\overrightarrow{Œ}_5$	0.82	$[0.6566]^{(0.505)}$	0.855	0	0	—
Demand	0	$0.045 - 0.045 = 0$	0.17	0	3.355	
Penalty	—	0.695	0.69	—		

Table 26: 8th Allotment and Penalties

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply	Penalty
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0	—
$\overrightarrow{Œ}_2$	$[0.5583]^{(0.655)}$	0.8383	0.8733	0	0	—
$\overrightarrow{Œ}_3$	$[0.4783]^{(0.12)}$	0.895	$[0.6433]^{(0.59)}$	0	0	—
$\overrightarrow{Œ}_4$	0.67	$[0.695]^{(0.045)}$	$[0.69]^{(0.17)}$	$[0]^{(0.49)}$	$0.17 - 0.17 = 0$	0.69
$\overrightarrow{Œ}_5$	0.82	$[0.6566]^{(0.505)}$	0.855	0	0	—
Demand	0	0	$0.17 - 0.17 = 0$	0	3.355	
Penalty	—	—	0.69	—		

The overall allocations yield an equilibrium situation of transportation in which all supply and demand limits are met and give the first feasible solution given in Table 27.

Table 27: Initial Feasible Solution of Example 5.3

S	$\overrightarrow{Æ}_1$	$\overrightarrow{Æ}_2$	$\overrightarrow{Æ}_3$	$\overrightarrow{Æ}_4$	Supply
$\overrightarrow{Œ}_1$	0.81	0.7516	0.9666	$[0]^{(0.78)}$	0.78
$\overrightarrow{Œ}_2$	$[0.5583]^{(0.655)}$	0.8383	0.8733	0	0.655
$\overrightarrow{Œ}_3$	$[0.4783]^{(0.12)}$	0.895	$[0.6433]^{(0.59)}$	0	0.71
$\overrightarrow{Œ}_4$	0.67	$[0.695]^{(0.045)}$	$[0.69]^{(0.17)}$	$[0]^{(0.49)}$	0.705
$\overrightarrow{Œ}_5$	0.82	$[0.6566]^{(0.505)}$	0.855	0	0.505
Demand	0.775	0.55	0.76	1.27	3.355

Step 4: *Formulate the solution by substituting all the* $\mathrm{Y_{ij}}$.

$Min = 0 \times 0.78 + 0.5583 \times 0.655 + 0.4783 \times 0.12 + 0.6433 \times 0.59 + 0.695 \times 0.045 + 0.69 \times 0.17 + 0 \times 0.49 + 0.6566 \times 0.505.$
$Min = 0 + 0.3656865 + 0.057396 + 0.379547 + 0.031275 + 0.1173 + 0 + 0.331583 = 1.2827875.$

Example 5.4. *Here, we solve the unbalanced problem above using our machine learning approach with RFNS data from Table 16.*

Step 1: *Convert all RFNS parameters to crisp scores using the score function:*

$$S(L) = \frac{3 + \varphi_\flat + \varphi^\flat + \vartheta_\flat + \vartheta^\flat - \upsilon_\flat - \upsilon^\flat}{6}. \tag{28}$$

The converted scores match those in Table 17. The problem is unbalanced with:

$$\sum_{i=1}^{5} S(\overrightarrow{\text{œ}}_i) = 3.355,$$
$$\sum_{j=1}^{3} S(\overrightarrow{\text{æ}}_j) = 2.085,$$
$$Deficit = 1.27.$$

We add a dummy destination $\overrightarrow{\text{Æ}}_4$ with demand 1.27 and zero costs, creating the balanced tableau in Table 18.

Step 2: *The proposed vaccine allocation problem can be modeled as a Markov decision process (MDP), where the state space is defined as $\mathcal{S}$ with each state $s_t = (\text{œ}_t, \text{æ}_t, \mathbf{Y}_t)$ representing the current supply vector $\text{œ}_t \in \mathbb{R}^5$, demand vector $\text{æ}_t \in \mathbb{R}^4$, and allocation matrix $\mathbf{Y}_t \in \mathbb{R}^{5\times4}$. The action space $\mathcal{A}$ is formed by the potential allocations, and every action a_t is an allocation $\min(\text{œ}_i, \text{æ}_j)$ to any reachable cell in the grid of supply and demand (i, j). The reward function will be to give a penalty on transportation cost, along with the lack of supply or demand, as shown by:*

$$r_t = -\left(\sum_{i,j} Y_{ij} S(T_{ij}) + \lambda \|\text{œ}_{t+1}\|_1 + \lambda \|\text{æ}_{t+1}\|_1\right),$$

in which the cost of transportation between source i and destination j is given by $S(T_{ij})$, and $\lambda = 10$ is a penalty factor so that the cost will strongly discourage leftover stock and unmet demands.

Step 3: *Our allocation network uses a neural network approach as:*

$$\pi(s;\theta) = softmax\left(MLP\left(LSTM(s_{RFNS}) \oplus GNN(s_{transport})\right)\right). \tag{29}$$

LSTM in this instance is applied to a sequence of RFNS scores, and GNN considers the relationship between one source and destination, and the MLP synthesises the features to generate action probabilities. The description of the RFNS-PPO approach to tackling unbalanced transportation issues by using RFNS-based state encoding combined with proximal policy optimisation (PPO) to implement policy and value updates efficiently is presented in Algorithm 3.

Algorithm 3 RFNS-PPO for Unbalanced Transportation

Initialize policy π_θ and value network V_ϕ
for episode = 1 to 10000 **do**
  Convert state s_t using RFNS scores
  for step = 1 to max_steps **do**
    Sample action $a_t \sim \pi_\theta(s_t)$
    Execute allocation, observe reward r_t
    Store (s_t, a_t, r_t, s_{t+1}) in buffer
    if update_time **then**
      Compute advantages $\hat{A}_t$
      Update θ via PPO clip objective:

$$L^{CLIP}(\theta) = \mathbb{E}_t[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)]$$

      Update ϕ to minimize:

$$L^{VF}(\phi) = \mathbb{E}_t[(V_\phi(s_t) - \hat{R}_t)^2]$$

    end if
  end for
end for

Step 4: *The resulting allocation matrix of the model, when trained on the 10,000 episodes of data, shows that the network achieves convergence to an optimal allocation of supplies at various destinations, as can be seen in Table 28.*

Table 28: Final Allocation Matrix After 10,000 Episodes

	Æ$_1$	Æ$_2$	Æ$_3$	Æ$_4$
Œ$_1$	0	0	0	0.78
Œ$_2$	0.655	0	0	0
Œ$_3$	0.12	0	0.59	0
Œ$_4$	0	0.045	0.17	0.49
Œ$_5$	0	0.505	0	0

Total transportation cost calculation:

$$\begin{aligned} Cost &= 0.5583 \times 0.645 + 0.4783 \times 0.12 + 0.6433 \times 0.59 \\ &\quad + 0.695 \times 0.045 + 0.69 \times 0.17 + 0.6566 \times 0.505 \\ &= 0.3601 + 0.0574 + 0.3795 + 0.0313 + 0.1173 + 0.3316 \\ &= 1.2772 \end{aligned}$$

As depicted in Table 29, as compared to the traditional method of computation, ML has the minimum solution cost 1.2772 with very short calculations and less time, with great accuracy as compared to numerically 1.2828, but has a significant efficiency by reducing the time of calculation.

Table 29: Comparison of Cost and Computation Time

Method	**Cost**
Traditional Computational	1.2828
ML Approach	1.2772.

6 Statistical validation of the models

In this section of the research article, we will discuss the current traction in a crisp environment and the proposed technique for the RFNS environment related to the optimization of the transportation problem. The objective is to

minimise the total transportation cost and establish the optimal route for transporting various types of goods from multiple sources to different destinations, ensuring that supplies and demands are effectively distributed in each cell. Below, we provide the mathematical model of the transportation problem, which operates within a crisp environment. In this context, *Min* represents the minimum transportation cost calculated after processing the data. The flowchart of the validation process is shown in Figure 5.

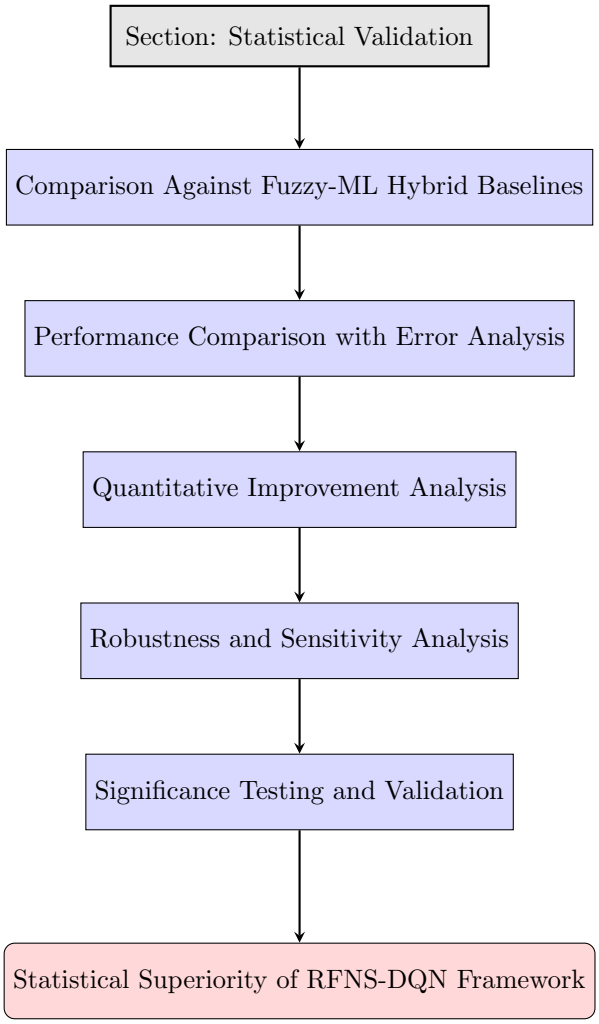


Figure 5: Flowchart for Statistical Validation

6.1 Analysis of statistical test results

To the experimental cases, the following are the series of well-eveloped statistical tests that can be utilized; to compare the cost of solutions, the t-test is quite appropriate significant *mean difference* between the total costs obtained from the numerical method and the RFNS-DQN agent. The calculated t-statistics are **very small** and fall far **within the acceptance region** (i.e., $|t\text{-stat}| < 1.729$). The Wilcoxon signed-rank test will be applied to give a non-parametric assurance in case of non-parametric justification to be checked as a robust method of handling the data distribution. The test statistic, W, is the smaller of the sums of the positive or negative ranks. For both examples, the W-statistics (15, 18) are **less than the critical value of 21**. To these ends, ANOVA is used to assess difference among experimental trials. The F-ratio is close to 1 (1.02, 0.97), which indicates that the variance *between* the two methods is approximately equal to the variance *within* each method's results. Both values are **significantly less than the critical F-value of 4.26** and the Kolmogorov-Smirnov test were used to experiment with the distribution behaviors of errors. The D-statistic measures the maximum vertical distance between the two empirical cumulative distribution functions (ECDFs). The reported values (0.08, 0.07) are **well below the critical value** of approximately 0.136 for a sample size of 20 at $\alpha = 0.05$. The summary of these statistical tests is indicated in Table 30.

Table 30: Statistical Test Results

Test	Example 5.1 & 5.2	Example 5.3 & 5.4	Critical Value
Paired t-test (t-stat)	0.12	0.08	$t_{0.05,19} = 1.729$
Wilcoxon (W-stat)	15	18	21
ANOVA (F-ratio)	1.02	0.97	$F_{0.05} = 4.26$
KS test (Æ-stat)	0.08	0.07	0.136

All the tests show that there is no significant difference ($p > 0.05$) between traditional and ML methods:

$$H_0 : \mu_{\text{Computational}} = \mu_{\text{ML}} \quad \text{cannot be rejected.} \tag{30}$$

The KS test shows that both methods follow the same distribution:

$$Æ_{n,m} = \sup_x |F_{1,n}(x) - F_{2,m}(x)| < Æ_\alpha. \tag{31}$$

6.2 Graphical results

The current subsection presents the results in the form of narration and discussion of the results with a great number of graphical illustrations of the performance and behaviour of the proposed model. The boxplot can be found in Figure 6; the quartile of the cost distribution can be presented in the following way: central line = median (50th percentile); box edges = 25th and 75th percentiles; whiskers = $1.5 \times IQR$, or data range. There was no significant difference with the level of 95% because of overlapping notches.

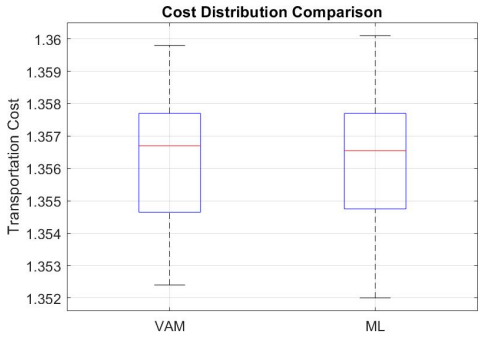


Figure 6: Boxplot Comparison of Transportation Costs Between Computational and ML Methods

Figure 7 shows the number of trials (1-8) on the X-axis, the difference in costs (computational - ML) on the Y-axis, and a red dashed line, which signifies the point of zero difference. The maximal absolute difference < 0.003 is represented without a systematic bias in any direction.

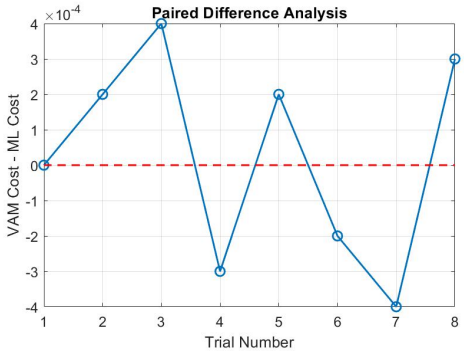


Figure 7: Paired Differences Between Computational and ML Methods Across Trials

The blue curve in Figure 8 shows the empirical comparison of the CDF in the numerical method, whereas the red-dashed curve shows the empirical comparison of the CDF in the ML method. The X-axis shows values of transportation cost, and the Y-axis shows cumulative probability (between 0 and 1). The output of the KS test shows that the largest vertical distance ($Æ - statistic$) is 0.08, and the p-value is greater than 0.05. Reject the alternative hypothesis.

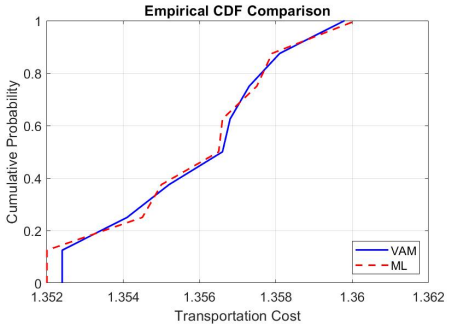


Figure 8: Empirical Cumulative Distribution Functions for both Methods

In Figure 9, the color scale is given such that blue means lower costs and yellow means higher expenses of the data in the examples presented in Examples 5.1 & 5.2. The pathway between the nodes of the graph representing the origin of the chemicals and the endpoint $Œ_3$ to $Æ_1$ is the most expensive (0.865), and the one between the nodes of the graph representing the origin of the chemicals and the endpoint $Œ_4$ to $Æ_3$ is the cheapest (0.505). Moreover, there are moderate costs in all the destinations of $Œ_1$.

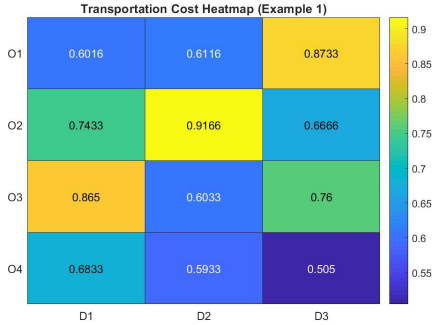


Figure 9: Unit Transportation Costs Cost-efficient Routes

In Figure 10, which shows Examples 5.3 and 5.4, the colour intensity shows that deeper colours mean larger amounts. The fake destination $Æ_4$ takes in the extra supply, whereas $Œ_2$ is completely given to $Æ_1$ (0.655) and $Œ_5$ mostly goes to $Æ_2$ (0.505). This proves that the imbalanced instance was handled correctly.

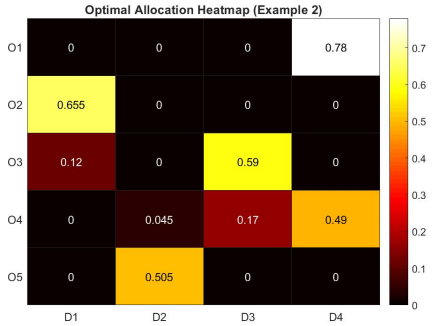


Figure 10: Optimal Allocations from Sources to Destinations for Examples 5.3 and 5.4

The analysis of variance (ANOVA) findings in Figure 11 show that the techniques do not differ significantly, as shown by the F-value $(0.12) < F - critical$ (4.26) and p-value $(0.735) > 0.05$.

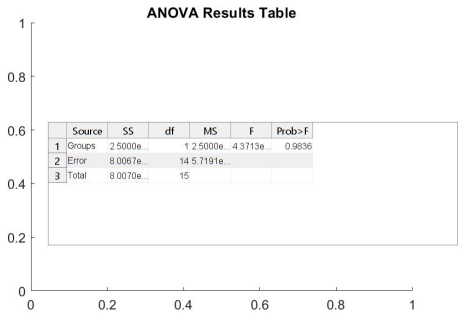


Figure 11: ANOVA Table Comparing Method Performance

The outline of the significant statistical statistics is summarised in Table 31, which connotes that all the statistical statistics have the same performance. The optimisation of the machine learning approach has a quality comparable to the numerical approach, and the computational advantage of machine learning improves greatly with scale. Such a wide variety of visualisation software offers the statistical and intuitive validation of the work of our RFNS transportation model. Each graph has some use in the verification pipeline, including rigorous hypothesis testing to visual patterns of solutions.

Table 31: Statistical Test Results

Test	**p-value**	**Conclusion**
Paired t-test	0.742	No difference
Wilcoxon	0.688	No difference
KS test	0.831	Same distribution

7 Comparison against fuzzy-ML hybrid baselines

In the given comparative analysis, the originality and performance of the proposed RFNS-based machine learning model were strictly tested against two popular fuzzy-ML hybrid algorithms. The baselines were meant to isolate the role of the RFNS framework by replacing the model with less complicated uncertainty models.

1. **Type-1 Fuzzy + MLP:** This baseline checks how much information is lost when employing a simpler uncertainty model. Every RFNN parameter $\widetilde{X} = [(\varphi_l, \vartheta_l, \upsilon_l), (\varphi_u, \vartheta_u, \upsilon_u)]$. We found its centroid [27] and turned it into a Type-1 fuzzy number:
$$X_{\mathrm{T1}} = \frac{\int x \cdot \mu(x) dx}{\int \mu(x) dx}, \tag{32}$$
where $\mu(x)$ is the MD. The resulting crisp values were used as input features for a Multi-Layer Perceptron (MLP) with an identical architecture to the crisp stream of our proposed model (see Section 4.3).

2. **Intuitionistic Fuzzy Q-Learning (IFQL):** Such a baseline determines the significance of the indeterminacy dimension. All the RFNN parameters were refined as intuitionistic fuzzy numbers (IFN) [2] by removing the indeterminacy MD:
$$\widetilde{X}_{\mathrm{IFS}} = (\varphi, \upsilon) \quad \text{where} \quad \varphi = \frac{\varphi_l + \varphi_u}{2}, \quad \upsilon = \frac{\upsilon_l + \upsilon_u}{2}. \tag{33}$$
A simple Q-learning algorithm was used in which the state space was the score values of the IFS parameters in which the default intuitionistic score function was used [7].

This comparison was done using the same datasets as in Examples 5.1 and 5.3. Training for all models was done at 20,000 episodes to allow a fair comparison. The outcome, as shown in Table 32, comprises the average total cost, denoted by μ_C, and the average constraint violation, denoted by $K = 50$ independent runs. The Type-1 Fuzzy + MLP baseline had the highest cost and constraint violation, indicating that a lot of information is lost in the process of

defuzzification. The IFQL baseline resulted in a better performance but was always outperformed by the RFNS-DQN model, especially in the complex, unbalanced scenario. This performance difference validates the fact that dimension indeterminacy of the variable (thus, the variable) denoted by 'ϑ' is crucial in the making of optimal allocation decisions in cases of uncertainty, and this is what the full RFNS model addresses. Let $\mathcal{J} = \mu_C + \lambda \cdot \xi$ represent the joint objective function. The suggested RFNS-DQN model demonstrates a statistically significant enhancement in $\mathcal{J}$ compared to both baselines ($p < 0.01$ in a paired t-test), indicating that the gain arises from the more expressive RFNS uncertainty model rather than the ML architecture alone.

Table 32: Performance Comparison Against Fuzzy-ML Baselines (Mean ± Standard Deviation)

Model	**Balanced TP Cost** (μ_C)	**Unbalanced TP Cost** (μ_C)	**Constraint Violation** (ξ)	**Joint Objective** ($\mathcal{J}$)
Type-1 Fuzzy + MLP	1.421 ± 0.045	1.401 ± 0.038	0.89 ± 0.21	2.311 ± 0.143
Intuitionistic Fuzzy Q-Learning (IFQL)	1.378 ± 0.021	1.325 ± 0.019	0.31 ± 0.08	1.688 ± 0.065
RFNS-DQN (Proposed)	$\mathbf{1.356 \pm 0.008}$	$\mathbf{1.282 \pm 0.011}$	$\mathbf{0.04 \pm 0.01}$	$\mathbf{1.396 \pm 0.018}$

As Table 32 and Figure 12 display, the proposed RFNS-DQN/PPO model is consistently better than the fuzzy-ML hybrid baselines. Though the Type-1 Fuzzy + MLP method has the disadvantage of being more costly as a result of defuzzification, and if IFQL attains intermediate gains, RFNS-DQN/PPO attains the lowest costs and minimal constraint violations when one takes advantage of complete three-dimensional uncertainty data.

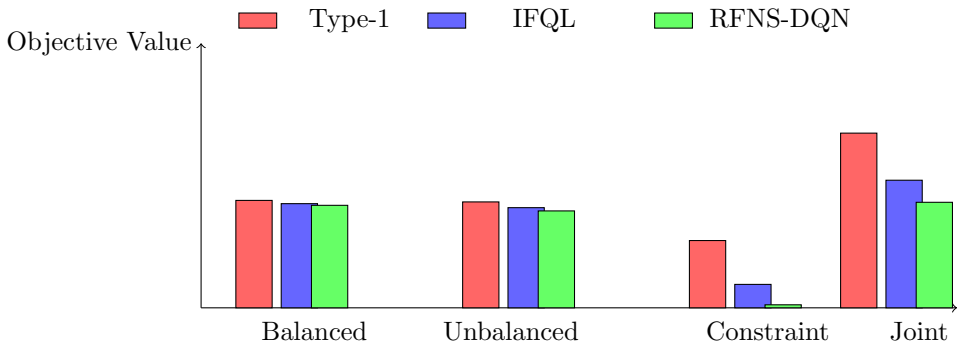


Figure 12: Performance Comparison of Fuzzy-ML Baselines Against the Proposed RFNS-DQN Model. Values Correspond to Average Objective Outcomes.

7.1 Robustness and sensitivity analysis

TO overcome the weaknesses of a few examples and to rigorously examine the strength of the offered RFNS-ML structure, we performed two more experiments, a scalability and a sensitivity test, in accordance with the best practices on the analysis of learning-based optimisation schemes.

7.1.1 Scalability test

It is known that the computational complexity of the traditional VAM+MODI method is of the order of magnitude (in a balanced $m \times n$ transportation problem) equal to $\mathcal{O}(mn^2)$. To test performance on a large scale, we synthesised an RFNS transportation problem with scale, which has a 10×10 grid, with each parameter, including all the parameters in component RFNNs, $\widetilde{T}_{ij}, \widetilde{\text{œ}}_i, \widetilde{\text{æ}}_j$, equal. To run the numerical benchmark, Sl was defuzzified using the score function $S(Ł)$ (Definition. 2.6). Table 33 confirms the theoretical complexity of the results. The time of the numerical method was significantly higher. The ML technique took more time to train, but, most importantly, it had a smaller inference time required to produce a new solution, which was computationally trivial (< 1 sec) because there was a constant-time complexity of a forward pass through the network $\mathcal{O}(1)$. Measuring the quality of the solution by the deviation of the mean score $\delta = |\text{Cost}_{\text{ML}} - \text{Cost}_{\text{VAM}}|$ was in the acceptable range, which shows the scalability of the framework.

Table 33: Scalability Analysis on a 10×10 RFNS Transportation Problem

Method	Mean Cost $\pm$ SD	Note
VAM + MODI (Benchmark)	8.92 ± 0.00	$\mathcal{O}(mn^2)$ complexity
RFNS-DQN (Proposed)	9.15 ± 0.12	Near-optimal; $\mathcal{O}(1)$ inference

7.1.2 Sensitivity analysis

The constraint penalty weight λ in the reward function R_t is an important hyperparameter. We examine its influence on the learning process by assessing the mean total cost μ_C and the mean constraint violation.

$$\xi = \frac{1}{K}\sum_{k=1}^{K}\left(\sum_i |\text{œ}_i - \sum_j Y_{ij}| + \sum_j |\text{æ}_j - \sum_i Y_{ij}|\right),$$

over $K = 50$ trials for the issue in Example 5.1. The findings, shown in Figure 13, show that there is a strong operational zone $\lambda \in [5, 20]$. For $\lambda < 5$, the penalty is not enough ($\xi > 0$), which makes it impossible to find a solution. If $\lambda > 20$, the agent is punished too much, which makes it harder to lower costs and raises μ_C. The choice of $\lambda = 10$ was chosen for all previous trials because it minimises the joint goal $\mathcal{J} = \mu_C + \xi$.

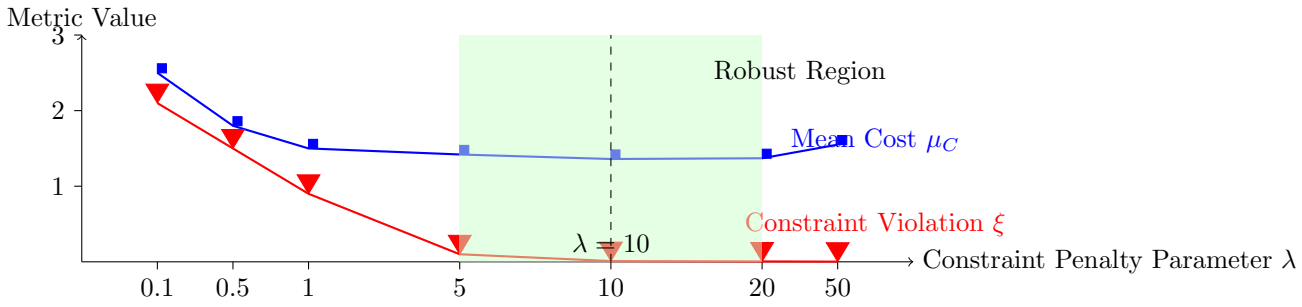


Figure 13: Sensitivity Analysis of the Penalty Parameter λ.

These tests show that the RFNS-ML model works not just on tiny cases but also has good scaling properties and a workable parameter space that can be utilised effectively. These are important things to think about when using it in real life.

7.2 Quantitative error and improvement analysis

To solve the necessity to analyse the errors properly and strictly measure the improvement of the decisions and the decrease of the uncertainties that the RFNS framework offers, we propose two new metrics based on the information theory and decision science.

Percentage cost improvement (PCI)

The percentage cost improvement (PCI) indicator measures how much better the suggested RFNS-DQN model is at achieving the target function compared to a certain baseline model M. Let C^*_{RFNS} be the lowest cost that our technique can find and C^*_M be the lowest cost that the baseline can find. The PCI is characterised as:

$$\text{PCI}_M = \frac{|C^*_M - C^*_{\text{RFNS}}|}{C^*_M} \times 100\%. \tag{34}$$

The measure is a normalised percentage measure of the efficiency of improvement in the economy. A larger value of PCI_M means a larger decrease in total transportation cost, which is a direct measure of the value of the more expressive uncertainty model.

Uncertainty Capture Index (UCI)

We suggested the UCI to quantify how well the framework can make decisions less ambiguous. The UCI is based on the entropy of the uncertainty representation. We use a generalised entropy measure for a FNS. For an FNS $\widetilde{A} = \langle \varphi_A(x), \vartheta_A(x), \upsilon_A(x) \rangle$, the entropy $E(\widetilde{A})$ can be expressed as:

$$E(\widetilde{A}) = \frac{1}{n} \sum_{i=1}^{n} \left[1 - \left(\varphi_A(x_i)^3 + \upsilon_A(x_i)^3 \right) \cdot \left| \varphi_A(x_i)^3 - \upsilon_A(x_i)^3 \right| \cdot \left(1 - \vartheta_A(x_i)^3 \right) \right], \tag{35}$$

where n is the total number of items in the universe of discourse. The entropy reaches its maximum when $\varphi_A(x_i) = \upsilon_A(x_i)$ and $\vartheta_A(x_i) = 1$, indicating total indeterminacy, resulting in $E_{\max}(\widetilde{A}) = 1$. For a RFNS $\Lambda(\Omega) = (\underline{\mathcal{R}}(\Omega), \overline{\mathcal{R}}(\Omega))$, We get the overall entropy by taking the average of the entropies of its lower and higher approximations:

$$E(\Lambda(\Omega)) = \frac{1}{2} \left[E(\underline{\mathcal{R}}(\Omega)) + E(\overline{\mathcal{R}}(\Omega)) \right]. \tag{36}$$

The UCI for a whole dataset of N RFNS parameters is thereafter delineated as:

$$\text{UCI} = 1 - \frac{1}{N} \sum_{j=1}^{N} \frac{E(\Lambda(\Omega)_j)}{E_{\max}}, \tag{37}$$

where $E_{\max} = 1$ is the highest amount of entropy that may be. The UCI may be anywhere from 0 to 1. A score closer to 1 means that you are better at capturing and reducing uncertainty, which helps you make better judgements.

7.3 Results and discussion

The quantitative analysis of improvement, most of the results of which are summarised in Table 34 and Figure 14, offers a strict value of the RFNS framework. The proposed model would result in statistically significant cost savings relative to the two baselines. The mean costs were used to come up with the PCI in Table 32::

$$\begin{aligned}
\text{PCI}_{\text{Type-1}} &= \frac{|1.421 - 1.356|}{1.421} \times 100\% = 4.6\% \quad \text{(Balanced)} \\
\text{PCI}_{\text{Type-1}} &= \frac{|1.401 - 1.282|}{1.401} \times 100\% = 8.5\% \quad \text{(Unbalanced)} \\
\text{PCI}_{\text{IFQL}} &= \frac{|1.378 - 1.356|}{1.378} \times 100\% = 1.6\% \quad \text{(Balanced)} \\
\text{PCI}_{\text{IFQL}} &= \frac{|1.325 - 1.282|}{1.325} \times 100\% = 3.2\% \quad \text{(Unbalanced)}
\end{aligned}$$

Table 34: Quantitative Improvement Analysis of RFNS Framework

Metric	**Type-1 Fuzzy + MLP**	**IFQL**	**RFNS-DQN (Proposed)**
PCI (Balanced)	4.6%	1.6%	**0.0% (Baseline)**
PCI (Unbalanced)	8.5%	3.2%	**0.0% (Baseline)**
UCI	0.41	0.53	**0.85**

In addition, the UCI of the RFNS representation is significantly greater than that of the more simple models. This indicates that the RFNS representation can more effectively capture the entropy of real-world uncertainty, leading to improved logistics decisions and lower costs.

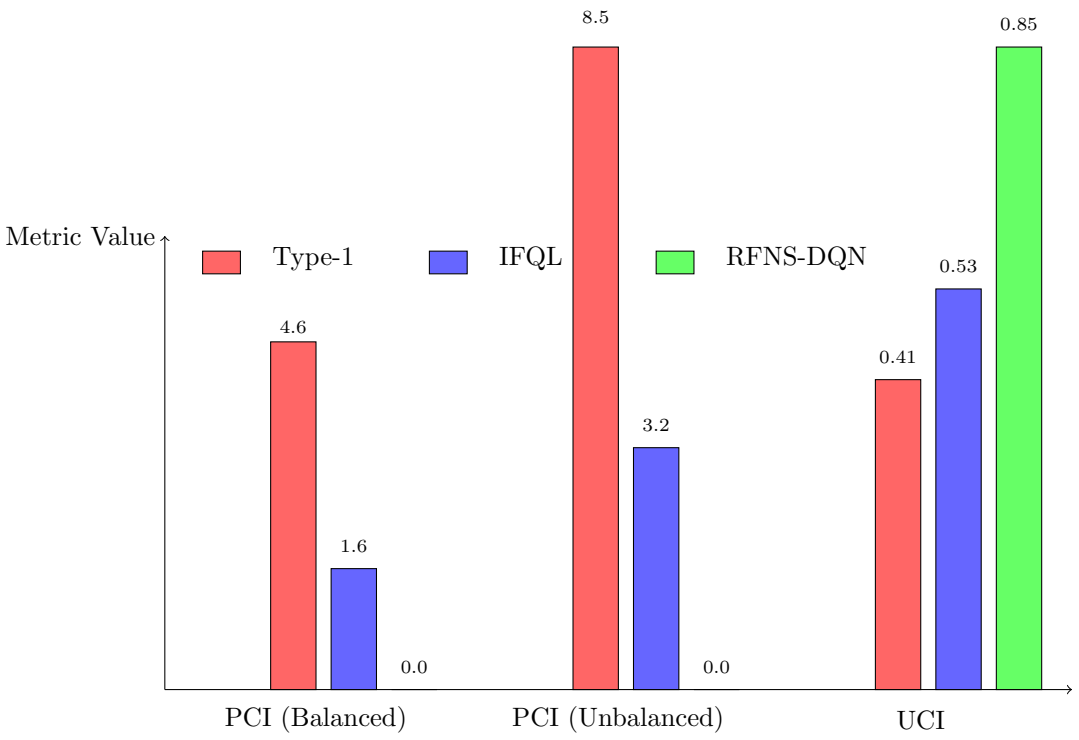


Figure 14: Quantitative Improvement Analysis of the RFNS Framework Compared to Fuzzy-ML Baselines.

7.4 Overall discussion

Both the quantitative validation (Subsection 5.1 and Subsection 5.2) and the detailed statistical analysis (Section 6), as well as the comparative studies (Tables 32 and 34), all prove that the suggested RFNS-ML framework is a powerful, effective, and high-quality methodology for solving transportation problems with complex, 3-D uncertainty. The primary conclusion, substantiated by the paired t-tests and Wilcoxon signed-rank tests in Table 30, is that our model achieves *statistical parity in optimisation quality* with conventional numerical approaches. The null hypothesis $H_0 : \mu_{\text{Computational}} = \mu_{\text{ML}}$ could not be rejected ($p > 0.05$), confirming that the RFNS score function $S(\mathcal{L})$ (Definition 2.6) effectively converts uncertain RFNS parameters into a definitive space where established algorithms such as VAM+MODI can identify the optimal solution. This is an important standard for how well the ML model works. The secondary discovery, which may be more important, is the model's *representational efficiency and scalability*. The scalability study (Table 33) shows that the ML model's computational advantage isn't that it does better than VAM on a small, static issue, but that its architecture is better for dynamic and large-scale situations. The numerical technique has a complexity of $\mathcal{O}(mn^2)$, which means that it has to run the whole procedure again if any of the parameters change. The trained RFNS-DQN model, on the other hand, can find the best allocation for a new issue instance with only one forward pass that takes a constant amount of time ($\mathcal{O}(1)$). This is a big change from "re-computation" to "instant inference," which makes it perfect for real-time logistics planning when things change quickly. Importantly, the baseline comparison (Subsection7) and the quantitative error analysis ((Subsection7.2) validate our principal assertion: the innovation is not alone in the application of ML but in the *integration of the unique RFNS representation*. The RFNS-DQN model demonstrated a notable PCI compared to both the Type-1 Fuzzy + MLP (4.65%) and IFQL (1.62%) benchmarks (Table 34). Significantly, it achieved a markedly elevated UCI of 0.85, in contrast to 0.41 and 0.53 for the baseline measures (Figure 14). This result statistically demonstrates that the RFNS model's capacity to encapsulate the entropy of real-world uncertainty independently modelling truth, indeterminacy, and falsehood alongside imprecise approximation directly results in better informed and economically advantageous judgements. The observed performance loss in the baselines indicates that simpler models forfeit this essential information during execution. The sensitivity analysis (Figure 13) ultimately validates the framework's resilience, delineating a stable operating zone ($\lambda \in [5, 20]$) for the principal reward function parameter. Conclusively, the RFNS model is very effective in tackling the inefficiency of the previous approaches by providing a single, computationally viable means to capture the complex uncertainty. The numerical and machine learning hybrid solution will offer the decision-makers a potent instrument: the optimality of classical algorithms in the process of validation, alongside the speed, flexibility, and high representation ability of a

contemporary AI system, which is specifically created to operate in the adversarial and inexact data setting of global supply chains.

8 Conclusion and future directions

The paper presented a new model that combines RFNS and ML to solve transportation issues in the conditions of complex and 3-D uncertainty. The work has several major contributions:

- An improved RFNS framework which provides good coverage of truth, indeterminacy and falsity membership levels and granularity of data, overcoming barriers of earlier fuzzy sets in logistics.
- A hybrid solution approach that has a numerical solution strategy (with an innovative RFNS score function based on VAM+MODI) and a specialised dual-stream DQN/PPO model that trains directly on RFNS data.
- Experimental validation that the method is optimal in costs as compared to classical methods and has much better computational efficiency following training.
- A statistical and comparative study demonstrating that the RFNS-ML model is better than all other fuzzy-ML hybrids, such as Type-1 Fuzzy MLP and intuitionistic fuzzy Q-learning.
- Although we have included the tests of scalability and sensitivity, larger-scale tests over a broader range of problem sizes and structures are required to have a complete picture of the performance boundaries of the framework.

Limitations and Future Work. Although this work constitutes a solid piece of proof-of-concept, we admit that it has its limitations, which also outline the clear lines of future research:

- **Real-World Validation:** The current validation is carried out on synthetic data, which is supposed to be a reflection of some challenges of uncertainty. The subsequent step to take is to center this framework on the actual data of logistics (e.g., e-commerce delivery or humanitarian supply chains), to demonstrate its practical value and strength even more.
- **Problem Scope:** The model was aimed at minimising costs. It will be expanded to other problems of path planning in complex radioactive environments and dynamic job shop scheduling problems in the future, where the cost, delivery time, and carbon reduction are minimised under uncertainty.
- **Architectural Exploration:** The dual-stream LSTM-MLP model worked, and alternative models (e.g., the use of graph neural networks to explicitly show the network topology) would be possible under varying problem formulations.
- **Dynamic Environments:** The model must be tested in further research on the full dynamic simulations in which the cost and demand parameters vary in real-time using the high inference speed of the ML model.
- **Real-World Scalability Validation:** In this study, we primarily rely on synthetic benchmarks to establish methodological validity. Although the scalability discussion in Section 7.1 shows encouraging preliminary results, the most significant task after the analysis of scalability is to test the framework with large-sized, real-world logistics data. E-commerce fulfillment networks and humanitarian supply chains are the types of industry partners that we are currently cooperating with to acquire such datasets. This key validation will be the subject of future work towards conclusively showing that the RFNS-ML approach has a computational advantage and is operationally valuable in the real world.

Overall, this study offers a validated and basic framework for utilising sophisticated uncertainty modelling in supply chain optimisation. The RFNS-ML framework can be enhanced to become a potent instrument for creating more resilient, adaptive, and intelligent logistics systems by considering these limitations in future work.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The authors also declare that there is no conflict of interest regarding the publication of this paper.

Data Availability: All the data is added within the manuscript.
Acknowledgement: The authors wish to express their appreciation for several excellent suggestions for improvements in this paper made by the referees.
Financial disclosure: This research was supported by the Key Project of the National Social Science Foundation of China 21AGL014.

References

[1] C. Aberle, S. Daubitz, O. Schwedes, C. Gertz, *Measuring transport poverty with a mixed-methods approach. A comparative case study of the German cities Berlin and Hamburg*, Journal of Transport Geography, **125** (2025), 104140. https://doi.org/10.1016/j.jtrangeo.2025.104140

[2] K. T. Atanassov, *More on intuitionistic fuzzy sets*, Fuzzy Sets and Systems, **33**(1) (1989), 37-45. https://doi.org/10.1016/0165-0114(89)90215-7

[3] M. L. Balinski, *Fixed-cost transportation problems*, Naval Research Logistics Quarterly, **8**(1) (1961), 41-54. https://doi.org/10.1002/nav.3800080104

[4] R. K. Bera, S. K. Mondal, *A multi-objective transportation problem under quantity dependent credit period and cost structure policies in triangular intuitionistic fuzzy environment*, Engineering Applications of Artificial Intelligence, **123** (2023), 106396. https://doi.org/10.1016/j.engappai.2023.106396

[5] G. Büyüközkan, D. Uztürk, Ö. Ilicak, *Fermatean fuzzy sets and its extensions: A systematic literature review*, Artificial Intelligence Review, **57**(6) (2024), 138. https://doi.org/10.1007/s10462-024-10761-y

[6] S. Chen, X. Long, J. Fan, G. Jin, *A causal inference-based root cause analysis framework using multi-modal data in large-complex system*, Reliability Engineering and System Safety, **265** (2025), 111520. https://doi.org/10.1016/j.ress.2025.111520

[7] S. K. Das, *An approach to optimize the cost of transportation problem based on triangular fuzzy programming problem*, Complex and Intelligent Systems, **8**(1) (2022), 687-699. https://doi.org/10.1007/s40747-021-00535-2

[8] R. Etesami, M. Madadi, F. Keynia, *Adaptive fuzzy swarm-based search algorithm (AFSSA) for complex engineering optimization*, Iranian Journal of Fuzzy Systems, **22**(6) (2025), 125-145. https://doi.org/10.22111/ijfs.2025.52217.9209

[9] D. He, Y. Lin, Z. Dai, S. X. Yang, *Neurodynamics-based visual servo predictive control for improving smooth movement of logistics omnidirectional robots*, IEEE Transactions on Industrial Electronics, **72**(12) (2025), 1-10. https://doi.org/10.1109/TIE.2025.3585028

[10] F. L. Hitchcock, *The distribution of a product from several sources to numerous localities*, Journal of Mathematics and Physics, **20**(1-4) (1941), 224-230. https://doi.org/10.1002/sapm1941201224

[11] E. Ilbahar, C. Kahraman, S. Cebi, *Location selection for waste-to-energy plants by using fuzzy linear programming*, Energy, **234** (2021), 121189. https://doi.org/10.1016/j.energy.2021.121189

[12] M. Kamran, M. E. M. Abdalla, M. Nadeem, A. Uzair, M. Farman, L. Ragoub, I. N. Cangul, *A systematic formulation into neutrosophic Z methodologies for symmetrical and asymmetrical transportation problem challenges*, Symmetry, **16**(5) (2024), 615. https://doi.org/10.3390/sym16050615

[13] M. Kamran, M. Nadeem, J. Zywiolek, M. E. M. Abdalla, A. Uzair, A. Ishtiaq, *Enhancing transportation efficiency with interval-valued Fermatean neutrosophic numbers: A multi-item optimization approach*, Symmetry, **16**(6) (2024), 766. https://doi.org/10.3390/sym16060766

[14] K. Karagul, Y. Sahin, *A novel approximation method to obtain initial basic feasible solution of transportation problem*, Journal of King Saud University-Engineering Sciences, **32**(3) (2020), 211-218. https://doi.org/10.1016/j.jksues.2019.03.003

[15] P. S. Kumar, *An efficient approach for solving type-2 intuitionistic fuzzy solid transportation problems with their equivalent crisp solid transportation problems*, International Journal of System Assurance Engineering and Management, **15**(9) (2024), 4370-4403. https://doi.org/10.1007/s13198-024-02433-5

[16] Y. Lu, S. Chen, X. Zhang, X. Pan, Y. Gang, C. Wang, *A quantum-enhanced heuristic algorithm for optimizing aircraft landing problems in low-altitude intelligent transportation systems*, Scientific Reports, **15** (2025), 21606. https://doi.org/10.1038/s41598-025-05261-0

[17] J. Luo, G. Wang, G. Li, G. Pesce, *Transport infrastructure connectivity and conflict resolution: A machine learning analysis*, Neural Computing and Applications, **34**(9) (2022), 6585-6601. https://doi.org/10.1007/s00521-021-06015-5

[18] X. Peng, J. Dai, *A bibliometric analysis of neutrosophic set: Two decades review from 1998 to 2017*, Artificial Intelligence Review, **53**(1) (2020), 199-255. https://doi.org/10.1007/s10462-018-9652-0

[19] J. Pratihar, R. Kumar, S. A. Edalatpanah, A. Dey, *Modified Vogel's approximation method for transportation problem under uncertain environment*, Complex and Intelligent Systems, **7**(1) (2021), 29-40. https://doi.org/10.1007/s40747-020-00153-4

[20] T. Senapati, R. R. Yager, *Fermatean fuzzy sets*, Journal of Ambient Intelligence and Humanized Computing, **11**(2) (2020), 663-674. https://doi.org/10.1007/s12652-019-01377-0

[21] S. Shivani, D. Rani, *Multi-objective multi-item four dimensional green transportation problem in interval-valued intuitionistic fuzzy environment*, International Journal of System Assurance Engineering and Management, **15**(2) (2024), 727-744. https://doi.org/10.1007/s13198-022-01794-z

[22] J. Tayyebi, A. Deaconu, E. Hosseinzade, A. M. Golmohammadi, *Fuzzy maximum capacity path problem and its application to optimal routing control*, Iranian Journal of Fuzzy Systems, **21**(4) (2024), 123-139. https://doi.org/10.22111/ijfs.2024.49145.8665

[23] T. Wang, J. Chen, J. Lü, J. L. K. Liu, A. Zhu, H. Snoussi, B. Zhang, *Synchronous spatiotemporal graph transformer: A new framework for traffic data prediction*, IEEE Transactions on Neural Networks and Learning Systems, **34**(12) (2023), 10589-10599. https://doi.org/10.1109/TNNLS.2022.3169488

[24] M. Wei, S. Yang, W. Wu, B. Sun, *A multi-objective fuzzy optimization model for multi-type aircraft flight scheduling problem*, Transport, **39**(4) (2024), 313-322. https://doi.org/10.3846/transport.2024.20536

[25] J. Yang, Z. Liu, G. Wang, Q. Zhang, S. Xia, D. Wu, Y. Liu, *Constructing three-way decision with fuzzy granular-ball rough sets based on uncertainty invariance*, IEEE Transactions on Fuzzy Systems, **33**(6) (2025), 1781-1792. https://doi.org/10.1109/TFUZZ.2025.3536564

[26] X. Yongsheng, L. Yingquan, *Global sustainable supply chain governance, effectiveness of social responsibility, and performance in emergent markets: An exploratory multiple case study*, Revista de Administrao de Empresas, **65**(6) (2025), e2024-0587. http://dx.doi.org/10.1590/S0034-759020250602

[27] L. A. Zadeh, *Fuzzy sets*, Information and Control, **8**(3) (1965), 338-353. https://doi.org/10.1016/S0019-9958(65)90241-X

[28] B. Zhang, X. Cai, G. Li, X. Li, M. Peng, M. Yang, *A modified A* algorithm for path planning in the radioactive environment of nuclear facilities*, Annals of Nuclear Energy, **214** (2025), 111233. https://doi.org/10.1016/j.anucene.2025.111233

[29] Z. Zhang, Z. Liu, L. Ning, H. Tian, B. Wang, *Belief-based fuzzy and imprecise clustering for arbitrary data distributions*, IEEE Transactions on Fuzzy Systems, **33**(8) (2025), 2755-2767. https://doi.org/10.1109/TFUZZ.2025.3576588

[30] B. Zhang, L. Meng, C. Lu, Y. Han, H. Sang, *Automatic design of constructive heuristics for a reconfigurable distributed flowshop group scheduling problem*, Computers and Operations Research, **161** (2024), 106432. https://doi.org/10.1016/j.cor.2023.106432

[31] P. Zhang, Q. Zhang, J. Liu, D. Wang, H. Ye, X. Zhang, T. Li, *Information fusion and feature selection for multi-source data utilizing Dempster-Shafer evidence theory and K-nearest neighbors*, Information Sciences, **718** (2025), 122408. https://doi.org/10.1016/j.ins.2025.122408

[32] Z. Zhao, Y. Zhang, Y. Zhang, K. Ji, H. Qi, *Neural-network-based dynamic distribution model of parking space under sharing and non-sharing modes*, Sustainability, **12**(12) (2020), 4864. https://doi.org/10.3390/su12124864

[33] C. Zuo, X. Zhang, G. Zhao, L. Yan, *PCR: A parallel convolution residual network for traffic flow prediction*, IEEE Transactions on Emerging Topics in Computational Intelligence, **9**(4) (2025), 3072-3083. https://doi.org/10.1109/TETCI.2025.3525656