

Optimal Transportation System & Resilience: A Study Of Sindhudurg District Cashew Industry

Dr. Shilpa C. Shinde¹, N. Balasubramanian²

¹Assistant Professor, Lala Lajpatrai Institute of Management ²Head of Department and Assistant Professor, Lala Lajpatrai Institute of Management

Abstract

Value chains have increased in intricacy and length in recent decades as firms prepare to tackle expanding globalisation with increased peripheral advancements. This involves the adoption of leaner supply chains as well as the formation of ecosystems that provide a stable environment and a constant flow of operations. However, because disruptions are inevitable in today's world, the operational models must be tuned to handle any risks. Complex production networks are designed for a variety of reasons, including cost, proximity to markets, and mass standardisation, but not necessarily for transparency or resilience. Any organization's supply chain operations can be a cause of vulnerability or resilience, depending on its capacity to assess risks, adopt risk mitigation methods, and develop effective business continuity plans. Transportation is the most important component in value chains, and transportation resilience is critical in recovering production networks through precise scheduling and achieving resilience indicators such as lowest trip time, minimum cost, and route optimization, among others.

The goal of this research is to clarify the key issues in network restoration scheduling and to offer a unique resilience-based optimization model for post-disaster transportation network restoration, in order to clear up theoretical and empirical ambiguity.Cashew industry which is seasonal as well as face many disruptions in production and processing stages was considered for the study.The study's objectives are (a) Study resilience indexes and its influence on transportation system optimization and (b)Study influence of resilience indexes on industry-based challenges with cashew product.The study objectives were addressed utilising an optimization model based on OR techniques and computer programming.The ideal solution for transportation cost, time, and efficiency can be obtained with the least amount of adjustment and analysis time, allowing cashew farmers to take advantage of transportation resilience and earn financial and environmental benefits.

Keywords: - Cashew, Disruption, Empirical, Optimal, Resilience, Risk, TransportationOperations Complexity

1. INTRODUCTION

Supply Chain Management operations is getting leaner thanks to growing globalization requirements and competitive environments. The leaner supply chains need to be more cost effective and high time responsive chains; it requires maintenance of low-cost inventory, timely retrieval of goods and information, shorter lead times, on-time-in-full deliveries,



optimal scheduling of transportation, just in time production and as well as collaborative offshoring and outsourcing operations. The entire ecosystems of the supply chain demand a very stable environment and continuous flow of operations.

Any disruption brings in vulnerability, thereby risk and can rapidly impact the operations and finance of the stakeholders. At the same time, disruption is unavoidable, and organizations require to build in resiliency in their business processes. The last decade witnessed massive disruptions all over the world and COVID 19 pandemic has engulfed and impacted the entire world. The supply resilience has become focal point in mitigating the risks created by disruptions. Supply Chain resilience is the ability to react to problems and recovering from them without significant impact to operations and customer timelines.

1.1. Transportation important cog in Supply Chain Management

Interestingly, both disruption and resilience are dependent on precise transportation schedule. The transportation network is essential for city maintenance and restoration, particularly after extreme events. Considerations of choices on the transportation network restoration plan must be made quickly following an occurrence, while taking into account a variety of decision-making factors and resource restrictions. Transportation systems, water supply systems, electrical grids, communication systems, and other key infrastructure systems are all used heavily in India's industrial industry. These systems work together to deliver vital goods and services in our daily lives. These systems have gotten increasingly sophisticated and interdependent over the last few decades, making them sensitive to disruptions and challenging to recover from. As a result, unanticipated disasters can have a significant financial and human consequence, and a transportation system is one of the most vulnerable infrastructure systems during disasters.

National transportation of manufacturing industrial raw materials is reliant mostly on public transportation infrastructure, and current roadway design standards stress optimal vehicle movement via a transportation network. In this context, efficiency may entail determining the shortest or fastest route, as well as the path that causes the least amount of traffic and bottleneck.

Many concepts were used for studying the efficiency of transport system when exposed to a variety of risks from daily fluctuations to rare natural disasters, including increasing use of the term resilience in the literature.Resilience focuses upon performance reduction and recovery, when experiencing indispensable disturbances, as compared to other conditions including robotics, reliability, survival and flexibility.The key criteria on which road networks are analyzed and design alternatives are examined are mostly presented here.Resilient infrastructure systems are those that respond to stress in a flexible and adaptive manner.Because of the critical role of transportation in emergency response, crucial service delivery, and economic well-being, policymakers are paying more attention to the resilience of roadway networks.

Climate fluctuations, market volatility, and political unrest all contribute to transportation concerns and disruptions, and globalisation has made anticipating and mitigating adverse occurrences difficult.Scholars have yet to agree on a common definition of resilience that may be used to guide the design, operation, and rebuilding of roadway networks when transportation attributes, such as the supply chain base structure, are significant.



The goal of this research is to learn how transportation resilience can assist cashew growers in navigating through disruptive occurrences and seasonal variations. A cashew producer and a transportation system might be able to build their supply chain base to be more resilient to plausible kinds of disruptive occurrences using the resilience perspective, which provides a conceptually unique 'capacity-oriented' approach. As a result, demand for transportation resilience is increasing. Cashew producers can minimise risks by reorganising their supply chains, but previous research has yielded insufficient guidance; hence, theoretical advances and empirical tools are required.

The creation of paradigms and quantification methodologies are currently the focus of transportation resilience research. Characterization of resilience metrics, such as traffic congestion, financial damage, post-disaster maximum flow, and autonomous system components, are part of these initiatives. Technical challenges with this type of resilience assessment include the reliance on unreliable performance data and the exclusion of unquantifiable indicators. Additional resilience approaches apply traffic network modelling to detect critical geo-location locations, minimise transport distances and minimise system-wide journey time.

The goal of the research was to outline the various problems in network restore planning and present a new model of resilience optimisation for the restoration of the post-catastrophe transport network to alleviate empirical and conceptual confusion.

Consequently, the following research question was posed:

• Is response diversity of producers positively associated with supply chain resilience, more positively than type diversity is?

• If so, response diversity assessments by cashew companies such as cashew raw material process companies would be an appropriate means in the management of their supply base structure and thus supply chain resilience.

The assessment was exemplified empirical study by validating the results through analysis of data from SindhudurgDistrict cashew product demand indices, each with a different relevant disruption and, consequently, a different criterion for the response diversity structure. The assessment, while borrowing theory from adjacent disciplines relative to cashew supply chain management, such as biodiversity, ecology and geography, also responds to calls to explore different types of event studies on transportation resilience and for the use of secondary and archival data.

2. LITERATURE REVIEW

Lean supply chain management (SCM) practises have become very popular (Blackhurst et al. 2005) as a result of the globalisation of markets and the competitive business environment, calling for continuous flow processing with low inventory quantities, leveled and just-in-time production, and precise transportation scheduling for cross-docking operations, resulting in more cost-effective and responsive supply chains (SCs). To attain leaner chains, stable and undistracted environments are essential, but it is extremely difficult when vulnerability hits due to any type of interruption. This has an impact on the company's operations and finances (Zsidisin et al. 2005). Given that more than 56% of enterprises worldwide experience a SC disruption each year, businesses have begun to take SC disruptions more seriously (BCI-



Business Continuity Institute 2019). To withstand interruptions, every company must design supply chain resilience.

The term "resilience" is derived from the Latin word "resiliere," which means "to rebound or spring back."Following Holling's (C. S. Holling, 1973) conceptualization of resilience in the context of ecological systems and classification of the distinction between resilience and stability, the concept of resilience has been introduced to various disciplines such as organisation (Y. Sheffi, 2005), economics (A. Rose, and S. Y. Liao, 2005), social science (W. N. Adger, 2000), supply chain (J. W.Wang et al., 2017 and J. (E. Hollnagel et. al., 2007; G.Y. Xu et. al., 2017; J. W.Wang et.al., 2019).Though multiple interpretations of resilience exist in different fields, most of them are founded on the same concept: resilience is a system's ability to recover to its regular state after being disrupted (S. Hosseini et. al., 2016).In recent years, resiliency has received a lot of attention in the field of transportation engineering.

To evaluate the resilience and measure the possible repercussions of a disruption scenario to a transportation system, researchers used a variety of parameters, performance characteristics, and assessment criteria. Though most research on transportation system resilience used a particular parameter or resilience index, transportation system performance-based studies infrequently used this term. Many performance measurements in the industry can be classified as resilience indexes.

Resilience Index	Influencing Factors
Travel time	(i) Travel time
Traver time	(ii) Place
	(i) Travel time
Reliability	(ii) Capacity
	(iii) Flow
	(i) Travel time
Vulnerability	(ii) Capacity
vullerability	(iii) Flow
	(iv) Large, connected component
Restoration time	(i) Restoration time
Restoration time	(ii) Delay
Travel demand	(i) Travel demand
	(ii) Flow
	(i) Recovery budget
Pudget / Cost	(ii) Delay cost
Budget / Cost	(iii) Operating cost
	(iv) Damage cost
	(i) Capacity
Capacity	(ii) Flow
	(iii) Travel demand
Accessibility	(i) Connected link

The following resilience indices and their numerous variables have an impact on transportation mode-based models.



Source: Ahmed, S., Dey, K. (2020)

Researchers chose the two resilience measures of Travel Time and Cost or Budget for this investigation.

(a) Travel time

Travel time is the time spent traveling from the point of source to the destination is heavily influenced by the topographical features of the transportation system as well as meteorological conditions during a disaster scenario.Because disasters can have a significant impact on topographical features (e.g., road Network links) as well as travel time, travel time is one of the most commonly used parameters to quantify system performance in a disaster scenario.

Travel time was also used to find out the difference in performance of a transportation system before and after a disaster (Faturechi R, Miller-Hooks E., 2014; Soltani-Sobh A. et. al., 2015)Scholars have used travel time to identify the most significant links in a highway network (Ukkusuri SV, Yushimito WF, 2009; Yin, Y., H. Leda, 2001), which provides valuable information on the effects of a probable disruption situation and allows them to develop strategies and recovery plans based on the link on crucial ranking.Furthermore, travel time can be interpreted differently to describe the resilience characteristics of the transportation system.

(b) Cost or budget

A number of investigators highlighted costs associated with the disruption, which can include operating costs (e.g. travel time costs, fuel costs).For example, (Li et. al.. 2008) suggested using vehicle expense (i.e. travel and idle time expenses) and delay (i.e. time to allocate backup trips), in order to fix problems with bus rescheduling due to any interference. Similarly, the problem of transit planning (Kliewer etc., 2006) was solved based on minimization of runway costs.(Kliewer et al. 2006) considered a multiple depots scenario (i.e. multiple depots (nodes) of passenger pick-up and drop-off locations).Operational costs are utilised (Asadabadi and Miller Hooks, 2018) for the purpose of increasing resilience of the marine transport networking system as a requirement for relative investment.

3. MATERIAL AND METHODS

(a) *Operationalization*

To test our hypotheses empirically using a case study approach, researchers have applied the following to implement the key concepts.Due to the interruption in the supply of cashews (the pandemic period of the year 2020), transport resilience was operationalized; the core function of supply chains was considered by sales (monetary sales as well as by the sales of product quantity).

(b) Data

The primary data sources comprised of a cashew industry demand index, shipment operating cost, geo mapping transport distance time, production capacity and public reports – accessible



through the Internet – of cashew producers and processors companies, and media regarding the enterprises encompassed and actions during disruptions (archival data).

Primary data collected during period of Jan to Dec 2020 during Pandemic Lockdown Period.The primary data presented the weekly shipping and sales in monetary value percashew producers and trademark for Sindhudurg District in west costal Konkan region Maharashtra.

The study's objectives -

(a) Study resilience indexes and its influence on transportation system optimization

(b) Studyinfluence of resilience indexes on industry-based challenges with cashew product.

This study observed hypothesis of resilience indexes which tested for its significant influence on Indian cashew Industry transportation optimization (H1).Further it also studiescost indexes significant influence on Indian cashew Industry transportation optimization (H1a) and time indexes significant influence on Indian cashew Industry transportation optimization (H1b).

(C) Method obtained for feasible solution

Selected Taluka from Sindhudurg District for study are Vengurla, Kudal, Malvan, Kankavli, Devgad and Sawantwadi. Researcher uses the following code for the destination Vengurla, Kudal, Malvan, Kankavli. D1* for Vengurla, D2* for Kudal, D3* for Malvan, D4* for Kankavli, D5* for Devgad and D6* for Sawantwadi.

Cashew producer (i.e., A, B, C, D) who distributes processed cashews to the six regional distributors indicated above. The average transportation cost per quintal of different producers to the same destination varies to some extent due to their own characteristics, capacity policies, and requirements in terms of quintals as per geographical regions, as shown in the table below.

	D1	D2	D3	D4	D5	D6	supply
Α	70	130	75	190	78	180	370
В	56	110	60	165	62	125	220
С	63	120	65	180	67	190	320
D	67	115	80	195	83	150	140
Demand	150	200	150	250	200	100	

(d) Formulation of Model based on cost

In this study, the researcher creates a transportation schedule for cashew, which is the district of Sindhudurg's essential commodity (one of the primary producing commodity).

The following transportation model is being developed to identify an effective schedule for transporting cashew to various markets in Sindhudurg for further distribution of the product on a domestic and worldwide level.

	D1	D2	D3	D4	D5	D6	Supply
Α	70	130	75	190	78	180	370
	(150)		(150)		(70)		
В	56	110	60	165	62	125	220
		(90)			(130)		
С	63	120	65	180	67	190	320



		(110)		(210)			
D	67	115	80	195	83	150	140
				(40)		(100)	
Demand	150	200	150	250	200	100	

Transportation business model delivers non-degeneracy solution (m+n-1=9) when using Row minimum approach because priority of commodity dispatch is given to producer's capacity of production based on lowest shipping cost distributors (destination).

The total cost of transportation is Rs.1, 18,970.Also, because to the lockdown restrictions imposed by the global industry scenario, low demand is observed in the export market throughout the pandemic year 2020 era.

The above-mentioned findings are investigated again using Object Oriented C++ modelling pseudo code, and the results are presented for a cost minimum transportation model. The findings reveal that the resilience indexes and their factor have a significant impact on the optimization of commodity transportation systems.

<pre>define row_max R; define col_max C; /* create initial matrix float cost_matrix[R][C]; for i:0 to R-1 for y:0 to C-1 cin>>cost_matrix[ij] end loop end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_matrix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix[R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minr_array[C]=0; float minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int min_loc; int c = j;</r-c+4)] </pre>	
<pre>/* create initial matrix float cost_matrix[R][C]; for i:0 to R-1 for y:0 to C-1 cin>>cost_matrix[i[j] end loop end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_matrix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix[R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] <="" create_minr_array(cost_matrix[i][j]);="" float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" min_loc;="" minc_array[r]="0;" minr_array[c]="0;" pre="" {=""></r-c+4)]></pre>	
<pre>float cost_matrix[R][C]; for i:0 to R-1 for y:0 to C-1 cin>>cost_matrix[i[j] end loop end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int min_loc; int b = min_loc;</r-c+4)] </pre>	define col_max C;
<pre>for i:0 to R-1 for y:0 to C-1 cin>>cost_matrix[i[j] end loop end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R}; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" min_loc;<="" minr_array(cost_matrix[i][j]);="" minr_array[c]="0;" pre="" {=""></r-c+4)]></pre>	/* create initial matrix
<pre>for y:0 to C-1 cin>>cost_matrix[i[j] end loop end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix[R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" min_loc;<="" minr_array(cost_matrix[i][j]);="" minr_array[c]="0;" pre="" {=""></r-c+4)]></pre>	float cost_matrix[R][C];
<pre>cin>>cost_matrix[i[j] end loop end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minr_array[R]=0; create_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int min_loc; int b = min_loc;</r-c+4)] </pre>	for i:0 to R-1
<pre>end loop end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minr_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	for y:0 to C-1
<pre>end loop i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minr_array[C]=0; float minr_array[C]=0; float minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	cin>>cost_matrix[i[j]
<pre>i=0; j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minr_array[C]=0; float minr_array[C]=0; create_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	end loop
<pre>j=0; /* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minr_array[C]=0; float minr_array[C]=0; float minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	end loop
<pre>/* Create quantity-matrix and initialize it to zero float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] a="i;" b="min_loc;</pre" create_minr_array(cost_matrix[i][j]);="" float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" minc_array[r]="0;" minr_array[c]="0;" {=""></r-c+4)]></pre>	i=0;
<pre>float quantity_matrix[R][C]; for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	j=0;
<pre>for i:0 to R-1 for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] a="i;" b="min_loc;</pre" create_minr_array(cost_matrix[i][j]);="" float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" minc_array[r]="0;" minr_array[c]="0;" {=""></r-c+4)]></pre>	/* Create quantity-matrix and initialize it to zero
<pre>for j:0 to C-1 quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R}; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] a="i;" b="min_loc;</pre" create_minr_array(cost_matrix[i][j]);="" find_minr_array(cost_matrix[i][j]);="" float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" minc_array[r]="0;" minr_array[c]="0;" {=""></r-c+4)]></pre>	float quantity_matrix[R][C];
<pre>quantity_martix[i][j]=0; end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	for i:0 to R-1
<pre>end loop end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	for j:0 to C-1
<pre>end loop /* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] a="i;" b="min_loc;</pre" create_minr_array(cost_matrix[i][j]);="" find_minr_array(cost_matrix[i][j]);="" float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" minc_array[r]="0;" minr_array[c]="0;" {=""></r-c+4)]></pre>	quantity_martix[i][j]=0;
<pre>/* create capacity array and requirement array float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] a="i;" b="min_loc;</pre" create_minr_array(cost_matrix[i][j]);="" find_minr_array(cost_matrix[i][j]);="" float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" minc_array[r]="0;" minr_array[c]="0;" {=""></r-c+4)]></pre>	end loop
<pre>float capacity_matrix{R]; float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	end loop
<pre>float require_matrix[C]; float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	/* create capacity array and requirement array
<pre>float*cost_matrix_ptr; cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	float capacity_matrix {R];
<pre>cost_matrix_ptr = &cost_matrix[0][0]; int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	
<pre>int count=0; while(count<r-c+4)] { float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</r-c+4)] </pre>	float*cost_matrix_ptr;
<pre>while(count<r-c+4)] a="i;" b="min_loc;</pre" create_minr_array(cost_matrix[i][j]);="" find_minr_array(cost_matrix[i][j]);="" float="" int="" min_loc="find_min_loc(cost_matrix[i][j]);" minc_array[r]="0;" minr_array[c]="0;" {=""></r-c+4)]></pre>	cost_matrix_ptr = &cost_matrix[0][0];
<pre>{ float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</pre>	int count=0;
<pre>float minr_array[C]=0; float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</pre>	· / -
<pre>float minc_array[R]=0; create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</pre>	
<pre>create_minr_array(cost_matrix[i][j]); find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</pre>	float minr_array[C]=0;
<pre>find_minr_array(cost_matrix[i][j]); int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</pre>	• •
<pre>int min_loc = find_min_loc(cost_matrix[i][j]); int a = i; int b = min_loc;</pre>	create_minr_array(cost_matrix[i][j]);
<pre>int a = i; int b = min_loc;</pre>	find_minr_array(cost_matrix[i][j]);
int b = min_loc;	
int $c = j$;	
	int $c = j$;



```
int r = 0;
if (require_array[min_loc].c>capacity_array[min_loc].r)
int x = \min_{x \in x} \log_{x} \frac{1}{x}
int y = 0;
quantity_matrix[a[b] = capacity_array[y];
require_array[min_loc]=require_array[min_loc]-quantity_matrix[a][b];
capacity_array[y]=capacity_array[y]-quantity_matrix[a][b];
i=i+1;
y=I;
count++;
cost=cost+quantity_matrix[a][b]*cost_matrix[a][b];
continue;
}
If(require_array[min_loc]<capacity_array[y])
quantity_matrix[i][b]=require_array[min_loc];
require_array[min_loc]=require_array[min_loc]-quantity_matrix[i][b];
capacity_array[y]=capacity_array[y]-quantity_matrix[a][b];
cost=cost+quantity_matrix[a][b]*cost_matrix[a][b];
y=y+1;
continue;
}
if (capacity_array[0]==require_array[min_loc]&==0)
ł
find 2_min_array(cost_matrix[i][j]);
int loc2= find loc2min_array(cost_matrix[i][j]);
quantity_matrix[a][loc2]=0;
i=i+1;
j=1;
cost=cost+quantity_matrix[a][b]*cost_matrix[a][b];
x = x + 1;
continue;
}
}
/* display quantity matrix
int l.m:
for 1:0 to R-1
for m:0 to C-1
cout<<quantity_matrix[l][m];</pre>
/* display the final min_cost
cout<<"the final minimal cost is" <<cost;
{
find2_min_array(cost_matrix[i][j]);
int loc2= find loc2min_array(cost_matrix[i][j]);unit_matrix[a][loc2]=0;
i=i+1;j=1;
cost=cost+unit_matrix[a][b]*cost_matrix[a][b];x=x+1;
continue;
```



}
/*displayunit_matrixintl,m;
forl:0toR-1form:0toC-1
cout<<unit_matrix[l][m];
/*displaythefinalmin_cost
cout<<"thefinalminimalcostis"<<cost;</pre>

Optimality Test applied for business solution (MODI Method)

We proceed to optimality using the MODI method after obtaining a basic feasible solution using the row minima method. Starting with u1=0, we determine a set of ui and vj for occupied basic cells using the relation cij = ui + vj, as shown below.

Iteration 1 for Occupied cell cost with optimality test									
c11	=u1+v1⇒v1	=c11-u1⇒v1	=70-0⇒v1	=70					
c13	=u1+v3⇒v3	=c13-u1⇒v3	=75-0⇒v3	=75					
c15	=u1+v5⇒v5	=c15-u1⇒v5	=78-0⇒v5	=78					
c25	=u2+v5⇒u2	=c25-v5⇒u2	=62-78⇒u2	=-16					
c22	$=u2+v2\Rightarrow v2$	=c22-u2⇒v2	=110+16⇒v2	=126					
c32	=u3+v2⇒u3	=c32-v2⇒u3	=120- 126⇒u3	= - 6					
c34	=u3+v4⇒v4	=c34-u3⇒v4	=180+6⇒v4	=186					
c44	=u4+v4⇒u4	=c44-v4⇒u4	=195- 186⇒u4	=9					
c46	=u4+v6⇒v6	=c46-u4⇒v6	=150-9⇒v6	=141					

We now find net evaluations for unoccupied cells by using the relation dij = zij - cij. dij for all unoccupied cells(i,j), where dij=cij-(ui+vj)

Iteration 1 for unoccur	Iteration 1 for unoccupied cell cost with optimality test								
d12	=c12-(u1+v2)	=130-(0+126)	=4						
d14	=c14-(u1+v4)	=190-(0+186)	=4						
d16	=c16-(u1+v6)	=180-(0+141)	=39						
d21	=c21-(u2+v1)	=56-(-16+70)	=2						
d23	=c23-(u2+v3)	=60-(-16+75)	=1						
d24	=c24-(u2+v4)	=165-(-16+186)	=-5						
d26	=c26-(u2+v6)	=125-(-16+141)	=0						
d31	=c31-(u3+v1)	=63-(-6+70)	=-1						
d33	=c33-(u3+v3)	=65-(-6+75)	=-4						
d35	=c35-(u3+v5)	=67-(-6+78)	=-5						
d36	=c36-(u3+v6)	=190-(-6+141)	=55						



d41	=c41-(u4+v1)	=67-(9+70)	=-12
d42	=c42-(u4+v2)	=115-(9+126)	=-20
d43	=c43-(u4+v3)	=80-(9+75)	=-4
d45	=c45-(u4+v5)	=83-(9+78)	=-4

We obtain the Initial Iteration optimum table, as well as the closed path and plus/minus sign allocation, as shown below:-

	D1	D2	D3	D4	D5	D6	Supply	ui
Α	70 (150)	130 [4]	75 (150)	190 [4]	78 (70)	180 [39]	370	u1=0
В	56 [2]	110 (90)	60 [1]	165 [-5]	62 (130)	125 [0]	220	u2=-16
С	63 [-1]	120 (110) (-)	65 [-4]	180 (210) (+)	67 [-5]	190 [55]	320	u3=-6
D	67 [-12]	115 [-20] (+)	80 [-4]	195 (40) (-)	83 [-4]	150 (100)	140	u4=9
Demand	150	200	150	250	200	100		
vj	v1=70	v2=126	v3=75	v4=186	v5=78	v6=141		

Closed path is DD2 \rightarrow DD4 \rightarrow CD4 \rightarrow CD2 and Revised cost allocation we get as follows:

	D1	D2	D3	D4	D5	D6	Supply
Α	70 (150)	130	75 (150)	190	78 (70)	180	370
В	56	110 (90)	60	165	62 (130)	125	220
С	63	120 (70)	65	180 (250)	67	190	320
D	67	115 (40)	80	195	83	150 (100)	140
Demand	150	200	150	250	200	100	

After starting the Modi technique and going through 7 cost iterations, we arrive at the following final optimal table:

	D1	D2	D3	D4	D5	D6	Supply
Α	70 (150)	130 (60)	75 (150)	190 (10)	78	180	370



	D1	D2	D3	D4	D5	D6	Supply
В	56	110	60	165 (120)	62	125 (100)	220
С	63	120	65	180 (120)	67 (200)	190	320
D	67	115 (140)	80	195	83	150	140
Demand	150	200	150	250	200	100	

The minimum total transportation cost :-= $70 \times 150 + 130 \times 60 + 75 \times 150 + 190 \times 10 + 165 \times 120 + 125 \times 100 + 180 \times 120 + 67 \times 200 + 115 \times 140 =$ Rs. 1,14,850/-. Here alternate solution is available with unoccupied cell CD2:d32 = [0], but with the same optimal value.

This study result shows that fluctuation in resilience indexes (cost) has significant influence on Indian cashew Industry transportation optimization (H1a).

	D1	D2	D3	D4	D5	D6	Capacity
A(i)	30	35	25	45	50	42	370
В	35	30	25	35	45	25	220
С	45	25	45	40	50	27	320
D	30	29	55	32	35	30	140
Requirement	150	200	150	250	200	100	

(e) Formulation of Model based on time factor

The above time matrix is mathematically expressed as follow:

Minimize (Total Time) $\mathbf{Z} = Max (i.j) \{t: x > 0\} ---- (-1)$

Subject to constraints

 $\sum_{j=1}^{n} x_{ij} = a_i$, i = 1, 2, ..., m (Supply constraints) ------ (2)

 $\sum_{i=1}^{m} x_{ij} = bj$, j = 1, 2, ..., n (Demand constraints)-----(3)

 $xij \ge 0$; a > 0; b > 0 for all i and j -----(4)

Here initial basic feasible solution is denoted by **Optimum (minimum) time of transportation**

 $Z^{(n)}(x) = \{Max_{ij}t_{ij}: x_{ij} > 0\} = t_{rs}^{(n)}$

where n (n=1,2,3,4...) is number of iterations. Note that (r,s) cell may or may not be unique.

Also calculated **Total transportation time** $\mathbf{T}(\mathbf{x}) = \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbf{t}_{ij} \mathbf{u}_{ij}$ Where auxiliary function $\mathbf{u}_{ij} = \{1; \mathbf{x}_{ij} > 0 \text{ and } 0; \mathbf{x}_{ij} = 0\}$

Efficiency of Transportation $\mathbf{E}(\mathbf{x}) = \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbf{t}_{ij} \mathbf{x}_{ij}$

Improved optimization after 3 revisions with NWC (North-West Corner) method is present here below-

	D1	D2	D3	D4	D5	D6	Capacity
A(i)	30 (150)	35 (200)	25 (20)	45	50	42	370
В	35	30	25 (130)	35 (30)	45 (60)	25	220
С	45	25	45	40	50	27	320



	D1	D2	D3	D4	D5	D6	Capacity
				(220)		(100)	
D	30	29	55	32	35 (140)	30	140
Requirement	150	200	150	250	200	100	

Here the total time required for shipment is minimum Z(x), Total time of transportation T(x) and Transportation efficiency E(x).

Improved	Basic solution	Transp	ortation	Does it	
optimum		Z(x)	T(x)	E(x)	opt?
Ι	x11=150;x12=200,x13=20;	50	305	37200	No
	x23=130; x24=90; x34=160;				
	x35=160; x45=40; x46=100				
II	x11=150; x12=200; x13=20;	50	302	35400	No
	x23=130; x24=90; x34=160;				
	x35=60; x36=100; x45=140				
II	x11=150; x12=200; x13=20;	45	297	32827	Yes
	x23=130;x24=30; x25=60;				
	x34=220; x36=100; x45=140				

This study result shows that fluctuation in resilience indexes (time) has significant influence on Indian cashew Industry transportation optimization (H1b).

4. DISCUSSION AND CONCLUSION

In any supply chain response diversity represents a novel structuring theory of transportation base complexity inside each purchase category to improve resilience, and it suggests advancement to supply chain system and supply base management. In this paper, the proposed model demonstrates that an optimal solution for cost, time, and transportation efficiency may be achieved with the least amount of adjustment and analysis time. Here the Cost based transportation model which has priority of commodity dispatch based on producer's capacity of production with lowest shipping cost distributors (destination)shows that it has influence on resilience indexes and cost factor has evidential influence on commodity transportation system optimization.Cost model with optimality test (Modi method) show that iteration in cost is possible and producers can achieve business objective of lowest shipping cost effectively.Another resilience index i.e. time factor, in this study shows amazing results, model focus 3 optimization of 3 times i.e. efficiency of transportation time, total transportation time and optimum (minimum) time of transportation system also has significant influence on commodity transportation system optimization.

Resilience indexes optimization is favorable to the financial and ecological well-being of cashew producers, as well as in times of adversity. Additionally, producer-specific empirical analyses can be utilised to design the supplier composition in order to reduce the impact of supply disruptions. Beyond supply base management, the transportation diversity method has novel theory and application potential, such as in the management of demand uncertainties and variations. Here research highlights that more research is needed on this component of supply chain resilience, as well as the precise qualities of diversity that are useful for innovation.



5. REFERENCES

- A. Rose, and S. Y. Liao (2005), "Modeling regional economic resilience to disasters: A computable general equilibrium analysis of water service disruptions," Journal of Regional Science, vol. 45, no. 1, pp. 75-112.
- [2]. Ahmed, S., Dey, K. (2020) Resilience modeling concepts in transportation systems: a comprehensive review based on mode, and modeling techniques. J Infrastruct Preserv Resil 1, 8 (2020). https://doi.org/10.1186/s43065-020-00008-9
- [3]. Asadabadi A, Miller-Hooks E (2018) Co-opetition in enhancing global port network resiliency: a multi-leader, common-follower game theoretic approach. Transp Res B Methodol 108:281–298, 2018.
- [4]. S. Holling (1973), "Resilience and stability of ecological systems," Annual review of ecology and systematics, vol. 4, no. 1, pp. 1-23.
- [5]. E. Hollnagel, D. D. Woods, and N. Leveson, (2007) Resilience engineering: Concepts and precepts: Ashgate Publishing, Ltd.
- [6]. Faturechi R, Miller-Hooks E (2014) Travel time resilience of roadway networks under disaster. Transportation Research Part B 70:47–64
- [7]. G.Y. Xu, J.W. Wang, G.Q. Huang, and C.H, Chen, (2017) "Data-driven resilient fleet management for cloud asset-enabled urban flood control," IEEE Transactions on Intelligent Transportation Systems, vol.19, no. 6, pp.1827-1838.
- [8]. J. W.Wang, R.L.Dou, R.R. Muddada, and W.J.Zhang, (2017) "Management of a holistic supply chain network for proactive resilience: theory and case study," Computers & Industrial Engineering, vol. 125, pp. 668-677.
- [9]. J.W. Wang, W.H.Ip, R.R.Muddada, J.L.Huang, and W.J.Zhang, (2012) "On Petri Net implementation of proactive resilient holistic supply chain networks," International Journal of Advanced Manufacturing Technology, vl.69 no.1-4, pp.427-437.
- [10]. J.W.Wang, H.F.Wang, Y.M.Zhou, Y. Wang, and W.J.Zhang, (2019) "On an integrated approach to resilient transportation systems in emergency situations," Natural computing, doi:10.1007/s11047-016-9605-y.
- [11]. Kezhi Liu, Changhai Zhai & You Dong (2020) Optimal restoration schedules of transportation network considering resilience, Structure and Infrastructure Engineering, DOI: 10.1080/15732479.2020.1801764
- [12]. Kliewer, N., T. Mellouli, L. Suhl, (2006) A time-space network based exact optimization model for multi-depot bus scheduling, Eur J Operational Res, Vol: 175, issue: 3, pp: 1616-1627
- [13]. Li, J. Q., P. B. Mirchandani, D. Borenstein, (2008) Parallel Auction Algorithm for Bus Scheduling, Computer-aided System Public Transport, pp: 281–299
- [14]. S. Hosseini, K. Barker, and J. E. Ramirez-Marquez, (2016) "A review of definitions and measures of system resilience," Reliability Engineering & System Safety, vol. 145, pp. 47-61.
- [15]. Soltani-Sobh A, Heaslip K, Khoury JE (2015) Estimation of road network reliability on resiliency: An uncertain based model. Int J Disaster Risk Reduction 14:536–544
- [16]. Supply chain disruptions and resilience: a major review and future research agenda K. Katsaliaki1 · P. Galetsi1 · S. Kumar pp 1-38
- [17]. Ukkusuri SV, Yushimito WF (2009) A methodology to assess the criticality of highway transportation networks. J Transportation Security 2(1–2):29–46
- [18]. W. N. Adger, (2000) "Social and ecological resilience: are they related?," Progress in human geography, vol. 24, no. 3, pp. 347-364.



- [19]. Y. Sheffi (2005), "The resilient enterprise: overcoming vulnerability for competitive advantage," MIT Press Books, vol. 1.
- [20]. Yin, Y., H. Leda, (2001) Assessing performance reliability of road networks under nonrecurrent congestion, Transportation Research Record 1771, pp: 148–155