Development of a Dynamic Information Fractal Framework to Monitor and Optimise Sustainability in Food Distribution Network

SAAD, Sameh <http://orcid.org/0000-0002-9019-9636> and BAHADORI, Ramin <http://orcid.org/0000-0001-6439-7033>

Sameh M. Saad[1,2] / Ramin Bahadori[1]

Development of a Dynamic Information Fractal Framework to Monitor and Optimise Sustainability in Food Distribution Network

[1] Department of Engineering and Mathematics, Sheffield Hallam University, Sheffield, S1 1WB, UK, E-mail: S.Saad@shu.ac.uk, b2047010@my.shu.ac.uk
[2] Faculty of ACES, Sheffield Hallam University, Harmer 2413, Howard Street, Sheffield S1 1WB, UK, E-mail: S.Saad@shu.ac.uk

Abstract:
The aim of this research paper is to develop a new framework for an information fractal to improve the food distribution network sustainability through two variables; Greenfield service constraints and minimum vehicle weight fill level on board. This paper applies the proposed framework to a hypothetical distribution network. Further, Supply Chain GURU Software is adapted to implement Greenfield analysis to identify the optimal number and location for setting up the new facilities through different Greenfield service constraints. A new Green Split Delivery-Vehicle Routing Problem also is developed to minimise CO_2 emission and implemented using the simulated annealing algorithm. The results revealed that the proposed dynamic control system has led to an enhancement in both collaboration and integration to decide upon the optimal number and location of distribution facilities as well as optimal vehicle weight fill levels to improve the sustainability throughout the food distribution chain.

Keywords: food supply chain, sustainable distribution network, information fractal, Greenfield analysis, Green Vehicle Routing Problem, split delivery

DOI: 10.1515/ijfe-2018-0061

Received: February 15, 2018; **Revised:** April 30, 2019; **Accepted:** June 14, 2019

1 Introduction

In recent decades, environmental and social considerations such as reducing raw materials, increasing both waste and pollutants have become significant issues for organisations. Thus, the sustainable supply chain has motivated many of the industrialists to meet customers, stakeholders and government's expectation through three dimensions of sustainable development; economic, environmental and social [1] in various policies of the organisation, such as purchasing, design, manufacturing, distribution and logistics.

The food industry is one such example for a dynamic type of environment where the expectation of customers for quality, availability and sustainability of food is high [2]. The food supply chain can be distinguished from other kinds of supply chains, in terms of parties which are involved, process and product features, and alternative redesign strategies [3]. It includes a vast variety of process centres like procurement and manufacturing companies, distributors, wholesalers, retailers and food service firms dealing with a vegetable or animal-based products which each should acquire sustainability to develop long-term relationships with the customers [4].

The downstream distribution of the food products to retailers through transportation is known as one of the major sources of environmental concern within food supply chains [5]. Transportation has irreparable effects on the environment; Consumption of resources, toxic effects on ecosystems and humans, noise and emissions of greenhouse gases (GHG) and pollutants are examples of these risks. Apart from these negative effects, emissions of GHG and carbon dioxide (CO_2) are directly linked to the health of the community and, indirectly, to the destruction of the ozone layer [6]. Most research has taken into account economic goals by minimising the distance, the time required or the number of vehicles needed and etc. and has neglected attention to environmental goals. Hence, Green Vehicle Routing Problem has received the attention by scholars since 2006 and two categories including Green-VRP [7–11] and Pollution Routing Problem (PRP) [12–15], are predominantly focused on reducing the energy consumption and CO_2 emissions, respectively, however, this research paper focuses on CO_2 emission.

Sameh M. Saad is the corresponding author.

Unlike previous research, this paper presents a new framework for an information fractal to dynamically improve distribution network sustainability in food retail supply chains. In comparison to the other information structure, information fractal is distinguished due to its capabilities such as self-similarity, self-optimisation, self-organisation, goal orientation and dynamics. The information fractal is composed of different fractal units named as the basic fractal unit (BFU) which are identical to each other and have the ability to make decisions, use appropriate methods, generate the goals and adapt to the dynamic environment changing by themselves [16, 17].

To achieve the aim of this study, two methodologies are used; Greenfield analysis is used to identify the optimal number and location of facilities with different service constraints. Greenfield analysis can be useful for determining the location of a new facility in a regional configuration. This method of analysis is quite frequently used in industry to determine the best location for a new and existing facility by which the location is indicated by latitude and longitude. This will be involved to optimise the travelling distance, travelling time, transportation routes etc. to consummate sustainability in the food supply chain. In this paper, also, a new approach using Split Delivery-Vehicle Routing Problem (SD-VRP) is introduced to minimise the CO_2 emission by considering minimum shipment weight that must be on the vehicle in length of each route and implemented using simulated annealing algorithm which is programmed in MATLAB software.

1.1 SD-VRP

In 1990, the SD-VRP and its mathematical model were introduced and presented by Dror et al. [18], in which the economic aspect of the problem of when a customer is served with more than one vehicle was considered. Dror et al. [19] provided an integer program for the above problem and used the branch and bound algorithm to solve it. The real application of this problem was studied by Mullaseril et al. [20] which they raised this problem for a food distribution network at a dairy farm in Arizona, USA, when the delivery of goods to the customer was associated with a time limitation and used heuristics algorithm which was proposed by Dror et al. [19]. Belfiore et al. [21] applied SD-VRP in a case study in Brazil, for a distribution network consisting of a central warehouse and 519 customers in 11 sectors using Neighbourhood Search algorithm. Tavakkoli-Moghaddam et al. [22] developed the Simulated Annealing algorithm for SD-VRP with the heterogeneous fleet. In this study, SD-VRP formulation is modified to consider the CO_2 emission and guarantee minimum vehicle weight fill level on board in order to formulate the new Green Vehicle optimisation model.

2 The proposed framework for Information Fractal Distribution Network

The new proposed framework for Information Fractal Distribution Network is displayed in Figure 1. As can be seen, it has two levels including an *Information Fractal – Reconfiguration Centre* as a top-level fractal and the *Information Fractal – Distribution Centres* as bottom level fractal with their own assigned retailers.

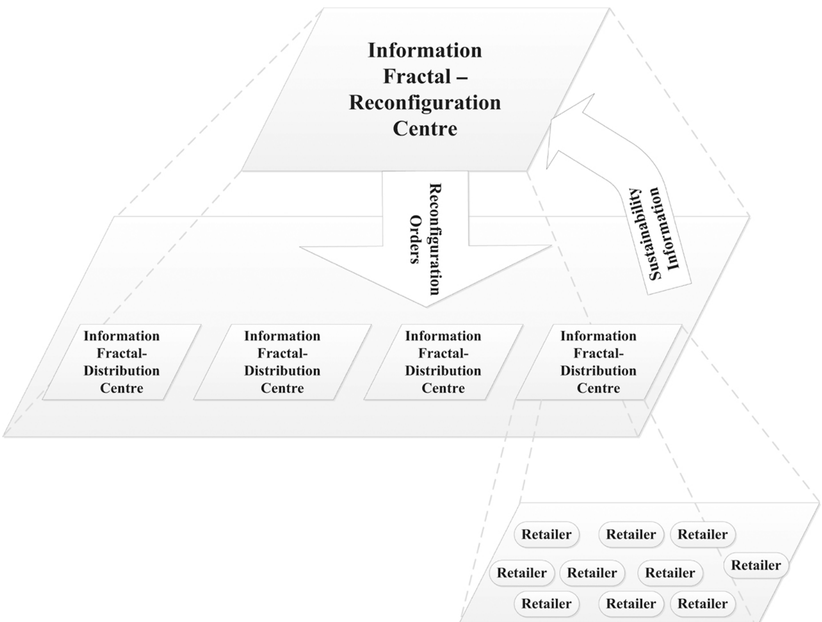

Figure 1: The proposed framework for an Information Fractal Distribution Network (IFDN).

According to Ryu et al. [23], each information fractal unit consists of five function models including observer, analyser, resolver, organiser and reporter as a BFU, see Figure 2.

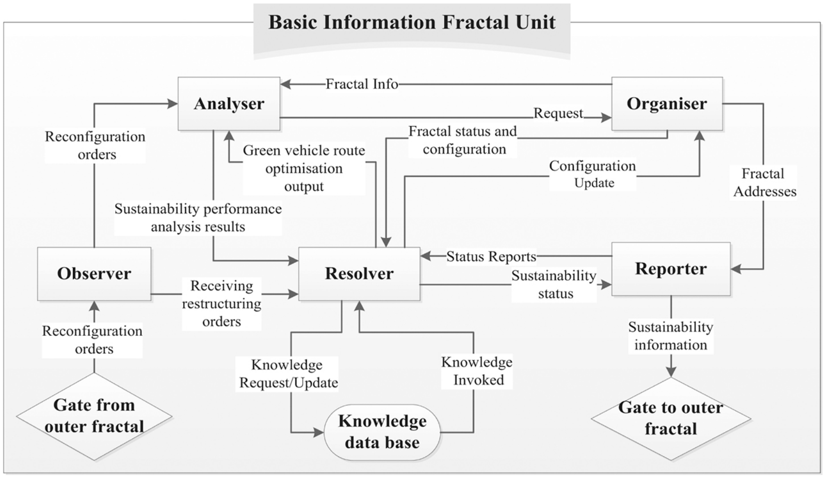

Figure 2: Basic information fractal unit structure for bottom level fractal.

In the bottom level fractal, observers in the distribution centres (DCs) trace and receive the reconfiguration orders from reconfiguration centre, transmit the orders to analysers and notify resolvers to receive the new restructuring orders. Resolvers transmit the orders to organisers to apply the reconfiguration. Once the fractal reconfiguration is done, resolvers apply green vehicle routing optimisation through their assigned retailers. Analysers use output data which is transmitted from resolvers to investigate sustainability performance measures and return analysis results. Then, resolvers transmit the fractal sustainability information to the reconfiguration centre through the reporter function.

In the top-level fractal, the observer traces and receives reconfiguration outputs from the bottom level shown as “Gate from outer fractal” (see Figure 2), then transmits them to the analyser and notifies the resolver. The analyser investigates and analyses the distribution network sustainability status and transmits the analysis results to the resolver. The resolver may make decisions for any further improvement and network restructuring regarding the analyser’s investigation. If the reconfiguration is specified by the resolver, the order should be sent to the organiser to apply the network reconfiguration. Then, the organiser notifies the resolver of which order is performed. Finally, resolvers transmit the reconfiguration orders to each DCs located in the bottom level through reporter function which is shown as “Gate to outer fractal” (see Figure 3). This structure is demonstrated in Figure 3 and clearly explains the internal relationships between these five function models.

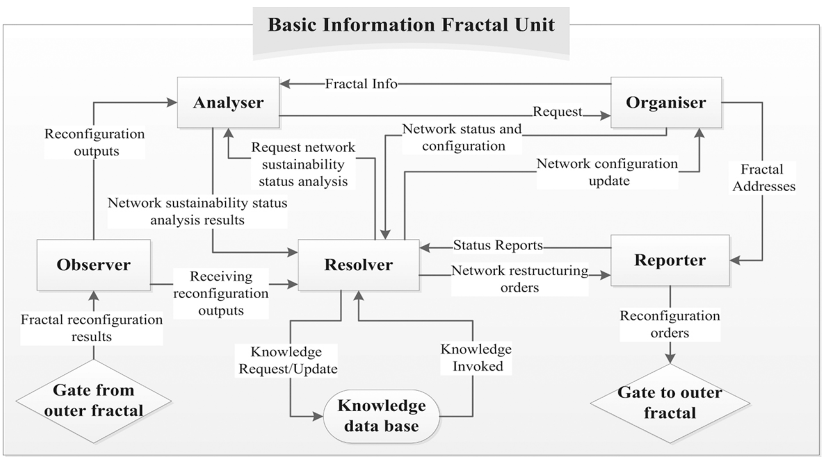

Figure 3: Basic information fractal unit structure for top level fractal.

As part of the top-level’s information fractal performance, LlamaSoft, Supply Chain GURU Software [24] was adapted to implement Greenfield analysis to identify the optimal number and location for setting up the new facilities, given the location and demand of customers with different service constraints aiming to improve distribution network sustainability.

In this method, the objective is to minimise the total weighted distance. The Greenfield service constraints such as customer demand or percentage of customers to be served within specified distances from the Greenfield site, which is a new site as the current sites are not sustainable in the long term, has a significant relationship with transportation costs, CO_2 emissions, transportation time and the number of vehicles in the required fleet [25].

As part of the information fractal performances, which are in the bottom level, an integer mathematical model is proposed and presented in the next section with which the simulating annealing algorithm is used as a heuristic technique to identify the optimum/near-optimum solution.

2.1 Green vehicle routing optimisation mathematical model

In this research, a PRP with a homogeneous fleet of vehicles and considering the possibility of split delivery and constraint of minimum shipment weight that must be on the vehicle during its service in each route is investigated simultaneously and its' integer linear programming model of the problem is described as follows:

2.1.1 Input parameters

V: Total number of nodes; with vertex set $V = \{0, 1, \ldots, n\}$; Where node 0 corresponds to the depot and the other nodes in this set of vertex represent the customers.

A: sets of edges; $A = \{(i,j) \mid i, j\} \in V$ and $i \neq j\}$.

K: Number of available vehicles; $K = \{1, \ldots, k\}$.

Q_k = Capacity of kth vehicle ($k \in K$).

D_i = Customers demand ($i \in V$).

d_{ij} = Length of edge between the nodes i and j $(i,j) \in A$

M_{sk} = Minimum Shipment weight that must be on the kth vehicle in length of each route during its service

C_{ijk} = CO_2 emission of moving kth vehicle ($k \in K$) between the nodes i and j

Where:

$$C_{ijk} = \left(\left(TW_k + W_{ijk}\right) E_k\right) \times d_{ij}$$

And

TW_k = Tare Weight of kth vehicle, which is the weight of empty vehicle.

W_{ijk} = Weight of shipments on board of kth vehicle between the nodes i and j

E_k = CO_2 Emission rate of kth vehicle

2.1.2 Decision variables

$$x_{ijk} = \begin{cases} 1 \text{ if } j\text{th customer is served by } k\text{th vehicle after } i\text{th customer} \\ 0 \text{ otherwise} \end{cases}$$

y_{ik} = The quantity of the demand of ith customer which is delivered by the kth vehicle.

2.1.3 Formulation

Therefore, the vehicle routing problem formulation by Dror and Trudeau [19] can be modified in order to consider the CO_2 emission and guarantee minimum vehicle weight fill level on board in order to formulate the proposed Green Vehicle optimisation model in this study.

The objective function represents minimisation of the total CO_2 emissions generated by the transportation fleet can be written as follows:

$$Min \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{K} C_{ij} x_{ijk}, \quad i \neq j \tag{1}$$

The model constraints are:

Constraint (2) ensures that each customer is visited at least once which guarantees the possibility of a split delivery.

$$\sum_{i=0}^{n}\sum_{k=1}^{K} x_{ijk} \geq 1\,, \qquad j = 1, \ldots, n, \tag{2}$$

Constraint (3) is about entrance and exit flows (p), where if a node i is visited by vehicle k, then the amount of product from vehicle k that enters and leaves that node must equal the demand at that node. Conversely, if node i is not visited by vehicle k, then the amount of product from vehicle k that enters and leaves that node must be 0. In fact, these constraints guarantee that any vehicle enters each node will definitely leave it.

$$\sum_{i=0}^{n} x_{ipk} - \sum_{j=0}^{n} x_{pjk} = 0\,, \qquad p = 0, \ldots, n;\ k = 1, \ldots, K, \tag{3}$$

Constraint (4) guarantees that vehicle cannot continue to serve more customers in length of each route if the weight of its shipment on board, coming down is from a specified minimum shipment weight.

$$\sum_{i=1}^{n} W_{ijk} \geq M_{sk}\ , \qquad j = 2, \ldots, n;\ k = 1, \ldots, K \tag{4}$$

Constraint (5) ensures that the ith customer's demand is completed if at least one vehicle passes through it.

$$y_{ik} \leq D_i \sum_{j=0}^{n} x_{ijk}\ , \qquad i = 1, \ldots, n;\ k = 1, \ldots, K \tag{5}$$

Constraint (6) indicates that all customers demand is entirely fulfilled.

$$\sum_{k=1}^{K} y_{ik} = D_i\ , \qquad i = 1, \ldots, n \tag{6}$$

Constraint (7) imposes that the loading process on any route should not exceed the capacity of the vehicle.

$$\sum_{i=1}^{n} y_{ik} \leq Q, \qquad k = 1, \ldots, K \tag{7}$$

Constraint (8) presents the sub tour elimination constraints where (S) refers to any collection of customers having at least 2 and at most $n-1$ members.

$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1\,, \qquad (S \subset \{1, \ldots, n\})\,;\ |S| \geq 2 \tag{8}$$

Equation (9) guarantees the decision variables x_{ijk} to be binary.

$$x_{ijk} \in \{0, 1\}\ , \qquad i = 0, \ldots, n; j = 0, \ldots, n; k = 1, \ldots, K \tag{9}$$

Equation (10) guarantees that the decision variable y_{ik} is positive.

$$y_{ik} \geq 0\,, \qquad i = 1, \ldots, n;\ k = 1, \ldots, K \tag{10}$$

The vehicle routing optimisations model which is presented above, are determined by resolvers; to minimise the CO_2 emission. Moreover, analysers also start to measure other sustainability performances including transportation costs, transportation time and the number of required vehicles which are needed to meet customers' demands. For these purposes, the following equations are developed.

2.1.4 Total transportation cost

$$TTC = \sum_{i=0}^{n}\sum_{j=1}^{n}\sum_{k=1}^{K} d_{ij}x_{ijk} \times ATC \tag{11}$$

Where
TTC = Total Transportation Cost
ATC = Average Transportation Cost per km
d_{ij} = The length of the edge between nodes i and j travelled by a vehicle.

2.1.5 Total transportation time

$$T_t = \sum_{i=0}^{n}\sum_{j=1}^{n}\sum_{k=1}^{K} \frac{d_{ij}x_{ijk}}{AFV_k} \tag{12}$$

Where
T_t = Total transportation time
AFV_k = Average Fleet Velocity (km/h) of vehicle k

2.1.6 Number of required vehicles

The proposed mathematical model allocates certain numbers of customers to be served according to its max load capacity until all customers' demand has been fulfilled. This will lead to the Total Number of Vehicles required (TNV) to be identified as an output from the proposed model.

3 Implementation of the proposed mathematical model using simulated annealing algorithm

Simulated annealing algorithm is an effective meta-heuristic optimisation algorithm for solving optimisation problems presented by Kirkpatrick et al. [26] and adapted from the Metropolis-Hastings algorithm [27]. They proposed a gradual freezing technique to solve the hard optimisation problems. The main advantage of the simulated annealing algorithm is its ability to not remain at the optimal local point and move to the global optimum point.

In generic term, the algorithm consists of two loops: one loop reduces the initial temperature to the final temperature and the second loop identifies the number of repetitions at each temperature. The factors affecting the timing of temperature reduction include the initial temperature, the final temperature, how to reduce the temperature and the number of repetitions in each temperature. Simulated annealing algorithm starts from an initial answer and then, in a repeated loop, it moves to neighbouring answers. If the neighbour's answer is better than the current one, the algorithm puts it as the current answer. Otherwise, the algorithm accepts that answer with the probability of $exp\ (-\Delta E/T)$ as the current answer. In this regard, ΔE is the difference between the objective function of the current answer and the neighbour's answer and T is a parameter called temperature. At each temperature, several repetitions are performed, and then the temperature is slowly reduced. In the initial steps, the temperature is set very high, so it is more likely to accept worse answers. With the gradual decrease of temperature, in the final steps, there will be fewer probabilities for accepting worse answers, and so the algorithm converges to a good answer. Thus, in this paper, in order to implement the proposed mathematical model using simulated annealing algorithm, in the beginning, an initial solution (x) and neighbourhood solution (x_{new}) for the problem are created and then, aligned with simulated annealing algorithm. Figure 4 displays a logical implementation flowchart for the proposed mathematical model using simulating annealing algorithm followed by a descriptive structure of the six steps involved. For more information, the MATLAB codes are also provided in Appendix 1.

I. At the first step, a model for each DC based on the assigned retailers and vehicles need to be developed.

II. Then, a discrete solution can be utilised to randomly allocating the retailers to an available vehicle to receive the service (i. e. generate the random routes for the problem).

III. At the third step, an objective function and the problem constraints can be generated, the output consists of a solution with a wide range of information such as CO_2 emission generated per path, length of the paths (e. g. can be used to determine both transportation time and cost), list of customers who are assigned to a vehicle per service (e. g. can be used to determine the total number of required vehicle) and etc.

IV. Next, CO_2 emission function should be developed where total CO_2 emission can be determined.

V. Later, after the initial solution is developed, the neighbourhood solution as part of the Simulated Annealing algorithm must be created.

VI. Finally, the developed initial solution and neighbourhood solution can be aligned to the Simulated Annealing algorithm. It is noteworthy that, so far, there is no perfectly good instruction to determine the accurately simulated annealing parameters. Hence, the most practical way is to test a set of possible values to find the most optimum set. But, it is more advisable to carefully select the simulated annealing parameters to minimise the number of calculations and, consequently, reduce the time spend on vain perturbations [28].

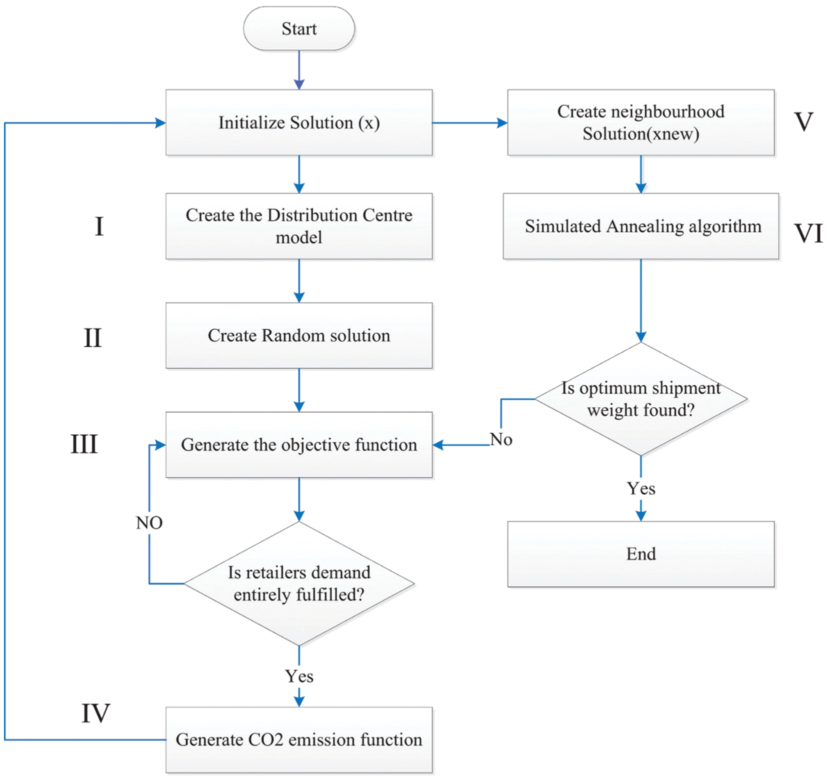

Figure 4: Implementation of the proposed mathematical model using simulated annealing algorithm.

4 Application of the proposed information fractal framework in food distribution network

In this research, a hypothetical distribution network and its data is considered: A large British food and beverage company wanted to determine the best number and location for DCs facilities as well as number of required fleet to meet customers demand for its national operations with multi-objective approach; minimisation of CO_2 emissions, transportation costs and maximise responsiveness. The company serves 340 stores around the country, the customers' daily demand weights (kg) are randomly selected from n (1,000, 4,000). Figure 5 displays the GURU snapshot of the stores' distribution.

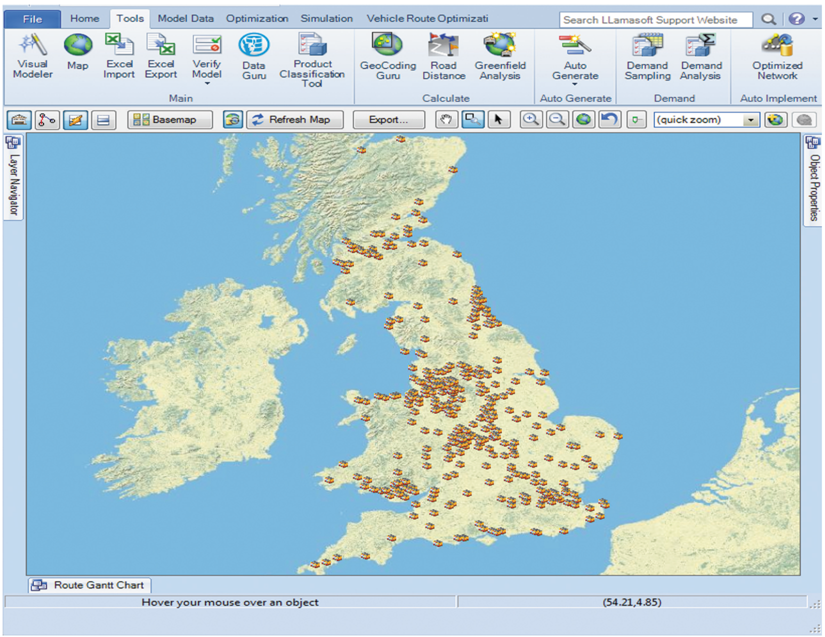

Figure 5: Supply chain guru screen shot of the considered retailers.

There is a homogeneous fleet available at the company (rigid 7.5 ton). The capacity of the vehicle is determined as 3,000 (kg) with a CO_2 emission rate of 0.0005442 kg per km [29]. Moreover, average transportation costs, average vehicle's velocities and vehicle's tare weight are considered to be £2.1 per km, 90 km/h (56 mph) and 3,000 kg, respectively.

5 Results analysis and discussion

5.1 Greenfield analysis results

As part of dynamic reconfiguration, to achieve the company's sustainability objectives, three reconfiguration scenarios are approved by the resolver in top-level fractal in which 100 % of customers are served within maximum sourcing distance of 113 km (first scenario), 161 km (second scenario) and 209 km (third scenario). Then, the proposed network reconfiguration scenarios are transmitted to the organiser function. Greenfield analysis is used by the organiser to determine the DC facilities within the best geographical locations with different service constraints. The obtained results from GURU Software are displayed in Table 1–Table 3 in which 12, 7 and 4 potential DC facilities with their assigned retailers are determined for first, second and third scenarios, respectively. For instance, Figure 6 also displays the snapshots of the GURU results for application to the reconfiguration scenarios.

Table 1: Greenfield analysis results for first scenario.

DC Facility	Latitude	Longitude	Number of assigned retailers
DC1	52.57657	−1.54377	65
DC2	55.90237	−3.64298	30
DC3	53.72346	−1.34595	35
DC4	54.66324	−3.36845	11
DC5	51.5389	0.14755	40
DC6	51.60858	−3.66043	31
DC7	52.41286	0.75166	15
DC8	57.64985	−3.31961	3
DC9	53.27981	−2.8974	65
DC10	50.37546	−4.14266	5
DC11	54.95469	−1.55084	23
DC12	50.98893	−1.49658	17

Table 2: Greenfield analysis results for second scenario.

DC Facility	Latitude	Longitude	Number of assigned retailers
DC1	50.71858	−3.532	15
DC2	55.6232	−2.81464	42
DC3	53.58013	−2.09142	116
DC4	51.48294	−0.38841	50
DC5	52.24223	−3.37758	55
DC6	56.4667	−2.9667	13
DC7	52.5695	−0.24053	49

Table 3: Greenfield analysis results for third scenario.

DC Facility	Latitude	Longitude	Number of assigned retailers
DC1	50.71858	−3.532	42
DC2	53.41493	−2.07702	161
DC3	51.87856	−0.41942	90
DC4	56.07189	−3.4537	47

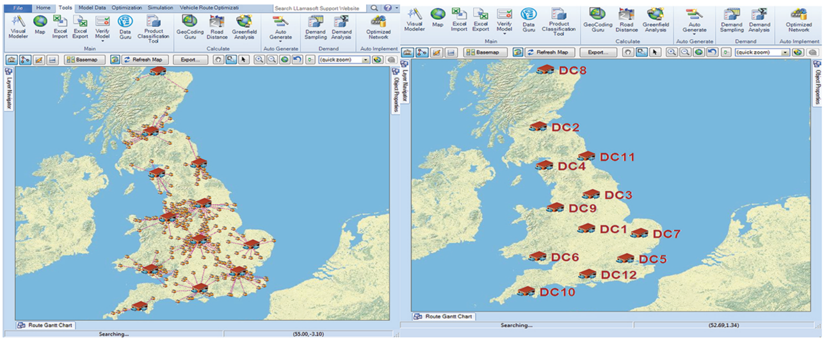

Figure 6: Supply chain guru screen shot of the Greenfield analysis result (first scenario).

5.2 Vehicle routing optimisation results

As soon as the configuration orders are received from the top level, resolvers in each bottom level notified the organisers to restructure the fractal to meet the orders. Then, in order to achieve the lowest CO_2 emission, the proposed green vehicle routing optimisation in this paper is applied by resolvers to examine the different minimum shipment weights using the simulating annealing heuristics search which is programmed in MATLAB Software. When the vehicle routing optimisation, within the specified minimum shipment weight, is complete, performance measures are investigated by analysers located in the bottom level fractals and the analysis results are returned to the resolvers. The above loop between resolver and analyser is continued until an optimum shipment weight is found.

Table 4 demonstrates the green vehicle routing optimisation results with split delivery through different scenarios which are obtained by determining the optimum minimum weight shipment.

Table 4: Green vehicle routing optimisation results.

	DC Facility	optimum M_{sk} (kg)	C (kg)	TTC (£)	T_t (h)	TNV (Q)
First Scenario	DC1	1,300	12,868	12,180	64	58
	DC2	1,550	5,970	5,643	30	28
	DC3	1,800	6,286	5,880	31	33
	DC4	1,000	2,523	2,371	13	11

	DC5	1,100	7,649	7,379	39	33
	DC6	1,200	6,247	5,863	31	29
	DC7	900	3,438	3,158	17	14
	DC8	1,300	706	628	3	3
	DC9	1,300	12,069	11,294	60	60
	DC10	2,000	1,113	1,061	6	5
	DC11	1,000	3,244	3,173	17	22
	DC12	1,400	3,826	3,641	19	14
Second Scenario	DC1	500	5,942	5,609	30	13
	DC2	1,000	18,023	16,336	86	39
	DC3	900	29,462	28,020	148	100
	DC4	1,100	13,207	12,590	67	42
	DC5	1,300	20,549	18,753	99	52
	DC6	1,600	3,901	3,715	20	13
	DC7	1,100	16,124	15,028	80	44
Third Scenario	DC1	1,000	20,013	19,236	102	37
	DC2	1,500	55,084	52,582	278	150
	DC3	1,400	34,460	32,645	173	85
	DC4	1,030	19,687	17,993	95	44

M_{sk} = Minimum shipment weight that must be on the kth vehicle in length of each route during its service.
$C = CO_2$ emission.
TTC = Total Transportation Cost.
T_t = Total transportation time.
TNV = Total Number of vehicles required.

5.2.1 Verification and efficiency of the proposed mathematical model

In order to evaluate the efficiency and verify the proposed model, it was also tested without considering the minimum weight of shipments on board (M_{sk}), which is the standard vehicle routing problem with split delivery (SD-VRP) proposed by Dror and Trudeau [19] and results were compared with the proposed model outputs using two criteria: mileage and CO_2 emissions. Comparison of the results proved that in all scenarios, the obtained values from the proposed model were improved in terms of both mileage and CO_2 emission:

- In the first scenario, the values obtained from the proposed model in terms of both criteria, the mileage and CO_2 emissions were reduced by 7.1 % and 5.9 %, respectively (see Figure 7 and Figure 8).
- In the second scenario, Figure 9 and Figure 10 display that there was also an improvement in both the mileage and CO_2 emissions by 7.4 % and 4.9 %, respectively.
- Finally, both the mileage and CO_2 emissions were reduced by 4.9 % and 3.3 % in the third scenario as shown in Figure 11 and Figure 12.

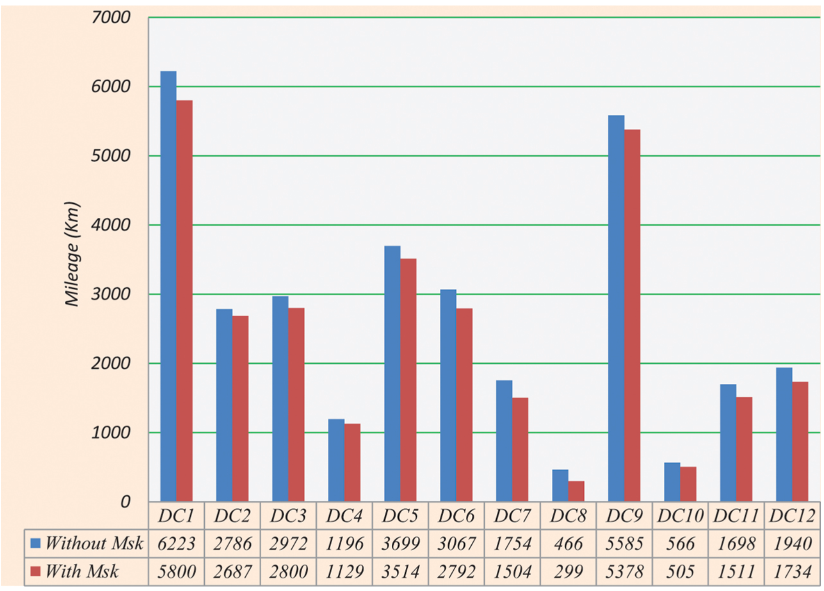

Figure 7: Comparison of the generated results in terms of the mileage criterion in the first scenario.

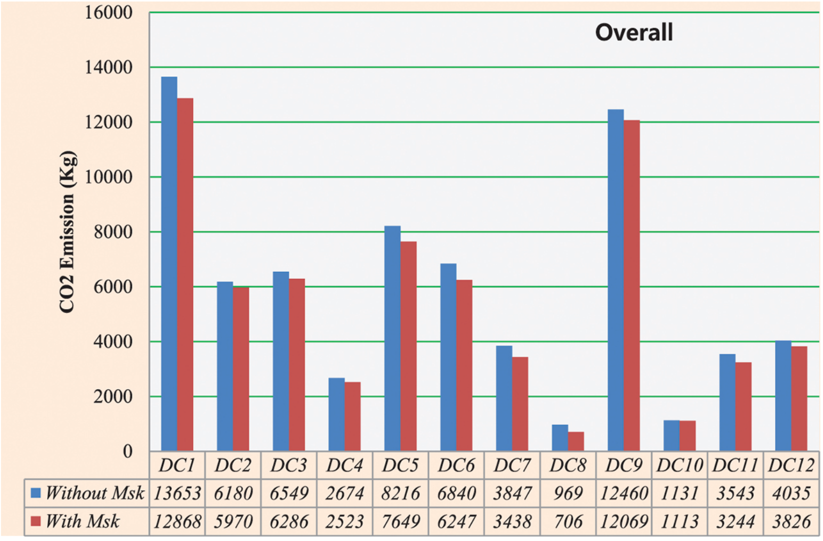

Figure 8: Comparison of the generated results in terms of the CO_2 emission criterion in the first scenario.

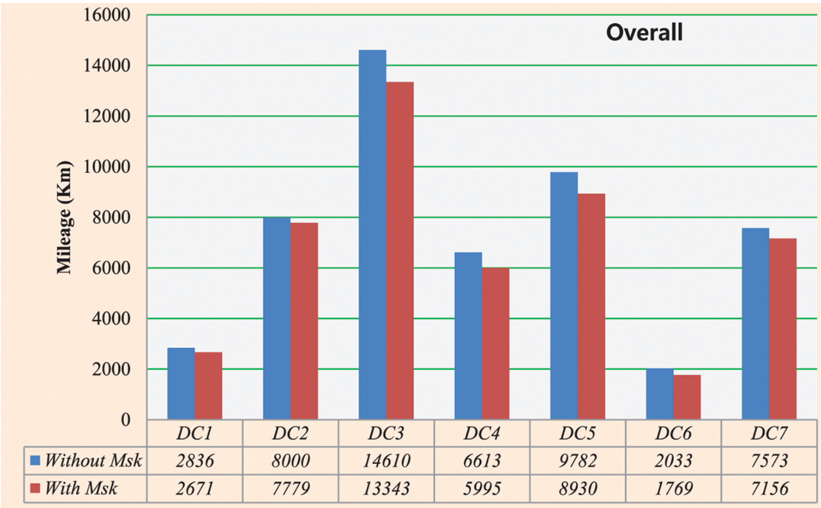

Figure 9: Comparison of the generated results in terms of the mileage criterion in the second scenario.

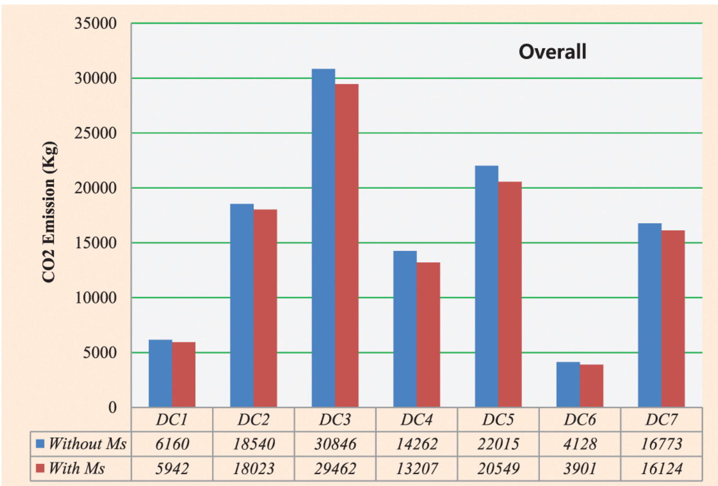

Figure 10: Comparison of the generated results in terms of the CO_2 emission criterion in the second scenario.

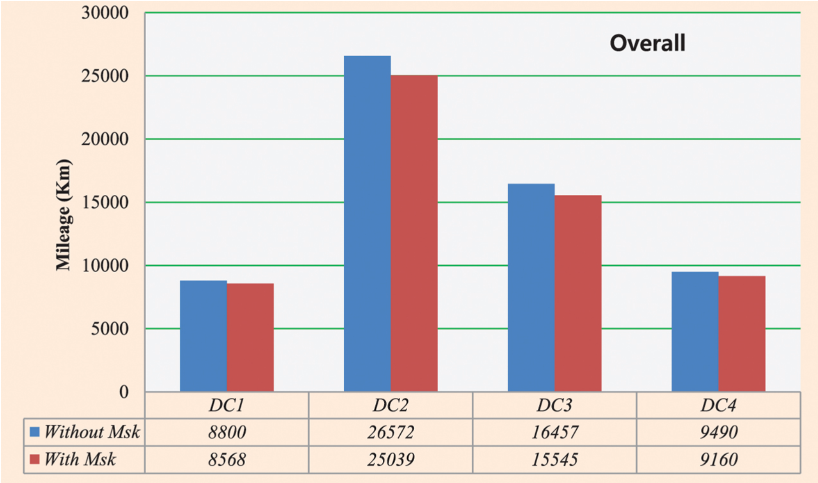

Figure 11: Comparison of the generated results in terms of the mileage criterion in the third scenario.

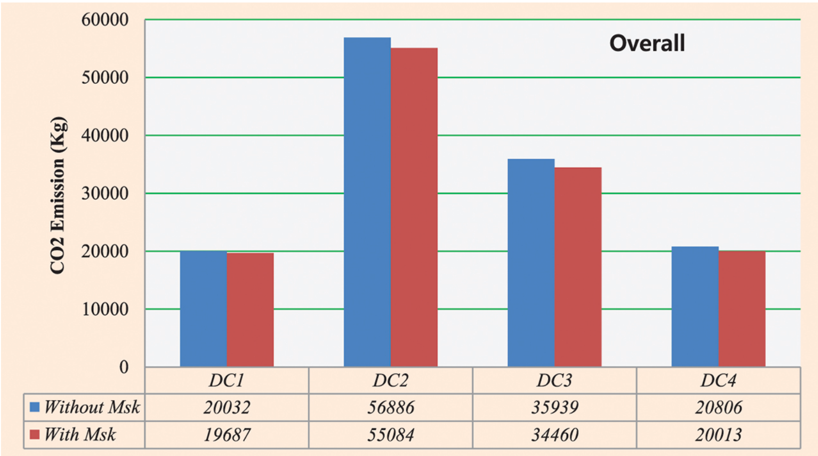

Figure 12: Comparison of the generated results in terms of the CO_2 emission criterion in the third scenario.

Furthermore, a programme has been developed to do the two-way analysis of variance (ANOVA) to explore the significant impact of the proposed model in which the minimum shipment has been applied. As shown in Table 5, the results clearly demonstrated that both mileage and CO_2 emission have significantly affected by the shipment condition in terms of "Minimum Shipment weight that must be on the k^{th} vehicle in length of each route during its service" and different scenarios in terms of "Greenfield analysis" at confidence level of 95 %.

Table 5: Two-way analysis of variance (ANOVA) results.

Source	Dependent variable	Sum of squares	DF	Mean square	F	Tail probability*
Scenario	Mileage	589,829,862.00	2	294,914,931.0	74.80	0.000
	CO_2 emission	2,794,993,596.0	2	1,397,496,798.	66.872	0.000
Shipment condition	Mileage	446,547,891.20	21	21,264,185.30	5.393	0.001
	CO_2 emission	2,089,444,203.0	21	99,497,342.98	4.761	0.001
Scenario * Shipment condition	Mileage	232,447,536.90	6	38,741,256.15	9.826	0.000
	CO_2 emission	1,021,415,511.0	6	170,235,918.5	8.146	0.000

(Tail Probability*) The source is significant at 95 %, if Tail probability ≤ 0.05.

5.3 Distribution network sustainability analysis results

As soon as the results of implementing the reconfiguration scenarios are received from the bottom level, the analyser in the reconfiguration centre starts to investigate the network sustainability for each scenario and, in turn, the analyser outputs are transmitted to the resolver.

- *First Scenario*: The result proved that 310 units of transportation assets are required to meet stores demand and the total CO_2 emission, transportation costs and transportation time are 65,939 kg, £62,271 and 329 hours, respectively.
- *Second Scenario*: The result showed that 303 units of transportation assets are required and total CO_2 emissions, transportation costs and transportation time are 107,208 kg, £100,050 and 529 hours, respectively.
- *Third Scenario*: In terms of service constraint, with 100 % of customer served within max sourcing distance of 209 km, 316 units of transportation assets are required for meeting the store's demand and total CO_2 emissions, transportation costs and transportation time are 129,244 kg, £122,455 and 648 hours, respectively.

In summary, as illustrated in Figure 13, CO_2 emissions, transportation costs and transportation time display rising trends from the first scenario to the third scenario, whilst, the Total Number of Vehicles required (TNV) to meet the store's demand does not follow the same trend. The reason behind this it could be due to the different number of customers/retailers allocated for each DC and the distance between them which is defined as scenarios in this paper. In addition, the identified minimum shipment weight on board which is may vary from DC to another that is defined as "Shipment condition" in this work. This means that the scenarios should have no impact on the number of vehicles required, hence, to support this justification, a full statistical factorial ANOVA technique was used to analyse the related results obtained from the proposed model at 95 % confidence interval. As shown in Table 6, the results revealed that Total Number of Vehicles required (TNV) was not significantly affected by the shipment condition in terms of "Minimum Shipment weight that must be on the vehicle in length of each route during its service" and different scenarios in terms of "Greenfield analysis" at confidence level of 95 %.

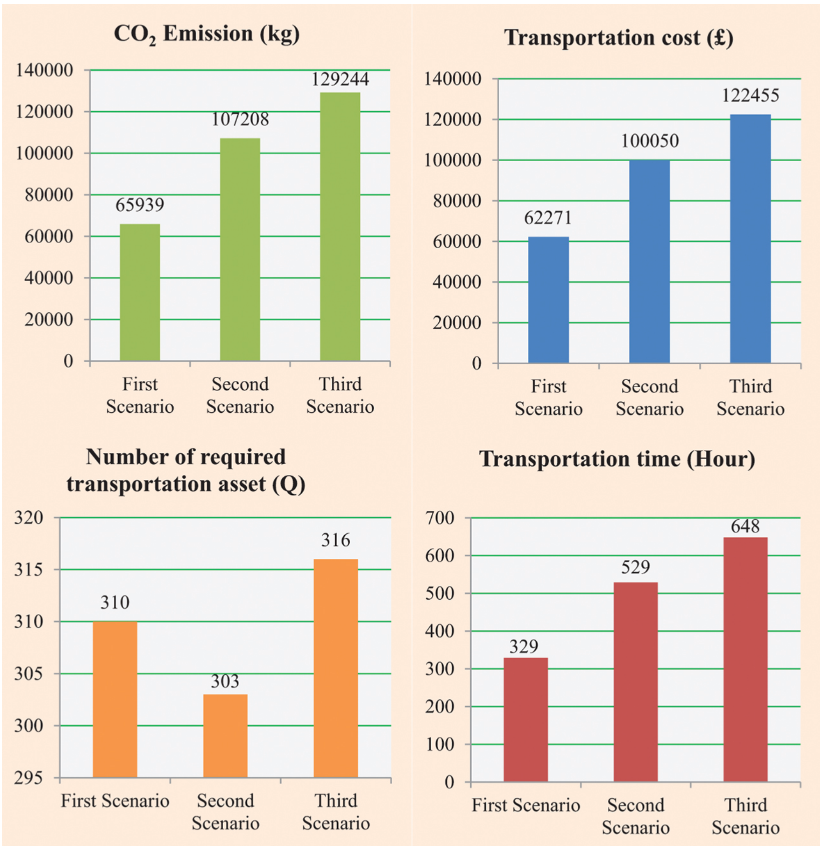

Figure 13: Performance measures trends through reconfiguration scenarios.

Table 6: Full factorial ANOVA results.

Source	Dependent variable	Sum of squares	DF	Mean square	*F*	Tail probability*
Scenario	TNV	432.982	2	216.491	0.169	0.846
Shipment condition	TNV	15,188.290	21	723.252	0.563	0.892
Scenario * Shipment condition	TNV	13,677.617	6	2,279.603	1.775	0.168

(Tail Probability*) The source is significant at 95 %, if Tail probability ≤0.05.

Nevertheless, the Greenfield service constraint, with 100 % of customers served within the maximum sourcing distance of 113 km is identified as the optimum scenario to have the lowest CO_2 emissions, transportation costs and transportation time.

6 Conclusions

In this paper, a new framework for the information fractal with two levels, named top and bottom level fractals was proposed to optimise the food distribution network sustainability through two variables; Greenfield service constraints and minimum weight of shipments on board.

The Fractal in the top level traced, observed and analysed the sustainability status of the distribution network, determined the optimum reconfiguration solution and, then, shared with fractals in the bottom level. Based on this information, the fractals in the bottom level implemented the reconfiguration orders and applied green vehicle routing optimisation and then transmitted the sustainability performance information to the top-level fractal.

The proposed framework was applied to the hypothetical food distribution network. The Supply Chain GURU Software was adapted to implement the Greenfield analysis to identify the optimal number and location for setting up the new facilities. The new Green Split Delivery-Vehicle Routing Problem (GSD-VRP) was developed and implemented using the simulated annealing algorithm which was programmed in the MATLAB software.

Application of the proposed framework has introduced a dynamic control system for the distribution network sustainability which has led to the increase of both collaboration and integration throughout the food distribution network.

Moreover, it provides a systematic method through which practitioners should be able to decide upon the optimal number and location of distribution facilities as well as optimal vehicle weight fill levels to improve the sustainability throughout the food distribution chain.

The focus of this research paper was the environmental impact as one of the sustainability dimensions. However, for future work, the other dimensions of sustainability should be considered, and the proposed green vehicle routing model should be developed further to take into consideration the time window, heterogeneous fleet and its availability for further evaluation and its effectiveness.

Appendix

A MATLAB codes

I Create the Distribution Centre model

```
function model=CreateDCModel(I,J) % I= number of Retailers, J= number of
Vehicle
E=[];     % CO2 Emission rate of vehicle
TW =[];    % Tare Weight of vehicle
r=[];     % Retailer Demands
c=[];     % Vehicle Capacity
x=[];     % Longitudinal coordinates of retailers
x0=[];    % Latitude coordinates of distribution centre
```

```
    y=[];     % Longitudinal coordinates of retailers
    y0=[];    % Latitude coordinates of distribution centre
    d=zeros(I,I);
    d0=zeros(1,I);
          for i=1:I
      %%% Distance among retailers
               for i2= i+1:I
                 d(i,i2)=distdim(distance(x(i),y(i),x(i2),y(i2)),'deg','kilometers
    ');
                 d(i2,i)=d(i,i2);
               end
           %%%Distance from depot to each retailers
               d0(i)=distdim(distance(x0,y0,x(i),y(i)),'deg','kilometers');
          end
    end
```

II Create Random solution

```
  function q=CreateRandomSolution(model)
        q=randperm(I+J-1);
        DelPos=find(q>I); %DelPos= Delimiter Position
        From=[0 DelPos]+1;
        To=[DelPos I+J]-1;
        L=cell(J,1); %L= List of retailers who received the service from the
    vehicle j
           for j=1:J
            L{j}=q(From(j):To(j))
           end
    end
```

III Generate the objective function. In the below code only CO_2 emission calculation is presented

```
function sol=CO2C(q,model)
       CH=0;
       ucap=zeros(J,1);
       C=zeros(J,1);
       DC=0;
       sh=0;     %%% Vehicle load moments
      for j=1:J
           if ~isempty(L{j})
              last_costm=L{j}(end);
               %%% Output loading weight from the depot
               s(j)=0;
               for ii=1:length(L{j})
                    s(j)=s(j)+r(L{j}(ii));
                     ucap(j)=sum(r(L{j}));
                     CH=CH+max(ucap(j)-c,0); %%%Vehicle capacity constraint
               end
                 sh=s(j);
                %%%CO2 emission from depot to first retailer
               C(j)=((TW+sh)*E)*d0(L{j}(1));
               sh=sh-r(L{j}(1));
               r(L{j}(1))=0;

               %%% CO2 emission among retailers
              for k=2:numel(L{j})
              %%% Apply constraint to guarantee that vehicle cannot continue to
     serve more customers in length of each route if the weight of its shipment
    on board, coming down is from a specified minimum shipment weight.
                  DC=DC+max(Ms-sh,0);
```

```
                         if sh>=r(L{j}(k))
                            sh=sh-r(L{j}(k));
                            r(L{j}(k))=0;
                        else
                            r(L{j}(k))=r(L{j}(k))-sh;
                            sh=0;
                            last_costm=L{j}(k);
                        end
              C(j)=C(j)+((vw+sh)*E)*d(L{j}(k-1),L{j}(k));
              end
               %%% CO2 emission from last retailer to depot
               C(j)=C(j)+(TW*E)*d0(last_costm)
           end
            %%% Identify retailers which their demand is not fully fulfilled
            rn=nonzeros(r);
            rr=find(r==0);
            A=d;
            A(rr,:)=[];
            A(:,rr)=[];
            A0=d0;
            A0(:,rr)=[];
            In=numel(rn);
            Jn=numel(rn);
            rb=zeros(In,1);
            rb=rn;
       end
   end
```

IV Generate CO_2 emission function

```
 function [z sol]=MyCO2(q,model)
      global NFE;
      NFE=NFE+1;
      sol=CO2C(q,model);
      eta=[];
      beta=[];
      z=sol.TotalC;
      z=z+ beta*sol.CH+ eta*sol.DC;

   end
```

V Create neighbourhood Solution (xnew)

```
function qnew=CreateNeighbor(q)
       m=randi([1 3]);
      switch m
          case 1
              % Do Swap
              qnew=Swap(q);
          case 2
              % Do Reversion
              qnew=Reversion(q);
          case 3
              % Do Insertion
              qnew=Insertion(q);
      end
   end

   function qnew=Swap(q)
      n=numel(q);
```

```
    i=randsample(n,2);
    i1=i(1);
    i2=i(2);
    qnew=q;
    qnew([i1 i2])=q([i2 i1]);
end
function qnew=Reversion(q)
    n=numel(q);
    i=randsample(n,2);
    i1=min(i(1),i(2));
    i2=max(i(1),i(2));
    qnew=q;
    qnew(i1:i2)=q(i2:-1:i1);
end
function qnew=Insertion(q)
    n=numel(q);
    i=randsample(n,2);
    i1=i(1);
    i2=i(2);
    if i1<i2
        qnew=[q(1:i1-1) q(i1+1:i2) q(i1) q(i2+1:end)];
    else
        qnew=[q(1:i2) q(i1) q(i2+1:i1-1) q(i1+1:end)];
    end
end
```

VI Simulated Annealing algorithm

```
clc;
clear;
close all;
global NFE;
NFE=0;
```

Problem Definition

```
model=SelectModel();                    % Select Model of the Problem
CO2Function=@(q) MyCO2(q,model);       % Objective Function, CO2 emission
                                              function
```

SA Parameters

```
MaxIt=1000;       % Maximum Number of Iterations by default
MaxIt2=80;        % Maximum Number of Inner Iterations by default
T0=100;           % Initial Temperature by default
alpha=0.99;       % Temperature Damping Rate by default
```

Initialisation

```
% Create Initial Solution
x.Position=CreateRandomSolution(model);
[x.CO2 x.Sol]=CO2Function(x.Position);
% Update Best Solution Ever Found
BestSol=x;
% Array to Hold Best CO2 Values
BestCO2=zeros(MaxIt,1);
% Array to Hold NFEs
nfe=zeros(MaxIt,1);
% Set Initial Temperature
T=T0;
```

SA Main Loop

```
for it=1:MaxIt
     for it2=1:MaxIt2
          % Create Neighbor
          xnew.Position=CreateNeighbor(x.Position);
          [xnew.CO2 xnew.Sol]=CO2Function(xnew.Position);
          if xnew.CO2<=x.CO2
               % xnew is better, so it is accepted
               x=xnew;
          else
               % xnew is not better, so it is accepted conditionally
               delta=xnew.CO2-x.CO2;
               p=exp(-delta/T);
               if rand<=p
                    x=xnew;
               end
          end
     end
     % Update Best Solution
          if x.CO2<=BestSol.CO2
               BestSol=x;
          end
     % Store Best CO2
     BestCO2(it)=BestSol.CO2;
     if BestSol.Sol.IsFeasible
          FLAG=' *';
     else
          FLAG='';
     end
     % Store NFE
     nfe(it)=NFE;
     % Display Iteration Information
     disp(['Iteration ' num2str(it) ': Best CO2 = ' num2str(BestCO2(it)) FLAG
]);
     % Reduce Temperature
     T=alpha*T;
     %Plot Solution
%      figure(1);
%      PlotSolution(BestSol.Sol,model);
%      pause(0.01);
end
```

Results

```
figure;
plot(nfe,BestCO2,'LineWidth',2);
xlabel('NFE');
ylabel('Best CO2');
```

References

[1] Seuring S, Müller M. From a literature review to a conceptual framework for sustainable supply chain management. J Clean Prod. 2008;16:1699–710.
[2] Beske P, Land A, Seuring S. Sustainable supply chain management practices and dynamic capabilities in the food industry: a critical analysis of the literature. Int J Prod Econ. 2014;152:131–43.
[3] Van Der Vorst JG, Tromp SO, Zee DJ. Simulation modelling for food supply chain redesign; integrated decision making on product quality, sustainability and logistics. Int J Prod Res. 2009;47:6611–31.
[4] Mattevi M, Jones JA. Traceability in the food supply chain: awareness and attitudes of UK small and medium-sized enterprises. Food Control. 2016;64:120–7.
[5] Validi S, Bhattacharya A, Byrne PJ. A case analysis of a sustainable food supply chain distribution system – a multi-objective approach. Int J Prod Econ. 2014;152:71–87.
[6] Bektaş T, Laporte G. The pollution-routing problem. Transp Res Part B: Methodol. 2011;45:1232–50.

[7] Kara İ, Kara BY, Yetis MK. Energy minimizing vehicle routing problem. In: Dress A, Xu Y, Zhu B, editors. Combinatorial optimization and applications. COCOA 2007. Lecture notes in computer science, vol 4616. Berlin, Heidelberg: Springer, 2007.
[8] Peng Y, Wang X. Research on a vehicle routing schedule to reduce fuel consumption. Proceedings of 2009 international conference on measuring technology and mechatronics automation. 11 Apr 2009;3:825–7. IEEE.
[9] Xiao Y, Zhao Q, Kaku I, Xu Y. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. Comput Oper Res. 2012;39:1419–31.
[10] Kuo Y. Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. Comput Ind Eng. 2010;59:157–65.
[11] Norouzi N, Sadegh-Amalnick M, Tavakkoli-Moghaddam R. Modified particle swarm optimization in a time-dependent vehicle routing problem: minimizing fuel consumption. Optim Lett. 2017;11:121–34.
[12] Palmer A. The development of an integrated routing and carbon dioxide emissions model for goods vehicles. 2007.
[13] Maden W, Eglese R, Black D. Vehicle routing and scheduling with time-varying data: a case study. J Oper Res Soc. 2010;61:515–22.
[14] Demir E, Bektaş T, Laporte G. A comparative analysis of several vehicle emission models for road freight transportation. Transp Res Part B: Transp Environ. 2011;16:347–57.
[15] Lin C, Choy KL, Ho GT, Chung SH, Lam HY. Survey of green vehicle routing problem: past and future trends. Expert Syst Appl. 2014;41:1118–38.
[16] Saad S, Bahadori R. An information fractal for supply network inventory optimisation. In: Affenzeller M, Longo F, Merkuryev Y, Bruzzone AG, Piera MA, editors. HMS2016: proceedings of the 18th international conference on harbor, maritime & multimodal logistics modelling and simulation, 2016 Sep 26–28. Larnaca, Cyprus. Red Hook, NY: Curran Associates, Inc, 2016: 28–34.
[17] Saad S, Bahadori R. Development of an information fractal to optimise inventory in the supply network. Int J Serv Computing Oriented Manuf. 2018;3:127–50.
[18] Dror M, Trudeau P. Savings by split delivery routing. Transp Sci. 1989;23:141–5.
[19] Dror M, Laporte G, Trudeau P. Vehicle routing with split deliveries. Discrete Appl Math. 1994;50:239–54.
[20] Mullaseril PA, Dror M, Leung J. Split-delivery routing heuristics in livestock feed distribution. J Oper Res Soc. 1997;48:107–16.
[21] Belfiore P, Yoshida Yoshizaki HT. Scatter search for a real-life heterogeneous fleet vehicle routing problem with time windows and split deliveries in Brazil. Eur J Oper Res. 2009;199:750–8.
[22] Tavakkoli-Moghaddam R, Safaei N, Kah MM, Rabbani M. A new capacitated vehicle routing problem with split service for minimizing fleet cost by simulated annealing. J Franklin Inst. 2007;344:406–25.
[23] Ryu K, Moon I, Oh S, Jung M. A fractal echelon approach for inventory management in supply chain networks. Int J Prod Econ. 2013;143:316–26.
[24] LlamaSoft Incorporated. Supply Chain Guru. 2017. Available at: http://www.llamasoft.com/. Accessed 7 Nov 2017.
[25] Saad S, Bahadori R. Modelling the sustainable distribution system in food retail sector – a simulation approach. In: Bruzzone AG, Piera MA, Longo F, Vignali G, editors. FoodOPS2017: proceedings of the international food operations and processing simulation workshop. 2017 Sep 18–20. Barcelona, Spain. Red Hook, NY: Curran Associates, Inc, 2017: 45–50.
[26] Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science. 1983;220:671–80.
[27] Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. Equation of state calculations by fast computing machines. J Chem Phys. 1953;21:1087–92.
[28] Ledesma S, Aviña G, Sanchez R. Practical considerations for simulated annealing implementation. In Simulated annealing. 2008. InTech.
[29] Consultancy AE. 2010 guidelines to Defra/DECC\'s GHG conversion factors for company reporting; produced by AEA for the Department of Energy and Climate Change (DECC) and the Department for Environment, Food and Rural Affairs (Defra), Version 1.2.1. Available at: http://archive.defra.gov.uk/environment/business/reporting/conversion-factors. 2010:1–49