Journal of Open Innovation: Technology, Market, and Complexity

Article

A Novel Enhanced Differential Evolution Algorithm for Outbound Logistics of the Poultry Industry in Thailand

Karn Moonsri [1], **Kanchana Sethanan** [1,*] and **Kongkidakhon Worasan** [2]

[1] Research Unit on System Modelling for Industry, Department of Industrial Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen 40002, Thailand; hotphai@gmail.com
[2] Faculty of Business Administration and Accountancy, Khon Kaen University, Khon Kaen 40002, Thailand; kongwo@kku.ac.th
* Correspondence: skanch@kku.ac.th

Abstract: Outbound logistics is a crucial field of logistics management. This study considers a planning distribution for the poultry industry in Thailand. The goal of the study is to minimize the transportation cost for the multi-depot vehicle-routing problem (MDVRP). A novel enhanced differential evolution algorithm (RI-DE) is developed based on a new re-initialization mutation formula and a local search function. A mixed-integer programming formulation is presented in order to measure the performance of a heuristic with GA, PSO, and DE for small-sized instances. For large-sized instances, RI-DE is compared to the traditional DE algorithm for solving the MDVRP using published benchmark instances. The results demonstrate that RI-DE obtained a near-optimal solution of 99.03% and outperformed the traditional DE algorithm with a 2.53% relative improvement, not only in terms of solution performance, but also in terms of computational time.

Keywords: multi-depot vehicle routing problem; novel enhanced differential evolution algorithm; outbound logistics planning

Citation: Moonsri, K.; Sethanan, K.; Worasan, K. A Novel Enhanced Differential Evolution Algorithm for Outbound Logistics of the Poultry Industry in Thailand. *J. Open Innov. Technol. Mark. Complex.* **2022**, *8*, 15. https://doi.org/10.3390/joitmc8010015

Received: 23 October 2021
Accepted: 31 December 2021
Published: 10 January 2022

1. Introduction

In recent years, competitive markets, challenging customer demands, and increased awareness of logistics and transportation activities have increased the importance of technology for Thailand's poultry industry. In order to develop new technology for the marketplace, companies "can and should use external ideas and internal ideas, as well as internal and external paths to market," according to the open innovation paradigm [1]. In addition to logistics management being influenced by innovations, new technologies are evolving in response to businesses' goals and competitiveness conditions, increasing their complexity [2]. Currently, logistics, also known as logistics 4.0, is receiving the most attention in the fourth industrial revolution trend. Artificial intelligence, real-time tracking, data-driven network logistics, the internet of things, optimization software, and so on are examples of technologies [3–5]. These logistics 4.0 technologies will significantly impact Thailand's poultry industry's outbound logistics planning.

The outbound logistics of the poultry industry in Thailand are considered in this paper, as agricultural products comprise a significant part of Thailand's development, impacting Thailand's economic growth. Outbound logistics is the shipping of finished products to customers from a distribution center. At this stage, transportation is typically carried out by trucks. Distribution planning can be a challenging problem and adhering to distribution center best practices is also crucial for ensuring the efficient transportation of products. The importance of poultry distribution planning has increased, due to rising transportation costs and opportunities for decreasing costs in incorporating optimal distribution planning. A flow process of outbound logistics for the poultry industry in Thailand is depicted in Figure 1. In short, the outbound logistics of the poultry industry in Thailand consists of three principal distributions: (1) The old hens are slaughtered and then sold as poultry meats to customers; (2) eggs are mainly sold directly to the end-consumers; and (3) broken eggs are sent to a processing plant [6,7]. Various egg products (e.g., egg powder) are produced, which are used in the food industry. Poultry distribution, regarding the design of the routing scheme for routing problems, is known as the multiple-depot vehicle-routing problem (MDVRP). One characteristic of this problem is that many customers need to be served by vehicles from many depots (e.g., egg distribution centers 1,2 and a slaughterhouse). Each vehicle must start at its depot, visit customers in order, and return to the same depot. Therefore, poultry distribution planning is intended to route the vehicle at each depot, with respect to the orders of customers.

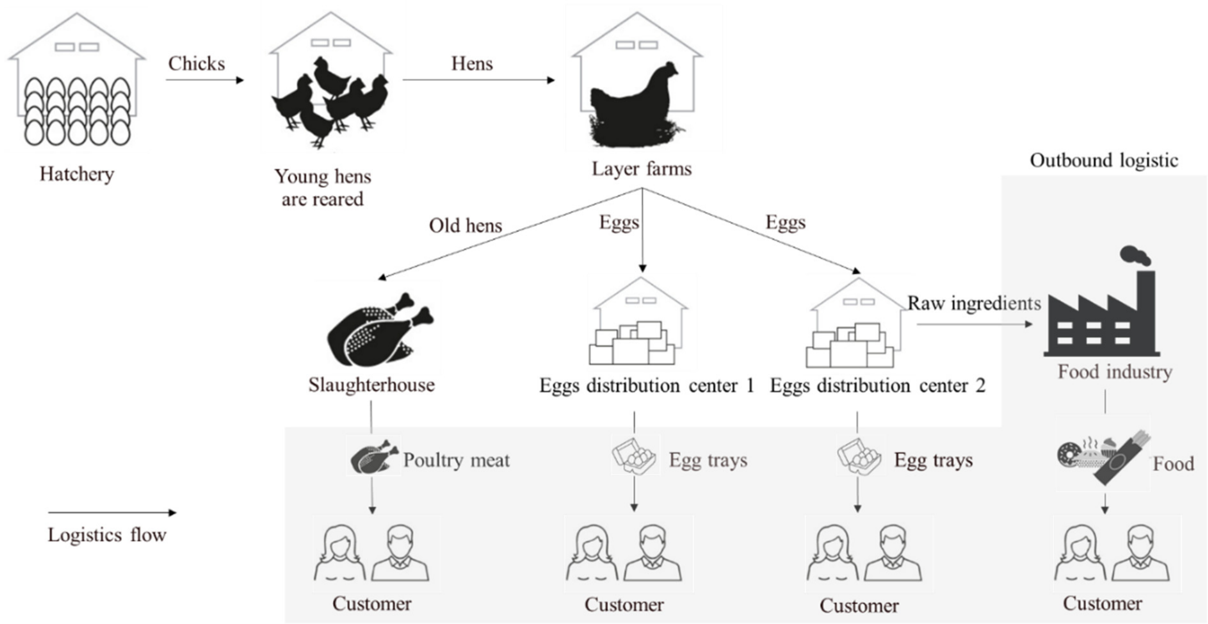

Figure 1. Logistics flow of the poultry industry.

The MDVRP has been the focus of many studies, as it is widely applicable to many real-world situations, including logistics distribution problems for optimizing total transportation costs. After all, an optimal route can minimize the total distance of each route, thereby leading to cost savings. Hence, it is essential to have an optimized plan for vehicle routing to complete poultry distribution.

In this paper, we focus on outbound logistics planning of the poultry industry in Thailand, with the objective being to minimize transportation costs. We developed a mixed-integer linear programming (MILP) model for small-sized instances. In terms of large-sized instances, we developed a new enhanced differential evolution (DE) algorithm, which should make the best decision to minimize the distance between each depot and the customers. The main contributions of this paper are twofold. First, we developed a new mutation formula for re-initializing solutions for the DE algorithm in the context of protecting movements to a local optimal. Secondly, we designed an algorithm to further enhance the DE algorithm. The local search techniques were used in the *k*-variable move to improve the ability to search for the best solutions to enhance the exploitation search capability, called the local vector. Hence, using the re-initialization solution formula on the DE algorithm is known as the novel re-initialization solution and local search function on the DE algorithm (RI-DE), which will be implemented in Thailand's Poultry Industry.

This paper is organized as follows: In Section 2, a review of the related literature is presented. Section 3 introduces the problem statement and formulation of MDVRP. Section 4 presents our proposed algorithm for solving the MDVRP. The computational results are discussed in Section 5. Finally, our conclusion is detailed in Section 6.

2. Literature Review

Developing an efficient production planning system for the poultry industry is challenging and interesting to manage, in terms of the supply chain and logistics; this is especially the case in the transportation planning for poultry, which is a process that covers activities involving both inbound, production, and outbound logistics. Optimizing operational and production planning results in lower operating costs and links to the outbound logistics system, where each customer receives the product according to their quantity demand. Therefore, this research aims to develop decision-making guidelines for solving poultry transportation planning problems based on obtaining globally optimal solutions, then developing a mathematical model for a mixed-integer linear program considering a single objective optimization problem to minimize the total cost of transportation, solving poultry transportation planning problems based on obtaining globally optimal solutions. By developing a mathematical model of a mixed-integer linear program considering a single objective optimization problem, the minimization of the total distance from each depot to all customers can be achieved.

The literature on open-innovation dynamics and multiple-depot vehicle-routing problems mainly considers three factors: MDVRP, metaheuristics in routing problems, and local search problems. In the following, we review the related papers in terms of these three directions.

Open innovation is intended for exploring and exploiting [8] new opportunities to obtain and develop new knowledge and technology [9], specifically in the poultry industry in Thailand, in which Small and Medium Enterprises (SMEs) assist in overcoming innovation obstacles [8,10,11]. In addition, previous research on open innovation investigates the importance of management in SMEs [12].

Given the novelty of open innovation dynamics in various research domains such as SMEs, as well as small and medium industries (SMIs), numerous studies have attempted to establish a precise description of this concept through research techniques by using qualitative methods such as system dynamics [13–19], theory-building [20–22] content analysis [3,23,24] optimization [12,25] and etc. Open innovation dynamics vary across different ecosystems [26], Moreover businesses' attitudes toward open innovation result from a mix of the ecosystem's various aspects [27,28].

Open innovation appears to be an ideal strategy for promoting a firm's operations for knowledge exploration and exploitation to produce optimization software for routing problems in the context of Industry 4.0 technologies and their execution and integration [2,3]. Theeraviriga et al. [29] presented a new optimization technique for the location-routing problem of agriculture in Thailand. They developed a mathematical model and variable neighborhood strategy adaptive search (VaNSAS). They compared the solution of the proposed algorithm with the probability of selecting a black box in four different equations. Theeraviriga et al. [30] studied location decision making and transportation for the palm oil collection center. Firstly, they proposed a mixed-integer linear programming model (MILP) and adaptive large neighborhood search (ALNS). Then, the results were compared between the solution from the MILP by the LINGO program and ALNS. Supattananon and Akararungruangkul [31] presented a combination of a web application and the modified differential evolution (MDE) algorithm for the vehicle dispatching problem (VDP). They modified the DE with the probability of accepting the solution in the four different equations. The results demonstrated that the MDE outperformed the traditional DE.

Reviewing of the open innovation dynamics literature demonstrates that a combined approach of metaheuristic techniques such as genetic algorithm (GA), particle swarm optimization (PSO), and differential evolution (DE) algorithm in the field, planning the distribution of SMEs/SMIs, has not been used yet.

In 2015, Montoya-Torres et al. [32] provided a state-of-the-art survey on vehicle routing with multiple depots (MDVRP). Most of the initial papers on MDVRP considered transportation cost minimization. Moreover, due to the intricacy of transportation issues, there are numerous elements which should be as near to a real-world scenario as possible, including time windows, split delivery, heterogeneous fleets, periodic deliveries, and pickup and delivery.

Ho et al. (2008) [33] considered the topic of reverse logistics for a company that wants to collect cores from dealers during campaign seasons. Moreover, they proposed the distribution of products from multiple warehouses to sample customers in the transportation industry. This study developed two hybrid genetic algorithms (HGAs) to deal with the challenge efficiently. The primary distinction between the HGAs was that, in HGA1, the initial solutions were produced at random. For the initialization technique, HGA2 used the Clarke and Wright savings approach and the closest neighbor heuristic. Computational modeling was carried out in order to compare the methods with different issue sizes. In terms of overall delivery time, HGA2 outperformed HGA1. Aras, Aksen, and Tekin (2011) [34] presented two mixed-integer linear programming (MILP) models in MDVRP. They developed a Tabu search-based heuristic technique to modify medium- and large-sized cases, as the issue is NP-hard.

Sombuntham and Kachitvichyanukul (2010) [35] developed a particle swarm optimization algorithm with multiple social-learning structures (GLNPSO) to solve MDVRP with simultaneous pickup and delivery time windows. They created a new decoding technique, and their preliminary findings suggested that the proposed method could effectively solve most of the test issues.

Subsequently, many studies have focused on local search algorithms in MDVRP. A heuristic approach can tackle issues that are computationally difficult to solve. A local search may be employed to address issues when moving from one solution to the next in the context of candidate solutions, by making local adjustments until a solution which is deemed optimal is discovered, or a time limit has passed. Kuo and Wang (2012) [36] offered a variable neighborhood search (VNS) solution for the MDVRP with loading costs (LC). There were three phases in the proposed VNS: (1) Initial solution generation; (2) random neighborhood solutions; and (3) neighborhood solution acceptance by simulated annealing (SA). Their findings demonstrated that the proposed technique is both efficient and successful in addressing the related issues. In 2016, Alinaghian and Shokouhi [37] presented a hybrid algorithm composed of an adaptive large neighborhood search (ALNS) and VNS for a multi-compartment MDVRP. In the same year, Bezerra et al. [38] presented a modified randomized variable neighborhood descent (RVND) for solving MDVRP.

Sadati, Çatay, and Aksen (2021) [39] have recently developed a hybrid Tabu search and variable neighborhood search to escape from local optima; the algorithm was called the variable tabu neighborhood search (VTNS). The VTNS was used to solve three problems: MDVRP, MDVRPTW, and multi-depot open vehicle routing problem (MDOVRP). The study found that VTNS was competitive, in terms of the speed of solution, compared to state-of-the-art solution approaches published in the literature. Sethanan and Pitakaso (2016) [40] indicated a DE metaheuristic for the transportation of raw milk. In order to increase the quality of the solution, they developed five modified DE algorithms, including two new steps: reincarnation and survival processes. The modified DE algorithms offered higher efficiency in minimization of the total costs. Dechampai et al. (2015) [41] used the Multifactor Based Evolving Self-Organizing Maps with Differential Evolution for the General Q-Delivery Vehicle Routing Problem (G-Q-DVRP) with considerations of flexibility in mixing pickup and delivery services and the maximum duration of a route constraint (MESOMDE_G-Q-DVRP-FD) algorithm for the egg industry. The algorithm was beneficial for minimizing overall costs, compared to real-world cases, and for the efficient handling of a poultry production system.

Therefore, in the field of work related to the DE algorithm [40,41] we developed, in contrast to the above-mentioned works, the new mutation formula for re-initialization and new operation of using the *k*-valuable move algorithm in the DE algorithm.

Stodola (2018) [42] developed the ant colony optimization (ACO) theory to minimize the length of the longest route of all vehicles in the standard MDVRP. Later, Stodola (2020) [43] used a hybrid ant colony optimization algorithm. Mutual colony optimization was conducted twice. ACO applies the local optimization process, and updates pheromone trails according to selected solutions in a current generation, using the simulated annealing technique for decisions. The algorithm was found to minimize the total distance and the longest route for all vehicles. Zhang et al. (2020) [44] presented a multi-depot green vehicle routing problem considering alternative fuel-powered vehicles with limited fuel tank capacity. They proposed a two-stage ant colony system (TSACS). The proposed TSACS is distinguished through the utilization of two types of ants for two different objectives: the first type of ant was used to assign customers to depots, while the second type of ant was assigned to find the routes. The overall goal was to reduce total carbon emissions as much as possible. This approach could effectively reduce the total carbon emissions. Shi, Hu, and Han (2020) [45] considered waste collection problems from waste collection points to waste disposal plants formulated in terms of MDVRP. They presented the sector combination optimization (SCO) algorithm to generate initial solutions and used the merge-head and drop-tail (MHDT) strategy in the process of updating solutions to minimize the total transportation distance. This algorithm provided more effective solutions compared to the other algorithms presented, and used the lowest computational time in the tabu search to obtain near-optimal solutions. Zhen et al. [46] investigated a multi-depot multi-trip vehicle routing problem with time windows and release dates. They developed a mixed integer programming model for small-sized instances, and a hybrid particle swarm optimization algorithm and a hybrid genetic algorithm for large-sized instances. A summary of the past work on MDVRP is shown in Table 1.

Table 1. Summary of past work on MDVRP.

Author	Year	Solution Approach
Ho et al. [33]	2008	Initial solution with Hybrid based on GA and LS
Sombuntham and Kachitvichyanukul [35]	2010	PSO with multiple social learning structures
Aras, Aksen, and Tekin [34]	2011	A rich neighborhood TS (TS-RN) heuristic
Kuo and Wang [36]	2012	Initial solution and VNS
Dechampai et al. [41]	2015	DE algorithm
Sethanan and Pitakaso [40]	2016	DE algorithm
Alinaghian and Shokouhi [37]	2016	Hybrid based on ALNS and VNS
Bezerra et al. [38]	2018	Randomized VND
Stodola [42]	2018	Ant Colony Optimization (ACO)
Shi, Hu, and Han [45]	2020	Sector Combination Optimization (SCO) algorithm with merge-head and drop-tail (MHDT) strategy.
Stodola [43]	2020	Hybrid based on ACO and SA
Zhen et al. [46]	2020	Hybrid based on PSO and GA
Zhang et al. [44]	2020	A Two-stage ACO
Sadati, Çatay, and Aksen [39]	2021	Variable Tabu Neighborhood Search (VTNS)

3. Problem Statement and Mathematical Model

3.1. The Problem Statement

The logistic flow problem is also known as the vehicle routing problem (VRP). The VRP is an NP-hard combinatorial optimization problem that is designed to improve logistic planning to minimize the total cost. In this paper, we considered the outbound logistics for the poultry industry in Thailand. The outbound logistic in this case study is characteristic of multiple-depot vehicle routing problems. In summary, transportation from the depot to the customers must follow a circular route, with the depot serving as both a starting node and a destination node. When considering only one depot, the problem is known as a single-depot problem, or VRP (see Figure 2a). Meanwhile, when the number of depots increases, it is known as the multi-depot vehicle problem (see Figure 2b). The objective of this paper is to find the best plan for the poultry industry that minimizes transportation costs.

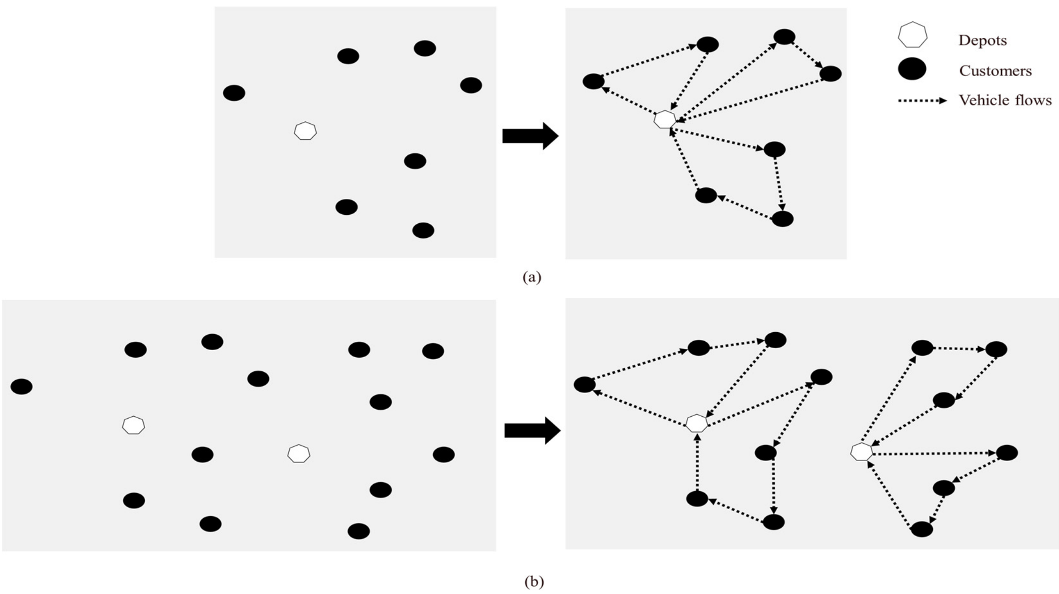

Figure 2. The logistic flow problem: (**a**) Vehicle routing problem (VRP); and (**b**) multi-depot vehicle routing problem (MDVRP).

3.2. The Mathematical Model

This paper used the mixed-integer linear programming of MDVRP based on the first presented by Kulkarni and Bhave [47]. The mathematical model of the MDVRP defines a three-index binary decision, where the binary decision variable x_{ijk} is equal to 1 when linking the arc between customers i and j in the route of vehicle k, and 0 otherwise. We consider assumptions that are used in the poultry distribution industry based on MDVRP:

(1) Each vehicle must depart from and return to the same depot;
(2) Each customer is served exactly once by one vehicle;
(3) The customer's demand does not exceed the vehicle capacity;
(4) All customers needs are met; and
(5) The total time of a route does not exceed the maximum duration time of the route.

The following notation is used in the model:

Sets and indices:

M	Set of poultry distribution center (depot) nodes
N	Set of customer nodes
K	Set of vehicles
i, j	Index of nodes; $i, j = 1, \ldots, N + M$
k	Index of vehicles; $k = 1, \ldots, K$

Parameters:

c_{ij}	Transportation cost between customers i and j
t_{ij}	Transportation time between customers i and j
Q_i	Demand at customer i
P_k	Capacity of vehicle k
T_k	Maximum duration time of route allowed for vehicle k

Decision variables:

x_{ijk}	Equal to 1 if the vehicle k is traveling from customer i to customer j, zero otherwise
u_i	An integer variable that defines the order of vertices visited on a tour for the elimination of sub-tours

The formulation of poultry distribution based on MDVRP can be stated as follows:

$$Minimize\ Z = \sum_{i=1}^{N+M} \sum_{j=1}^{N+M} \sum_{k=1}^{K} c_{ij} x_{ijk} \tag{1}$$

Subject to:

$$\sum_{i=1}^{N+M} \sum_{k=1}^{K} x_{ijk} = 1 \qquad j = 1, \ldots, N \tag{2}$$

$$\sum_{j=1}^{N+M} \sum_{k=1}^{K} x_{ijk} = 1 \qquad i = 1, \ldots, N \tag{3}$$

$$\sum_{i=1}^{N+M} x_{ihk} - \sum_{j=1}^{N+M} x_{hjk} = 0 \qquad \begin{array}{l} k = 1, \ldots, K \\ h = 1, \ldots, N+M \end{array} \tag{4}$$

$$\sum_{i=1}^{N+M} \sum_{j=1}^{N+M} Q_i x_{ijk} \leq P_k \qquad k = 1, \ldots, K \tag{5}$$

$$\sum_{i=1}^{N+M} \sum_{j=1}^{N+M} t_{ij} x_{ijk} \leq T_k \quad k = 1, \ldots, K \tag{6}$$

$$\sum_{i=N+1}^{N+M} \sum_{j=1}^{N} x_{ijk} \leq 1 \qquad k = 1, \ldots, K \tag{7}$$

$$\sum_{j=N+1}^{N+M} \sum_{i=1}^{N} x_{ijk} \leq 1 \qquad k = 1, \ldots, K \tag{8}$$

$$u_i - u_j + (N+M) x_{ijk} \leq (N+M) - 1 \quad 1 \leq i \neq j \leq N\ and\ 1 \leq k \leq K \tag{9}$$

$$x_{ijk} \epsilon \ \{0, 1\} \qquad \begin{array}{l} k = 1, \ldots, K \\ i, j = 1, \ldots, N+M \end{array} \tag{10}$$

$$u_i \geq 0 \qquad i = 1, \ldots, N+M \tag{11}$$

The objective function (1) of the poultry industry distribution is to minimize the transportation cost. Constraints (2) and (3) denote a single visit, and that a single vehicle serves the customer. Constraints (4) states that any vehicle that visits a customer should also depart from that customer. Constraints (5) ensure that the demand at customer i (Q_i) cannot exceed its vehicle capacity (P_k). Constraint (6) ensures that a vehicle k is returned to the depot no later than the maximum duration time (T_k). Vehicle availability is verified by constraints (7) and (8). Constraint (9) is for sub-tour elimination. Finally, constraint (10) and (11) serves as the basis for the decision variables.

4. New Enhanced Differential Evolution Algorithms

The differential evolution (DE) algorithm is a population-based search, which was proposed by Storn and Price [48]. The DE algorithm includes three operations: Mutation, recombination, and selection. The vector containing dimension (D) variables is denoted by $x_{i,j}^t$. The pseudocode of the traditional DE algorithm is shown in Algorithm 1. This section presents a new, enhanced DE algorithm with a new mutation formula to adapt between the mutation formula DE/rand/1 and the local mutation vector and local search function. The proposed algorithm is named RI-DE, and its steps are shown in Algorithm 2. The details of the procedures are as follows.

Algorithm 1 The pseudocode of DE

Define the objectives function *f(x)*
Generate the initial population of *NP*
while t < Max(number of iteration)
  for each x_i^t in the population do
  Random vector $r_1, r_2, r_3 \in$ (1,NP), with $r_1 \neq r_2 \neq r_3$
  Generete a random integer $j_{rand} \in (1, D)$
    for each parameter j do
      Generate v_i^t through Equation (12)
      Generate u_i^t through Equation (13)
    end for
  Replace x_i^t with the u_i^t if u_i^t is better through Equation (16)
  end for
end while

4.1. Initialization Operation

The initial individuals vector $x_{i,j}^0$ ($i = 1, \ldots, NP$) generates random numbers within the range [0, 1), and the maximum number of iterations t_{max} for the current generation t is set as 0 (note: For understanding x_i^t is a target vector i iteration t does not reference dimension j, which $x_{i,j}^t$ references dimension j, that the same variable).

4.1.1. New Mutation Operation

For each x_i^t in the population, the mutant vector v_i^{t+1} is generated by the mutation formula. The traditional DE uses the following mutation formula:

$$\text{DE/rand/1}: v_i^{t+1} = x_{r1}^t + F\left(x_{r2}^t - x_{r3}^t\right), \tag{12}$$

where r_1, r_2, and r_3 form a random vector chosen within the range [1,*NP*], and each r must be different (i.e., $r_1 \neq r_2 \neq r_3$). F is the is the scaling factor. The new enhanced DE is developed considering the following mutation operation.

The new mutation operation was developed considering the concept of protecting the moving trap solution, as the search performance of DE still needs to be improved. The mutant vector, v_i^{t+1}, is generated by the mutation formula as follows:

$$v_i^{t+1} = \begin{cases} x_{r_{new}}^t + F\left(x_{best}^t - x_i^t\right) + F\left(x_{r1}^t - x_{r2}^t\right), & if\ NI \geq RI \\ x_{r1}^t + F\left(x_{r2}^t - x_{r3}^t\right), & otherwise \end{cases} \tag{13}$$

where $x_{r_{new}}^t$, x_{best}^t, NI, and RI are the new random vector, the vector that obtains the best solution at iteration t (called the best vector), the number of iterations after which the solution does not improve, and the number of cumulative iterations in which the solution does not improve (function re-initialization is used), respectively. Moreover, the percentage of the number of populations (Ps) for random populations used Equation (13). We set the parameter Ps equal to 20% based on previous work [49]. We hope to use a new mutation formula to protect against trapping in local optima.

4.1.2. The Recombination Operation

For this study, we adopted the binominal recombination operation. For each target vector $x_{i,j}^0$, the trial vector $u_{i,j}^t$ was generated as follows:

$$u_{i,j}^{t+1} = \begin{cases} v_{i,j}^{t+1}, & if\ r_j \leq CR\ or\ j = randn(i) \\ x_{i,j}^t, & otherwise, \end{cases} \tag{14}$$

where r_j, j, and CR are a random number within the range [0, 1], a random integer within the number of dimensions D, and the crossover probability within the range [0, 1], respectively.

Algorithm 2: The pseudocode of RI-DE

Define the objectives function *f(x)*
Generate the initial population of *NP*
while t < Max(number of iteration)
  for each x_i^t in the population do
   Random vector $r_1,\ r_2,\ r_3\ \in (1{,}NP)$, with $r_1 \neq r_2 \neq r_3$
  Generete a random integer $j_{rand}\ \in (1, D)$
    for each parameter j do
      if $NI \leq RI$
      Generate v_i^t through Equation (12)
      else
        if Random each x_i^t in population with $Ps\%$
         Generate v_i^t through Equation (13)
         Generate u_i^t through Equation (14)
        else
         Generate v_i^t through Equation (12)
         Generate u_i^t through Equation (14)
        end
      end
      Call local search function
      Generate l_i^t through local search function
      Replace u'^t_i with the l_i^t if l_i^t is better, and u_i^t otherwise through Equation (15)
    end for
  Replace x_i^t with the u'^t_i if u'^t_i is better through Equation (16)
  end for
end while

4.1.3. The Local Search Functions

The RI-DE algorithm used a local search algorithm (e.g., swap function, insert function, and 2-opt function), in order to improve the exploitation ability of the search in the context of protecting movements to a local optimum. To create a new vector, the local search function (called the local vector, l_i^t) from Equation (15), is operated in each dimension j by the local search function to create the $l_{i,j}^t$. The new trial vectors $u'^t_{i,j}$ are selected as follows:

$$u'^t_{i,j} = \begin{cases} l_{i,j}^t\ , & if\ f\left(l_{i,j}^t\right) \leq f\left(u_{i,j}^t\right) \\ u_{i,j}^t\ , & otherwise \end{cases} \tag{15}$$

Our proposed algorithm uses a local search name k-variable move algorithm [50]. The k-variable move is an extended version of the swap algorithm, in which k continues to move from a k position to the next k position until the last k moves to the first k in D.

For example, after the trial vector $u_{i,j}^t$ is created (see Figure 3), a k-variable move (where k = 3) is shown in Figure 3. The trial vector is randomized in three positions {3,7,5} (see Figure 3a), swapping from the first position to the next position. Finally, the third position must move to the first position (see Figure 3b).

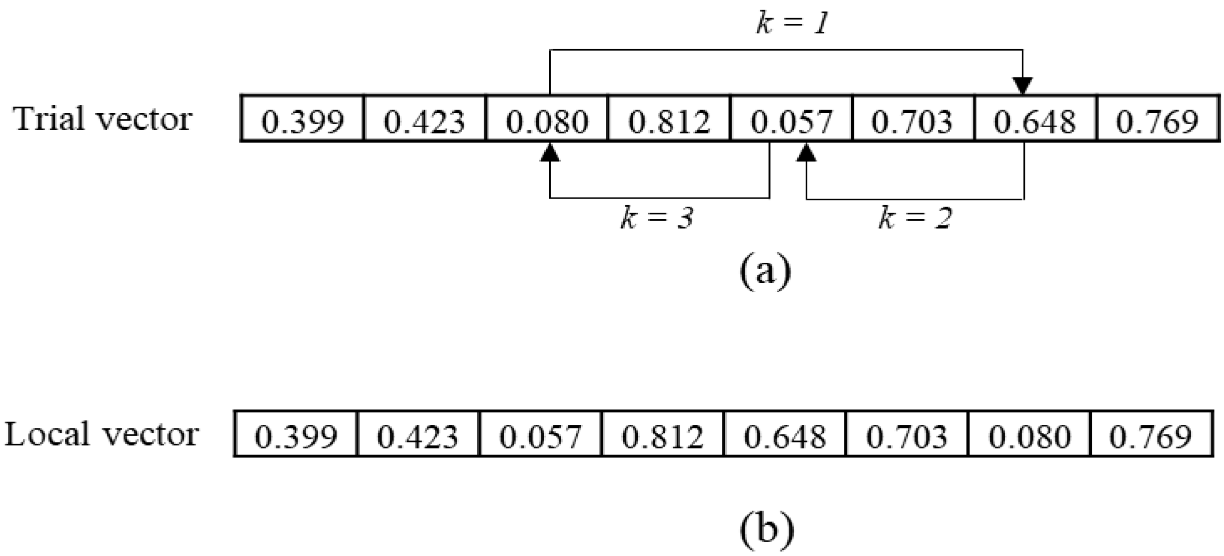

Figure 3. The example for *k*-variable move function: (**a**) The trial vector; and (**b**) trial vector after the *k*-variable move (local vector).

In this paper, we used k = 20% based on previous work [49], which means that k should be the random number for moving up to around 20% of the customers.

4.1.4. Selection Operation

If the fitness value of the trial vector is better than the target vector, then the trial vector replaces the target vector with a pre-target vector in the next iteration based on a greedy selection.

$$x_{i,j}^{t+1} = \begin{cases} {u'}_{i,j}^{t}, & if\ f\left({u'}_{i,j}^{t}\right) \leq f\left(x_{i,j}^{t}\right) \\ x_{i,j}^{t}, & otherwise \end{cases} \tag{16}$$

In the brief discussion of Algorithm 1, each individual of the trial vector is compared to the objective function with its target vector, and the better one is selected for the next iteration. Then, these steps are repeated until the stopping criterion is reached. Moreover, the critical difference between Algorithms 1 and 2 is that in Algorithm 2, when the solution does not develop, Equation (13) is used, the individual target vector is random with $PS\%$ chance to be the local vector, and the better one between the trial vector and local vector is chosen to be new trial vector that is compared with its target vector.

4.2. The Decoding Method

The solution to our proposed problem is the use of multiple vehicle routes for the poultry distribution problem to deliver products from the multi-depot to customers within the limited number of vehicles at each depot. The following example shows the decoding method: suppose we have two depots. Then, the limited number of vehicles at each depot is two, the vehicle capacity in a homogeneous fleet is equal to 40, and there are 10 customers. The demand of the 10 customers is shown in Table 2.

Table 2. The demand of 10 customers and depot assigned.

Customers	Demands	Depot Assigned
1	10	1
2	12	2
3	10	1
4	15	2
5	18	2
6	10	1
7	12	2
8	16	1
9	10	2
10	14	1

In our proposed algorithm, the dimension in the vector represents the solutions. In the first array, each dimension is identified by a customer sequence, which is sorted by increasing values. In the second array, each dimension is identified by a corresponding random real-number sequence (see Figure 4a). Therefore, the customer sequence of each depot can be obtained by rank order value (ROV), which decodes the ascending order (see Figure 4b).

Customers	1	2	3	4	5	6	7	8	9	10
Random real number	0.26	0.14	0.78	0.54	0.13	0.64	0.89	0.34	0.47	0.67

(a)

5	2	1	8	9	4	6	10	3	7
0.13	0.14	0.26	0.34	0.47	0.54	0.64	0.67	0.78	0.89

ROV with ascending order

(b)

Figure 4. The decoding method of our proposed algorithm: (**a**) Vector that has not yet been operated with the decoding method; and (**b**) decoded vector.

4.2.1. The Customer Assigned

The customer assigned to each depot is chosen based on the greedy algorithm. The customers must select a closer depot for delivering products for them. The objective of this paper is to minimize the overall transportation costs. Thus, the customers are assigned to the nearest depots. For example, each of the 10 customers should be assigned to nearby depots, as shown in Table 2, column 3.

4.2.2. Solution Representation of RI-DE Algorithm

Each vector is structured as a double array, with the length being the number of customers in this algorithm (∣N∣). Figure 4 depicts an example of the decoding method of RI-DE, where ∣N∣ equals 10. The vector with random positioning of customers is shown in Figure 4a. The customers in {1, 3, 6, 8, 10} are assigned to depot 1. Similarly, customers in {2, 4, 5, 7, 9} are assigned to depot 2. Therefore, the customer sequences are depicted as {5, 2, 1, 8, 9, 4, 6, 10, 3, 7} (see Figure 4b), in which the customer sequences of depot 1 and depot 2 are {1, 8, 6, 10, 3} and {5, 2, 9, 4, 7}, respectively. The routing of vehicles for each customer is available with respect to the vehicle capacity and the maximum number of vehicles at each depot.

(1) Route No. 1: Vehicle No. 1 at Depot 1. The routing assignment: The vehicle has to deliver products to customers {1,8,6} with amounts of 10, 16, and 10, respectively, as

the products loaded are inadequate to meet the demand of the following customers: customer number 10 had a demand of 14, but there were only four remaining products (40–10−16–10 = 4). Hence, the vehicle must then return to Depot 1.

(2) Route No. 2: Vehicle No. 2 at Depot 1. This is similar to Route No. 1 at Depot 1, but Vehicle No. 2 has to deliver to the remaining customers ({10,3}).

(3) Route No. 3: Vehicle No. 3 at Depot 2. The routing assignment: The vehicle has to deliver products to customers {5, 2, 9} with amounts of 18, 12 and 10, respectively; however, the products loaded were not enough to meet the demand of the following customers: customer number 4 had a demand of 15, but there were no remaining products (40–18−12–10 = 0). The vehicle must return to Depot 2.

(4) Route No. 4: Vehicle No. 4 at Depot 2. This is similar to Route No. 3 Vehicle No. 3 at Depot 2., but Vehicle No. 2 at Depot 2 has to deliver to the remaining customers ({4, 7}).

Suppose the number of vehicles in service is insufficient to meet the demands of the customers. In that case, it is necessary to move the number of customers that cannot be served to the next nearest depot. For example, customers {5,3,2,1,6,7} are assigned to be serviced by depot 1 (see Figure 5a), while customers {4,8,9,10} must be serviced by depot 2. It is found that the number of vehicles at depot 1 can only serve customers {5,3,2,1}, while customers {6,7} cannot be serviced as the number of vehicles at depot 1 are inadequate. Thus, it is necessary to assign customers {6,7} to the next nearest depot to receive service. Suppose the next nearest depot is depot 2, where the sequence will be appended to the original order, becoming {4,8,9,10,6,7} (see Figure 5b).

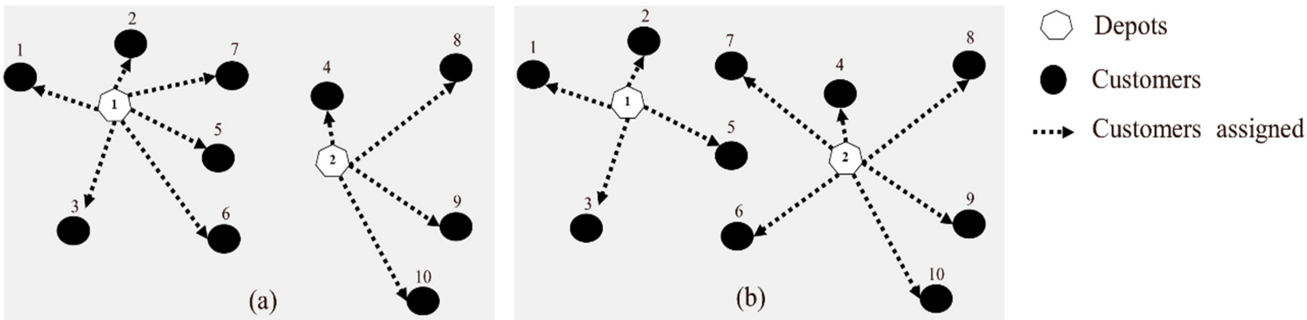

Figure 5. Example of customers assigned with 2 depots and 10 customers: (**a**) Customers {6,7} assign to depot 1; and (**b**) customers {6,7} assign to depot 2.

5. Computational Experiment

In this section, the experiments were executed on a computer with the following parameters: Intel® Core™ i7-8750H CPU @ 2.20 GHz, 2.21 GHz RAM, and 16.0 GB. We developed the mathematical model in LINGO software, based on the branch and bound method, and our proposed algorithms were coded in MATLAB (R2018a). The implementation of our proposed algorithm requires parameters, as provided in Table 3 [7,51–53]. We divided the parameters into small- and large-sized instances, shown in the third and fourth columns, respectively. For each experimental set, we attempted 15 replicates. Comparisons based on several population-based algorithms were discovered as well, such as traditional differential evolution (DE) algorithm, genetic algorithm (GA) [54,55], and particle swarm optimization (PSO) [55].

The results of our proposed algorithms were compared with LINGO for small-sized instances, as illustrated in Table 4, which is organized as follows. The first column contains the number of the instance *ID*. Column 2 contains the instance *ID*. In addition, each instance comprises three parts: the number of depots, the number of customers, and the number of vehicles. In column three, we show the optimal solution obtained by LINGO. Columns 4–12 show the best, average, and worst solutions obtained by the genetic algorithm (GA), particle swarm optimization (PSO), traditional differential evolution (DE), and re-initialization differential evolution (RI-DE) algorithm. We tested a total of 25 instances with 5, 10, 15, 20, and 25 customers. The number of the depots was 2, 3, and 4 depots. The numbers of vehicles at each depot were 3, 4, 5, and 6 vehicles. We used the best solution obtained from the 15 replicates of our proposed algorithm to demonstrate the transportation cost. The computational time of our proposed algorithms is shown in Table 5.

Table 3. Parameters of our proposed algorithm.

Symbol	Meaning	Small-Sized Instance	Large-Sized Instance
T_{max}	Maximum number of iterations	300	500
NP	Number of populations	25	50
F	Scaling factor	2.0	2.0
CR	Crossover Rate	0.8	0.8
RI	Re-initial factor	30	50

The case study (CS) on this research was motivated by the small and medium-sized enterprises (SMEs) in northeastern Thailand. Currently, the total number of distribution centers is 6 with customers numbering about 500. To deliver to the customers, the case-study company has a contract with a third-party delivery company, which provides about 250 vehicles.

In Table 6, the heuristic performance (HP) percentage is shown, which was calculated as $HP = (S_L/S_o) \times 100$. Here, S_L and S_o are the solutions obtained by LINGO and the solution obtained by our proposed algorithm in Table 4, respectively. The statistical test results in the solution obtained by our proposed algorithm using the Wilcoxon test are shown in Table 7. We used the Wilcoxon test because it neither depends on the form of the parent distribution nor its parameters and does not depend on any assumptions about the shape of the distribution or on being normally distributed. The statistical test was performed based on the transportation cost of LINGO and our proposed algorithm, and the SPSS V14 software for Windows was used to carry out the statistical analysis.

Table 4. Transportation cost of our proposed algorithm for small-sized instances.

No.	Instance *ID*	Transportation Cost												
		LINGO	GA			PSO			DE			RI-DE		
			Best	Avg.	Worst	Best	Avg.	Worst	Best	Avg.	Worst	Best	Avg.	Worst
1	In2-10-3	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**	**4754.79**
2	In2-10-3	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**	**3419.17**
3	In2-10-3	**3355.35**	**3355.35**	3361.91	3371.68	**3355.35**	**3355.35**	**3355.35**	**3355.35**	**3355.35**	**3355.35**	**3355.35**	**3355.35**	**3355.35**
4	In2-10-3	**5811.17**	**5811.17**	5824.88	5854.59	**5811.17**	5821.27	5848.89	**5811.17**	**5811.17**	**5811.17**	**5811.17**	**5811.17**	**5811.17**
5	In2-10-3	**5405.38**	**5405.38**	5420.44	5452.22	**5405.38**	5414.85	5437.56	**5405.38**	5412.15	5435.45	**5405.38**	**5405.38**	**5405.38**
6	In2-15-4	**6292.73**	**6292.73**	6302.73	6328.22	**6292.73**	6303.81	6336.78	**6292.73**	6299.78	6323.76	**6292.73**	**6292.73**	**6292.73**
7	In2-15-4	**7775.00**	**7775.00**	7795.30	7832.59	**7775.00**	7788.91	7842.48	**7775.00**	7786.51	7822.35	**7775.00**	7781.95	7816.56
8	In2-15-4	**8323.10**	**8323.10**	8339.58	8376.87	**8323.10**	8337.36	8380.55	**8323.10**	8333.38	8365.84	**8323.10**	8325.47	8335.13
9	In2-15-4	**6263.96**	**6263.96**	6278.28	6304.50	**6263.96**	6272.03	6283.38	**6263.96**	6272.60	6301.04	**6263.96**	6269.66	6298.98
10	In2-15-4	**10,708.31**	**10,708.31**	10,733.77	10,785.34	**10,708.31**	10,728.70	10,792.46	**10,708.31**	10,723.25	10,762.20	**10,708.31**	10,717.51	10,759.03
11	In2-20-5	11,752.80	12,366.13	12,396.17	12,452.71	12,125.96	12,163.71	12,210.02	12,048.52	12,069.69	12,114.07	11,765.05	11,778.60	11,814.05
12	In2-20-5	17,256.92	18,272.71	18,330.35	18,449.57	18,447.77	18,504.03	18,603.07	17,741.07	17,770.48	17,817.13	17,617.10	17,650.48	17,710.62
13	In2-20-5	11,333.43	12,247.92	12,286.08	12,323.79	11,572.32	11,611.25	11,668.84	11,646.04	11,668.50	11,726.48	11,460.67	11,479.88	11,517.08
14	In2-20-5	10,752.50	11,476.59	11,505.77	11,551.50	11,021.77	11,062.89	11,120.57	10,993.35	11,032.68	11,050.87	10,997.47	11,021.31	11,066.12
15	In2-20-5	13,712.95	14,464.44	14,490.47	14,523.75	14,260.33	14,297.85	14,356.45	14,373.83	14,401.39	14,461.48	14,192.19	14,219.51	14,261.92
16	In3-20-5	13,123.55	13,845.44	13,884.53	13,952.46	13,438.09	13,487.34	13,557.08	13,558.32	13,601.09	13,657.74	13,249.50	13,274.50	13,337.62
17	In3-20-5	10,777.28	11,337.92	11,364.47	11,392.98	11,017.87	11,343.60	11,391.68	11,293.19	11,319.52	11,365.95	11,061.50	11,078.58	11,119.02
18	In3-20-5	10,902.53	11,535.25	11,573.11	11,625.81	11,440.83	11,474.40	11,522.53	11,218.21	11,250.43	11,299.06	10,937.57	10,950.59	11,002.75
19	In3-20-5	8511.74	9083.85	9112.99	9153.57	8924.62	8959.61	8994.24	8602.79	8610.54	8633.11	8595.39	8609.01	8646.73
20	In3-20-5	12,674.18	13,369.58	13,403.44	13,464.31	13,231.46	13,266.27	13,307.45	13,093.37	13,115.04	13,173.06	13,041.96	13,058.42	13,096.90
21	In4-25-6	17,493.36	18,567.10	18,612.52	18,716.55	18,038.85	18,184.90	18,253.68	17,707.57	18,104.00	18,236.30	18,056.50	18,093.86	18,183.47
22	In4-25-6	11,540.92	12,304.64	12,348.42	12,397.44	12,021.83	12,057.42	12,133.14	11,790.48	11,815.73	11,856.01	11,775.83	11,796.24	11,827.22
23	In4-25-6	11,026.53	11,656.89	11,696.79	11,755.03	11,539.67	11,570.91	11,622.78	11,493.51	11,527.66	11,582.66	11,047.37	11,055.88	11,076.36
24	In4-25-6	20,052.29	21,233.29	21,300.74	21,377.40	20,511.95	20,556.37	20,639.39	20,528.55	20,570.91	20,648.21	20,344.64	20,391.66	20,469.14
25	In4-25-6	13,121.14	14,316.55	14,358.68	14,408.74	13,257.00	13,297.78	13,373.62	13,445.43	13,478.90	13,542.73	13,243.98	13,267.26	13,291.11
CS	In6-500-250	a	148,323.04	148,459.56	148,523.61	147,820.78	147,932.51	148,123.42	146,961.10	147,059.76	147,152.23	146,897.64	146,989.41	147,080.32
Avg (Number 1 to 25)		10,245.64	10,727.49	10,755.81	10,801.02	10,518.37	10,561.38	10,608.24	10,465.73	10,500.19	10,540.64	10,379.83	10,394.36	10,426.74

[a] Problem is too large to be solved by LINGO.

Table 5. Computational time of our proposed algorithm for small-sized instances.

Number	Instance ID	Computational Time (s.)				
		LINGO	GA (avg.)	PSO (avg.)	DE (avg.)	RI-DE (avg.)
1	In2-10-3	2.65	1.27	1.13	1.22	2.31
2	In2-10-3	1.79	1.18	1.22	1.32	2.12
3	In2-10-3	2.30	1.22	1.24	1.14	2.26
4	In2-10-3	2.77	1.19	1.29	1.27	2.21
5	In2-10-3	1.30	1.25	1.09	1.21	1.72
6	In2-15-4	311.31	1.81	1.93	1.91	2.52
7	In2-15-4	253.57	1.66	1.69	1.78	2.89
8	In2-15-4	226.19	1.64	1.69	1.79	3.54
9	In2-15-4	327.04	1.74	1.75	1.81	2.67
10	In2-15-4	327.53	1.79	1.76	1.82	2.84
11	In2-20-5	8632.18	2.49	2.42	2.74	4.08
12	In2-20-5	11,825.68	2.42	2.65	2.36	4.00
13	In2-20-5	13,446.76	2.15	2.16	2.76	4.20
14	In2-20-5	12,711.19	2.25	2.34	2.15	4.14
15	In2-20-5	9548.06	2.36	2.40	2.59	3.81
16	In3-20-5	16,024.76	3.56	4.13	3.86	5.18
17	In3-20-5	13,802.22	3.80	3.96	3.82	5.53
18	In3-20-5	14,967.13	3.28	3.80	3.25	4.94
19	In3-20-5	17,021.28	3.81	3.55	3.90	6.63
20	In3-20-5	17,902.19	3.77	3.62	3.91	6.15
21	In4-25-6	18,947.25	5.37	5.77	5.66	10.99
22	In4-25-6	24,137.45	6.26	5.51	6.57	10.94
23	In4-25-6	21,703.05	6.16	6.15	6.91	10.51
24	In4-25-6	24,492.81	5.91	5.90	6.37	9.24
25	In4-25-6	21,441.77	6.21	6.13	6.79	8.27
Case study	In6-500-250	[a]	174.83	201.69	180.37	324.92
Avg. (Number 1 to 25)		9922.40	2.98	3.01	3.16	4.95

[a] Problem is too large to be solved by LINGO.

Table 6. The heuristic performance of our proposed algorithm for small-sized instances.

Number	Heuristic Performance%			
	GA	PSO	DE	RI-DE
1	100.00	100.00	100.00	100.00
2	100.00	100.00	100.00	100.00
3	100.00	100.00	100.00	100.00
4	100.00	100.00	100.00	100.00
5	100.00	100.00	100.00	100.00
6	100.00	100.00	100.00	100.00
7	100.00	100.00	100.00	100.00
8	100.00	100.00	100.00	100.00
9	100.00	100.00	100.00	100.00
10	100.00	100.00	100.00	100.00
11	95.04	96.92	97.55	99.90
12	94.44	93.54	97.27	97.96
13	92.53	97.94	97.32	98.89
14	93.69	97.56	97.81	97.77
15	94.80	96.16	95.40	96.62
16	94.79	97.66	96.79	99.05
17	95.06	97.82	95.43	97.43
18	94.51	95.29	97.19	99.68
19	93.70	95.37	98.94	99.03
20	94.80	95.79	96.80	97.18
21	94.22	96.98	98.79	96.88
22	93.79	96.00	97.88	98.01
23	94.59	95.55	95.94	99.81
24	94.44	97.76	97.68	98.56
25	91.65	98.98	97.59	99.07
Avg.	96.48	97.97	98.33	99.03

Table 7. Statistical test results of differences of the solutions obtained from our proposed algorithm for small-sized instances.

	GA	PSO	DE	RI-DE
LINGO	0.001	0.001	0.001	0.001
GA		0.002	0.001	0.001
PSO			0.233	0.002
DE				0.011

Our proposed algorithm could also obtain the optimal solutions for instances 1 to 10 from Table 4 (see bold numbers in Table 4). However, for instances 11 to 25, our proposed algorithm found near-optimal solutions. The best solutions for our proposed algorithms were 12/15, 2/15, 1/15, and 0/15 times by RI-DE, DE, PSO, and GA, respectively, which implies the new formula of the re-initialization DE algorithm and the local search function can improve protection against trapping in local optima for small-sized instances. The average computational time of the mathematical model (LINGO) and our proposed algorithms were 9922.40, 2.98, 3.01, 3.16, and 4.95 s with LINGO, GA, PSO, DE, and RI-DE, respectively.

The statistical test results demonstrated that the RI-DE algorithm obtained solutions that differed in a statistically significant manner from the other algorithms (p-value ≤ 0.05). In addition, in our experiments, we found that the DE and PSO did not significantly differ. In terms of small-sized instances, RI-DE outperformed the other proposed algorithms.

In large-sized instances, where LINGO cannot obtain solutions, our proposed algorithm was evaluated on Cordeau's benchmark instances. The best knowledge solutions were taken from NEO Web, with all instances available for download at https://neo.lcc.uma.es/vrp/vrp-instances (23 September 2021).

Table 8 shows some properties of Cordeau's benchmark instances for each instance; the numbers of nodes (N) and depots (M) are shown, as well as the results of the GA, PSO, traditional DE and RI-DE algorithms, in terms of the best and average solutions found. Our proposed algorithm ran 10 replicates on all instances. The best solutions, average solutions, and average computational time of our proposed algorithm are recorded. The numbers in bold in the best solutions columns record the best-known solutions (BKS) taken from our proposed algorithm.

The results for the 23 instances of Cordeau's benchmark in Table 9 report the deviation from the best-known solutions (Δ) for our proposed algorithms. The last column reports the percentage relative improvement ($RI\%$) between the traditional DE and RI-DE algorithm, calculated by $RI = (S_{DE} - S_{RI\text{-}DE}/S_{DE}) \times 100$. Here, S_{DE} and $S_{RI\text{-}DE}$ are the solutions obtained by the traditional DE and RI-DE algorithms, respectively. The average deviation from the best-known solutions (Δ) for different GA, PSO, DE and RI-DE algorithms was 9.98%, 6.15%, 4.05% and 1.48%, respectively. In addition, the average percentage of relative improvement between the traditional DE and RI-DE algorithms was 2.53%. Also, when considering the large-sized instances from Cordeau's benchmark, it was demonstrated that the RI-DE algorithm outperforms the traditional DE algorithm.

The discussion of metaheuristics and open innovation is applied to the features of outbound logistics for distribution in order to develop the new optimized technique. in our opinion, the metaheuristic can be applied to minimize the transportation of the poultry industry in Thailand. One of the complexities of multiple vehicle routing problems is the NP-hard problem [47]. However, many researchers have considered the problem of open innovation [29–31]. To develop the optimization software for finding the optimal solution for the vehicle routing problem, it is discussed how the open innovation concept can be applied to the metaheuristics technique, such as our proposed algorithm to industrial dynamics perspective. Moreover, this paper is virtually the first case in which the differential evolution algorithm has been enhanced using the re-initialization mutation formula and local search function, in terms of the theoretical implications of open innovation among the poultry industry SMEs in Thailand via the concept of open innovation [1].

Table 8. Comparison of DE and RI-DE with Cordeau's benchmark instances.

Instance		GA		PSO		DE		RI-DE		Computational Time			
(N/M)	BKS	Best	Avg.	Best	Avg.	Best	Avg.	Best	Avg.	GA Avg. (s.)	PSO Avg. (s.)	DE Avg. (s.)	RI-DE Avg. (s.)
P01 (50/4)	576.87	590.14	837.94	**576.87**	747.66	**576.87**	755.30	**576.87**	654.16	17.63	24.24	17.59	25.94
P02 (50/4)	473.53	503.55	700.36	**473.53**	628.52	**473.53**	583.29	**473.53**	535.82	18.21	23.61	18.74	28.88
P03 (75/5)	641.19	684.45	931.38	657.38	837.07	656.42	845.45	**641.19**	762.40	30.61	31.68	32.84	43.55
P04 (100/2)	1001.59	1064.98	1446.00	1055.00	1366.98	1025.98	1343.12	1014.08	1213.49	16.19	18.57	17.47	24.16
P05 (100/2)	750.03	841.58	1018.74	833.15	960.77	794.45	963.18	776.84	908.12	18.25	16.34	18.51	27.13
P06 (100/3)	876.5	1030.21	1267.55	954.99	1205.12	958.66	1159.09	879.17	994.95	23.92	24.00	26.04	39.07
P07 (100/4)	885.8	959.81	1262.26	897.78	1188.61	903.33	1156.47	887.68	1016.06	33.85	40.85	34.63	57.32
P08 (249/2)	4437.68	4817.86	6264.81	4624.79	6187.56	4562.23	6177.58	4513.52	5196.59	44.52	47.22	43.60	64.83
P09 (249/3)	3900.22	4513.86	5138.81	4266.08	5046.08	4097.08	5036.16	4050.07	4603.25	63.14	61.10	64.93	104.55
P10 (249/4)	3663.02	4211.41	4725.28	3936.37	4668.79	3814.88	4660.47	3730.75	4276.60	86.24	90.53	86.86	134.81
P11 (249/5)	3554.18	3821.44	4717.28	3795.95	4666.45	3615.38	4622.07	3578.34	4148.83	110.43	116.38	110.07	181.02
P12 (80/2)	1318.95	1464.75	1750.45	1374.12	1668.67	**1318.95**	1654.12	**1318.95**	1540.19	12.53	22.85	14.32	19.79
P13 (80/2)	1318.95	1447.21	1815.23	1412.07	1721.91	1422.76	1735.26	1362.13	1541.95	15.03	19.49	17.05	22.92
P14 (80/2)	1360.12	1418.77	1818.75	1417.71	1753.70	1393.68	1736.83	1385.47	1592.85	15.24	12.89	14.17	26.63
P15 (160/4)	2505.42	2773.68	3476.40	2737.91	3381.22	2610.03	3364.40	2597.64	3030.39	55.62	56.45	57.35	88.64
P16 (160/4)	2572.23	2898.63	3419.55	2805.70	3365.35	2676.14	3368.55	2579.16	2988.30	55.19	56.22	55.74	95.78
P17 (160/4)	2709.09	2882.47	3546.22	2846.79	3471.24	2765.00	3449.03	2716.64	3197.79	56.95	56.87	56.30	87.89
P18 (240/6)	3702.85	4164.11	5168.42	4164.39	5103.74	4133.41	5079.11	3787.42	4221.70	124.55	122.66	125.69	226.10
P19 (240/6)	3827.26	3945.80	4972.83	3865.73	4904.05	3911.28	4879.44	3844.66	4523.50	124.93	124.56	126.81	159.10
P20 (240/6)	4058.07	4861.87	5524.00	4497.35	5427.46	4345.94	5388.65	4278.40	4690.10	136.23	142.77	137.54	206.44
P21 (360/9)	5474.84	6046.16	7217.81	5810.89	7117.98	5550.54	7091.34	5510.11	6645.58	283.01	288.82	282.04	433.30
P22 (360/9)	5702.16	6603.52	8190.11	6401.79	8132.93	6395.11	8129.50	5830.91	6874.11	251.67	259.88	251.89	440.18
P23 (360/9)	6095.46	6490.57	7802.19	6267.43	7750.28	6236.93	7738.99	6107.00	7245.07	279.75	286.15	279.28	419.31
									Avg.	81.46	84.53	82.15	128.58

Table 9. Performance of GA, PSO, DE and RI-DE with Cordeau's benchmark instance.

Instance	ΔGA (%)	ΔPSO(%)	ΔDE (%)	ΔRI-DE (%)	*RI* (%)
P01	2.30	0.00	0.00	0.00	0.00
P02	6.34	0.00	0.00	0.00	0.00
P03	6.75	2.52	2.38	0.00	2.38
P04	6.33	5.33	2.44	1.25	1.17
P05	12.21	11.08	5.92	3.57	2.27
P06	17.54	8.96	9.37	0.30	9.04
P07	8.36	1.35	1.98	0.21	1.76
P08	8.57	4.22	2.81	1.71	1.08
P09	15.73	9.38	5.05	3.84	1.16
P10	14.97	7.46	4.15	1.85	2.26
P11	7.52	6.80	1.72	0.68	1.04
P12	11.05	4.18	0.00	0.00	0.00
P13	9.72	7.06	7.87	3.27	4.45
P14	4.31	4.23	2.47	1.86	0.59
P15	10.71	9.28	4.18	3.68	0.48
P16	12.69	9.08	4.04	0.27	3.76
P17	6.40	5.08	2.06	0.28	1.78
P18	12.46	12.46	11.63	2.28	9.14
P19	3.10	1.01	2.20	0.45	1.73
P20	19.81	10.82	7.09	5.43	1.58
P21	10.44	6.14	1.38	0.64	0.73
P22	15.81	12.27	12.15	2.26	9.68
P23	6.48	2.82	2.32	0.19	2.13
Avg.	9.98	6.15	4.05	1.48	2.53

As a result, it is essential to remember that it extends beyond traditional logistical boundaries. A concept model of open innovation developed in [26] aims to investigate current open innovation channels, which might motivate engineering research to increase open innovation and the creation of new open-business models from meta-heuristics.

6. Conclusions

This paper focused on the planning of outbound logistics for the poultry industry in Thailand. The problem involves the planning of distribution of poultry products where the distribution center has more than one depot, making the situation characteristic of the multi-depot vehicle routing problem (MDVRP), which aims to minimize transportation costs.

Considering the NP-Hardness of our proposed problem, a new enhanced DE algorithm composed of a re-initialization solution and a local search function was developed.

In the computational results, our proposed algorithm reached the optimal solution in small-sized instances (numbers 1–10). The average transportation costs of our proposed algorithm for small-sized instances for LINGO (exact method), GA, PSO, DE, and RI-DE were equal to 10,245.64, 10,727.49, 10,518.37, 10,465.73, and 10,379.83, respectively. The average computational times for LINGO (exact method), GA, PSO, DE, and RI-DE were equal to 9922.40, 2.98, 3.01, 3.16, and 4.95 s, respectively. The statistical test showed that the RI-DE solution obtained based on transportation costs was significantly different from the solutions from the GA, PSO, and DE algorithms. Heuristic performance (*HP*) indicated that GA, PSO, DE, and RI-DE obtained near-optimal results, with an average of 96.48, 97.97, 98.33 and 99.03% respectively. The experimental results showed that the IR-DE algorithm obtained a near-optimal solution ranging from 96.62% to 100% of 15 replicated runs. When solving large-sized instances on Cordeau's benchmark instances, the enhanced DE algorithm (RI-DE) returned 1.48% error on average, which was significantly lower than that of the traditional DE algorithm. Moreover, given that the relative improvement (*RI*) comparing the transportation cost obtained from the traditional DE to that of RI-DE was equal to 2.53% on average, the results show that the RI-DE algorithm provides better transportation cost that the DE algorithm, ranging from 0.00% to 9.68%.

The RI-DE algorithm demonstrated an ability to obtain effective solutions by using the re-initialization mutation formula and local search function. The re-initialization solution could protect against trapping in local optima when the solution did not improve and create new vectors to find the best solution. In addition, the local search function was used to enhance the exploitation searchability of the DE algorithm. This implies that the re-initialization mutation formula and the local search function significantly improved the DE algorithm.

Future work can be extended in the following directions: firstly, there is still much opportunity to extend our work in many aspects when there are multiple periods, multiple products, heterogeneous fleets, and time window constraints, which may also provide an interesting topic for minimizing the total cost, including transportation, holding, and hiring costs. Secondly, the performance of the algorithm will be assessed in other real-world environments involving difficult combinatorial optimization problems in logistics and supply management. Lastly, our proposed algorithm could be extended to solve problems in other industries, i.e., agriculture and food.

Author Contributions: Conceptualization, K.M. and K.S.; methodology, K.M.; software, K.M.; validation, K.M. and K.W.; formal analysis, K.M.; data curation, K.M.; writing—original draft preparation, K.M.; writing—review and editing, K.M., K.S. and K.W.; supervision, K.M. and K.S.; project administration, K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received Faculty of Engineering, Khon Kaen University, Thailand.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: This work is supported by the Research Unit on System Modeling for Industry, Department of Industrial Engineering, Faculty of Engineering, Khon Kaen University, Thailand.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sengupta, A.; Sena, V. Impact of Open Innovation on Industries and Firms—A Dynamic Complex Systems View. *Technol. Forecast. Soc. Chang.* **2020**, *159*, 120199. [CrossRef]
2. Shcherbakov, V.; Silkina, G. Supply Chain Management Open Innovation: Virtual Integration in the Network Logistics System. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 54. [CrossRef]
3. Mubarak, M.F.; Petraite, M. Industry 4.0 Technologies, Digital Trust and Technological Orientation: What Matters in Open Innovation? *Technol. Forecast. Soc. Chang.* **2020**, *161*, 120332. [CrossRef]

4. Woschank, M.; Dallasega, P. The Impact of Logistics 4.0 on Performance in Manufacturing Companies: A Pilot Study. *Procedia Manuf.* **2021**, *55*, 487–491. [CrossRef]
5. Barykin, S.Y.; Kapustina, I.V.; Sergeev, S.M.; Yadykin, V.K. Algorithmic Foundations of Economic and Mathematical Modeling of Network Logistics Processes. *J. Open Innov. Technol. Mark. Complex.* **2020**, *6*, 189. [CrossRef]
6. Boonmee, A.; Sethanan, K.; Arnonkijpanich, B.; Theerakulpisut, S. Minimizing the Total Cost of Hen Allocation to Poultry Farms Using Hybrid Growing Neural Gas Approach. *Comput. Electron. Agric.* **2015**, *110*, 27–35. [CrossRef]
7. Boonmee, A.; Sethanan, K. A GLNPSO for Multi-Level Capacitated Lot-Sizing and Scheduling Problem in the Poultry Industry. *Eur. J. Oper. Res.* **2016**, *250*, 652–665. [CrossRef]
8. Keupp, M.M.; Gassmann, O. Determinants and Archetype Users of Open Innovation. *R D Manag.* **2009**, *39*, 331–341. [CrossRef]
9. Han, J.; Heshmati, A. Determinants of Financial Rewards from Industry-University Collaboration in South Korea. *Int. J. Entrep. Innov. Manag.* **2016**, *20*, 235–257. [CrossRef]
10. Heikkilä, M.; Bouwman, H.; Heikkilä, J. From Strategic Goals to Business Model Innovation Paths: An Exploratory Study. *J. Small Bus. Enterp. Dev.* **2018**, *25*, 107–128. [CrossRef]
11. Radicic, D.; Djalilov, K. The Impact of Technological and Non-Technological Innovations on Export Intensity in SMEs. *J. Small Bus. Enterp. Dev.* **2019**, *26*, 612–638. [CrossRef]
12. Hakaki, A.; Shafiei Nikabadi, M.; Heidarloo, M.A. An Optimized Model for Open Innovation Success in Manufacturing SMES. *RAIRO-Oper. Res.* **2021**, *55*, 3339–3357. [CrossRef]
13. Amrina, U.; Hidayatno, A.; Zagloel, T.Y.M. A Model-Based Strategy for Developing Sustainable Cosmetics Small and Medium Industries with System Dynamics. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 225. [CrossRef]
14. Yuana, R.; Prasetio, E.A.; Syarief, R.; Arkeman, Y.; Suroso, A.I. System Dynamic and Simulation of Business Model Innovation in Digital Companies: An Open Innovation Approach. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 219. [CrossRef]
15. Peñarroya-Farell, M.; Miralles, F. Business Model Dynamics from Interaction with Open Innovation. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 81. [CrossRef]
16. Skordoulis, M.; Ntanos, S.; Kyriakopoulos, G.L.; Arabatzis, G.; Galatsidas, S.; Chalikias, M. Environmental Innovation, Open Innovation Dynamics and Competitive Advantage of Medium and Large-Sized Firms. *J. Open Innov. Technol. Mark. Complex.* **2020**, *6*, 195. [CrossRef]
17. Lee, M.H.; Yun, J.H.J.; Pyka, A.; Won, D.K.; Kodama, F.; Schiuma, G.; Park, H.S.; Jeon, J.; Park, K.B.; Jung, K.H.; et al. How to Respond to the Fourth Industrial Revolution, or the Second Information Technology Revolution? Dynamic New Combinations between Technology, Market, and Society through Open Innovation. *J. Open Innov. Technol. Mark. Complex.* **2018**, *4*, 21. [CrossRef]
18. Yun, J.J.; Park, K.B.; Hahm, S.D.; Kim, D. Basic Income with High Open Innovation Dynamics: The Way to the Entrepreneurial State. *J. Open Innov. Technol. Mark. Complex.* **2019**, *5*, 41. [CrossRef]
19. Casquejo, M.N.; Himang, C.; Ocampo, L.; Ancheta, R.; Himang, M.; Bongo, M. The Way of Expanding Technology Acceptance-Open Innovation Dynamics. *J. Open Innov. Technol. Mark. Complex.* **2020**, *6*, 8. [CrossRef]
20. Barrett, G.; Dooley, L.; Bogue, J. Open Innovation within High-Tech SMEs: A Study of the Entrepreneurial Founder's Influence on Open Innovation Practices. *Technovation* **2021**, *103*, 102232. [CrossRef]
21. Dahlander, L.; Gann, D.M.; Wallin, M.W. How Open Is Innovation? A Retrospective and Ideas Forward. *Res. Policy* **2021**, *50*, 104218. [CrossRef]
22. Almeida, F. Open-Innovation Practices: Diversity in Portuguese Smes. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 169. [CrossRef]
23. Iqbal, M.; Suzianti, A. New Product Development Process Design for Small and Medium Enterprises: A Systematic Literature Review from the Perspective of Open Innovation. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 153. [CrossRef]
24. Baierle, I.C.; Siluk, J.C.M.; Gerhardt, V.J.; Michelin, C.D.F.; Junior, Á.L.N.; Nara, E.O.B. Worldwide Innovation and Technology Environments: Research and Future Trends Involving Open Innovation. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 229. [CrossRef]
25. Shmatko, A.; Barykin, S.; Sergeev, S.; Thirakulwanich, A. Modeling a Logistics Hub Using the Digital Footprint Method—The Implication for Open Innovation Engineering. *J. Open Innov. Technol. Mark. Complex.* **2021**, *7*, 59. [CrossRef]
26. Yun, J.H.J.; Zhao, X.; Jung, K.H.; Yigitcanlar, T. The Culture for Open Innovation Dynamics. *Sustainability* **2020**, *12*, 5076. [CrossRef]
27. Lyu, Y.; He, B.; Zhu, Y.; Li, L. Network Embeddedness and Inbound Open Innovation Practice: The Moderating Role of Technology Cluster. *Technol. Forecast. Soc. Chang.* **2019**, *144*, 12–24. [CrossRef]
28. Radziwon, A.; Bogers, M. Open Innovation in SMEs: Exploring Inter-Organizational Relationships in an Ecosystem. *Technol. Forecast. Soc. Chang.* **2019**, *146*, 573–587. [CrossRef]
29. Theeraviriya, C.; Pitakaso, R.; Sethanan, K.; Kaewman, S.; Kosacka-Olejnik, M. A New Optimization Technique for the Location and Routing Management in Agricultural Logistics. *J. Open Innov. Technol. Mark. Complex.* **2020**, *6*, 11. [CrossRef]
30. Theeraviriya, C.; Pitakaso, R.; Sillapasa, K.; Kaewman, S. Location Decision Making and Transportation Route Planning Considering Fuel Consumption. *J. Open Innov. Technol. Mark. Complex.* **2019**, *5*, 27. [CrossRef]
31. Supattananon, N.; Akararungruangkul, R. Modified Differential Evolution Algorithm for a Transportation Software Application. *J. Open Innov. Technol. Mark. Complex.* **2019**, *5*, 84. [CrossRef]
32. Montoya-Torres, J.R.; Franco, J.L.; Isaza, S.N.; Jiménez, H.F.; Herazo-Padilla, N. A Literature Review on the Vehicle Routing Problem with Multiple Depots. *Comput. Ind. Eng.* **2015**, *79*, 115–129. [CrossRef]

33. Ho, W.; Ho, G.T.S.; Ji, P.; Lau, H.C.W. A Hybrid Genetic Algorithm for the Multi-Depot Vehicle Routing Problem. *Eng. Appl. Artif. Intell.* **2008**, *21*, 548–557. [CrossRef]
34. Aras, N.; Aksen, D.; Tuğrul Tekin, M. Selective Multi-Depot Vehicle Routing Problem with Pricing. *Transp. Res. Part C Emerg. Technol.* **2011**, *19*, 866–884. [CrossRef]
35. Sombuntham, P.; Kachitvichyanukul, V. Multi-Depot Vehicle Routing Problem with Pickup and Delivery Requests. *AIP Conf. Proc.* **2010**, *1285*, 71–85. [CrossRef]
36. Kuo, Y.; Wang, C.-C. A Variable Neighborhood Search for the Multi-Depot Vehicle Routing Problem with Loading Cost. *Expert Syst. Appl.* **2012**, *39*, 6949–6954. [CrossRef]
37. Alinaghian, M.; Shokouhi, N. Multi-Depot Multi-Compartment Vehicle Routing Problem, Solved by a Hybrid Adaptive Large Neighborhood Search. *Omega* **2016**, *76*, 85–99. [CrossRef]
38. Bezerra, S.N.; de Souza, S.R.; Souza, M.J.F. A GVNS Algorithm for Solving the Multi-Depot Vehicle Routing Problem. *Electron. Notes Discret. Math.* **2018**, *66*, 167–174. [CrossRef]
39. Ehsan Hesam Sadati, M.; Çatay, B.; Aksen, D. An Efficient Variable Neighborhood Search with Tabu Shaking for a Class of Multi-Depot Vehicle Routing Problems. *Comput. Oper. Res.* **2021**, *133*, 105269. [CrossRef]
40. Sethanan, K.; Pitakaso, R. Differential Evolution Algorithms for Scheduling Raw Milk Transportation. *Comput. Electron. Agric.* **2016**, *121*, 245–259. [CrossRef]
41. Dechampai, D.; Tanwanichkul, L.; Sethanan, K.; Pitakaso, R. A Differential Evolution Algorithm for the Capacitated VRP with Flexibility of Mixing Pickup and Delivery Services and the Maximum Duration of a Route in Poultry Industry. *J. Intell. Manuf.* **2015**, *28*, 1357–1376. [CrossRef]
42. Stodola, P. Using Metaheuristics on the Multi-Depot Vehicle Routing Problem with Modified Optimization Criterion. *Algorithms* **2018**, *11*, 74. [CrossRef]
43. Stodola, P. Hybrid Ant Colony Optimization Algorithm Applied to the Multi-Depot Vehicle Routing Problem. *Nat. Comput.* **2020**, *19*, 463–475. [CrossRef]
44. Zhang, W.; Gajpal, Y.; Appadoo, S.S.; Wei, Q. Multi-Depot Green Vehicle Routing Problem to Minimize Carbon Emissions. *Sustainability* **2020**, *12*, 3500. [CrossRef]
45. Shi, Y.; Lv, L.; Hu, F.; Han, Q. A Heuristic Solution Method for Multi-Depot Vehicle Routing-Based Waste Collection Problems. *Appl. Sci.* **2020**, *10*, 2403. [CrossRef]
46. Zhen, L.; Ma, C.; Wang, K.; Xiao, L.; Zhang, W. Multi-Depot Multi-Trip Vehicle Routing Problem with Time Windows and Release Dates. *Transp. Res. Part E Logist. Transp. Rev.* **2020**, *135*, 101866. [CrossRef]
47. Kulkarni, R.V.; Bhave, P.R. Integer Programming Formulations of Vehicle Routing Problems. *Manag. Sci.* **1985**, *20*, 58–67. [CrossRef]
48. Storn, R.; Price, K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
49. Moonsri, K.; Sethanan, K.; Worasan, K.; Nitisiri, K. A Hybrid and Self-Adaptive Differential Evolution Algorithm for the Multi-Depot Vehicle Routing Problem in Egg Distribution. *Appl. Sci.* **2022**, *12*, 35. [CrossRef]
50. Sethanan, K.; Pitakaso, R. Improved Differential Evolution Algorithms for Solving Generalized Assignment Problem. *Expert Syst. Appl.* **2016**, *45*, 450–459. [CrossRef]
51. Qin, A.K.K.; Huang, V.L.L.; Suganthan, P.N.N. Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization. *IEEE Trans. Evol. Comput.* **2009**, *13*, 398–417. [CrossRef]
52. Moonsri, K.; Sethanan, K.; Sangsawang, C. Metaheuristics for Scheduling Unrelated Parallel Machines with Sequence-Dependent Setup Time and Machine Eligibility. *Chiang Mai Univ. J. Nat. Sci.* **2015**, *14*, 431–446. [CrossRef]
53. Fan, Q.; Wang, W.; Yan, X. Differential Evolution Algorithm with Strategy Adaptation and Knowledge-Based Control Parameters. *Artif. Intell. Rev.* **2019**, *51*, 219–253. [CrossRef]
54. Surekha, P.; Sumathi, S. Solution to Multi-Depot Vehicle Routing Problem Using Genetic Algorithms. *World Appl. Program.* **2011**, *1*, 118–131.
55. Pintarič, M.; Karakatič, S. Solving Multi-Depot Vehicle Routing Problem with Particle Swarm Optimization. In Proceedings of the 2019 6th Student Computer Science Research Conference, Koper, Slovenia, 10 October 2019; pp. 53–56. [CrossRef]