



Article Ant Colony Optimization and Genetic Algorithm for Fuzzy Stochastic Production-Distribution Planning

Umit Sami SAKALLI ^{1,*} and Irfan ATABAS²

- ¹ Department of Industrial Engineering, Kırıkkale University, Kırıkkale 71451, Turkey
- ² Department of Computer Engineering, Kırıkkale University, Kırıkkale 71451, Turkey; irfan@kku.edu.tr
- * Correspondence: ssakalli@kku.edu.tr; Tel.: +90-318-357-4242

Received: 20 August 2018; Accepted: 18 October 2018; Published: 24 October 2018



Abstract: In this paper, a tactical Production-Distribution Planning (PDP) has been handled in a fuzzy and stochastic environment for supply chain systems (SCS) which has four echelons (suppliers, plants, warehouses, retailers) with multi-products, multi-transport paths, and multi-time periods. The mathematical model of fuzzy stochastic PDP is a NP-hard problem for large SCS because of the binary variables which determine the transportation paths between echelons of the SCS and cannot be solved by optimization packages. In this study, therefore, two new meta-heuristic algorithms have been developed for solving fuzzy stochastic PDP: Ant Colony Optimization (ACO) and Genetic Algorithm (GA). The proposed meta-heuristic algorithms are designed for route optimization in PDP and integrated with the GAMS optimization package in order to solve the remaining mathematical model which determines the other decisions in SCS, such as procurement decisions, production decisions, etc. The solution procedure in the literature has been extended by aggregating proposed meta-heuristic algorithms. The ACO and GA algorithms have been performed for test problems which are randomly generated. The results of the test problem showed that the both ACO and GA are capable to solve the NP-hard PDP for a big size SCS. However, GA produce better solutions than the ACO.

Keywords: production-distribution; genetic algorithm; ant colony optimization; fuzzy; stochastic

1. Introduction

In nowadays, as a result of globalization and technological developments, products have become more complex structure; the numbers and varieties of the components in products have increased, the number of suppliers has enlarged, and production systems have been transformed into a more complex system to be flexible. On the other side, distribution of products on time and at required quantity to the customer, who are located in different geographic areas in the world, became an important issue for the companies to improve the customer satisfaction level and profitability. Therefore, managers need the support of efficient approaches and methodologies to make right decisions in dynamic and complex market conditions. Production-Distribution Planning (PDP), which enables optimizing the supplying, manufacturing, and distributing process simultaneously, is one of the techniques to support the managers.

PDP aims to determine the varieties and quantities of raw materials that will be supplied and go on making decisions related to the production planning and distribution of final products to the customers [1]. It can be operated by two different approaches in supply chain systems (SCS); centralized and decentralized. In centralized approaches, a single member of the chain makes decision on PDP for the whole system. In contrary to the first approach, all the members of the SCS manage their own PDP in decentralized approach [2].

PDP can be used for different decision problems. Akbari and Karimi [3], Yu and et al. [4] and Mousazadeh and et al. [5] adopted PDP to the supply network design problem. Niknamfar and et al. [6] and Su and et al. [7] developed PDP models for capacity and resource planning problems. Nishi and et al. [8] proposed a PDP model for scheduling and routing problems.

There are two challenging issues to solve the PDP: size and complexity of the SCS and type of the parameters. The number of echelons, which represents the components in SCS such as suppliers, plants, warehouses, and retailers, the number of transportation path between components, number of products and number of time periods affect the problem size and complexity. Haq and et al. [9], Yilmaz and Çatay [10], Boudia and et al. [11], and Hamedi and et al. [12] considered single product models. Lee and Kim [13] and Safaei and et al. [14] developed multi products and single plant models. Mohamed [15] proposed multiple products, plants, and single (or no) warehouse models. Tang and et al. [16] handled multi products, plants, warehouses, and single (or no) end-user models. Dhaenens-Flipo and Finke [17], and Aliev and et al. [18] focused on multi products, plants, warehouses, end-users, and single transport path models. Gunnarsson and et al. [19], and Ferrio and Wassick [20] considered multiple products, plants, warehouses, end-users, transport paths, and no time period models. Bashiri and et al. [21], and Raa and et al. [22] handled multi products, plants, warehouses, end-users, time periods and single transport path models. Sakalli [1] proposed multi suppliers, products, plants, warehouses, retailers, transport paths, and time periods model.

PDP models can include different type of parameters such as deterministic [23,24], stochastic [25,26], fuzzy [27,28], and fuzzy stochastic [29,30]. On the other side, the numbers of echelons, products, transportation paths, customers and time periods define the size and complexity of SCS [1].

The PDP, which is small size and less complex, can be solved optimally by using analytic techniques whatever the type of the parameter is such as linear programming [24,31], mixed integer programming [32], fuzzy programming [33,34], stochastic programming [35,36]. When the parameters are stochastic and the SCS is larger and more complex, simulation is an efficient technique to analyze the PDP performance. However, as indicated by Fahimnia [37], simulation is expensive and needs more effort to evaluate the results. On the other hand, it is possible to develop meta-heuristic algorithms for the solution of PDP mathematical model in complex SCS.

Meta-heuristic algorithms, inspired by natural process and systems, are powerful methods for solving single or multi-objective engineering problems by means of using biological knowledge [38]. Recently, there are many successful meta-heuristic approaches that have been used to solve complex problems such as Genetic Algorithm (GA), Simulated Annealing Algorithm (SA), Ant Colony Optimization (ACO), Tabu search (TS), Particle Swarm Optimization (PSO), and etc. [39–41]. However, they are successful in obtaining feasible solutions and cannot guarantee the optimality of solutions.

Chang and et al. [42] proposed a mathematical model for integrated PDP and a solution algorithm using ACO. Cheng and et al. [43] developed a 0–1 integer mathematical model for PDP and ACO to solve the problem. Calvete and et al. [44] proposed an ACO approach for decentralized PDP. Fahimnia [45] proposed a mixed integer non-linear model for a PDP and solved the problem by using GA. Amirtahari and et al. [2] modeled the PDP as two nonlinear bi-level programming and proposed solution algorithms combining GA and PSO. Khalifehzadeh and et al. [46] developed a multi-objective mixed integer mathematical model for PDP in a fuzzy environment and proposed two meta-heuristic algorithms based on GA and Concessive Variable Neighborhood Search (CVNS).

Armentano et al. [47], proposed a 0–1 mixed integer model for PDP that aims to minimize the total cost and proposed two TS algorithms. Chan [48] developed a GA for PDP in a multi plants SCS in order to allocate the jobs to the plants. Gen [49] considered a PDP that handles inventory management, facility location and distribution process and proposed a GA based meta-heuristic algorithm. Altiparmak et al. [50] developed a steady-state GA to solve PDP for a SCS network design problem.

In this paper, we considered a PDP for big size SCS by using centralized approach at tactical decision level in a fuzzy stochastic environment. The SCS has four echelons (suppliers, plants,

warehouses, retailers) with multi-products, multi-transport paths, and multi-time periods. The PDP handled in this study has been investigated by only Sakalli [1] in the literature according to our best knowledge. He developed a fuzzy stochastic 0–1 mixed integer mathematical model and converted it into a deterministic multi-objective mixed integer model by integrating fuzzy programming and chance-constrained programming techniques. Finally, he proposed a solution approach to solve deterministic multi-objective mix integer linear programming (MOMILP). The proposed solution approach can be successfully implemented into PDP for a small SCS. However, the mathematical model includes binary variables that select the transportation paths between the echelons of the SCS. Therefore, it cannot be solved optimally by using optimization packages when the numbers of the binary variables increase, which is a challenge.

The main contribution of this study is developing meta-heuristic algorithms to deal with this challenge. Two meta-heuristic algorithms have been proposed for big size PDP in a fuzzy and stochastic environment; ACO and GA. The proposed meta-heuristic algorithms have been designed for determining routing problem (selecting transportation paths) in PDP. The solution procedure mentioned by Sakalli [1] has been extended by integrating proposed meta-heuristic algorithms.

The paper is organized as follows: the problem definition and mathematical model are given in Section 2. Section 3 presents the proposed solution approach. The computational experiment is performed in Section 4. Finally, in Section 5, we make some concluding remarks.

2. Problem Formulation

The SCS for PDP includes four echelons. Plants manufacture products in regular and overtime by using raw materials supplied by different suppliers. The products are delivered to the customer by using warehouses and retailers. Customers only pick up their products from retailers. The SCS considered is drawn in Figure 1.



Figure 1. SCS network.

Sakalli [1] developed a 0–1 mixed integer programming model for PDP which includes uncertain parameters in both objective function and constraints. In order to solve uncertain model, it is required to transform the uncertain parameters into an equivalent deterministic one [51]. Therefore, he converted the fuzzy stochastic 0–1 mixed integer programming into a deterministic MOMILP by using fuzzy and chance-constraint approaches.

Fuzzy parameters, which are symbolized by "~", are modeled by using triangular fuzzy numbers. A triangular fuzzy number is denoted by the triplet (a^p, a^m, a^o) where a^m represents the most possible value, a^o represents the most optimistic value, and a^p represents the most pessimistic value.

Fuzzy random parameters (transportation capacities), which are symbolized by "-", are modeled by using the triangular fuzzy number as follows:

$$\varepsilon = \begin{cases} (a^{p}, a^{m}, a^{o}) \text{ with probability } Pr_{a} \\ (b^{p}, b^{m}, b^{o}) \text{ with probability } Pr_{b} \\ (c^{p}, c^{m}, c^{o}) \text{ with probability } Pr_{c} \end{cases}$$
(1)

 Pr_a , Pr_b , and Pr_c represent the probabilities of transportation capacity situations such as high, medium, and low capacities. Triangular fuzzy numbers (a^p, a^m, a^o) , (b^p, b^m, b^o) and (c^p, c^m, c^o) , define the amounts of each transportation.

There is one random fuzzy parameter $\overline{CDP_{jrtc}}$ that represents the customer demand. The customer demand is a discrete variable which can occur in three situations such as high, medium, and low.

The probabilities of each situation are modeled as random fuzzy variables which are defined in Equation (2) for only high demand:

$$\zeta \Big(Pr(D)_{high} \Big) = \begin{cases} 0.8 \text{ with possibility 1.0} \\ 0.6 \text{ with possibility 0.8} \end{cases}$$
(2)

On the other hand, demand quantities for each situation follow normal distributions with different fuzzy mean parameters and variances which is defined in Equation (3) for only high demand:

$$\zeta\left(\theta_{high}\right) \sim N\left(X\left(\theta_{high}\right), Y\left(\theta_{high}\right)\right)$$
(3)

where $X(\theta_{high})$ is a fuzzy variable. Therefore, demand quantity is a random fuzzy variable. Interested readers can be referred to Sakalli [1] for more detailed discussion on modeling uncertainties:

$$Z_{1} = \left[\sum_{l=1}^{T} \sum_{j=1}^{L} \left(\left(POP_{jl}^{m} - POP_{jl}^{m} \right) * \sum_{i=1}^{C} \sum_{j=1}^{K} SPQ_{jrt} \right) \right] - \left[\sum_{l=1}^{T} \sum_{i=1}^{S} \sum_{i=1}^{L} \left(\left(RUP_{ist}^{m} - RUP_{ist}^{m} \right) * \sum_{p=1}^{P} \sum_{i=1}^{K} TRQ_{ispl} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{L} \sum_{p=1}^{P} \sum_{i=1}^{S} \left(\left(TCS_{ipt} + RUS_{ipt} \right) \right) + TRQ_{ispl} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{p=1}^{L} \sum_{i=1}^{L} \left(\left(SCC_{jrt}^{m} - SRC_{jrt}^{p} \right) * SRP_{ipt} \right) \right] - \left[\sum_{l=1}^{T} \sum_{p=1}^{P} \sum_{i=1}^{L} \left(\left(RUC_{jrt}^{m} - RPC_{jrt}^{p} \right) * RPQ_{jrt} \right) \right] \right] \\ - \left[\sum_{l=1}^{T} \sum_{p=1}^{L} \sum_{i=1}^{L} \left(\left(SCC_{jrt}^{m} - SRC_{jrt}^{p} \right) * SRP_{ipt} \right) \right] - \left[\sum_{l=1}^{T} \sum_{p=1}^{P} \sum_{i=1}^{L} \left(\left(RUC_{jrt}^{m} - RPC_{jrt}^{p} \right) * SLP_{jrt} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{P} \sum_{p=1}^{L} \left(\left(COC_{jrt}^{m} - OPC_{jrt}^{p} \right) * OPQ_{jrt} \right) \right] - \left[\sum_{l=1}^{T} \sum_{p=1}^{P} \sum_{i=1}^{L} \left(\left(RUC_{jrt}^{m} - RPC_{jrt}^{p} \right) * SLP_{jrt} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{P} \sum_{p=1}^{L} \left(\left(RUC_{jrt}^{m} - RPC_{jrt}^{p} \right) * SLP_{jrt} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{P} \sum_{p=1}^{L} \left(\left(RUC_{jrt}^{m} - RPC_{jrt}^{p} \right) * SLP_{jrt} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{P} \sum_{p=1}^{L} \left(\left(RUC_{jrt}^{m} - RPC_{jrt}^{p} \right) * SLP_{jrt} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{K} \sum_{p=1}^{R} \sum_{j=1}^{R} \left(\left(RUC_{jrt}^{m} - RUC_{jrt}^{p} \right) * SLP_{jrt} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{K} \sum_{p=1}^{R} \sum_{j=1}^{R} \left(\left(RUC_{jrt}^{m} - RUC_{jrt}^{p} \right) * SLPQ_{jrrsk} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{R} \sum_{p=1}^{R} \sum_{j=1}^{R} \left(\left(RUC_{jrt}^{m} - RUC_{jrt}^{p} \right) * SLPQ_{jrrsk} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{R} \sum_{p=1}^{R} \sum_{j=1}^{R} \left(\left(RUC_{jrt}^{m} - RUC_{jrt}^{p} \right) * SLPQ_{jrrsk} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{R} \sum_{p=1}^{R} \sum_{j=1}^{R} \left(\left(RUC_{jrt}^{m} - RUC_{jrt}^{p} \right) * SLPQ_{jrrsk} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{R} \sum_{j=1}^{R} \sum_{j=1}^{R} \left(\left(RUP_{jrt}^{m} - RUC_{jrt}^{p} \right) * SLPQ_{jrrsk} \right) \right] \\ - \left[\sum_{l=1}^{T} \sum_{k=1}^{R} \sum_{k=1}^{R} \sum_{p=1}^{R} \sum_{j=1}^{R} \left(\left(RUP_{jrt}^{m} - RUC_{jrt}^{p} \right) * SLPQ_{jrrs} \right) \right] \\ - \left[\sum_$$

$$Z_{3} = \left[\sum_{t=1}^{T} \sum_{j=1}^{L} \left(\left(POP_{jt}^{o} - POP_{jt}^{m} \right) * \sum_{c=1}^{C} \sum_{r=1}^{R} SPQ_{jrtc} \right) \right] - \left[\sum_{t=1}^{T} \sum_{s=1}^{S} \sum_{t=1}^{L} \left(\left(RUP_{ist}^{o} - RUP_{ist}^{m} \right) * \sum_{p=1}^{P} \sum_{k=1}^{K} TRQ_{ispkt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{p=1}^{P} \sum_{s=1}^{S} \left(FTCS_{spkt} * UKS_{spkt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{p=1}^{P} \sum_{s=1}^{L} \left(\left(VTCS_{ispkt}^{o} - VTCS_{ispkt}^{m} \right) * TRQ_{ispkt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{p=1}^{F} \sum_{j=1}^{L} \left(\left(CPC_{jpt}^{o} - OPC_{jpt}^{m} \right) * OPQ_{jpt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{p=1}^{P} \sum_{j=1}^{L} \left(\left(CPC_{jpt}^{o} - OPC_{jpt}^{m} \right) * OPQ_{jpt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{p=1}^{F} \sum_{j=1}^{L} \left(\left(PHC_{jpt}^{o} - PHC_{jpt}^{m} \right) * SLP_{jpt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{m=1}^{K} \sum_{j=1}^{M} \left(\left(VTCP_{jpukt}^{o} + WTCP_{jpukt}^{m} \right) \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{m=1}^{K} \sum_{p=1}^{K} \left(\left(VTCP_{jpukt}^{o} - VTCP_{jpukt}^{m} \right) * TPQP_{jpukt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{m=1}^{K} \sum_{m=1}^{K} \left(\left(VTCP_{jpukt}^{o} - WTCP_{jpukt}^{m} \right) * TPQP_{jpukt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{m=1}^{K} \sum_{j=1}^{K} \left(\left(VTCP_{jpukt}^{o} - WTCP_{jpukt}^{m} \right) * TPQP_{jpukt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{m=1}^{K} \sum_{j=1}^{K} \left(\left(VTCP_{jpuk}^{o} - WTCP_{jpukt}^{m} \right) * TPQP_{jpukt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{m=1}^{K} \sum_{j=1}^{K} \left(\left(VTCP_{jpuk}^{o} - WTCP_{jpukt}^{m} \right) * TPQP_{jpukt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{m=1}^{K} \sum_{j=1}^{K} \left(\left(VTCP_{jpu}^{o} - WTCP_{jpukt}^{m} \right) * TPQW_{jurkt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{j=1}^{K} \left(\left((VTCP_{jpt}^{o} - WTCP_{jpuk}^{m} \right) * TPQW_{jurkt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{j=1}^{K} \left(\left((CPC_{jpt}^{o} - WTCP_{jpt}^{m} \right) * SLW_{jpt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{j=1}^{K} \left(\left((CPC_{jpt}^{o} - WTCP_{jpt}^{m} \right) * SLW_{jpt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{j=1}^{K} \left(\left((CPC_{jpt}^{o} - WTCP_{jpt}^{m} \right) * SLW_{jpt} \right) \right] - \left[\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{j=1}^{K} \left(\left((CPC_{jpt}^{o} - WTCP_{jpt}^{m} \right) * SLW_{jpt} \right) \right] \right]$$

$$\sum_{k=1}^{K} \sum_{p=1}^{P} TRQ_{ispkt} \le RC_{\infty,ist}^{p} \quad \forall i, \forall s, \forall t$$
(7)

$$\sum_{k=1}^{K} \sum_{p=1}^{P} TRQ_{ispkt} \le RC_{\alpha,ist}^{o} \quad \forall i, \forall s, \forall t$$
(8)

$$\sum_{i=1}^{l} \left(RRC_i * TRQ_{ispkt} \right) \le \left(TCSP_U \right)_{spkt} + \left(1 - UKS_{spkt} \right) * M \qquad \forall s, \forall p, \forall k, \forall t$$
(10)

$$\sum_{i=1}^{l} \left(RRC_i * TRQ_{ispkt} \right) \le UKS_{spkt} * M \quad \forall s, \forall p, \forall k, \forall t$$
(11)

$$\sum_{k=1}^{K} \sum_{s=1}^{S} TRQ_{ispkt} + SRP_{ip(t-1)} - \sum_{j=1}^{J} \left(RPQ_{ij} * \left(RPQ_{jpt} + OPQ_{jpt} \right) \right) = SRP_{ipt} \ \forall i, \forall p, \forall t$$
(12)

$$\sum_{i=1}^{I} (IRC_i * SRP_{ipt}) \le TSC_p \qquad \forall p, \forall t$$
(13)

$$\sum_{i=1}^{J} (UPT_{jpt} * RPQ_{jpt}) \le ARC_p \quad \forall p, \forall t$$
(14)

$$\sum_{i=1}^{J} (UPT_{jpt} * RPQ_{jpt}) \le ARC^{o}_{\alpha,pt} \qquad \forall p, \forall t$$
(15)

$$\sum_{i=1}^{J} (UPT_{jpt} * OPQ_{jpt}) \le AOC_{\alpha,pt}^{p} \qquad \forall p, \forall t$$
(16)

$$\sum_{i=1}^{J} (UPT_{jpt} * OPQ_{jpt}) \le AOC^{o}_{\alpha,pt} \qquad \forall p, \forall t$$
(17)

$$\sum_{i=1}^{J} (RHC_i * SLP_{ipt}) \le PIC_p \qquad \forall p, \forall t$$
(18)

5 of 25

$$RPQ_{jpt} + OPQ_{jpt} + SLP_{jp(t-1)} - \sum_{k=1}^{K} \sum_{w=1}^{W} TPQP_{jpwkt} = SLP_{jpt} \qquad \forall j, \forall p, \forall t$$
(19)

$$\sum_{i=1}^{J} \left(RTC_{j} * TPQP_{jpwkt} \right) \le \left(TCPW_{L} \right)_{pwkt} + \left(1 - UKP_{pwkt} \right) * M \qquad \forall p, \forall w, \forall k, \forall t$$
(20)

$$\sum_{j=1}^{J} \left(RTC_j * TPQP_{jpwkt} \right) \le \left(TCPW_U \right)_{pwkt} + \left(1 - UKP_{pwkt} \right) * M \qquad \forall p, \forall w, \forall k, \forall t$$
(21)

$$\sum_{j=1}^{J} \left(RTC_j * TPQP_{jpwkt} \right) \le UKP_{pwkt} * M \qquad \forall p, \forall w, \forall k, \forall t$$
(22)

$$\sum_{k=1}^{K} \sum_{p=1}^{P} TPQP_{jpwkt} + SLW_{jw(t-1)} - \sum_{k=1}^{K} \sum_{r=1}^{R} TPQW_{jwrkt} = SLW_{jwt} \qquad \forall j, \forall w, \forall t$$
(23)

$$\sum_{j=1}^{J} (RHC_j * SLW_{jwt}) \le WIC_w \qquad \forall w, \forall t$$
(24)

$$\sum_{j=1}^{J} \left(RTC_j * TPQW_{jwrkt} \right) \le (TCWR_L)_{wrkt} + (1 - UKW_{wrkt}) * M \qquad \forall w, \forall r, \forall k, \forall t$$
(25)

$$\sum_{j=1}^{J} \left(RTC_j * TPQW_{jwrkt} \right) \le \left(TCWR_U \right)_{wrkt} + \left(1 - UKW_{wrkt} \right) * M \qquad \forall w, \forall r, \forall k, \forall t$$
(26)

$$\sum_{j=1}^{J} \left(RTC_j * TPQW_{jwrkt} \right) \le UKW_{wrkt} * M \qquad \forall w, \forall r, \forall k, \forall t$$
(27)

$$\sum_{k=1}^{K} \sum_{w=1}^{W} TPQW_{jwrkt} + SLR_{jr(t-1)} - \sum_{c=1}^{C} SPQ_{jrtc} = SLR_j - \sum_{c=1}^{C} BLR_{jrtc} \qquad \forall j, \forall r, \forall t$$
(28)

$$\sum_{j=1}^{J} (RHC_j + SLR_{jrt}) \le RIC_r \qquad \forall r, \forall t$$
⁽²⁹⁾

$$SPQ_{jrtc} \ge (\mu_{lo})_{jrtc} + K_{1-\beta}\sqrt{\sigma^2} \qquad \forall j, \forall r, \forall t, \forall c$$
(30)

$$SPQ_{jrtc} \le (\mu_{up})_{jrtc} + K_{1-\beta}\sqrt{\sigma^2} \qquad \forall j, \forall r, \forall t, \forall c$$
(31)

$$TRQ_{ispkt}, SRP_{ipt}, RPQ_{jpt}, OPQ_{jpt}, SLP_{jpt}, TPQP_{jpwkt} \ge 0$$

$$SLR_{jrt}, BLR_{jrtc}, SPQ_{jrtc}, SLW_{jwt}, TPQW_{jwrkt} \ge 0$$

$$UKS_{spkt}, UKP_{pwkt}, UKW_{wrkt} = 0, 1$$
(32)

The objective functions, Z_1 , Z_2 , and Z_3 define lower profit risk, possible profit, and higher profit possibility, respectively. Therefore, the decision maker aims to maximize Z_1 and Z_3 and minimize Z_2 . Equations (7) or (8) ensures that the total transported quantity from *s* for *i* at *t* will be less than, or equal to, on-hand quantity. Equations (9)–(11) are modeled for selecting the k from *s* to *p* and not to exceed its capacity where M is a big number. Equation (12) is a stock balance constraint for *i* in *p* at *t*. Equation (13) is a capacity constraint for inventory level of *i* in *p*. Equations (14)–(18) are capacity constraints for regular production, overtime production and product inventory levels in *p*, respectively. Equation (19) is an inventory balance constraint for *j* in *p*. Equations (20)–(22) are constructed to select the *k* from *p* to the *w* and not to exceed its capacity. Equation (23) is an inventory balance constraint in *w*. Equation (24) is an inventory capacity constraint for a *w*. Equations (25)–(27) are constructed to select *k* from *w* to *r* and not to exceed its capacity. Equation (28) is a balance constraint for inventory and backorder level in *r*. Equation (29) is an inventory capacity constraint for *r*. Equations (30) or (31) ensures to meet customer demand at a given probability level. Equation (32) is the definition of the decisions variables.

In order to solve PDP, the MOMILP is converted into an equivalent single-objective 0–1 mixed integer programming model by using Zimmermann's [52] fuzzy programming method. According to the fuzzy programming method, first, the positive ideal solutions (PIS) and negative ideal solutions (NIS) of the objective functions can be specified as Equation (33) [53]:

$$Z_{1}^{PIS} = min(C^{m} - C^{p})x, \qquad Z_{1}^{NIS} = max(C^{m} - C^{p})x$$

$$x \in X \qquad x \in X$$

$$Z_{2}^{PIS} = min(C^{m})x, \qquad Z_{2}^{NIS} = max(C^{m})x$$

$$x \in X \qquad x \in X$$

$$Z_{3}^{PIS} = min(C^{o} - C^{m})x, \qquad Z_{3}^{NIS} = max(C^{o} - C^{m})x$$

$$x \in X \qquad x \in X$$
(33)

Second, the linear membership function of each objective function is defined by Equations (34) and (35):

$$f_1(Z_1) = \begin{cases} 1 & Z_1 < Z_1^{PIS} \\ (Z_1^{NIS} - Z_1) / (Z_1^{NIS} - Z_1^{PIS}) & Z_1^{PIS} \le Z_1 \le Z_1^{NIS} \\ 0 & Z_1 > Z_1^{NIS} \end{cases}$$
(34)

$$f_{2}(Z_{2}) = \begin{cases} 1 & Z_{2} > Z_{2}^{PIS} \\ (Z_{2}^{NIS} - Z_{2}) / (Z_{2}^{NIS} - Z_{2}^{PIS}) & Z_{2}^{NIS} \le Z_{2} \le Z_{2}^{PIS} \\ 0 & Z_{2} < Z_{2}^{NIS} \end{cases}$$
(35)

 $f_{3(Z_3)}$ is similar to $f_{2(Z_2)}$

Finally, the equivalent single-objective 0–1 mixed integer programming model is constructed as follows:

s.t.
$$f_i(z_i) \ge \lambda$$
, $i = 1, 2, 3 \text{ and } x \in X$ (36)

Sakalli [1] proposed a procedure for the solution of PDP which is given as follows: *Step 1*: Construct the PDP model.

Step 2: Formulate the fuzzy and fuzzy random parameters as triangular fuzzy numbers and triangular fuzzy numbers with probability values, respectively. Model the demand quantities and situations as random fuzzy parameters.

Step 3: Transform the fully fuzzy objective function into three new crisp objective functions

Step 4: Convert triangular fuzzy numbers into deterministic close intervals for a fixed α value. Use the lower bound of the close interval for constructing Equations (7), (14) and (16). Use the upper bound of the close interval for constructing Equations (8), (15) and (17).

Step 5: Convert the discrete fuzzy random parameters into deterministic close intervals. Use the lower bound of the close interval for constructing Equations (9), (20) and (25). Use the upper bound of the close interval for constructing Equations (10), (21) and (26).

Step 6: Convert the random fuzzy parameters into normally distributed random parameters with a deterministic close interval mean parameters

Step 7: Fix an acceptable probability value (β). According to the β and chance-constraint approach, convert the demand constraint which is obtained at step 6, into deterministic linear constraints by considering lower and upper bounds of close interval separately.

Step 8: Find the maximum and minimum values for Z_1 , Z_2 , and Z_3 and construct the membership functions. Formulate a single-objective 0–1 mixed integer programming model (Z_4) by using membership functions.

For: Z_1^{PIS} : Min Z_1

s.t. Equations (7), (9), (11)–(14), (16), (18)–(20), (22)–(25), (27)–(30) and (32)

For Z_1^{NIS} : Max Z_1 s.t. Equations (8), (10–(13), (15), (17)–(19), (21)–(24), (26)–(29), (31) and (32) For Z_2^{PIS} : Max Z_2 s.t. Equations (8), (10–(13), (15), (17)–(19), (21)–(24), (26)–(29), (31) and (32) For Z_2^{NIS} : Min Z_2 s.t. Equations (7), (9), (11)–(14), (16), (18)–(20), (22)–(25), (27)–(30) and (32) For Z_3^{PIS} : Max Z_3 s.t. Equations (8), (10)–(13), (15), (17)–(19), (21)–(24), (26)–(29), (31) and (32) For Z_3^{NIS} : Min Z_3 s.t. Equations (7), (9), (11)–(14), (16), (18)–(20), (22)–(25), (27)–(30) and (32) Step 9: Solve Z₄ For Z_4 : $\max \lambda$ s.t. $f_i(z_i) \ge \lambda$, i = 1, 2, 3 and Equations (8), (10)–(13), (15), (17)–(19), (21)–(24), (26)–(29), (31) and (32)

Step 10: If the DM is not satisfied with the initial solution, return to Step 4 and update the α and β values and repeat the remaining steps until a satisfactory solution is found.

The proposed modeling and solution approaches can be solved global optimally by using GAMS's Cplex solver. However, if the numbers of echelons (s, p, w, r, k and t) increase, the number of binary variables in the model will increase exponentially. Therefore, optimization packages cannot solve the PDP global optimum, which is a reason to develop a meta-heuristic algorithm.

3. The Proposed Solution Algorithms

The 0–1 mixed integer PDP model is an NP-hard optimization problem for large SCS. In this study, the solution approaches of NP-hard PDP under fuzzy stochastic environments for a large SCS has been considered. Therefore, two meta-heuristic algorithms have been developed to solve PDP; the Ant Colony Algorithm (ACA) and Genetic Algorithm (GA).

The GA and ACO have been developed for route optimization process in PDP. The route optimization process aims to find optimal transportation paths between suppliers-plants (*UKS*), plants-warehouses (*UKP*), and warehouses-retailers (*UKW*). At each iteration of the GA and ACO, the obtained routes are sent to the GAMS as parameters and GAMS solves the remaining problem with an objective function value which is returned to GA/ACO as a fitness function value. The framework of the proposed solution approach is given in Figure 2.



UKS, UKP, UKW

Figure 2. The framework of the proposed approach.

In order to integrate the proposed solution meta-heuristic algorithms, solution procedure, proposed by Sakalli [1], has been extended by reorganizing Steps 8 and 9, which is given in Section 2, as follows:

Step 8: Find the maximum and minimum values for Z_1 , Z_2 , and Z_3 by using ACO or GA and construct the membership functions. Formulate a single-objective 0–1 mixed integer programming model (Z_4) by using membership functions.

Step 9: Solve Z_4 by using the ACO or GA.

3.1. Ant Colony Optimization (ACO)

ACO is an approximate optimization approach, proposed by Colorni, Dorigo, and Maniezzo [54], that was inspired by the behavior of ants. Ants use pheromones for sharing information in order to reach food by using the shortest path. ACO has been successfully implemented to several combinatorial optimization problems, such as Travelling Salesman [55], vehicle routing [56], and scheduling problems [57].

Chang and et al. [42] proposed a mathematical model for integrated PDP and a solution algorithm using ant colony optimization. Cheng and et al. [43] developed a 0–1 integer mathematical model for PDP and ant colony optimization to solve the production part. Calvete and et al. [44] proposed an ant colony optimization approach for decentralized PDP.

At the beginning of the ACO process, whole constants, variables (iteration limit, tolerance etc.), and weight matrix are defined. A weight matrix which will be updated according to the choices of ants and pheromone amounts is defined by the size of input variables (suppliers, plants, retailers, and warehouses).

Iteration limits and ant population sizes are defined in compliance with the objective function type. The optimization process proceeds till the iteration limit is reached. Each step of the iteration begins with a transportation route determination. Routes are defined by the weight matrix in each step of the iteration. At the beginning of the process, first route is defined randomly because of equality of the weight matrix. The existing route is sent to GAMS solver module and the success rate or with another saying objective function fitness value (z_j) is calculated and saved. Calculated objective value is compared to the best fitness value. If a new fitness value is equal or better than the tolerance level, the pheromone amount (ph) is calculated and each active relation of the weight matrix is updated and rewarded using the calculated ph value. If the fitness value is worse than the tolerance level, the weight matrix is updated by using the calculated ph value and the selection probability of the existing route is decreased. If GAMS finds an infeasible solution, the optimization process carries on with previous objective function value (z_{j-1}). ACO process pseudo code for maximization problem is given in Table 1.

Table 1. Pseudo-code for ACO.

3.2. Genetic Algorithm (GA)

GAs are nature-inspired evolutionary search algorithms based on natural selection and genetics [58]. GAs give promising performance results for optimizing complex, large-scaled, single or multi-objective engineering problems and provides an optimal or a near-optimal solution [59]. The multi-objective optimization process is similar to a single-objective operation. However, in the single-objective process, there is only one global optimal solution; in the multi-objective process, there are more than one superior to the rest of the solutions in the search area [60]. Fundamentals of GA were developed by Holland [61]. Goldberg et al. presented the detailed research and implementation of GAs in various fields [62,63].

In a GA, each solution is called individual which is one of the global candidate solutions. The population consists of individual solutions and each individual carries ancestors' genetic information in their chromosomes.

The initial population is generally created with the randomly distributed dataset. The GA uses crossover and mutation operators to find the optimal solution. The crossover operator is used for information transference to new individuals and populations. Genetic diversity is held with the mutation operation.

3.2.1. Selection Operator

Selection operation is based on the idea of the "survival of the fittest". In general, the selection is the process of transferring the characteristics of the most powerful or individual having the closest fitness value to the next generation. There are many different selection methods in the literature such as roulette wheel, tournament, or truncation selection.

3.2.2. Crossover Operator

Global search called exploration is held with a crossover operator. Every individual consists of gene and chromosome sets and these genes and chromosomes carry over meaningful information coming from their ancestors. This operator chooses a crossing position or layout to diversify genetic pool. New individuals are produced by using this crossing combination. Breeding by using multi-point crossing is given in Figure 3.



Figure 3. Multi-point crossover operation.

3.2.3. Mutation Operator

The mutation operator is responsible for abstaining local optimal solutions and helps to prevent the recessive population. This operator randomly changes one or more genes values 1 to 0 or 0 to 1. The mutation constant is defined at the beginning of the optimization process.

The genetic algorithm-based optimization process begins with the transportation path (tp) relation base matrix creation. Each tp matrix involves a combination of three different relation matrices, which are UKS, UKP, and UKW. The initial population is involving four different randomly uniform distributed individuals. After the base population creation phase, all of the transportation paths (tp₁ to tp₄) are evaluated by using GAMS optimization package, respectively, and objective values are obtained. The roulette wheel selection method is used for elitism and provides transferring more powerful individuals' features to the next generations. Selected paths are used for generating new individuals by using single point cross-over operation. The crossing point is defined as the middle position of the path. After the crossing operation, each bit of the new individuals is controlled with the mutation constant. The mutation constant is defined as 1% before the optimization process at the beginning of the script. If randomly produced variable for each bit is greater than mutation value, it is changed to 1 to 0 or 0 to 1 randomly. After the mutation process, a new population is created and the optimization process continues while the iteration limit or deserved objective value is not reached. The optimization script is evaluated several times with 500–1000 iteration. The GA process pseudo-code for maximization problem is given in Table 2.

Table 2. Pseudo-code for GA.

```
Begin;
Define iteration limit, mutation constant, cross-over position,
Generate initial Transportation Relation Population tppop
              Generate UKS
                         For i = 1 to (s * p)
                                   For j = 1 to (t * k)
                                                 tp_{(i,j)} = rand[0\ 1]
                                   End
                         End
              Generate UKP
                         For i = (s * p + 1) to (s * p + p * w)
                                   For j = 1 to (t * k)
                                                 tp_{(i,j)} = rand[0\ 1]
                                   End
                         End
              Generate UKP
                         For i = (s * p + p * w + 1) to ((s * p + p * w + w * r)
                                   For j = 1 to (t * k)
                                                 tp_{(i,j)} = rand[0\ 1]
                                   End
                         End
Evaluate each individual solution using GAMS and save fitness values
For i = 1 to Iteration Limit
  Select parental individuals by Roulette Wheel Selection operator
  Generate new candidate offspring solutions by Crossover operator
  Improve candidate solutions using Mutation Operator
  Evaluate each offspring solution using GAMS and save new fitness values
End
Return Best Solution
End
```

4. Computational Experiments

The computational experiments described in this section have been conducted in order to analyze the performance of the proposed solution procedures. According to our best knowledge, this is the first study that handles PDP for a large SCS which includes multiple-supplier, multiple-product, multiple-plant, multiple-warehouse, multiple-retailers, multiple-transport paths, and multiple time periods. Thus, any data for comparison does not exist. Therefore, five test problems have been considered to evaluate the proposed algorithms which are randomly generated.

On the other side, proposed ACO and GA have been compared with a Simulated Annealing (SA), which is a well-known and widely used algorithm in order to analyze and observe the superiorities (or weakness) of them. SA is a general form of optimization which refers to tempering materials such as alloys of metal, glass etc. by heating and cooling to reach perfect atomic structure [64]. SA is based on finding global optima using hill climbing method. Firstly, movement (position) is selected randomly from a pool of all possible movements and applied to the objective function. Then, if the current movement improves the solution it is accepted and the search area is moved to this position. Otherwise, the next movement is chosen from other ones, which are probably less than current [65]. The pseudo-code of SA is given in Table A5 in Appendix A.

In each test problem, there are six parameters: the number of suppliers (s), the number of plants (p), the number of warehouses (w), the number of retailers (r), the number of time periods (t), and the number of transportation paths (k) between echelons of the SCS. The characteristics of test problems are given in Table 3. There are five raw materials and one product in each test problem.

Problem	s	p	w	r	t	k
1	5	1	5	5	3	3
2	5	1	5	10	3	3
3	5	1	10	10	3	3
4	5	2	10	10	3	3
5	10	2	10	10	3	3

Table 3. Characteristics of test problem.

The proposed algorithms were implemented in MATLAB (The MathWorks Inc., New York, NY, USA), using the COIN-OR (Computational Infrastructure for Operations Research) solver of GAMS (GAMS Software GmbH, Frechen, Germany) optimization package. The computational experiments and validation were performed on a PC with Intel ®Core i7-7700K CPU at 4.20 GHz having 64 GB RAM.

In each test problem, α and β values were set as 0.4 and 0.95 respectively. Test problems were solved by COIN-OR and the optimal solutions for NIS and PIS of Z_1 , Z_2 , Z_3 , and Z_4 are given in rows which are entitled "Optimal" in Table 4. On the other hand, the proposed ACO and GA were repeated five times with 250 iterations for each test problem. The best solutions for NIS and PIS of Z_1 , Z_2 , Z_3 , and Z_4 obtained by the proposed approaches are reported in Table 4. The performances of the proposed algorithms are measured in terms of the solution gap (in %). The proposed algorithms achieved to obtain optimal solutions for Problem 1 and 2 which are small in size. When the size of the problem increases, such as in Problems 3, 4, and 5, the proposed approaches are capable of obtaining near-optimal solutions in an acceptable gap. It can be stated that the results of the ACO and GA approaches are close to each other while the GA finds relatively better solutions.

According to the gaps (%) given in Table 4, ACO and GA obtained lower values than SA in test Problems 3, 4, and 5. It is observed that the proposed ACO and GA approaches are more promising methods for fuzzy stochastic PDP than SA.

After the validation and verification of proposed approaches, a new test problem (Problem 6), which has a larger size than the other problems, was generated. Only ACO and GA have been performed to Problem 6 because of their robustness. Problem 6 includes 40 suppliers (s = 40) 12 raw materials (i = 12), three plants (p = 3), three products (j = 3), 12 warehouses (w = 12), 40 retailers

(r = 40), and four transportation paths between each of echelons (supplier-plant, plant-warehouse, warehouse-retailer). Therefore, Problem 6 is more suitable for real-life problems. The parameters in the mathematical model which were randomly generated by using domains are given in Tables A1–A4 in Appendix A. Problem 6 could not be solved by using COIN-OR within the maximum limits of iteration and time.

				Problem 1	Problem 2	Problem 3	Problem 4	Problem 5
			Optimal	1,108,835	1,404,026	1,659,658	1,776,098	1,972,490
		ACO	Best	1,108,835	1,404,026	1,721,895	1,864,015	2,094,193
			Gap (%)	0	0	0.0375	0.0495	0.0617
	PIS	GA	Best	1,108,835	1,404,026	1,712,435	1,851,760	2,087,683
			Gap (%)	0	0	0.0318	0.0426	0.0584
		SA	Best	1,108,835	1,404,026	1,736,500	1,879,467	2,126,739
71			Gap (%)	0	0	0.0463	0.0582	0.0782
Ζ1			Optimal	2,709,941	3,017,161	3,286,451	3,480,057	3,837,795
		ACO	Best	2,709,941	3,017,161	3,134,946	3,293,178	3,556,101
			Gap (%)	0	0	0.0461	0.0537	0.0734
	NIS	GA	Best	2,709,941	3,017,161	3,175,369	3,315,798	3,625,949
			Gap (%)	0	0	0.0338	0.0472	0.0552
		SA	Best	1108835	1,404,026	3,481,338	3,709,045	4,177,056
			Gap (%)	0	0	0.0593	0.0658	0.0884
			Optimal	15,658,380	16,184,285	16,301,772	16,831,979	1,7763,954
		ACO	Best	15,658,380	16,184,285	15,468,751	15,845,625	16,632,390
			Gap (%)	0	0	0.0511	0.0586	0.0637
	PIS	GA	Best	15,658,380	16,184,285	15,511,136	15,862,457	16,678,576
			Gap (%)	0	0	0.0485	0.0576	0.0611
		SA	Best	1108835	1,404,026	17,260,316	17,885,661	18,979,008
70			Gap (%)	0	0	0.0588	0.0626	0.0684
Z2			Optimal	11.063.345	11.864.926	12.303.159	12.627.619	12,966,067
		ACO	Best	11.063.345	11.864.926	12,754,685	13,199,650	13.642.896
			Gap (%)	0	0	0.0367	0.0453	0.0522
	NIS	GA	Best	11063345	11.864.926	12.744.842	13,168,081	13.609.184
	1110		Gap (%)	0	0	0.0359	0.0428	0.0496
		SA	Best	1108835	1,404,026	12,912,165	13,312,036	13,767,370
			Gap (%)	0	0	0.0495	0.0542	0.0618
			Optimal	2,365,633	2.760.450	2,980,651	3,277,653	3.413.026
		ACO	Best	2,365,633	11.864.926	2,893,020	3.125.898	3,240,668
			Gap (%)	0	0	0.0294	0.0463	0.0505
	PIS	GA	Best	2,365,633	11864926	2,888,251	3,148,513	3,262,853
			Gap (%)	0	0	0.031	0.0394	0.044
		SA	Best	1108835	1,404,026	3,098,983	3,462,185	3,612,347
70			Gap (%)	0	0	0.0397	0.0563	0.0584
Ζ3			Optimal	1,083.653	1,377.620	1,648.027	1,740.722	1,915.385
		ACO	Best	1.083.653	1.377.620	1.722.188	1.834.025	2.034.713
			Gap (%)	0	0	0.045	0.0536	0.0623
	NIS	GA	Best	1.083.653	1.377.620	1.713.618	1.815.747	2.019.199
			Gap (%)	0	0	0.0398	0.0431	0.0542
		SA	Best	1,108,835	1,404,026	1,733,395	1,844,469	2,046,397
			Gap (%)	0	0	0.0518	0.0596	0.0684
			Optimal	0.685	0.693	0.618	0.635	0.709
		ACO	Best	0.685	0.693	0.581909	0.594932	0.65859
			Gap (%)	0	0	0.0584	0.0631	0.0711
	Z4	GA	Best	0.685	0.693	0.582156	0.596138	0.660504
			Gap(%)	0	0	0.058	0.0612	0.0684
		SA	Best	1,108.835	1,404.026	0.658912	0.682181	0.763735
			Gap (%)	0	0	0.0662	0.0743	0.0772

Table 4.	Test problem	results.
----------	--------------	----------

Z1^{PIS}, Z2^{NIS}, and Z3^{NIS} are minimization problems. The minimization process generally lasts longer than maximization process because of GAMS' solving procedures for the proposed methods and problems. Therefore, the iteration process is limited to 500 and 1000 for minimization and maximization

problems, respectively. Comparative results of the proposed meta-heuristic algorithms are represented in Figure 4a–c Figures 5a–c and 6.



Figure 4. (a) Minimization process of the *Z*1; (b) minimization process of the *Z*2; and (c) minimization process of the *Z*3.



Figure 5. (a) Maximization process of the *Z*1; (b) maximization process of the *Z*2; and (c) maximization process of the *Z*3.



Figure 6. Maximization process of Z4.

According to the negative and positive ideal solutions for Z_1 , Z_2 , and Z_3 , it can be observed that the GA shows better performance than ACO in four of six problems. However, these solutions are used as lower and upper limits for the membership function in Z_4 . The optimal solution of Z_4 cannot be obtained at the lower or upper bounds of membership functions because of conflicting objectives. Therefore, the comparative solutions of ACO and GA for Z_1 , Z_2 , and Z_3 are not significant indicators in determining the best meta-heuristic approach. The main problem that determines the best performance for proposed meta-heuristic algorithms is Z_4 .

According to the ACO, the positive ideal solution and negative ideal solution values of the objective functions are found as (15,742,124, 4,421,268), (45,856,319, 78,583,274), and (4,964,094, 15,906,761) for Z_1 , Z_2 , and Z_3 , respectively. When the equivalent single-objective 0–1 mixed integer programming model of the auxiliary MOMILP problem is solved, the total profit is obtained as a triangular possibility distribution with (54,920,893, 66,474,301, 78,332,275) and the overall degree of decision maker (DM) satisfaction with multiple goal values is achieved at 0.63. On the other hand, GA obtained the positive ideal solution and negative ideal solution values of the objective functions as (15,757,821, 4,694,713), (45,781,117, 79,004,432), and (4,852,275, 15,781,522) for Z_1 , Z_2 , and Z_3 , respectively. When the equivalent single-objective 0–1 mixed integer programming model is solved, the total profit is obtained as a triangular possibility distribution with (56,243,986, 68,505,864, 80,833,744) and the overall degree of DM satisfaction with multiple goal values is achieved at 0.684.,

Total profits in triangular fuzzy numbers can be defuzzified by using the expected value (*EV*) of fuzzy numbers which is given in Equation (37):

$$EV(\tilde{c}) = \frac{c^p + 2c^m + c^o}{4} \tag{37}$$

The expected profit values of the ACO and GA have been calculated as 66,550,442 and 68,522,364 Turkish Lira, respectively, by using Equation (37). According to the total profit and DM satisfaction level, it can be concluded that GA found better solutions than ACO.

The solutions of the proposed algorithms are summarized in Tables 5–7 because of large sizes of indices. Although ACO produces more product 1 and 2 (j1 and j2), its production quantity for product 3 is significantly smaller than the GA. Therefore, GA produced and sold more products than the ACO in total. The available capacities of the plants are not adequate to meet all demand. This situation caused to backorders. Therefore, there are no inventories in warehouses and retailers. Total backorder level of GA is smaller than ACO which enables reducing the cost in GA.

According to the total purchased raw material quantities by plants, it can be absorbed that the numbers of raw materials supplied GA and ACO are close to each other. However, while GA supplied raw materials from 13 different suppliers, ACO used 23 suppliers. Cooperating with more suppliers means extra fixed transportation.

j1.t1

j1.t2

j1.t3

j2.t1

j2.t2

j2.t3

j3.t1

j3.t2

j3.t3

5822.66

5822.66

5822.66

6664.43

6664.43

6664.43

6008.03

6024.95

6012.11

6064.51

6064.51

6064.51

834.75

834.75

834.75

0.00

0.00

0.00

649.37

632.46

645.29

599.92

599.92

599.92

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

0.00

	Table 5. Summary of results.												
Total Quan Product a	Sold tity of at Periods	Back Quan Product a	order tity of 1t Periods	Inventor Product in Re	y Level of at Periods tailers	Inventor Product in War	y Level of at Periods ehouses	Total Production Quantity in Each Plant at Periods				ods	
GA	400	GA	400	GA	ACO	GA	400	GA	ACO	GA	ACO	GA	ACO
0A	neo	U A	Aco	бA	neo	ОЛ	Aco	p1	p1	p2	p2	p3	p3
6699.43	6619.75	574.61	654.30	0.00	0.00	0.00	0.00	1834.28	0.00	2004.95	3345.89	2860.20	3273.86
6364.37	6619.75	909.67	654.30	0.00	0.00	0.00	0.00	2748.15	895.47	1840.83	3345.89	1775.39	2378.39
6553.76	6619.75	720.28	654.30	0.00	0.00	0.00	0.00	2568.38	2400.90	1957.15	978.87	2028.22	3239.98

0.00

0.00

0.00

0.00

0.00

0.00

2411.49

1680.00

1680.00

3108.48

3182.22

3397.94

2793.11

2808.02

1257.98

4699.02

3600.60

4274.15

2106.34

2921.17

2780.30

1694.07

0.00

313.08

1794.34

1509.36

3024.45

0.00

483.02

878.78

1304.83

1221.49

1362.36

1861.88

3482.21

2953.41

1420.59

1707.57

1729.68

1365.49

1980.89

911.59

	Table 6. Summary of results.											
	Total Purchased Raw Material Quantities by Plants at Periods											
	GA	ACO	GA	ACO	GA	ACO	GA	ACO	GA	ACO	GA	ACO
	i1	i1	i2	i2	i3	i3	i4	i4	i5	i5	i6	i6
p1,t1	6080	2793	7234	8379	1834	0	13,571	16,890	3669	0	17,817	19,683
p1,t2	7176	4599	5040	8424	2748	895	13,975	14,505	5496	1791	18,403	18,209
p1,t3	6817	6060	5040	3774	2568	2401	14,442	16,481	5137	4802	18,691	20,140
p2,t1	6116	8486	6319	5383	2005	3346	9193	5140	4010	6692	13,305	10,280
p2,t2	6603	8201	8764	4528	1841	3346	4762	6304	3682	6692	9524	11,160
p2,t3	6695	4982	8341	9073	1957	979	5677	6640	3914	1958	10,414	10,643
p3,t1	7025	7968	3914	4262	2860	3274	9751	8791	5720	6548	13,916	13,485
p3,t2	4772	6464	3664	5123	1775	2378	13,444	10,029	3551	4757	16,440	14,115
p3,t3	5419	8210	4087	5189	2028	3240	12,251	7704	4056	6480	15,641	12,674
	i7	i7	i8	i8	i9	i9	i10	i10	i11	i11	i12	i12
p1,t1	3669	1791	11,737	16,890	8492	5586	10,903	8379	7234	8379	8628	12,191
p1,t2	5496	4802	11,227	13,610	8856	7407	10,536	10,215	5040	8424	8044	10,009
p1,t3	5137	6692	11,874	14,080	8497	7318	10,177	8576	5040	3774	8476	9806
p2,t1	4010	6692	7189	1794	8223	10,280	10,329	12,075	6319	5383	5494	1794
p2,t2	3682	1958	2921	2958	9524	9710	12,445	11,220	8764	4528	2921	2475
p2,t3	3914	6548	3720	5661	9475	8007	12,255	11,031	8341	9073	3406	4782
p3,t1	5720	4757	6890	5517	8330	9389	9635	10,809	3914	4262	5029	4152
p3,t2	3551	6480	11,668	7650	5994	8172	7215	9879	3664	5123	8186	5669
p3,t3	4056	0	10,223	4464	6781	9939	8144	11,669	4087	5189	7269	3553

Table 6 Summary of regults

 Table 7. Summary of results.

	Cooperated Suppliers
GA	s2, s5, s7, s8, s13, s17, s18, s27, s28, s29, s31, s34, s37
ACO	s2, s5, s6, s7, s8, s9, s10, s11, s12, s13, s14, s17, s18, s19, s20, s21, s23, s25, s27, s28, s29, s31

5. Conclusions

In this paper, a fuzzy stochastic PDP has been considered for an SCS which includes four echelons. The PDP handled in this study was considered by Sakalli [1] and modeled as a 0–1 mixed integer model. He developed a solution procedure and successfully implemented it for small SCS. However, it is not possible to solve the mathematical model global optimally by using optimization packages for large SCS because of the binary variables which are designed for selecting routes between echelons in SCS. Therefore, it is required to develop meta-heuristic algorithms to solve NP-hard PDP.

In this study, we have developed two meta-heuristic algorithms, GA and ACO, in order to solve Sakalli's model for large size SCS. In the solution process of PDP, meta-heuristic algorithms (GA/ACO) are performed for route optimization. By determining routes, 0–1 mixed integer model is converted into a deterministic linear programming model and can be solved optimally by the COIN-OR solver of GAMS.

The proposed solution approaches have been performed for randomly generated test problems. The results of the test problems have showed that both proposed meta-heuristic algorithms are capable of solving the problem where the GA obtain better solutions than the ACO.

In real-life applications, some parameters of the PDP can be affected by the conditions of dynamic market structure and it is not possible to define them precisely in the decision making process at the tactical or strategic level. This challenge can be overcome by including the manager's expertise and judgments into the modeling process by using fuzzy, random fuzzy, and fuzzy random parameters. Therefore, decision or policy makers can manage the large production and distribution systems more effectively by using the proposed approach. The proposed approach can be used in several industries, such as automotive, electronic devices, and textiles, by implementing the proposed extended procedure in ten steps. The proposed approach is implemented iteratively for real-life problems. In each solution, the proposed approach produces several performance values that show capacity usage proportions in the production-distribution system. In this way, decision makers can detect the bottlenecks in the system and take preventive measures for balancing the capacities of supply, transport, manufacture, and storage. On the other hand, the proposed approach can be used to manage customer demand. The proposed approach enables the analysis of the results of changes in demand quantity on profitability by using random fuzzy parameters. Therefore, managers can specify their marketing and pricing strategies. Consequently, decision makers can evaluate the effects of different predictions and judgments by performing the proposed approach several times.

There are two main limitations of the proposed approach. The first one is determining the minimum acceptable possibility degree (α) which represents the decision maker satisfaction level and takes values between 0 and 1. A very low possibility degree may cause obtaining an optimistic and unrealistic solution. On the contrary, a very high possibility degree may cause obtaining a conservative solution which can be feasible, but not robust. Therefore, the proposed approach has to be performed with different α values in real-life applications, which means longer processing time. The second one is starting with the GA/ACO with a better feasible solution. According to the nature of proposed approaches, the starting solutions have just been selected randomly. This situation directly affects the processing time which is required to approximate the optimal solution. If GA/ACO start from a better feasible solution, they reach the best solution which is close to the optimal value in a shorter processing time.

With respect to future research, it is possible to integrate other heuristic algorithms, such as a Flower Pollination Algorithm (FPA), Krill Herd Algorithm (KH), and Particle Swarm Optimization etc., into the GA/ACO in order to obtain better starting solutions. On the other hand, PDP can be modeled as bi-objective functions by considering the maximization of the customer satisfaction level or minimization of the total transportation time.

Author Contributions: Conceptualization, U.S.S.; Methodology, U.S.S.; Software, I.A. and U.S.S.; Validation, U.S.S. and I.A.; Formal Analysis, I.A.; Investigation, U.S.S.; Resources, U.S.S.; Data Curation, I.A.; Writing-Original Draft Preparation, U.S.S. and I.A..; Writing-Review & Editing, U.S.S.; Visualization, I.A.; Supervision, U.S.S.

Funding: This research received no external funding.

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

Abbreviations

The MOMILP is given at below.

Indices	
i	Raw materials
j	Products
S	Suppliers
р	Plant
w	Warehouses
r	Retailers
k	Transportation path
С	Customers
t	Time period
Parameters	
\widetilde{RUP}_{ist}	raw material i price in s at t
\widetilde{RC}_{ist}	on hand quantity of i in s at t
TCSP _{spkt}	transportation capacity of k between s and p at t
FTCS _{spkt}	fixed cost of using k between s and p at t
<i>VTCS</i> _{ispkt}	variable cost of transporting i from s to p by using k at t
RRC _i	required transporting capacity of i
RRM _{ij}	required i quantity for producing j
IRC _i	required storing capacity for i
TSC_p	raw material storage capacity of p
SRC _{ipt}	holding cost of i in p at t
UPT_{ipt}	manufacturing time of j in p at t
RPC _{ivt}	manufacturing cost of j in p at t (regular time)
\widetilde{ARC}_{pt}	regular capacity (time) in p at t
\widetilde{OPC}'_{int}	manufacturing cost of <i>j</i> in <i>p</i> at <i>t</i> (overtime)
\widetilde{AOC}_{pt}	overtime capacity (time) in p at t
PIC_{p}	products storage capacity in p
PHC _{ipt}	holding cost of j in p at t
RHC_i	required storing capacity for j
TCPW _{pwkt}	transportation capacity of k from p to w at t
FTCP _{pwkt}	fixed cost of using k from p to w at t
<i>VTCP</i> _{jpwkt}	variable cost of transporting j from p to w by using k at t
RTC_j	required transporting capacity of j
TCWR _{wrkt}	transportation capacity of k from w to r at t
FTCW _{wrkt}	fixed cost of using k from w to r at t
VTCW _{jwrkt}	variable cost of transporting j from w to r by using k at t
WHC _{jwt}	holding cost of j in w at t
WICw	products storage capacity in w
<i>HCR</i> _{irt}	holding cost of j in r at t
RICr	storage capacity in r
\widetilde{BCR}_{irt}	backorder cost of j in r at t
$\overline{\overline{CDP}}_{irt}$	demand of c for j from r at t
\widetilde{POP}_{it}	price of j at period t
Decision Varial	bles
TRQ _{ispkt}	transported quantity of i from s to p by using k at t
SRP _{ipt}	stored quantity of <i>i</i> in <i>p</i> at the end of t
RPQ_{jpt}	quantity of j produced at regular time in p at t

OPQ_{jpt}	quantity of j produced at over time in part
TPQP _{jpwkt}	transported quantity of j from p to w by using k at t
SLP _{jpt}	stored quantity of j in p at the end of t
SLW _{jwt}	stored quantity of j in w at the end of t
TPQW _{jwrkt}	transported quantity of j from w to r by using k at t
SLR _{jrt}	stored quantity of j in r at the end of t
BLR _{jrtc}	backorder quantity of j in r for c at the end of t
SPQ _{jrtc}	sold quantity of j from r to c at t
$\int 1$	if k is used between s and p at t
UKS _{spkt}	0 otherwise
, IKP ∫ 1	if k is used between p and w at t
UKP _{pwkt}	0 otherwise
1	<i>if k is used between w and r at t</i>
UKW _{wrkt}	0 otherwise
(

Appendix A

 Table A1. Deterministic parameters.

Parameter	Domain
FTCS _{spkt}	U~[1000; 3500]
RRC_i	U~[0.85; 1.53]
RRM_{ij}	[0, 1, 2, 3]
IRC_i	U~[0.55; 1.20]
TSC_p	U~[900; 2500]
UPT_{ipt}	U~[5; 8.70]
PIC_{p}	[1000, 2200, 2800]
RHĊi	[2, 2.3, 2.9]
FTCP _{pwkt}	U~[670; 1700]
RTC_i	[2.5, 2, 3]
FTCW _{wrkt}	U~[900; 2300]
WIC_w	U~[16000; 24000]
RICr	U~[3500; 4100]

Table A2. Fuzzy parameters.

		Domain	
Parameter	Pessimistic	Possibilistic	Optimistic
RUP _{ist}	U~[27; 30]	U~[30; 50]	U~[50; 55]
RC_{ist}	U~[144,000; 168,000]	U~[180,000; 210,000]	U~[216,000; 273,000]
VTCS _{ispkt}	U~[3; 6]	U~[7; 10]	U~[11; 14]
SRC _{ipt}	U~[0.72; 0.98]	U~[1; 1.25]	U~[1.3; 1.54]
RPC_{jpt}	U~[10.8; 11.9]	U~[12; 17]	U~[17.1; 19.8]
ARC_{pt}	U~[62,560; 66,240]	U~[68,000; 72,000]	U~[7,3440; 77,760]
OPC_{jpt}	U~[0.8; 0.98]	U~[1;2]	U~[2; 2.4]
AOC_{pt}	U~[20,240; 23,000]	U~[23,000; 30,000]	U~[30,000; 33,600]
PHC _{jpt}	U~[4.5; 5]	U~[5;7]	U~[7; 8.8]
VTCP _{ipwkt}	U~[1.7; 2]	U~[2; 8]	U~[8; 9.2]
VTCW _{jwrkt}	U~[0.85; 1]	U~[1;3]	U~[3; 3.45]
WHCjwt	U~[5.95; 7]	U~[7; 10]	U~[10; 11.5]
HCR _{jrt}	U~[6.3; 7]	U~[7; 10]	U~[10; 11]
BCR _{jrt}	U~[1.8; 2]	U~[2;5]	U~[5; 5.5]
POP _{jt}	U~[800; 960]	U~[1000; 1200]	U~[1200; 1440]

Domain							
Parameter	Pessimistic	Possibilistic	Optimistic	Description			
TCSP _{spkt}	U~[182,400; 211,200];	U~[228,000; 264,000]	U~[273,600; 316,800]	High (Pr _a)			
	U~[152,000; 168,000]	U~[190,000; 220,000]	U~[228,000; 264,000]	Medium (Pr _b)			
	U~[106,400; 123,200]	U~[133,000; 15,4000]	U~[159,600; 184,800]	Low (Pr_c)			
TCPW _{pwkt}	U~[33,120; 36,800] U~[28,800; 32,000] U~[23,040; 25,600]	U~[36,800; 43,700] U~[32,000; 38,000] U~[25,600; 30,400]	U~[43,700; 48,070] U~[38,000; 41,800] U~[30,400; 33,440]	High (Pr _a) Medium (Pr _b) Low (Pr _c)			
TCWR _{wrkt}	U~[38,400; 46,080] U~[32,000; 40,000] U~[25,600; 30,720]	U~[48,000; 57,600] U~[40,000; 48,000] U~[32,000; 38,400]	U~[57,600; 69,120] U~[48,000; 57,600] U~[38,400; 46,080]	High (Pr _a) Medium (Pr _b) Low (Pr _c)			

Table	A3.	Fuzzv	random	parameters.
				p our our course

Where $Pr_a = 0.5$, $Pr_b = 0.35$ and $Pr_c = 0.15$.

Table A4. Random fuzzy parameters.

Domain							
Parameter	Pessimistic	Possibilistic	Optimistic	Description			
Mean Parameter of CDP _{jrt}	U~[10,400; 12,480] U~[8000; 9600] U~[6000; 7200]	U~[13,000; 15,600] U~[10,000; 12,000] U~[7500; 9000]	U~[15,600; 18,720] U~[12,000; 14,400] U~[9000; 10,800]	High (Pr_{high}) Medium (Pr_{medium}) Low $((Pr_{low})$			
Parameter		Domain		Description			
Standard Deviation of CDP _{jrt}		U~[1800; 2500] U~[2528; 3300] U~[1000; 1800]		High (Pr _{high}) Medium (Pr _{medium}) Low ((Pr _{low})			

Table A5. Pseudo-code for SA.

Start

Initialize transportation routes (UKPZ₀, UKSZ₀, UKWZ₀), Iteration limit *N*, cooling rate β , *j* = 0, neighborhood parameter *a*, maximum neighborhood limit *K*, starting temperature *T*₀ and maximum temperature *T*_{end} **Evaluate objective function with current transportation routes.** Compute F(UKPZ₀, UKSZ₀, UKWZ₀) and set *F*_{best} = *F*(UKPZ₀, UKSZ₀, UKWZ₀) **while**(*T* > *T*_{end})

```
while(j < N)
           set neighbor routes
           i = 1
           while (i \leq K)
             a = random [1, -1]
(\text{UKPZ}'_i = \text{UKPZ}_i \pm a)
(\mathrm{UKSZ}_{i}^{\dagger} = \mathrm{UKSZ}_{i} \pm a)
(UKWZ'_i = UKWZ_i \pm a)
           Compute F(UKPZ'_i, UKSZ'_i, UKWZ'_i)

IF F(UKPZ'_i, UKSZ'_i, UKWZ'_i) \ge F(UKPZ_i, UKSZ_i, UKWZ_i)
accept the new solution vector and j = j + 1
     end
     ELSE accept or reject with acceptance probability
        p = e^{\left(\frac{-\Delta}{T}\right)}
        Generate a uniform random number p' with a range [0,1]
        IF p' < p accept the new solution and j = j + 1
        end
ELSE reject the new solution
end
                         end
                      i = i + 1
                   end
                end
             T_{next} = \beta x T
end
Stop Procedure
```

References

- 1. Sakalli, U.S. Optimization of Production-Distribution Problem in Supply Chain Management under Stochastic and Fuzzy Uncertainties. *Math. Probl. Eng.* **2017**, 2017, 4389064. [CrossRef]
- 2. Amirtaheri, O.; Zandieh, M.; Dorri, B.; Motameni, A.R. A bi-level programming approach for production-distribution supply chain problem. *Comput. Ind. Eng.* **2017**, *110*, 527–537. [CrossRef]
- Akbari, A.A.; Karimi, B. A new robust optimization approach for integrated multi-echelon, multi-product, multi-period supply chain network design under process uncertainty. *Int. J. Adv. Manuf. Technol.* 2015, 79, 229–244. [CrossRef]
- 4. Yu, V.F.; Normasari, N.M.E.; Luong, H.T. Integrated location-production-distribution planning in a multiproducts supply chain network design model. *Math. Probl. Eng.* **2015**, 2015, 473172. [CrossRef]
- 5. Mousazadeh, M.; Torabi, S.A.; Zahiri, B. A robust possibilistic programming approach for pharmaceutical supply chain network design. *Comput. Chem. Eng.* **2015**, *82*, 115–128. [CrossRef]
- Niknamfar, A.H.; Niaki, S.T.A.; Pasandideh, S.H.R. Robust optimization approach for an aggregate production–distribution planning in a three-level supply chain. *Int. J. Adv. Manuf. Technol.* 2015, 76, 623–634. [CrossRef]
- 7. Su, W.; Huang, S.X.; Fan, Y.S.; Mak, K.L. Integrated partner selection and production–distribution planning for manufacturing chains. *Comput. Ind. Eng.* **2015**, *84*, 32–42. [CrossRef]
- 8. Nishi, T.; Konishi, M.; Ago, M. A distributed decision making system for integrated optimization of production scheduling and distribution for aluminum production line. *Comput. Chem. Eng.* **2007**, *31*, 1205–1221. [CrossRef]
- 9. Haq, A.N.; Vrat, P.; Kanda, A. An integrated production–inventory–distribution model for manufacture of urea: A case. *Int. J. Prod. Econ.* **1991**, 25, 39–49. [CrossRef]
- 10. Yilmaz, P.; Çatay, B. Strategic level three-stage production distribution planning with capacity expansion. *Comput. Ind. Eng.* **2006**, *51*, 609–620. [CrossRef]
- 11. Boudia, M.; Louly, M.A.O.; Prins, C. A reactive GRASP and path relinking for a combined production–distribution problem. *Comput. Oper. Res.* **2007**, *34*, 3402–3419. [CrossRef]
- 12. Hamedi, M.; Farahani, R.Z.; Husseini, M.M.; Esmaeilian, G.R. A distribution planning model for natural gas supply chain: A case study. *Energy Policy* **2009**, *37*, 799–812. [CrossRef]
- 13. Lee, Y.H.; Kim, S.H. Production–distribution planning in supply chain considering capacity constraints. *Comput. Ind. Eng.* **2002**, *43*, 169–190. [CrossRef]
- 14. Safaei, A.S.; Husseini, S.M.; Farahani, R.Z.; Jolai, F.; Ghodsypour, S.H. Integrated multi-site production–distribution planning in supply chain by hybrid modelling. *Int. J. Prod. Res.* **2010**, *48*, 4043–4069. [CrossRef]
- 15. Mohamed, Z.M. An integrated production–distribution model for a multinational company operating under varying exchange rates. *Int. J. Prod. Econ.* **1999**, *58*, 81–92. [CrossRef]
- 16. Tang, J.; Yung, K.-L.; Ip, A.W.H. Heuristics-based integrated decisions for logistics network systems. *J. Manuf. Syst.* **2004**, *23*, 1–13. [CrossRef]
- 17. Dhaenens-Flipo, C.; Finke, G. An integrated model for an industrial production–distribution problem. *IIE Trans.* **2001**, *33*, 705–715. [CrossRef]
- 18. Aliev, R.A.; Fazlollahi, B.; Guirimov, B.G.; Aliev, R.R. Fuzzy-genetic approach to aggregate production–distribution planning in supply chain management. *Inf. Sci.* 2007, 177, 4241–4255. [CrossRef]
- 19. Gunnarsson, H.; Rönnqvist, M.; Carlsson, D. Integrated production and distribution planning for Södra cell AB. *J. Math. Model. Algorithms* **2007**, *6*, 25–45. [CrossRef]
- 20. Ferrio, J.; Wassick, J. Chemical supply chain network optimization. *Comput. Chem. Eng.* **2008**, *32*, 2481–2504. [CrossRef]
- 21. Bashiri, M.; Badri, H.; Talebi, J. A new approach to tactical and strategic planning in production–distribution networks. *Appl. Math. Model.* **2012**, *36*, 1703–1717. [CrossRef]
- 22. Raa, B.; Dullaert, W.; Aghezzaf, E. A matheuristic for aggregate production–distribution planning with mould sharing. *Int. J. Prod. Econ.* **2013**, *145*, 29–37. [CrossRef]
- 23. Kanyalkar, A.P.; Adil, G.K. Aggregate and detailed production planning integrating procurement and distribution plans in a multi-site environment. *Int. J. Prod. Res.* **2007**, *45*, 5329–5353. [CrossRef]

- 24. Kanyalkar, A.P.; Adil, G.K. An integrated aggregate and detailed planning in a multi-site production environment using linear programming. *Int. J. Prod. Res.* 2005, *43*, 4431–4454. [CrossRef]
- 25. Nasiri, G.R.; Zolfaghari, R.; Davoudpour, H. An integrated supply chain production–distribution planning with stochastic demands. *Comput. Ind. Eng.* **2014**, 77, 35–45. [CrossRef]
- 26. Coronado, J.L. *An Optimization Model for Strategic Supply Chain Design under Stochastic Capacity Disruptions;* Texas A&M University: College Station, TX, USA, 2007.
- Demirli, K.; Yimer, A.D. Production-distribution with fuzzy costs. In Proceedings of the NAFIPS 2006—2006 Annual Meeting of the North American Fuzzy Information Processing Society, Montreal, QC, Canada, 3–6 June 2006; pp. 702–707.
- Gholamian, N.; Mahdavi, I.; Tavakkoli-Moghaddam, R.; Mahdavi-Amiri, N. Comprehensive fuzzy multi-objective multi-product multi-site aggregate production planning decisions in a supply chain under uncertainty. *Appl. Soft Comput.* 2015, *37*, 585–607. [CrossRef]
- Nazim, M.; Hashim, M.; Nadeem, A.H.; Yao, L.; Ahmad, J. Multi objective production–distribution decision making model under fuzzy random environment. In Proceedings of the Eighth International Conference on Management Science and Engineering Management, Lisbon, Portugal, 25–27 July 2014; pp. 591–601.
- 30. Xu, J.; Wei, P. Production-distribution planning of construction supply chain management under fuzzy random environment for large-scale construction projects. *J. Ind. Manag. Optim.* **2013**, *9*, 31–56. [CrossRef]
- Chen, M.; Wang, W. A linear programming model for integrated steel production and distribution planning. *Int. J. Oper. Prod. Manag.* 1997, 17, 592–610. [CrossRef]
- 32. Cohen, M.A.; Lee, H.L. Strategic analysis of integrated production-distribution systems: Models and methods. *Oper. Res.* **1988**, *36*, 216–228. [CrossRef]
- 33. Bilgen, B. Application of fuzzy mathematical programming approach to the production allocation and distribution supply chain network problem. *Expert Syst. Appl.* **2010**, *37*, 4488–4495. [CrossRef]
- 34. Selim, H.; Araz, C.; Ozkarahan, I. Collaborative production–distribution planning in supply chain: A fuzzy goal programming approach. *Transp. Res. Part E Logist. Transp. Rev.* **2008**, 44, 396–419. [CrossRef]
- 35. Pyke, D.F.; Cohen, M.A. Performance characteristics of stochastic integrated production-distribution systems. *Eur. J. Oper. Res.* **1993**, *68*, 23–48. [CrossRef]
- 36. Al-e-Hashem, S.M.J.M.; Baboli, A.; Sadjadi, S.J.; Aryanezhad, M.B. A multiobjective stochastic production-distribution planning problem in an uncertain environment considering risk and workers productivity. *Math. Probl. Eng.* **2011**, 2011, 406398.
- 37. Fahimnia, B.; Farahani, R.Z.; Marian, R.; Luong, L. A review and critique on integrated production–distribution planning models and techniques. *J. Manuf. Syst.* **2013**, *32*, 1–19. [CrossRef]
- Masulli, F.; Mitra, S. Natural computing methods in bioinformatics: A survey. *Inf. Fusion* 2009, 10, 211–216. [CrossRef]
- 39. Mitra, S.; Datta, S.; Perkins, T.; Michailidis, G. *Introduction to Machine Learning and Bioinformatics*; CRC Press: Boca Raton, FL, USA, 2008.
- 40. Zhang, X.; Feng, T.; Niu, Q.; Deng, X. A Novel Swarm Optimisation Algorithm Based on a Mixed-Distribution Model. *Appl. Sci.* **2018**, *8*, 632. [CrossRef]
- 41. Wang, Y.; Lu, J. Optimization of China crude oil transportation network with genetic ant colony algorithm. *Information* **2015**, *6*, 467–480. [CrossRef]
- 42. Chang, Y.-C.; Li, V.C.; Chiang, C.-J. An ant colony optimization heuristic for an integrated production and distribution scheduling problem. *Eng. Optim.* **2014**, *46*, 503–520. [CrossRef]
- 43. Cheng, B.-Y.; Leung, J.Y.-T.; Li, K. Integrated scheduling of production and distribution to minimize total cost using an improved ant colony optimization method. *Comput. Ind. Eng.* **2015**, *83*, 217–225. [CrossRef]
- 44. Calvete, H.I.; Galé, C.; Oliveros, M.-J. Bilevel model for production–distribution planning solved by using ant colony optimization. *Comput. Oper. Res.* **2011**, *38*, 320–327. [CrossRef]
- 45. Fahimnia, B.; Luong, L.; Marian, R. Genetic algorithm optimisation of an integrated aggregate production–distribution plan in supply chains. *Int. J. Prod. Res.* **2012**, *50*, 81–96. [CrossRef]
- Khalifehzadeh, S.; Seifbarghy, M.; Naderi, B. Solving a fuzzy multi objective model of a production–distribution system using meta-heuristic based approaches. J. Intell. Manuf. 2017, 28, 95–109. [CrossRef]
- 47. Armentano, V.A.; Shiguemoto, A.L.; Løkketangen, A. Tabu search with path relinking for an integrated production–distribution problem. *Comput. Oper. Res.* **2011**, *38*, 1199–1209. [CrossRef]

- Chan, F.T.S.; Chung, S.H.; Wadhwa, S. A hybrid genetic algorithm for production and distribution. *Omega* 2005, *33*, 345–355. [CrossRef]
- 49. Gen, M.; Syarif, A. Hybrid genetic algorithm for multi-time period production/distribution planning. *Comput. Ind. Eng.* **2005**, *48*, 799–809. [CrossRef]
- 50. Altiparmak, F.; Gen, M.; Lin, L.; Karaoglan, I. A steady-state genetic algorithm for multi-product supply chain network design. *Comput. Ind. Eng.* 2009, *56*, 521–537. [CrossRef]
- 51. Taha, H.A. *Operations Research: An Introduction (for VTU);* Pearson Education: Upper Saddle River, NJ, USA, 2007.
- 52. Zimmermann, H.-J. Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets Syst.* **1978**, *1*, 45–55. [CrossRef]
- 53. Lai, Y.-J.; Hwang, C.-L. Fuzzy mathematical programming. In *Fuzzy Mathematical Programming*; Springer: Berlin/Heidelberg, Germany, 1992; pp. 74–186.
- 54. Dorigo, M.; Maniezzo, V.; Colorni, A. Ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern. Part B* **1996**, *26*, 29–41. [CrossRef] [PubMed]
- 55. Cheng, C.-B.; Mao, C.-P. A modified ant colony system for solving the travelling salesman problem with time windows. *Math. Comput. Model.* **2007**, *46*, 1225–1235. [CrossRef]
- 56. Yu, B.; Yang, Z.-Z.; Yao, B. An improved ant colony optimization for vehicle routing problem. *Eur. J. Oper. Res.* **2009**, *196*, 171–176. [CrossRef]
- 57. T'kindt, V.; Monmarché, N.; Tercinet, F.; Laügt, D. An ant colony optimization algorithm to solve a 2-machine bicriteria flowshop scheduling problem. *Eur. J. Oper. Res.* **2002**, *142*, 250–257. [CrossRef]
- 58. Pandey, H.M.; Chaudhary, A.; Mehrotra, D. A comparative review of approaches to prevent premature convergence in GA. *Appl. Soft Comput.* **2014**, *24*, 1047–1077. [CrossRef]
- 59. Chowdhury, B.; Garai, G. A review on multiple sequence alignment from the perspective of genetic algorithm. *Genomics* **2017**, *109*, 419–431. [CrossRef] [PubMed]
- 60. Florentino, H.O.; Cantane, D.R.; Santos, F.L.P.; Bannwart, B.F. Multiobjective genetic algorithm applied to dengue control. *Math. Biosci.* **2014**, *258*, 77–84. [CrossRef] [PubMed]
- 61. Holland, J.H. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligenc; MIT Press: Cambridge, MA, USA, 1992.
- 62. Goldberg, D.E. Optimization & Machine Learning. In *Genetic Algorithm Search*; Addison-Wesley: Reading, MA, USA, 1989.
- 63. Tao, G.; Michalewicz, Z. Inver-over operator for the TSP. In Proceedings of the International Conference on Parallel Problem Solving from Nature, Amsterdam, The Netherlands, 27–30 September 1998; pp. 803–812.
- 64. Du, K.L.; Swamy, M.N.S. Particle Swarm Optimization. In *Search and Optimization by Metaheuristics*; Birkhäuser: Cham, Switzerland, 2016.
- 65. Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, 220, 671–680. [CrossRef] [PubMed]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).