



Article A Diploid Evolutionary Algorithm for Sustainable Truck Scheduling at a Cross-Docking Facility

Maxim A. Dulebenets

Florida A&M University-Florida State University, 2525 Pottsdamer Street, Building A, Suite A124, Tallahassee, FL 32310-6046, USA; mdulebenets@eng.famu.fsu.edu; Tel.: +1-(850)-410-6621

Received: 24 February 2018; Accepted: 21 April 2018; Published: 25 April 2018



Abstract: Supply chains have become more time-sensitive in recent years. Delays in supply chain operations may cause significant negative externalities, including lost sales and customers. In order to facilitate the product distribution process within supply chains, reduce the associated delays, and improve sustainability of the supply chain operations, many distribution companies started implementing the cross-docking technique. One of the challenging problems in management of the cross-docking facilities is efficient scheduling of the arriving trucks. This study proposes a novel Diploid Evolutionary Algorithm for the truck scheduling problem at a cross-docking facility, which—unlike the Evolutionary Algorithms presented in the cross-docking literature to date—stores the genetic information from the parent chromosomes after performing a crossover operation. The objective of the formulated mathematical model is to minimize the total truck service cost. The conducted numerical experiments demonstrate that the optimality gap of the developed algorithm does not exceed 0.18% over the considered small size problem instances. The analysis of the realistic size problem instances indicates that deployment of the developed solution algorithm reduces the total truck handling time, the total truck waiting time, and the total truck delayed departure time on average by 6.14%, 32.61%, and 34.01%, respectively, as compared to a typical Evolutionary Algorithm. Furthermore, application of the diploidy concept decreases the total truck service cost by 18.17%.

Keywords: supply chains; cross-docking; truck scheduling; evolutionary computation; diploidy concept; service cost savings

1. Introduction

Nowadays the "*just-in-time*" concept plays a critical role in supply chains [1–9]. The key supply chain players, including manufacturers, logistics companies, distribution facility operators, and retailers, aim to minimize delays in their operations, as those delays may negatively affect the whole supply chain and ultimately may even result in lost sales and customers. The cross-docking technique has been heavily used in industry to facilitate the flow of different products within distribution facilities, support "*just-in-time*" deliveries within supply chains, and improve sustainability of the supply chain operations [1]. Cross-docking distribution facilities operate as follows. The inbound trucks, delivering various products from different manufactures, are assigned to the inbound doors of a cross-docking facility (CDF), where the products are unloaded from the inbound trucks (typically by forklift operators) and moved to the dedicated storage areas. Then, the products are decomposed, sorted, consolidated, and loaded on the outbound trucks. The outbound trucks further deliver the products to the end customers. The products, delivered by the inbound trucks, generally do not spend more than 24 h at the CDFs [1]. Due to its efficiency, the cross-docking technique has been adopted by the largest retailers (e.g., Walmart, Target, COSTCO, Auchan), shipping companies (e.g., FedEx, UPS, USPS, DHL), and other private and public distribution companies across the world.

2 of 23

Managing a CDF is a quite challenging task. The CDF operators not only have to manage the available handling resources (e.g., forklift operators, conveyor belts, storage areas, etc.) efficiently, but also to ensure the proper scheduling of the arriving inbound and outbound trucks. This study focuses on the problem of scheduling the inbound and outbound trucks for service at the available doors of the CDF. Optimizing the truck scheduling will allow the CDF operators to achieve their major objectives [1], which include the following: reducing the inventory level, minimizing the total product stay time, minimizing the total truck loading and unloading time, maximizing the door utilization, and others.

The problem of scheduling trucks at the CDFs received a lot of attention from the community, especially in recent years due to increasing freight volumes within supply chains. The truck scheduling problem can be reduced to the unrelated machine scheduling problem, which belongs to the class of non-deterministic polynomial time hard (*NP*-hard) problems [1,10]. A large number of various Evolutionary Algorithms (EAs) were developed in the past to solve the truck scheduling problem at the CDFs [1]. In EAs, the solutions (i.e., the candidate truck schedules) are encoded into the chromosomes and changed using the crossover and mutation operators throughout the algorithm evolution until a certain convergence criterion is met [11,12]. The EAs, presented in the cross-docking literature to date, rely on the haploidy concept, where two parent chromosomes produce the offspring chromosomes. The crossover operator may cause significant genetic information from the parent chromosomes. The crossover operator may cause significant genetic changes [11,12] and ultimately worsen fitness of the offspring chromosomes in some cases. Deployment of the haploidy concept may cause the loss of parent chromosomes with high fitness values. Due to the latter fact, the population may be dominated by the low quality offspring chromosomes, which will negatively affect fitness of the solutions at the EA convergence.

On the other hand, many living organisms (e.g., humans, most animal species, plants, fungi, algae) are diploid in their nature [13] and contain two copies of each chromosome in a cell. Throughout the crossover operation in diploid organisms, the offspring chromosomes are produced, and a copy of each parent chromosome is stored. Based on the latter biological principle, this study proposes a novel Diploid EA (DEA) for the truck scheduling problem at the CDF. Unlike EAs presented in the cross-docking literature, the developed DEA stores a copy of each parent chromosome after the crossover operation, which is expected to prevent the dominance of low quality offspring chromosomes in the population and will improve the solution quality at convergence of the algorithm.

The remaining sections of this manuscript are ordered in the following manner. The second section provides an overview of the relevant CDF literature with a primary focus on the EA algorithms, which were developed for the truck scheduling problem. The third section describes the problem, addressed in this study, with a focus on the key attributes of the considered CDF. The fourth section presents a mathematical model for the truck scheduling problem at the CDF. The fifth section focuses on the main features of the developed DEA algorithm, while the sixth section presents a set of numerical experiments that were performed in this study to evaluate the proposed solution algorithm and draw certain managerial insights. The last section of the manuscript provides a summary of findings and outlines potential extensions for the future research.

2. Literature Review

The CDF truck scheduling problem has received a lot of attention from researchers in recent years. Due to *NP*-hard complexity of the problem, a number of heuristic and metaheuristic algorithms were proposed in the literature, including: (1) Evolutionary Algorithms (EAs); (2) Particle Swarm Optimization (PSO); (3) Tabu Search (TS); (4) Ant Colony Optimization (ACO); (5) Differential Evolution (DE); (6) Squeaky Wheel Optimization (SWO); (7) Simulated Annealing (SA); and others. The literature review, conducted as a part of this study, primarily focuses on the studies that applied EA-based algorithms to solve the CDF truck scheduling problem. For a more detailed review of the cross-docking literature, this study refers to Ladier and Alpan [1]. The collected studies were

3 of 23

classified into the following groups: (a) general CDF truck scheduling; (b) uncertainty modeling in the CDF truck scheduling; (c) multi-objective CDF truck scheduling; and (d) comparative analysis of the solution approaches.

2.1. General CDF Truck Scheduling

Li et al. [14] presented a mathematical model for the CDF truck scheduling problem, minimizing the total penalty due to earliness and lateness in processing the inbound and outbound containers. Due to the problem complexity, the authors developed an algorithm, which combined features of the SWO and EA algorithms, to obtain good quality truck schedules. It was found that the presented solution approach outperformed CPLEX in terms of the computational time. Furthermore, the developed algorithm returned the truck schedules, which were close to the optimal ones, for the considered problem instances. McWilliams et al. [15] focused on the truck scheduling problem at freight consolidation terminals. The objective function minimized the time span of the product transfer operation. A Simulation-based Algorithm, which relied on the EA, was developed to solve the problem. Based on the conducted numerical experiments, it was found that the proposed solution approach outperformed the arbitrary truck scheduling, where the best truck schedule was selected after generating a large number of random truck schedules. A similar study was conducted by McWilliams et al. [16]. The authors introduced a list-scheduling heuristic for the inbound trailer assignment within the Simulation-based Algorithm, which applied the EA. The computational experiments demonstrated that introduction of the local search heuristic improved the objective function value on average by 3.5%.

Yu et al. [17] proposed a mathematical model for the CDF truck scheduling problem, aiming to minimize the total man-hour recruitment. An EA was developed to solve the problem. The numerical experiments, conducted using the CDF operational data, indicated that the presented EA was able to reduce the total man-hour recruitment by 20% as compared to the current practice. Chmielewski et al. [18] studied the problem of scheduling the inbound and outbound trucks at less-than-truckload terminals. The problem was formulated as a mixed integer programming model, aiming to minimize the total cost associated with unloading the inbound trucks, internal cargo transport, and loading the outbound trucks. The authors proposed the decomposition-and-column-generation approach and the EA to solve the problem. It was found that the computational time of the decomposition-and-column-generation approach did not exceed 30 min for the considered problem instances. Furthermore, the EA was found to be efficient for the cases in a dynamic environment.

McWilliams [19] presented a mathematical model for the truck scheduling problem at the parcel hub, aiming to minimize the sum of maximum parcel workloads. Due to the complexity of the presented formulation, an EA was developed to solve the problem. The computational experiments demonstrated that the proposed EA outperformed the Simulation-based Scheduling Algorithm and Random Scheduling in terms of the solution quality and required an acceptable computational time. Choy et al. [20] studied the truck scheduling problem at the space-constrained industrial CDFs. The objective of the proposed mathematical model minimized the total truck waiting time. The authors developed an EA, which allowed insertion of the inbound orders without causing a significant disturbance in the outbound order schedule. The numerical experiments showed that the proposed solution approach was able to reduce the total truck waiting time by 10% and 20% under heavy and normal conditions, respectively.

Golias et al. [21] proposed a mathematical model for scheduling the inbound and outbound trucks at the CDF, aiming to minimize the total truck service time and the total cost due to early and late truck departures. The authors developed a hybrid EA, which relied on a local search heuristic. The heuristic continuously executed the mutation operation until certain improvement in the fitness of solutions was achieved or the CPU time limit was met. The computational experiments illustrated the efficiency of the developed solution approach. Joo and Kim [22] modeled the CDF, where three types of trucks could be served, including outbound-only trucks, inbound-only trucks, and compound trucks. The objective

minimized the last truck service completion time (or makespan). The EA and Self-Evolution Algorithm were developed to solve the problem. The algorithms were executed for the randomly generated problem instances. The results indicated that the Self-Evolution Algorithm was generally able to return superior solutions. Mohtashami [23] studied the CDF truck scheduling problem with temporary storage, aiming to minimize the makespan. It was assumed that preemption was allowed (i.e., the truck could leave the dock, so the other truck could park at that dock and start the service). An EA algorithm was designed to solve the problem. The numerical experiments indicated that preemption could significantly reduce makespan and improve efficiency of the cross-docking operations.

2.2. Uncertainty Modeling in the CDF Truck Scheduling

A number of researchers considered uncertainty in the CDF truck scheduling. Konur and Golias [24] studied the CDF truck scheduling problem with the truck arrival time uncertainty. Only truck arrival time windows (i.e., truck arrival time lower and upper bounds) were assumed to be known. Three approaches for truck scheduling were evaluated using the EA algorithm: (a) deterministic, assuming the truck arrival at the mid-arrival time window; (b) pessimistic, assuming that the worst truck arrivals would be realized; and (c) optimistic, assuming that the best truck arrivals would be realized. The numerical experiments showcased that the hybrid truck scheduling approach, which relied on the aforementioned truck scheduling approaches, was found to be superior. Konur and Golias [25] proposed a bi-objective bi-level mathematical formulation for the CDF truck scheduling problem with the unknown truck arrival times. The first objective aimed to minimize the total average truck service cost, while the second one minimized the total truck service cost range. The authors used an EA to solve the problem. The computational experiments demonstrated that the proposed methodology outperformed the First Come First Served policy in terms of the average total truck service cost and the total truck service cost range.

Heidari et al. [26] studied the truck scheduling problem at the CDF, considering uncertainty in truck arrivals. The problem was formulated as a bi-objective bi-level mathematical model. The first objective minimized the total average truck service cost, while the second one aimed to minimize the total truck service cost range. The Multi-Objective Differential Evolution (MODE) and Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) were adopted to solve the problem. MODE was found to be more efficient. Amini and Tavakkoli-Moghaddam [27] addressed the truck scheduling problem at the CDF, considering potential breakdowns during the truck service. The truck service breakdowns were modeled using a Poisson distribution. A bi-objective mathematical model was proposed, where the first objective aimed to minimize the total tardiness of the outbound trucks, while the second objective maximized the reliability of cross-docking operations. Three metaheuristics were applied to solve the problem, including: (1) NSGA-II; (2) Multi-Objective Simulated Annealing (MOSA); and (3) MODE. The numerical experiments demonstrated efficiency of the developed methodology and the proposed solution algorithms. Ladier and Alpan [28] conducted a review of studies, which specifically focused on modeling uncertainty in the cross-docking operations. It was mentioned that EAs had been widely used to solve the robust optimization problems in the cross-docking literature.

2.3. Multi-Objective CDF Truck Scheduling

A number of studies addressed the multi-objective truck scheduling problem at CDFs. Boloori Arabani et al. [29] focused on a multi-objective CDF truck scheduling problem, where the first objective function minimized the truck service makespan, while the second one minimized the total lateness in truck service completion. A total of three multi-objective algorithms were used to solve the problem, including: (1) NSGA-II; (2) Strength Pareto Evolutionary Algorithm-II (SPEA-II); and (3) Sub-Population Genetic Algorithm-II (SPGA-II). The computational experiments demonstrated that the Pareto Fronts that were suggested by the NSGA-II and SPGA-II algorithms were inferior to the Pareto Fronts that were returned by the SPEA-II algorithm for the considered problem instances. A similar study was conducted by Boloori Arabani et al. [30], where the following algorithms were compared: (1) SPGA-II; (2) Sub-Population Particle Swarm Optimization-II (SPPSO-II); and (3) Sub-Population Differential Evolution Algorithm-II (SPDE-II). It was found that the SPPSO-II algorithm was more efficient as compared to the SPGA-II and SPDE-II algorithms. Golias et al. [31] proposed bi-objective and bi-level mathematical models for the inbound truck scheduling at the CDF. Two conflicting objectives were considered: (a) minimizing the total truck service time; and (b) minimizing the total truck service completion delays. An EA was used to solve the bi-objective problem, while the bi-level problem was solved using the *k*-th best algorithm. The numerical experiments demonstrated that both formulations could be advantageous for the multi-objective truck scheduling problems at CDFs.

2.4. Comparative Analysis of the Solution Approaches

Several studies compared performance of the EA algorithm against other metaheuristic algorithms for the CDF truck scheduling problem. Miao et al. [32] studied the CDF truck scheduling problem with operational time constraints, aiming to minimize the total truck service cost. Three solution approaches were evaluated, including: (1) CPLEX; (2) TS; and (3) EA. It was found that both metaheuristic algorithms were able to dominate CPLEX for the considered problem size instances. Boloori Arabani et al. [33] proposed a multi-criteria optimization model for the CDF truck scheduling problem, aiming to minimize the total earliness and tardiness in service completion of the arriving trucks. A total of three metaheuristic algorithms were proposed to solve the problem, including EA, PSO, and DE. The numerical experiments indicated that EA outperformed PSO and DE in terms of the solution quality, while PSO generally required the least computational time. Boloori Arabani et al. [34] formulated the CDF truck scheduling problem as a mixed integer programming model. The objective was to minimize the makespan. The EA, TS, PSO, ACO, and DE algorithms were applied to solve the problem. It was found that that performance of the developed algorithms was substantially affected by the problem size.

Vahdani et al. [35] presented the EA and Electromagnetism-like Algorithms to solve the truck scheduling problem at the CDF, assuming that there was no temporary storage. The objective of the proposed integer programming model minimized the total product flow time within the CDF. The results from computational experiments indicated that the developed metaheuristic algorithms were superior to the compound heuristic and the TS algorithm, proposed by Yu [36]. Liao et al. [37] and [38] evaluated a number of metaheuristic algorithms for the CDF truck scheduling problem, including SA, TS, ACO, DE, and two versions of the hybrid DE. The studies highlighted the importance of hybridization and demonstrated that introduction of the local search operators could improve performance of the metaheuristic algorithms. Shiguemoto et al. [39] proposed a hybrid EA, which relied on features of TS, for the truck scheduling at the CDF, aiming to minimize the makespan. The developed solution approach was tested against CPLEX, Firefly metaheuristic, and the algorithm proposed by Yu and Egbelu [40]. The conducted computational experiments indicated that the developed algorithm outperformed the alternative solution approaches.

2.5. Contribution

A detailed review of the cross-docking literature indicates that the CDF truck scheduling problem receives increasing attention from the community. Many studies applied EAs for solving the CDF truck scheduling problem and demonstrated their efficiency against the alternative metaheuristic algorithms [32–40]. A few studies used only mutation in order to generate the new solutions (i.e., creating the offspring chromosomes) within the proposed EAs [21,24,25,31], while the majority of the collected studies applied both crossover and mutation. It was found that the EAs presented in the CDF truck scheduling literature primarily relied on the haploidy concept, where the parent genetic information is not stored before deployment of the crossover operator. The crossover operations may cause significant genetic changes and, ultimately, can even worsen the quality of the produced CDF truck schedules. In order to address such shortcomings, this study proposes the EA algorithm, which relies on the diploidy concept and stores the parent genetic information before application of the crossover

operator (i.e., Diploid EA or DEA). Although the diploidy concept has been widely used in various applications (e.g., evolutionary computation [11], biology [13]), a detailed review of the cross-docking literature indicates that the studies proposing EAs for the CDF truck scheduling problem did not consider application of the diploidy concept. The developed DEA is expected to serve as an efficient decision support tool for the CDF operators, assist with the design of cost-effective CDF truck schedules, and improve sustainability of the CDF and supply chain operations.

3. Cross-Docking Facility Description

This section of the manuscript describes the main attributes of the CDF modeled in this study. A geometric layout of the CDF is presented in Figure 1. It is assumed that the CDF has an I-shape, which is the most common shape of the CDFs used in industry [1]. However, the proposed mathematical model and the solution algorithm (which will be described in the fourth and the fifth sections of the manuscript) can be applied to CDFs of other shapes as well. The available CDF doors are assumed to be in the mixed service mode, that is, each CDF door can serve either inbound or outbound trucks. A set of the available doors at the considered CDF will be further denoted as $D = \{1, \ldots, m\}$.



Figure 1. A cross-docking facility (CDF) layout.

The CDF operator is assumed to have a truck appointment system, where each truck is expected to arrive at a specific time, so the CDF operator can design the truck schedule accordingly. A set of the arriving trucks (inbound + outbound) will be further denoted as $T = \{1, ..., n\}$. Delays in truck arrivals due to various reasons (e.g., traffic congestion, traffic accidents, weather conditions, and others) are not modeled in this study. Once a given truck arrives at the CDF, it will travel to the assigned door. If the assigned door is not available upon the arrival of the truck (as another truck is being served at that door), it will be waiting for service in the dedicated waiting area until the door becomes available. The CDF operator will incur an additional cost for waiting trucks (c_t^{wt} , $\forall t \in T$ in USD/h). Once a given truck is docked at the assigned door, the forklift operators start unloading the products, packaged in boxes, pallets, or other units, from the truck (in case of the inbound truck) or moving the products to the truck (in case of the outbound truck). The scope of this study does not include modeling the deployment strategies for the forklift operators.

The considered CDF has designated storage areas (see Figure 1) to temporarily store the products, delivered by the inbound trucks, before they will be loaded on the assigned outbound trucks.

The capacity of the storage areas is assumed to be sufficient to accommodate the products, delivered by the inbound trucks throughout a given planning horizon (typically 1 shift, ranging from 8 h to 12 h). The CDF operator performs a preliminary planning of the cross-docking operations based on the truck appointment times and the amount of products delivered by each inbound truck, schedules a sufficient number of the forklift operators to serve the trucks, and performs a preliminary assignment of the inbound and outbound trucks to the CDF doors (i.e., "*desired doors*"). Based on the preliminary truck to door assignment, the CDF operator also allocates the storage space for the products to be delivered in the storage areas (generally, as close as possible to the "*desired doors*" to minimize the travel distance by the forklift operators). If a given truck is diverted for service from the "*desired door*" to another door of the CDF due to changes in the schedule, the total handling time for that truck is assumed to increase. The CDF operator incurs the handling cost of trucks (c_t^{ht} , $\forall t \in T$ in USD/h), which includes the equipment costs, insurance costs, compensation of the personnel, facility maintenance costs, etc.

Throughout scheduling of trucks, the CDF operator has to ensure that the service of a given outbound truck cannot start before the service start of the inbound trucks, which carry the products for that outbound truck. Note that a given inbound truck may carry the products for different outbound trucks. The products, transferred within the CDF, are not interchangeable (one product cannot replace another product in a given truck load). Furthermore, the operator of the considered CDF does not allow preemption, i.e. the loading or unloading process cannot be interrupted, once a given truck is docked.

To insure the "*just-in-time*" delivery of products to the customers, the service of each truck should be completed before a specific scheduled departure time (S_t , $t \in T$ in h), known to the CDF operator in advance. The CDF will incur a delayed departure $\cos(c_t^{dt}, \forall t \in T \text{ in USD/h})$, if service of a given truck is completed after its scheduled departure time. The objective of the CDF operator is to design an efficient schedule for the inbound and outbound trucks by minimizing the associated costs, including the following components: (i) the total truck handling cost; (ii) the total truck waiting cost; and (iii) the total truck delayed departure cost.

4. Mathematical Model

This section of the manuscript formulates a mixed integer mathematical model for the problem of scheduling the inbound and outbound trucks at the CDF (which will be referred to as TSPCDF from now on).

4.1. Nomenclature

4.1.1. Sets

| $T = \{1, \dots, n\}$ | set of trucks arriving at the facility (inbound + outbound) |
|------------------------|-------------------------------------------------------------|
| $D = \{1, \ldots, m\}$ | set of doors available at the facility |
| $O = \{1, \ldots, k\}$ | set of truck service orders |

4.1.2. Decision Variables

| $\mathbf{x}_{tdo} \in \{0,1\} \ \forall t \in T,$ | -1 if truck t is assigned in the a^{th} order to dear $d(-0)$ otherwise) |
|---------------------------------------------------|-------------------------------------------------------------------------------------------|
| $d \in D, o \in O$ | =1 if the <i>t</i> is assigned in the <i>t</i> of defined to door <i>u</i> (=0 otherwise) |

4.1.3. Auxiliary Variables

| $\boldsymbol{y}_{tdo} \in R^+ \; \forall t \in T, d \in D,$ | idle time of door <i>d</i> between service of truck <i>t</i> and its immediate |
|-------------------------------------------------------------|--------------------------------------------------------------------------------|
| $o \in O$ | predecessor served as $(o-1)^{th}$ truck (h) |
| $oldsymbol{	au}_t^{st} \in R^+ \ orall t \in T$ | service start time for truck t (h) |
| $\boldsymbol{\tau}_t^{ft} \in R^+ \ \forall t \in T$ | service finish time for truck t (h) |
| $\boldsymbol{\tau}_t^{wt} \in R^+ \; orall t \in T$ | waiting time of truck t (h) |
| $oldsymbol{	au}_t^{dt} \in R^+ \ orall t \in T$ | delayed departure time for truck t (h) |

4.1.4. Parameters

| $A_t \in R^+ \ \forall t \in T$ | arrival time of truck t (h) |
|---------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|
| $H_{td} \in R^+ \ \forall t \in T, d \in D$ | handling time for truck t at door d (h) |
| $S_t \in R^+ \ \forall t \in T$ | scheduled departure time of truck t (h) |
| $C_{t\bar{t}} \in \{0,1\} \ \forall \underline{t}, \overline{t} \in T, \underline{t} \neq \overline{t}$ | =1 if inbound truck \underline{t} carries the products for outbound truck \overline{t} (=0 otherwise) |
| $c_t^{ar{ht}} \in R^+ \ orall t \in T$ | unit handling cost for truck t (USD/h) |
| $c_t^{wt} \in R^+ \ \forall t \in T$ | unit waiting cost for truck <i>t</i> (USD/h) |
| $c_t^{dt} \in R^+ \ \forall t \in T$ | unit delayed departure cost of truck t (USD/h) |
| M | large positive number |

TSPCDF: Truck Scheduling Problem at the CDF

$$\min\left[\sum_{t\in T}\sum_{d\in D}\sum_{o\in O} (H_{td}\boldsymbol{x}_{tdo}\boldsymbol{c}_t^{ht}) + \sum_{t\in T} (\boldsymbol{\tau}_t^{wt}\boldsymbol{c}_t^{wt}) + \sum_{t\in T} (\boldsymbol{\tau}_t^{dt}\boldsymbol{c}_t^{dt})\right] \tag{1}$$

Subject to:

$$\sum_{d \in D} \sum_{o \in O} x_{tdo} = 1 \ \forall t \in T$$
(2)

$$\sum_{t \in T} \mathbf{x}_{tdo} \le 1 \; \forall d \in D, o \in O \tag{3}$$

$$\sum_{\tilde{t}\in T:\tilde{t}\neq t}\sum_{\tilde{o}\in O:\tilde{o}(4)$$

$$\boldsymbol{\tau}_{t}^{st} \geq \sum_{\tilde{t}\in T: \tilde{t}\neq t} \sum_{\tilde{o}\in O: \tilde{o}< o} \left(H_{\tilde{t}d}\boldsymbol{x}_{\tilde{t}d\tilde{o}} + \boldsymbol{y}_{\tilde{t}d\tilde{o}}\right) + \boldsymbol{y}_{tdo} - M(1-\boldsymbol{x}_{tdo}) \ \forall t \in T, d \in D, o \in O$$
(5)

$$\boldsymbol{\tau}_t^{ft} \ge \boldsymbol{\tau}_t^{st} + \sum_{d \in D} \sum_{o \in O} \left(H_{td} \boldsymbol{x}_{tdo} \right) \, \forall t \in T \tag{6}$$

$$\boldsymbol{\tau}_{\bar{t}}^{st} \geq \boldsymbol{\tau}_{\underline{t}}^{st} - M(1 - C_{\underline{t}\bar{t}}) \; \forall \underline{t}, \overline{t} \in T, \underline{t} \neq \overline{t}$$

$$\tag{7}$$

$$\boldsymbol{\tau}_t^{wt} \ge \boldsymbol{\tau}_t^{st} - A_t \; \forall t \in T \tag{8}$$

$$\boldsymbol{\tau}_t^{dt} \ge \boldsymbol{\tau}_t^{ft} - S_t \; \forall t \in T \tag{9}$$

The objective function (1) of the TSPCDF mathematical model minimizes the total cost, associated with service of the inbound and outbound trucks at the CDF. The following cost components are considered in the model: (i) the total truck handling cost; (ii) the total truck waiting cost; and (iii) the total truck delayed departure cost. Constraint set (2) ensures that each truck (either inbound or outbound), arriving at the CDF, will be assigned for service at one of the available doors in any service order. Constraint set (3) indicates that only one truck can be assigned for service at each door of the CDF in a given order. Constraint set (4) guarantees that a service of each truck should start only after its arrival at the CDF. Constraint sets (5) and (6) calculate the service start time and the service finish time for each truck, respectively. Constraint set (7) ensures that the service of a given outbound truck cannot start before the service start of the inbound trucks, which carry the products for that outbound truck. Constraint sets (8) and (9) estimate the waiting time and the delayed departure time for each truck, served at the CDF.

5. Solution Algorithm Description

Many of the mathematical formulations, which were presented in the CDF truck scheduling literature, can be reduced to one of the machine scheduling problems [14,16,19,21,24,25]. The machine scheduling problems are known to have *NP*-hard complexity [10]. Therefore, the approximate solution algorithms (i.e., heuristic and metaheuristic algorithms) have to be developed in order

to solve the realistic size problem instances of the machine scheduling problems within an acceptable computational time. The problem of scheduling trucks at the CDF, represented by the TSPCDF mathematical model, can be reduced to the unrelated machine scheduling problem, where the available jobs have to be assigned to one of the machines, and the machines have different properties. The small size instances of the TSPCDF mathematical model can be solved optimally within a reasonable computational time. On the other hand, since TSPCDF can be reduced to one of the machine scheduling problems (and, therefore, TSPCDF is NP-hard), the realistic size problem instances will require development of the heuristic or metaheuristic algorithms [10]. As discussed in the introduction section of the manuscript, a lot of researchers applied EAs in the past to solve the truck scheduling problem at the CDFs. However, the EAs proposed in the cross-docking literature to date typically rely on the haploidy concept, where the parent chromosomes are probabilistically altered via crossover without storing the parent genetic information. The crossover operation may cause significant genetic changes and ultimately worsen fitness of the offspring chromosomes in some cases. A potential loss of the genetic information from the parent chromosomes may further increase the number of the low quality offspring chromosomes in the population and negatively affect fitness of the solutions at the algorithm convergence.

In order to avoid the latter drawback, this study proposes a novel DEA, which—unlike those presented in the literature EAs—applies the concept of diploidy, where a copy of each parent chromosome is stored after performing the crossover operation. Such an approach would allow the genetic information from the parent chromosomes to be kept and prevent dominance of the low quality offspring chromosomes in the population. The main DEA steps are presented in Figure 2. The required data structures for variables of the algorithm are initialized in step 0. Then, the algorithm generates the chromosomes of the initial population in step 1, while the fitness of those chromosomes is estimated in step 2. After that, the DEA algorithm starts an iterative procedure, where the parent chromosomes are identified in step 3. Then, the DEA operations are conducted in step 4. The fitness of the offspring chromosomes is estimated in step 5. After that, the offspring selection is performed in step 6. The iterative procedure is terminated within the DEA algorithm, once the convergence criterion is satisfied, and the algorithm returns the truck schedule with the best possible service cost. The next sections of the manuscript describe in detail each component of the DEA algorithm.

5.1. Chromosome Representation

Chromosomes are used to represent the candidate solutions to the TSPCDF mathematical model in the proposed DEA algorithm. Specifically, the chromosomes have an integer encoding and contain the information regarding the truck to door assignment and the truck service order at each door of a given CDF. An example chromosome is illustrated in Figure 3, where a total of 9 inbound and outbound trucks are served at 3 doors of the CDF. Trucks "1", "4", and "5" are served at door "1" (in that service order), trucks "2", "7", and "9" are served at door "2" (in that service order), while "3", "6", and "8" are served at door "3" (in that service order). Note that the components of a chromosome will be referred to as genes [11,12]. Furthermore, the terms "*chromosome*", "*solution*", and "*individual*" will be used interchangeably throughout this manuscript as they have the same meaning.



Figure 2. The DEA algorithm steps.

| $Door \rightarrow$ | 1 | 1 | 1 | 2 | 2 | 2 | 3 | 3 | 3 |
|---------------------|---|---|---|---|---|---|---|---|---|
| Truck \rightarrow | 1 | 4 | 5 | 2 | 7 | 9 | 3 | 6 | 8 |

Figure 3. An example chromosome.

5.2. Initialization of Chromosomes and Population

A First Come First Served (FCFS) policy has been widely used for the machine scheduling problems in order to initialize the chromosomes within EAs [41–43]. Deployment of the FCFS policy or other heuristic procedures generally allows the production of higher quality individuals at the chromosome initialization stage as compared to the purely random chromosome initialization mechanism. This study applies the FCFS heuristic with Inbound Truck Priority (FCFS-ITP), which is an extension of the canonical FCFS policy. Unlike the canonical FCFS policy, the FCFS-ITP heuristic

takes into account the service order requirements for the inbound and outbound trucks, arriving at the CDF (specifically, the service of a given outbound truck cannot start before the service start of the inbound trucks, which carry the products for that outbound truck). The main FCFS-ITP steps are presented in Pseudocode 1 (**PS-1**).

The required data structures for variables of the FCFS-ITP heuristic are initialized in step 0. Then, the inbound trucks are sorted based on their arrival times in the ascending order in step 1, while the outbound trucks are sorted based on their arrival times in the ascending order in step 2. After that, the set of trucks sorted based on their arrival times is constructed in step 3, where the service priority is given to the inbound trucks (i.e., the service of all outbound trucks will start after the service start of all inbound trucks). Then, the FCFS-ITP heuristic starts an iterative procedure, where the first available door is identified in step 6 (notation τ_d^a , $d \in D$ is used to define availability of door *d*). The earliest truck service in the earliest order at the first available door in step 8. The service start and finish times are calculated for a given truck in steps 9 and 10, respectively, while the door availability (i.e., the time when a given door becomes available for service) is updated in step 11. The FCFS-ITP heuristic stops the iterative procedure, once all inbound and outbound trucks are scheduled for service at the available doors of the CDF.

PS-1: First Come First Served Policy with Inbound Truck Priority (FCFS-ITP)

FCFS-ITP $(T, T^{I}, T^{O}, D, O, A_{t}, H_{td})$ **in**: $T = \{1, n\}$ —set of trucks: T^{I}

in: $T = \{1, \dots, n\}$ —set of trucks; $T^I = \{1, \dots, q\}$ —set of inbound trucks; $T^O = \{1, \dots, p\}$ —set of outbound trucks; $D = \{1, ..., m\}$ —set of doors; $O = \{1, ..., k\}$ —set of truck service orders; A_t —arrival time of truck t; H_{td} —handling time for truck t at door d**out:** *x*—truck to door to service order assignment 0: $|T^{IS}| \leftarrow q$; $|T^{OS}| \leftarrow p$; $|T^{S}| \leftarrow n$; $|\boldsymbol{\tau}^{a}| \leftarrow m$; $|\boldsymbol{x}| \leftarrow n \cdot m \cdot k$; $|\boldsymbol{\tau}^{st}| \leftarrow n$; $|\boldsymbol{\tau}^{ft}| \leftarrow n$ 1: $T^{IS} \leftarrow Sort(T^{I}, A_{t})$ \lhd Sort the inbound trucks based on their arrival times 2: $T^{OS} \leftarrow Sort(T^O, A_t)$ ⊲ Sort the outbound trucks based on their arrival times 3: $T^S \leftarrow T^{IS} \cup T^{OS}$ ⊲ Initialize the set of trucks sorted based on their arrival times 4: $t \leftarrow 1$ 5: for all $t \in T^S$ do 6: $d \leftarrow argmin_d(\boldsymbol{\tau}_d^a)$ ⊲ Identify the first available door 7: $o \leftarrow argmin_o(\mathbf{x}_{tdo})$ ⊲ Identify the earliest truck service order at that door 8: $x_{tdo} \leftarrow 1$ ⊲Assign a truck for service at the first available door in the earliest order 9: $\boldsymbol{\tau}_t^{st} \leftarrow max(A_t, \boldsymbol{\tau}_d^a)$ Calculate the start service time of a truck 10: $\boldsymbol{\tau}_t^{ft} \leftarrow \boldsymbol{\tau}_t^{st} + H_{td}$ Calculate the finish service time of a truck 11: $\boldsymbol{\tau}_{d}^{a} \leftarrow \boldsymbol{\tau}_{t}^{ft}$ Update availability of the door, to which the truck was assigned 12: $t \leftarrow t + 1$ 13: end for 14: return *x*

Note that the FCFS-ITP heuristic is deterministic, and, therefore, it will produce identical individuals in the initial DEA population. The latter will negatively affect diversity of the population, and will limit the explorative DEA capabilities in the beginning of the search process. To avoid such drawbacks, the developed DEA generates half of the initial population using the FCFS-ITP heuristic, while the other half is initialized randomly. The size of the initial population (*Psize*) will be set based on the parameter tuning analysis (described in Section 6.2) and will not change from one generation to another.

5.3. Selection Procedures

Two selection procedures are applied within the developed DEA algorithm: (a) parent selection; and (b) offspring selection. The parent selection procedure aims to identify a group of individuals that

will participate in the DEA operations and produce the new offspring, while the offspring selection procedure aims to determine a group of individuals that will survive in a given generation and become potential parents in the next generation. The study adopts the fitness proportionate selection mechanism (which is also referred to as *"roulette wheel selection"* in the literature) for the parent selection, while the binary tournament selection mechanism is applied for the offspring selection.

The fitness proportionate selection mechanism is based on a stochastic iterative procedure, where individuals with higher fitness values have higher chances to become parents. However, the fitness proportionate selection mechanism does not fully eliminate chances of individuals with lower fitness values to become parents as well. As for the binary tournament selection mechanism, it is based on multiple tournaments, where two individuals are randomly selected from the population. Then, fitness values of the selected individuals are compared, and the individual with a higher fitness value will become a parent. The tournaments are continuously executed until the desired population size is reached. Both fitness proportionate selection and binary tournament selection mechanisms are widely used in EAs, and more details regarding both selection mechanisms can be found in the relevant EA and cross-docking literature [11,12,35].

5.4. DEA Operations

Once the parent chromosomes are selected, the DEA algorithm applies the crossover and mutation operations in order to produce the offspring chromosomes. Both crossover and mutation operations are described in Sections 5.4.1 and 5.4.2 of the manuscript.

5.4.1. Crossover Operation

Many different crossover operators were applied in EAs, including *N*-point crossover, arithmetic crossover, uniform crossover, partially mapped crossover, order crossover, and others [11,12,41–43]. However, some of those crossover operators may cause infeasibility for the integer chromosome representation, as the one adopted in this study. The infeasibility consists of the fact that some trucks may be scheduled for service multiple times, while some trucks may not be scheduled for service at all. The order crossover operator will be used in the proposed DEA to produce the offspring chromosomes and ensure their feasibility. An example of a crossover operation is illustrated in Figure 4. Two parent chromosomes are randomly selected from the list of candidate parent chromosomes. The probability of a crossover operation for a given chromosome is determined by parameter *Cprob*. The value of *Cprob* will be set based on the parameter tuning analysis (described in Section 6.2). After that, a segment of the first parent chromosome is copied to the first offspring chromosome. Note that the segment length of the first parent chromosome to be copied to the first offspring chromosome is set randomly. In the presented example, genes with trucks "3", "5", "6", and "8" are copied to the first offspring from the first parent (see Figure 4). Then, the genes with trucks that were not scheduled for service are copied to the first offspring chromosome from the second parent chromosome. In the presented example, genes with trucks "2", "1", "9", "4", and "7" are copied to the first offspring from the second parent (see Figure 4). The second offspring is created in a similar fashion: genes with trucks "1", "3", "6", and "9" are copied to the second offspring from the second parent, while missing trucks "2", "5", "8", "4", and "7" are copied to the second offspring from the first parent (see Figure 4).

Unlike typical EAs that rely on the haploidy concept (see Figure 5), where the parent genetic information is not stored after the crossover operation, the proposed DEA applies the diploidy concept. Specifically, a copy of each parent chromosome is stored in a separate data structure after the crossover operation, and then the parent chromosomes are added to the DEA population before the mutation operation. The latter will cause an increase in the population size throughout the DEA operations and the offspring fitness evaluation (i.e., steps 4 and 5 in Figure 2), but after application of the offspring selection mechanism the population size will become equal to the initial population size (*Psize*) again.









Notes: - chromosomes to be used after the crossover operation

Figure 5. An example of a haploid crossover operation.

5.4.2. Mutation Operation

The proposed DEA algorithm executes the mutation operator for each individual in the population after performing the crossover operation. The swap mutation operator will be used in this study to mutate the offspring chromosomes (note that copies of the parent chromosomes, generated as a result of the diploid crossover operation will be mutated as well). Selection of the swap mutation operator can be justified by its efficiency for the integer chromosomes [41–43]. An example of a mutation operation is illustrated in Figure 6. It can be noticed that in the mutated chromosome trucks "2" and "5", initially scheduled for service at door "1", are diverted for service to door "3". In the meantime, trucks "4" and "9", initially scheduled for service at door "3", are diverted for service to door "1". The mutation rate (*Mrate*) defines the number of genes swapped in each chromosome of the DEA population. In the presented example, the mutation rate is equal to Mrate = 4 genes. The value of *Mrate* will be set based on the parameter tuning analysis (described in Section 6.2).

Before Mutation Operation

| $Door \rightarrow$ | 1 | 1 | 1 | 1 | 2 | 2 | 3 | 3 | 3 |
|---------------------|-----|------|-----|---|---|---|---|---|---|
| $Truck \rightarrow$ | 1 | 2 | 3 | 5 | 6 | 8 | 4 | 7 | 9 |
| After Mutation | Ope | erat | ion | | | | | | |

| | - | | | | | | | | |
|---------------------------|---|---|---|---|---|---|---|---|---|
| $\text{Door} \rightarrow$ | 1 | 1 | 1 | 1 | 2 | 2 | 3 | 3 | 3 |
| $Truck \rightarrow$ | 1 | 9 | 3 | 4 | 6 | 8 | 5 | 7 | 2 |

Figure 6. An example of a mutation operation.

5.5. Fitness of Chromosomes and Infeasibility Avoidance

The fitness function (*F*) values of chromosomes within the proposed DEA are estimated using the following relationship:

$$F = \Psi\left[\sum_{t \in T} \sum_{d \in D} \sum_{o \in O} \left(H_{td} \boldsymbol{x}_{tdo} c_t^{ht} \right) + \sum_{t \in T} \left(\boldsymbol{\tau}_t^{wt} c_t^{wt} \right) + \sum_{t \in T} \left(\boldsymbol{\tau}_t^{dt} c_t^{dt} \right) \right]$$
(10)

Along with the components of the TSPCDF objective function, the fitness function includes a penalty term (Ψ), which was introduced to penalize the infeasible chromosomes, produced via the DEA operations. Specifically, the stochastic DEA operators (i.e., crossover and mutation) may produce the chromosomes, where service of the inbound trucks, carrying the products for a given outbound truck, is scheduled after the service start of that outbound truck. Since the latter scenario contradicts the realistic cross-docking operations, the proposed DEA considers such chromosomes as infeasible. Introduction of the penalty term allows significant reduction of the chances of the infeasible individuals to survive from one generation to another. The value of Ψ will be set based on the parameter tuning analysis (described in Section 6.2). Note that for the feasible individuals the penalty term will be set to $\Psi = 1.0$.

5.6. Elitism

The proposed DEA algorithm applies the elitist strategy, where the individual with the highest fitness value will be present in the population for more than one generation. The elitist strategy is implemented before the parent selection procedure at each generation to ensure that the fittest individual will survive in the given generation and will become a potential parent in the next generation.

5.7. Convergence Criterion

The maximum number of generations (gen^{max}) will be adopted as the convergence criterion for the developed DEA algorithm. The value of gen^{max} will be set based on the parameter tuning analysis (described in Section 6.2).

6. Numerical Experiments

This section of the manuscript focuses on description of the numerical experiments, which were conducted to assess efficiency of the proposed DEA algorithm. The developed DEA will be compared to a typical EA, which relies on the haploidy concept (i.e., in case of a crossover operation the genetic information from the parent chromosomes is not stored) and has been widely used in the CDF truck scheduling literature. The EA and DEA algorithms were coded in MATLAB 2016a and executed on a Dell Intel(R) Core[™] i7 Processor with 32 GB of RAM. The numerical experiments focus on three major steps, which include the following: (1) algorithmic parameter tuning; (2) optimality gap estimation; and (3) analysis of the realistic size problem instances. The first step of the numerical experiments aims to determine the appropriate values for parameters of the EA and DEA algorithms. The second step of the numerical experiments focuses on evaluation of the truck schedules, produced by the EA and

DEA algorithms, against the global optimal truck schedules for the small size problem instances. In this study, the global optimal truck schedules were obtained using the CPLEX optimization algorithm. The third step of the numerical experiments focuses on a comparative analysis of the EA and DEA algorithms for the realistic size problem instances in terms of the objective function and computational time values. Moreover, the truck schedules, proposed by the developed algorithms, are compared in terms of the truck handling time, waiting time, and delayed departure time, which are considered as critical CDF performance indicators from a practical standpoint. The next sections of the manuscript elaborate on the input data generation and describe all the aforementioned steps, which were undertaken throughout the numerical experiments.

6.1. Input Data Selection

The input data for the TSPCDF mathematical model were generated based on the available cross-docking and freight operations literature [1-9,14-51]. The parameter values, selected for the numerical experiments, are presented in Table 1. A total of six door availability scenarios were modeled for the considered CDF, including the following: (i) 2 doors; (ii) 3 doors; (iii) 4 doors; (iv) 6 doors; (v) 8 doors; and (vi) 10 doors. This study assumed that the inter-arrival pattern of the inbound and outbound trucks followed the exponential distribution with an average of 5 min. or 0.0833 h. Notation "*EXP*" will be adopted further in this study for the exponentially distributed pseudorandom numbers.

Table 1. Selected input parameter values.

| Parameter | Selected Value |
|--------------------------------------------------------------------|---------------------------------------|
| CDF door availability (doors) | [2; 3; 4; 6; 8; 10] |
| Average truck inter-arrival time (h) | EXP[0.0833] |
| Truck handling time at the "desired door": H_t^0 , $t \in T$ (h) | U[1.5; 2.0] |
| Requested truck departure time: $S_t, t \in T$ (h) | $S_t = A_t + H_t^0 \cdot U[1.2; 1.4]$ |
| Unit handling cost: c_t^{ht} , $t \in T$ (USD/h) | U[200;400] |
| Unit waiting cost: c_t^{wt} , $t \in T$ (USD/h) | U[100;200] |
| Unit delayed departure cost: c_t^{dt} , $t \in T$ (USD/h) | U[300; 500] |

The handling time of trucks at the "desired doors" $(H_t^0, t \in T)$ was assigned as follows: $H_t^0 = U[1.5; 2.0] \ \forall t \in T$ (h), where notation "U" will be adopted further in this study for the uniformly distributed pseudorandom numbers. The "desired door" for each truck was set randomly. It was assumed that the handling time of a given truck could increase from 5% to 10%, if that truck was not assigned to for service at its "desired door". The requested departure time of truck *t* was assigned based on its arrival time and handling time at the "desired door" as follows: $S_t = A_t + H_t^0 \cdot U[1.2; 1.4] \ \forall t \in T$ (h). It was assumed that each inbound truck, arriving at the considered CDF, could carry the products for up to 3 outbound trucks. The unit truck handling cost was set as follows: $c_t^{ht} = U[200; 400] \ \forall t \in T$ (USD/h). The unit truck waiting cost was assigned as follows: $c_t^{wt} = U[100; 200] \ \forall t \in T$ (USD/h). The unit truck delayed departure cost was set as follows: $c_t^{dt} = U[300; 500] \ \forall t \in T$ (USD/h).

A total of 60 problem instances were developed using the generated numerical data by changing the number of doors at the CDF and the number of arriving trucks. All the developed problem instances can be categorized into two groups: (i) small size problem instances (I-10 through I-30), where the number of trucks was varied from 6 to 15 with an increment of 1 truck, while the number of doors was varied from 2 to 4 with an increment of 1 door; and (ii) realistic size problem instances (I-31 through I-60), where the number of trucks was varied from 84 to 120 with an increment of 4 trucks, while the number of doors was varied from 6 to 10 with an increment of 2 doors.

6.2. Algorithmic Parameter Tuning

Before estimating the optimality gap values of the EA and DEA algorithms and analyzing the managerial insights for the realistic size problem instances, it is necessary to select the appropriate

parameter values for the algorithms based on the parameter tuning analysis [41,42]. A full factorial design methodology was adopted to conduct the parameter tuning analysis for the developed algorithms. According to the full factorial design, the parameter values were selected based on the numerical experiments, where each parameter combination was evaluated for a given algorithm. Both EA and DEA algorithms have only 3 major parameters (i.e., population size—Psize, crossover probability—*Cprob*, and mutation rate—*Mrate*). A total of 4 candidate values were considered for each parameter of a given algorithm. The candidate values of the algorithmic parameters were set based on the preliminary algorithmic runs and the corresponding parameter ranges, which are reported in the cross-docking studies that relied on EAs [15,16,22,26,27,33,34]. A total of 5 problem instances were chosen randomly from the generated realistic size problem instances to conduct the parameter tuning. Since both EA and DEA algorithms are stochastic, a total of 5 replications were performed to estimate the average values of the objective function for each parameter combination and each problem instance. Therefore, a total of (4 candidate values) $(3 \text{ parameters}) \cdot (5 \text{ problem instances}) \cdot (5 \text{ replications}) = 1600 \text{ runs}$ were performed throughout the parameter tuning analysis for each algorithm. The final parameter values were adopted based on the tradeoff between the computational time and the obtained objective function values (to ensure that the developed algorithms will be able to produce the solutions of a good quality within an acceptable computational time). The results are reported in Table 2, including the following data: (i) algorithm; (ii) parameter; (iii) description of the parameter; (iv) candidate values considered for a given parameter; and (v) adopted value.

| _ | | | | | |
|---|-----------|-----------|------------------------------|--------------------------|---------------|
| | Algorithm | Parameter | Description of the Parameter | Candidate Values | Adopted Value |
| | EA | Psize | Population size | [30; 40; 50; 60] | 60 |
| | EA | Cprob | Crossover probability | [0.40; 0.50; 0.60; 0.70] | 0.60 |
| | EA | Mrate | Mutation rate | [2; 4; 6; 8] | 2 |
| | DEA | Psize | Population size | [30; 40; 50; 60] | 30 |
| | DEA | Cprob | Crossover probability | [0.40; 0.50; 0.60; 0.70] | 0.70 |
| | DEA | Mrate | Mutation rate | [2; 4; 6; 8] | 2 |
| | | | | | |

Table 2. The results of the parameter tuning analysis.

Note that, based on the parameter tuning analysis results, the population size for the EA algorithm was set to Psize = 60 individuals, while the population size for the DEA algorithm was set to Psize = 30 individuals. Therefore, even after application of the crossover operation the DEA algorithm will not have advantages over the EA algorithm in terms of the population size (i.e., the population size will temporarily become Psize = 60 individuals for the DEA algorithm, as a copy of each parent chromosome will be present in the population until the offspring selection procedure). Furthermore, throughout the parameter tuning analysis, it was found that the penalty value of $\Psi = 2.0$ for the infeasible individuals was adequate, since none of the individuals suggested by the EA and DEA algorithms at convergence were infeasible for all the considered problem instances. Furthermore, no significant improvements in the fitness function values of the EA and DEA algorithms were recorded after 3000 generations. Thus, the maximum number of generations will be set to $gen^{max} = 3000$ generations for both EA and DEA algorithms.

6.3. Optimality Gap Estimation

As indicated in Section 5 of the manuscript, due to *NP*-hard complexity of the TSPCDF mathematical model only small size problem instances can be solved using the exact optimization algorithms within a reasonable computational time. The first set of numerical experiments in this study aimed to estimate the optimality gaps of the EA and DEA algorithms and compare the objective function values of the solutions, produced by the developed algorithms, against the optimal ones for the small size problem instances (I-1 through I-30). Such analysis will enable assessment of the quality of solutions (i.e., truck schedules), proposed by the EA and DEA algorithms. The latter aspect is critical from a practical standpoint, as the CDF operators aim to design the truck schedule with the

least possible cost (as close as possible to the minimum cost), associated with service of the inbound and outbound trucks. The TSPCDF mathematical model was coded in General Algebraic Modeling System (GAMS) and solved using CPLEX for each one of the generated small size problem instances. Throughout the analysis, the relative optimality gap for CPLEX was restricted to 1.00%, while the allowable computational time was set to 3600 s (i.e., 1 h). The developed EA and DEA algorithms were launched for the considered small size problem instances as well. A total of 5 replications were performed for each problem instance.

The results of the optimality gap analysis are summarized in Table 3, where the following data are presented: (i) instance number; (ii) number of doors at the CDF; (iii) number of trucks arriving at the CDF; (iv) the optimal objective function value, obtained by CPLEX; (v) average over 5 replications CPLEX computational time; (vi) average objective function values, obtained by the EA and DEA algorithms; (vii) average optimality gap values for the EA and DEA algorithms; and (viii) average computational time for the EA and DEA algorithms. Note that in Table 3 **Z** is a notation used for the objective function values, while **G** is a notation used for the optimality gap values. The optimality gap for algorithm *a* (*G*_{*a*}) was estimated as follows: $G_a = \frac{Z_a - Z_{CPLEX}}{Z_{CPLEX}}$, where Z_a is the average objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by algorithm *a*; Z_{CPLEX} is the optimal objective function value, obtained by CPLEX.

Table 3. The results of the optimality gap analysis.

| | | | CI | PLEX | | EA | | | DEA | |
|----------|---------|----------|---------------------------|----------|---------------------------|--------------------------------|-------|---------------------------|------|--------|
| Instance | # Doors | # Trucks | Z, 10 ³ USD | CPU, s | Z, 10 ³ USD | Z, 10 ³ G, % CPU, s | | Z, 10 ³ USD | G, % | CPU, s |
| I-1 | 2 | 6 | 10.024 | 43.24 | 10.024 | 0.00 | 19.60 | 10.024 | 0.00 | 20.00 |
| I-2 | 2 | 7 | 13.406 | 82.42 | 13.406 | 0.00 | 20.18 | 13.406 | 0.00 | 20.59 |
| I-3 | 2 | 8 | 17.115 | 169.05 | 17.115 | 0.00 | 21.09 | 17.115 | 0.00 | 21.53 |
| I-4 | 2 | 9 | 21.061 | 280.81 | 21.061 | 0.00 | 22.82 | 21.061 | 0.00 | 23.30 |
| I-5 | 2 | 10 | 26.442 | 536.58 | 26.442 | 0.00 | 23.42 | 26.442 | 0.00 | 23.93 |
| I-6 | 2 | 11 | 31.752 | 898.52 | 31.752 | 0.00 | 24.45 | 31.752 | 0.00 | 24.99 |
| I-7 | 2 | 12 | 37.268 | 1543.44 | 37.268 | 0.00 | 25.34 | 37.268 | 0.00 | 25.91 |
| I-8 | 2 | 13 | 42.927 | 2329.38 | 42.950 | 0.05 | 26.22 | 42.927 | 0.00 | 26.83 |
| I-9 | 2 | 14 | 48.995 | 3300.40 | 49.151 | 0.32 | 27.03 | 49.030 | 0.07 | 27.70 |
| I-10 | 2 | 15 | N/A | >3600.00 | 55.542 | N/A | 28.29 | 55.159 | N/A | 29.00 |
| I-11 | 3 | 6 | 6.970 | 68.96 | 6.970 | 0.00 | 20.67 | 6.970 | 0.00 | 21.19 |
| I-12 | 3 | 7 | 9.172 | 131.45 | 9.172 | 0.00 | 21.47 | 9.172 | 0.00 | 22.03 |
| I-13 | 3 | 8 | 11.535 | 269.64 | 11.535 | 0.00 | 22.39 | 11.535 | 0.00 | 22.97 |
| I-14 | 3 | 9 | 14.048 | 447.90 | 14.048 | 0.00 | 23.18 | 14.048 | 0.00 | 23.83 |
| I-15 | 3 | 10 | 17.481 | 855.84 | 17.481 | 0.00 | 24.22 | 17.481 | 0.00 | 24.94 |
| I-16 | 3 | 11 | 20.716 | 1433.14 | 20.716 | 0.00 | 25.17 | 20.716 | 0.00 | 25.94 |
| I-17 | 3 | 12 | 24.196 | 2261.78 | 24.196 | 0.00 | 25.77 | 24.196 | 0.00 | 26.58 |
| I-18 | 3 | 13 | 27.764 | 2915.35 | 28.020 | 0.92 | 26.81 | 27.764 | 0.00 | 27.66 |
| I-19 | 3 | 14 | 31.636 | 3593.63 | 31.938 | 0.96 | 27.56 | 31.666 | 0.09 | 28.45 |
| I-20 | 3 | 15 | N/A | >3600.00 | 36.105 | N/A | 28.84 | 35.450 | N/A | 29.82 |
| I-21 | 4 | 6 | 5.480 | 120.99 | 5.480 | 0.00 | 21.14 | 5.480 | 0.00 | 21.88 |
| I-22 | 4 | 7 | 6.956 | 230.62 | 6.956 | 0.00 | 21.77 | 6.956 | 0.00 | 22.53 |
| I-23 | 4 | 8 | 8.675 | 473.06 | 8.675 | 0.00 | 22.66 | 8.675 | 0.00 | 23.48 |
| I-24 | 4 | 9 | 10.643 | 785.81 | 10.643 | 0.00 | 23.52 | 10.643 | 0.00 | 24.37 |
| I-25 | 4 | 10 | 12.993 | 1501.53 | 12.993 | 0.00 | 24.33 | 12.993 | 0.00 | 25.23 |
| I-26 | 4 | 11 | 15.320 | 2314.36 | 15.320 | 0.00 | 25.50 | 15.320 | 0.00 | 26.47 |
| I-27 | 4 | 12 | 17.563 | 3068.17 | 17.890 | 1.86 | 26.25 | 17.563 | 0.00 | 27.25 |
| I-28 | 4 | 13 | 20.059 | 3594.81 | 20.519 | 2.29 | 27.30 | 20.096 | 0.18 | 28.38 |
| I-29 | 4 | 14 | N/A | >3600.00 | 23.741 | N/A | 27.87 | 22.804 | N/A | 28.97 |
| I-30 | 4 | 15 | N/A | >3600.00 | 26.502 | N/A | 29.19 | 25.577 | N/A | 30.35 |

It can be noticed that the CPLEX computational time is significantly affected with the problem size. CPLEX was not able to solve the TSPCDF mathematical model within the specified time limit for problem instances I-10, I-20, I-29, and I-30. On the other hand, both EA and DEA algorithms provided solutions to the TSPCDF mathematical model within a reasonable computational time. Specifically, the average computational time comprised 24.47 s and 25.20 s for the EA and DEA algorithms, respectively, over the considered small size problem instances. Moreover, the results from the optimality gap analysis demonstrate that the developed solution algorithms were able to

provide solutions, which were close to the optimal ones. The maximum optimality gap of the EA algorithm comprised 2.29%, while the maximum DEA optimality gap comprised 0.18%. Throughout the numerical experiments, it was also found that the DEA optimality gaps were generally smaller than the EA optimality gaps, which underlines a higher accuracy of the DEA algorithm as compared to the EA algorithm for the small size problem instances.

6.4. Analysis of the Realistic Size Problem Instances

The second set of numerical experiments in this study aimed to compare the solutions, provided by the EA and DEA algorithms for the realistic size problem instances (I-31 through I-60), and draw important managerial insights. The developed EA and DEA algorithms were launched for the considered realistic size problem instances. A total of 5 replications were performed for each problem instance. The results are reported in Table 4, where the following data are presented: (i) instance number; (ii) number of doors at the CDF; (iii) number of trucks arriving at the CDF; (iv) average objective function values, obtained by the EA and DEA algorithms; (v) average computational time for the EA and DEA algorithms; (vi) average total truck handling time (THT) for the solutions, provided by the EA and DEA algorithms; (vii) average total truck waiting time (TWT) for the solutions, provided by the EA and DEA algorithms; and (viii) average total truck delayed departure time (TDT) for the solutions, provided by the EA and DEA algorithms.

| | | | | | EA | | | | | DEA | | |
|----------|---------|----------|--------------------|--------|--------|---------|---------|--------------------|--------|--------|---------|---------|
| Instance | # Doors | # Trucks | Z, 10 ³ | CPU, | THT, | TWT, | TDT, | Z, 10 ³ | CPU, | THT, | TWT, | TDT, |
| | | | USD | s | h | h | h | USD | s | h | h | h |
| I-31 | 6 | 84 | 501.845 | 106.67 | 168.06 | 927.64 | 936.90 | 432.237 | 110.23 | 161.47 | 750.12 | 752.90 |
| I-32 | 6 | 88 | 547.382 | 109.37 | 175.29 | 972.37 | 981.03 | 479.831 | 113.11 | 170.15 | 834.67 | 838.44 |
| I-33 | 6 | 92 | 602.021 | 111.28 | 183.14 | 1181.90 | 1191.27 | 518.227 | 115.20 | 177.04 | 908.19 | 911.85 |
| I-34 | 6 | 96 | 659.985 | 115.97 | 191.49 | 1231.61 | 1242.34 | 570.792 | 120.22 | 184.97 | 998.96 | 1003.12 |
| I-35 | 6 | 100 | 711.989 | 119.86 | 199.41 | 1261.94 | 1272.32 | 620.088 | 124.29 | 192.36 | 1094.26 | 1098.12 |
| I-36 | 6 | 104 | 772.629 | 123.94 | 207.90 | 1473.63 | 1485.14 | 668.512 | 128.61 | 200.65 | 1184.89 | 1189.41 |
| I-37 | 6 | 108 | 829.513 | 127.56 | 214.49 | 1636.57 | 1647.30 | 720.325 | 132.63 | 208.26 | 1286.91 | 1291.49 |
| I-38 | 6 | 112 | 891.122 | 133.23 | 222.60 | 1745.40 | 1757.01 | 782.828 | 138.60 | 215.96 | 1402.01 | 1407.30 |
| I-39 | 6 | 116 | 955.486 | 138.41 | 230.29 | 1864.28 | 1876.20 | 838.507 | 144.03 | 224.00 | 1503.26 | 1508.86 |
| I-40 | 6 | 120 | 1016.981 | 142.62 | 239.03 | 2011.47 | 2023.99 | 903.379 | 148.87 | 232.34 | 1621.41 | 1627.63 |
| I-41 | 8 | 84 | 357.316 | 101.84 | 166.93 | 683.07 | 691.30 | 293.271 | 106.38 | 156.64 | 485.44 | 483.68 |
| I-42 | 8 | 88 | 393.770 | 105.75 | 176.19 | 683.89 | 693.49 | 321.700 | 110.49 | 164.27 | 532.34 | 530.56 |
| I-43 | 8 | 92 | 427.472 | 109.99 | 182.19 | 837.07 | 845.43 | 344.579 | 115.18 | 170.35 | 576.23 | 573.68 |
| I-44 | 8 | 96 | 468.842 | 114.38 | 191.65 | 890.53 | 900.84 | 383.101 | 119.81 | 178.95 | 645.82 | 644.00 |
| I-45 | 8 | 100 | 504.641 | 118.70 | 197.62 | 958.65 | 967.76 | 412.542 | 124.35 | 185.97 | 699.19 | 697.54 |
| I-46 | 8 | 104 | 540.792 | 123.48 | 204.26 | 1062.66 | 1070.91 | 454.287 | 129.60 | 194.06 | 768.50 | 766.81 |
| I-47 | 8 | 108 | 589.544 | 127.68 | 214.50 | 1098.72 | 1113.27 | 482.431 | 134.04 | 200.53 | 830.58 | 827.83 |
| I-48 | 8 | 112 | 627.948 | 133.42 | 225.57 | 1127.98 | 1138.72 | 518.353 | 140.20 | 208.76 | 897.35 | 895.65 |
| I-49 | 8 | 116 | 668.407 | 138.04 | 228.23 | 1358.38 | 1368.33 | 565.641 | 145.39 | 217.27 | 983.73 | 982.78 |
| I-50 | 8 | 120 | 721.946 | 142.81 | 239.65 | 1371.62 | 1384.77 | 605.739 | 150.96 | 224.10 | 1049.93 | 1048.17 |
| I-51 | 10 | 84 | 274.704 | 101.43 | 167.46 | 533.30 | 545.29 | 215.514 | 107.26 | 152.98 | 333.91 | 328.58 |
| I-52 | 10 | 88 | 301.144 | 104.93 | 178.59 | 555.70 | 564.26 | 238.171 | 111.18 | 162.02 | 371.92 | 367.72 |
| I-53 | 10 | 92 | 323.463 | 110.54 | 184.02 | 643.86 | 654.21 | 256.099 | 117.34 | 168.72 | 408.89 | 405.02 |
| I-54 | 10 | 96 | 350.691 | 115.07 | 191.95 | 695.29 | 707.18 | 293.451 | 122.16 | 178.69 | 469.84 | 468.36 |
| I-55 | 10 | 100 | 378.528 | 119.27 | 200.51 | 750.08 | 762.74 | 310.387 | 126.64 | 183.91 | 502.53 | 498.10 |
| I-56 | 10 | 104 | 410.092 | 123.86 | 208.98 | 764.52 | 775.39 | 338.309 | 131.62 | 192.65 | 552.22 | 549.13 |
| I-57 | 10 | 108 | 445.081 | 127.57 | 216.00 | 930.28 | 942.54 | 379.254 | 135.60 | 204.23 | 629.51 | 630.16 |
| I-58 | 10 | 112 | 476.928 | 133.19 | 225.18 | 978.19 | 992.39 | 385.152 | 142.20 | 205.94 | 637.44 | 633.25 |
| I-59 | 10 | 116 | 500.795 | 138.31 | 232.02 | 992.85 | 1006.57 | 419.952 | 147.77 | 212.45 | 699.80 | 694.19 |
| I-60 | 10 | 120 | 539.011 | 142.10 | 240.87 | 1157.33 | 1171.76 | 456.054 | 151.92 | 221.33 | 759.03 | 754.43 |

Table 4. The results of the analysis for realistic size problem instances.

Note that the truck schedules produced by the EA and DEA algorithms were compared in terms of the truck handling time, waiting time, and delayed departure time, as the latter performance indicators critically affect the CDF performance. Specifically, a significant increase in the handling time, waiting time, and delayed departure time of the inbound and outbound trucks may delay delivery of the products to the end customers. Delayed product deliveries will violate the "*just-in-time*" concept, which plays an important role for the CDF operators and other supply chain stakeholders [1–9].

Moreover, delayed product deliveries can also negatively affect the economic sustainability of the supply chain operations, as logistics companies may be imposed monetary penalties for violation of the agreed product delivery time. Based on analysis of the results, it can be observed that the DEA algorithm consistently outperforms the EA algorithm in terms of the objective function values for all the considered realistic size problem instances, and, therefore, provides more cost-efficient schedules for the inbound and outbound trucks at the CDF. The total cost of the DEA truck schedules was on average 18.17% lower as compared to the EA truck schedules. Furthermore, application of the diploidy concept within the developed DEA algorithm resulted in a significant reduction of the truck handling time, waiting time, and delayed departure time (see Figure 7).



Figure 7. The truck handling time, waiting time, and delayed departure time reduction.

Specifically, the total handling time for the DEA truck schedules was on average 6.14% lower as compared to the EA truck schedules over the considered realistic size problem instances. The latter finding can be explained by the fact that the DEA algorithm assigned the arriving trucks either directly to the *"desired doors"* or to the other doors, which were located close to the *"desired doors"* of the CDF. The numerical experiments also demonstrate that application of the diploidy concept within the proposed DEA allowed reducing the total truck waiting time and the total truck delayed departure

time on average by 32.61% and 34.01%, respectively. Therefore, the developed DEA algorithm can assist the CDF operators with the design of efficient schedules for the inbound and outbound trucks, facilitate movements of goods within CDFs, ensure timely product delivery, and ultimately improve economic sustainability within supply chains.

The numerical experiments also show that the storage of genetic information from the parent chromosomes increased the computational time of the DEA algorithm on average by 5.04% as compared to the EA algorithm. The maximum DEA computational time did not exceed 151.92 s over the realistic size problem instances, which can be considered as acceptable from a practical standpoint (i.e., the CDF operator will be able to develop the truck schedules relatively quickly).

6.5. Discussion

Efficient scheduling of the inbound and outbound trucks at CDFs plays a critical role for the product distribution process within supply chains. Since the CDF truck scheduling problem is of a high complexity, a large number of various heuristic and metaheuristic algorithms were proposed in the cross-docking literature to obtain truck schedules of a good quality within an acceptable computational time (including EAs, TS, PSO, ACO, DE, and other algorithms, as discussed in the literature review section of the manuscript). A lot of studies demonstrated superiority of EAs against the alternative metaheuristic algorithms, which were developed for the CDF truck scheduling problem. This study proposes an EA algorithm, which relies on the diploidy concept (i.e., the DEA algorithm), where the parent genetic information is stored before application of the proposed DEA algorithm against a typical EA algorithm, which relied on the haploidy concept, where the parent genetic information was not stored before deployment of the crossover operator.

Throughout the numerical experiments, it was found that both EA and DEA algorithms were able to obtain the truck schedules, which were close to the optimal truck schedules, for the small size problem instances. The EA and DEA optimality gaps did not exceed 2.29% and 0.18%, respectively. Analysis of the realistic size problem instances indicates that application of the diploidy concept reduced the total truck handling time, the total truck waiting time, and the total truck delayed departure time on average by 6.14%, 32.61%, and 34.01%, respectively, for the DEA truck schedules as compared to the EA truck schedules. Moreover, the DEA algorithm yielded the truck schedules which had 18.17% lower truck service costs, as opposed to the truck service costs associated with the EA truck schedules.

7. Conclusions and Future Research

Nowadays the majority of the existing supply chains are driven with the "*just-in-time*" concept. The key supply chain players put more emphasis on timely completion of processes to ensure customer satisfaction and avoid potential losses in sales. In order to facilitate the transportation and handling processes within supply chains, many manufacturing companies, large retailers, shipping companies, and other distribution companies started implementation of the cross-docking technique. The cross-docking facilities provide the fast flow of products from the inbound trucks to the assigned outbound trucks, which further deliver the products to the final customers. One of the most challenging planning problems for the cross-docking facility operators is the truck scheduling problem.

Due to complexity of the truck scheduling problems, many cross-docking studies applied different Evolutionary Algorithms that relied on the haploidy concept, where the parent genetic information could be lost throughout the algorithm evolution. To address the latter drawback, this study proposed a novel biologically-inspired Diploid Evolutionary Algorithm that applied the diploidy concept, where a copy of each parent chromosome was stored after the crossover operation. The numerical experiments demonstrated efficiency of the developed algorithm and promising results. Specifically, the maximum optimality gap of the Diploid Evolutionary Algorithm comprised 0.18%. Based on a comparative analysis against a typical Evolutionary Algorithm, it was found that the proposed algorithm could reduce the total truck waiting time and the total truck delayed departure time on average by 32.61% and 34.01%, respectively. Furthermore, application of the developed algorithm allowed the cross-docking operator decreasing the total truck service cost on average by 18.17%. Therefore, the proposed Diploid Evolutionary Algorithm can assist the cross-docking operators with the design of cost-efficient truck schedules, facilitate the flow of different products within the cross-docking facilities, support "*just-in-time*" deliveries within supply chains, and improve sustainability of the supply chain operations.

The scope of future research may focus on the following aspects: (i) model uncertainty in arrival times of the inbound and outbound trucks; (ii) accounting for the potential congestion issues inside the cross-docking facility; (iii) considering the limited capacity of storage areas; (iv) assessing the effect of preemption on the truck schedules; (v) proposing a multi-objective framework for the conflicting objectives; (vi) deploying additional local search heuristics at various steps of the algorithm; and (vii) comparing the proposed Diploid Evolutionary Algorithm against the alternative state-of-the-art metaheuristic algorithms (e.g., Particle Swarm Optimization, Tabu Search, Ant Colony Optimization, Simulated Annealing).

Author Contributions: M.A.D. conducted all research activities associated with this study, including review of the relevant literature, mathematical model development, design of the Diploid Evolutionary Algorithm, and evaluation of the algorithm via the numerical experiments.

Conflicts of Interest: The author declares no conflict of interest.

References

- Ladier, A.L.; Alpan, G. Cross-docking operations: Current research versus industry practice. *Omega* 2016, 62, 145–162. [CrossRef]
- 2. Salam, M.; Ali, M.; Kan, K. Analyzing Supply Chain Uncertainty to Deliver Sustainable Operational Performance: Symmetrical and Asymmetrical Modeling Approaches. *Sustainability* **2017**, *9*, 2217. [CrossRef]
- 3. Ma, K.; Wang, L.; Chen, Y. A Collaborative Cloud Service Platform for Realizing Sustainable Make-To-Order Apparel Supply Chain. *Sustainability* **2018**, *10*, 11. [CrossRef]
- 4. Li, Z.; Xu, Y.; Deng, F.; Liang, X. Impacts of Power Structure on Sustainable Supply Chain Management. *Sustainability* **2018**, *10*, 55. [CrossRef]
- 5. Zhang, Z.; Zhang, Q.; Liu, Z.; Zheng, X. Static and Dynamic Pricing Strategies in a Closed-Loop Supply Chain with Reference Quality Effects. *Sustainability* **2018**, *10*, 157. [CrossRef]
- 6. Wang, J.; Ran, B. Sustainable Collaborative Governance in Supply Chain. *Sustainability* **2018**, *10*, 171. [CrossRef]
- Han, Y.; Skibniewski, M.; Wang, L. A Market Equilibrium Supply Chain Model for Supporting Self-Manufacturing or Outsourcing Decisions in Prefabricated Construction. *Sustainability* 2017, 9, 2069. [CrossRef]
- 8. Guo, F.; Liu, Q.; Liu, D.; Guo, Z. On Production and Green Transportation Coordination in a Sustainable Global Supply Chain. *Sustainability* **2017**, *9*, 2071. [CrossRef]
- 9. Zhu, L.; Ren, X.; Lee, C.; Zhang, Y. Coordination Contracts in a Dual-Channel Supply Chain with a Risk-Averse Retailer. *Sustainability* **2017**, *9*, 2148. [CrossRef]
- Pinedo, M. Scheduling: Theory, Algorithms, and Systems; Springer International Publishing: New York, NY, USA, 2008; Available online: http://www.springer.com/in/book/9781489990433 (accessed on 2 January 2018).
- 11. Eiben, A.E.; Smith, J.E. *Introduction to Evolutionary Computing*; Springer International Publishing: Berlin, Germany, 2003; Available online: http://www.springer.com/us/book/9783642072857 (accessed on 2 January 2018).
- 12. Sivanandam, S.; Deepa, S. *Introduction to Genetic Algorithms*; Springer International Publishing: Berlin, Germany, 2008; Available online: http://www.springer.com/us/book/9783540731894 (accessed on 2 January 2018).
- 13. Gallardo, M.H.; González, C.A.; Cebrián, I. Molecular cytogenetics and allotetraploidy in the red vizcacha rat, *Tympanoctomys barrerae* (Rodentia, Octodontidae). *Genomics* **2006**, *88*, 214–221. [CrossRef] [PubMed]
- 14. Li, Y.; Lim, A.; Rodrigues, B. Crossdocking: JIT scheduling with time windows. *J. Oper. Res. Soc.* 2004, 55, 1342–1351. [CrossRef]

- 15. McWilliams, D.; Standfield, P.; Geiger, C. The parcel hub scheduling problem: A simulation-based solution approach. *Comput. Ind. Eng.* **2005**, *49*, 393–412. [CrossRef]
- McWilliams, D.; Standfield, P.; Geiger, C. Minimizing the completion time of the transfer operations in a central parcel consolidation terminal with unequal-batch-size inbound trailers. *Comput. Ind. Eng.* 2008, 54, 709–720. [CrossRef]
- 17. Yu, V.; Sharma, D.; Murty, K. Door allocations to origins and destinations at less-than-truckload trucking terminals. *J. Ind. Syst. Eng.* **2008**, *1*, 1–15.
- Chmielewski, A.; Naujoks, B.; Janas, M.; Clausen, U. Optimizing the door assignment in LTL-terminals. *Transp. Sci.* 2009, 43, 198–210. [CrossRef]
- McWilliams, D. Genetic-based scheduling to solve the parcel hub scheduling problem. *Comput. Ind. Eng.* 2009, 56, 1607–1616. [CrossRef]
- 20. Choy, K.L.; Chow, H.; Poon, T.C.; Ho, G. Cross-dock job assignment problem in space-constrained industrial logistics distribution hubs with a single docking zone. *Int. J. Prod. Res.* **2012**, *50*, 2439–2450. [CrossRef]
- 21. Golias, M.M.; Saharidis, G.; Ivey, S.; Haralambides, H. Advances in truck scheduling at a cross dock facility. *Int. J. Inf. Syst. Supply Chain Manag.* **2013**, *6*, 40–62. [CrossRef]
- Joo, C.M.; Kim, B.S. Scheduling compound trucks in multi-door cross-docking terminals. *Int. J. Adv. Manuf. Technol.* 2013, 64, 977–988. [CrossRef]
- 23. Mohtashami, A. Scheduling trucks in cross docking systems with temporary storage and repetitive pattern for shipping trucks. *Appl. Soft Comput.* **2015**, *36*, 468–486. [CrossRef]
- 24. Konur, D.; Golias, M.M. Analysis of different approaches to cross-dock truck scheduling with truck arrival time uncertainty. *Comput. Ind. Eng.* **2013**, *65*, 663–672. [CrossRef]
- 25. Konur, D.; Golias, M.M. Cost-stable truck scheduling at a cross-dock facility with unknown truck arrivals: A meta-heuristic approach. *Transp. Res. Part E* **2013**, *49*, 71–91. [CrossRef]
- 26. Heidari, F.; Zegordi, S.; Tavakkoli-Moghaddam, R. Modeling truck scheduling problem at a cross-dock facility through a bi-objective bi-level optimization approach. *J. Intell. Manuf.* **2018**, *29*, 1155–1170. [CrossRef]
- 27. Amini, A.; Tavakkoli-Moghaddam, R. A bi-objective truck scheduling problem in a cross-docking center with probability of breakdown for trucks. *Comput. Ind. Eng.* **2016**, *96*, 180–191. [CrossRef]
- Ladier, A.; Alpan, G. Robust cross-dock scheduling with time windows. *Comput. Ind. Eng.* 2016, 99, 16–28. [CrossRef]
- 29. Boloori Arabani, A.; Zandieh, M.; Fatemi Ghomi, S. Multi-objective genetic-based algorithms for a cross-docking scheduling problem. *Appl. Soft. Comput.* **2011**, *11*, 4954–4970. [CrossRef]
- 30. Boloori Arabani, A.; Zandieh, M.; Fatemi Ghomi, S. A cross-docking scheduling problem with sub-population multi-objective algorithms. *Int. J. Adv. Manuf. Technol.* **2012**, *58*, 741–761. [CrossRef]
- Golias, M.M.; Saharidis, G.; Boile, M.; Theofanis, T. Scheduling of inbound trucks at a cross-docking facility: Bi-objective vs.b-level modeling approaches. *Int. J. Inf. Syst. Supply Chain Manag.* 2012, *5*, 20–37. [CrossRef]
- 32. Miao, Z.; Lim, A.; Ma, H. Truck dock assignment problem with operational time constraint within crossdocks. *Eur. J. Oper. Res.* **2009**, *192*, 105–115. [CrossRef]
- 33. Boloori Arabani, A.; Fatemi Ghomi, S.; Zandieh, M. A multi-criteria cross-docking scheduling with just-in-time approach. *Int. J. Adv. Manuf. Technol.* **2010**, *49*, 741–756. [CrossRef]
- 34. Boloori Arabani, A.; Fatemi Ghomi, S.; Zandieh, M. Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage. *Expert Syst. Appl.* **2011**, *38*, 1964–1979. [CrossRef]
- 35. Vahdani, D.; Soltani, R.; Zandieh, M. Scheduling the truck holdover recurrent dock cross-dock problem using robust meta-heuristics. *Int. J. Adv. Manuf. Technol.* **2010**, *46*, 769–783. [CrossRef]
- 36. Yu, W. Operational Strategies for Cross Docking Systems. Ph.D. Thesis, Iowa State University, Ames, IA, USA, 2002.
- 37. Liao, T.W.; Egbelu, P.J.; Chang, P.C. Two hybrid differential evolution algorithms for optimal inbound and outbound truck sequencing in cross docking operations. *Appl. Soft Comput.* **2012**, *12*, 3683–3697. [CrossRef]
- Liao, T.W.; Egbelu, P.J.; Chang, P.C. Simultaneous dock assignment and sequencing of inbound trucks under a fixed outbound truck schedule in multi-door cross docking operations. *Int. J. Prod. Econ.* 2013, 141, 212–229. [CrossRef]
- 39. Shiguemoto, A.; Netto, U.; Bauab, G. An efficient hybrid meta-heuristic for a cross-docking system with temporary storage. *Int. J. Prod. Res.* **2014**, *52*, 1231–1239. [CrossRef]

- 40. Yu, W.; Egbelu, P.J. Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. *Eur. J. Oper. Res.* **2008**, *184*, 377–396. [CrossRef]
- 41. Dulebenets, M.A. Application of Evolutionary Computation for berth scheduling at marine container terminals: Parameter tuning versus parameter control. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 25–37. [CrossRef]
- 42. Dulebenets, M.A.; Moses, R.; Ozguven, E.E.; Vanli, A. Minimizing carbon dioxide emissions due to container handling at marine container terminals via hybrid evolutionary algorithms. *IEEE Access* **2017**, *5*, 8131–8147. [CrossRef]
- 43. Dulebenets, M.A. A novel Memetic Algorithm with a deterministic parameter control for efficient berth scheduling at marine container terminals. *Marit. Bus. Rev.* **2017**, *2*, 302–330. [CrossRef]
- 44. Dulebenets, M.A.; Ozguven, E.E. Vessel scheduling in liner shipping: Modeling transport of perishable assets. *Int. J. Prod. Econ.* 2017, *184*, 141–156. [CrossRef]
- 45. Dulebenets, M.A. Minimizing the total liner shipping route service costs via application of an efficient collaborative agreement. *IEEE Trans. Intell. Transp. Syst.* **2018**, 1–14. [CrossRef]
- 46. Dulebenets, M.A. The green vessel scheduling problem with transit time requirements in a liner shipping route with emission control areas. *Alex. Eng. J.* **2018**, *57*, 331–342. [CrossRef]
- 47. Dulebenets, M.A. The vessel scheduling problem in a liner shipping route with heterogeneous vessel fleet. *Int. J. Civ. Eng.* **2018**, *16*, 19–32. [CrossRef]
- 48. Dulebenets, M.A. Advantages and disadvantages from enforcing emission restrictions within emission control areas. *Marit. Bus. Rev.* 2016, *1*, 107–132. [CrossRef]
- 49. Dulebenets, M.A. A comprehensive multi-objective optimization model for the vessel scheduling problem in liner shipping. *Int. J. Prod. Econ.* **2018**, *196*, 293–318. [CrossRef]
- 50. Dulebenets, M.A. Green vessel scheduling in liner shipping: Modeling carbon dioxide emission costs in sea and at ports of call. *Int. J. Transp. Sci. Technol.* **2018**, *7*, 26–44. [CrossRef]
- 51. Naskaris, S.; Pudukadan, R.; Smith, K.; Maher, M. *Cost Analysis for a Cross Dock Alternative to Supply Local Produce from Food Hub to Grocery Store via a Large Regional Distribution Center*; Center for Environmental Farming Systems, North Carolina State University: Raleigh, NC, USA, 2014.



© 2018 by the author. 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/).