**Moving Products from Suppliers to Customers: Cross-Docking versus Direct Shipping**

Amalia I. Nikolopoulou, Christos D. Tarantilis and Emmanouil E. Zachariadis

*Department of Management Science,*

*Athens University of Economics and Business, Athens, Greece, and*

Panagiotis P. Repoussis

*Howe School of Technology Management,*

*Stevens Institute of Technology, Hoboken, NJ, USA*

**Abstract**

**Purpose –** The goal of this paper is to evaluate the suitability of two distribution strategies, i.e., the *direct shipping* and the *cross-docking*, under various network options and operational settings and examine how these variations are translated in terms of transportation costs.

**Design/methodology/approach –** An innovative local search based optimization method is proposed for addressing both problem settings, that employs static move descriptors to guide the search process. The proposed method is tested on existing as well as new benchmark data sets. To that end, various computational experiments have been performed so as to analyze the impact of several spatial and temporal characteristics, such as the geographic distribution of customers and suppliers, the proximity of individual pickup and delivery pairs, the positioning of the depot, the tightness of the capacity and duration constraints, and the time required to handle and consolidate the shipments at the cross dock.

**Findings –** The results obtained indicated that the effectiveness of each strategy heavily depends on various parameters as well as the geographic distribution of customers and suppliers, the supplier-customer pair proximity, and the node connectivity (or demand density).

**Research limitations/implications (if applicable) -**

**Practical implications (if applicable) -**

**Social implications (if applicable)-**

**Originality/value –** None of the existing works in the literature provides a cost benefit analysis between the direct shipping and the cross-docking strategies. The optimization method proposed in this work can be used as the toolset for evaluating the performance of each strategy.

**Keywords** Vehicle routing, Cross-docking, Direct shipping, Pickup-and-delivery, Tabu search

**Paper type** Research paper

xxx

Old ABSTRACT

This work considers the problem of satisfying transportation requests from a set of suppliers to a set of customers. Each request calls for moving products between a pickup and a delivery location pair. A common approach is the direct shipping and delivery of the products without using any intermediate transshipment points or in-transit merge of the shipments. Another alternative that often appears in practice is to use an intermediate cross-dock facility that will act as a consolidation point for the shipments. The goal of this paper is to evaluate these inherently different distribution options and to conduct a comprehensive comparative analysis. For this purpose, a new local search based optimization method has been developed that employs static move descriptors to guide the search process. The proposed method is tested on existing as well as new benchmark data sets. To that end, various computational experiments have been performed so as to analyze the impact of several spatial and temporal characteristics, such as the geographic distribution of customers and suppliers, the proximity of individual pickup and delivery pairs, the positioning of the depot, the tightness of the capacity and duration constraints, and the time required to handle and consolidate the shipments at the cross dock.

**Introduction**

Various network options and distribution strategies can be adopted for moving products from a set of suppliers (origin nodes) to a set of customers (destination nodes). One very common and simple practice is to directly ship products from suppliers to customers assuming a single-echelon process. This strategy is often referred as *direct shipping* (a.k.a. *drop shipping*). Alternatively, the distribution of products can take place in multiple echelons, via intermediate facilities that act as in-transit merging points for the consolidation of the shipments from the various suppliers. This network option and strategy is often referred as *cross-docking*. This paper seeks to evaluate these distribution network options and to analyze how spatial and temporal elements of the underlying network, such as the node connectivity, the geographical dispersion, and the depot centrality, affects their suitability.

Regarding the direct shipping strategy, the products are picked up from the supplier locations and are directly delivered to the customer locations. This is a typical one-to-one scheme where each vehicle route performs one or multiple pickups and deliveries. Besides the ordering (precedence) restrictions, routes must also satisfy coupling constraints, since the supplier(s) and the corresponding customer(s) have to be assigned to the same vehicle route. This problem setting corresponds to the well-known Pickup-and-Delivery Problem (PDP) (Dumas et al., 1991, Savelsbergh and Sol, 1995).

On the other hand, the cross-docking strategy involves the shipping of products downstream from origins to destinations via a single or multiple intermediate transshipment facilities (or cross-docks). The main motivation for applying this strategy is to achieve cost savings through flow consolidation at the cross-dock. This problem setting corresponds to the so-called Vehicle Routing Problem with Cross-Docking (VRPCD) (Wen et al., 2009, Tarantilis, 2013). Note that in the VRPCD with a single cross-dock, the shipping of products is performed in two separate echelons; at the first echelon, the pickup vehicles leave the cross-dock to visit the suppliers and return to the cross-dock. Then, the picked up products are unloaded from the vehicles and they are consolidated. At the second echelon, the consolidated products are sorted and loaded to the delivery vehicles to be distributed to the customer locations.

Although the PDP and VRPCD have been extensively studied in the literature, none of the existing works provide a cost benefit analysis between the direct shipping and cross-docking strategies. Indeed, there is an evident need to study the effect of the network characteristics and configuration on the routing costs for each strategy, and to compare the relative performance. To that end, our main aim is to evaluate the suitability of each strategy under various network options and operational settings, and examine how these variations are translated in terms of transportation costs. For this purpose, an innovative local search based optimization method is proposed for addressing both problem settings, and it is used as the toolset for evaluating the performance of each strategy. It is worth to highlight that the proposed optimization method outperforms most of the current state-of-the-art solution approaches both for the PDP and the VRPCD, while it matches and/or improves the best known solutions for many well-known benchmark problem instances of the literature.

Various computational experiments are also reported for comparing the two distribution strategies. In particular, several problem instances are constructed to cover cases where customers and suppliers are located in the same area as well as cases where suppliers and customers are positioned into different geographic regions. The role of the supplier-customer pair proximity is also examined. To do so, several problem sub-classes are created and solved, involving remote, close and mixed pickup and delivery pairs. In addition, we try to examine the effect of the vehicle capacity and the tightness of the maximum route duration constraints on the total transportation costs. The impact of the depot positioning with respect to the various supplier and customer nodes is also considered. Finally, we examine how the time required for cross-dock consolidation operations can affect the total transportation costs, and consequently the relative effectiveness of the direct and cross-docking strategies.

The remainder of this paper is organized as follows. We begin by presenting related work in the literature. We then give the description and notation of the problem settings considered. In the next section, the proposed solution methodology for each of the two settings is presented. We then report and discuss the computational results of our study. The concluding section presents the outcomes of this work.

**Related work**

The problem of transferring products between origin and destination pairs has received great research interest. Regarding previously published models and solution approaches, we distinguish between two main groups, namely direct shipping and shipping via intermediate facilities.

The first group includes problem settings where products are directly shipped from a set of suppliers to a set of associated customers, as considered by the classic archetypal Pickup-and-Delivery Problem (PDP) (Dumas et al., 1991; Savelsbergh and Sol, 1995). Numerous papers related to the PDP can be found in the literature. For an extensive review of PDP variants and solution methods the reader is referred to Cordeau et al. (2008) and Parragh et al. (2008). A significant part of the literature on the PDP considers time windows constraints for the service of customers. This problem variant corresponds to the Pickup and Delivery Problem with Time Windows (PDPTW). One of the earliest works on the PDPTW is due to Nanry and Barnes (2000) and a so-called Reactive Tabu Search is proposed. Li and Lim (2001) deal with the PDPTW and developed a hybrid metaheuristic combining Simulated Annealing and Tabu Search. Other metaheuristic approaches have been later developed for solving the PDPTW by Bent and Van Hentenryck (2006), Ropke and Pisinger, (2006), and Nagata and Kobayashi (2010). All of the aforementioned works consider a hierarchical objective that primarily minimizes the number of routes and secondarily minimizes the total travel distance.

The second group refers to problem settings where the distribution of products from a set of suppliers to a set of customers takes place via intermediate facilities. These facilities have stationary predefined locations, and they also referred as transshipment or transfer points. In this type of problems, the constraint ensuring that the pickup and delivery points of a given request should be serviced by the same vehicle route, does not necessarily hold. Depending on whether the transshipment takes place at a (cross-dock) facility or at a customer location(s), we distinguish between two problem variants, i.e., the VRPCD and the PDP with Transfers and Time Windows (PDPTWT), respectively.

The literature on the VRPCD is limited. Lee at al. (2006) are among the first that study a transportation problem with a cross-docking facility. They consider a two-level distribution network with a single cross-dock, assuming that all vehicles performing pickups arrive at the cross-dock simultaneously within a predefined time window. A mathematical model is developed and an algorithm based on Tabu Search is used to solve the problem. In a later work, Wen et al. (2009) have considered a more generalized extension of the VRPCD without imposing simultaneous arrivals at the cross-dock, while they also consider that the same set of vehicles is utilized for both pickups and deliveries. The authors present additional constraints in order to ensure synchronization of vehicles at the cross-dock. A mathematical formulation is presented and a Tabu Search heuristic embedded within an Adaptive Memory procedure is developed to solve the problem. Lower bounds are also reported. Tarantilis (2013) proposes a multi-restart Tabu Search approach for the VRPCD considering different consolidation scenarios, regarding the load exchange between pickup and delivery vehicles. The former work also examines open and close network configurations. In the first case, vehicles start/end their routes at any possible pickup/delivery location while in the close network configuration vehicles start and end their routes at the cross-dock. More recently, Morais et al. (2014) have proposed three Iterated Local Search metaheuristic algorithms to solve the VRPCD. They also generate a new set of instances in order to demonstrate the efficiency of the proposed algorithm on larger size instances.

Regarding the PDPTWT, one of the first studies is due to Mitrovic-Minic and Laporte (2006). The authors presented a local search method for the uncapacitated PDPTWT. Masson et al. (2013) have developed an Adaptive Large Neighborhood Search metaheuristic algorithm for the PDPTWT with multiple transfer points. This approach is tested on real-life instances involving the transportation of mentally or physically disabled people. With regard to exact methods, Cortés et al. (2010) presented a comprehensive mathematical formulation for the PDPTWT. Their solution method consists of a Branch-and-Cut scheme based on a Benders decomposition algorithm. Mues and Pickl (2005) solve the PDPTWT via a Column Generation approach. The proposed method is used for solving two versions of the problem: one with a single transshipment location and another with multiple transshipment locations.

Finally, a limited number of studies have appeared on mixed distribution schemes where both direct deliveries and deliveries via an intermediate transshipment facility are considered. The Pickup and Delivery Problem with Cross-Docking (PDPCD) is an extension of the VRPCD where the constraint requiring all vehicles to visit the cross-dock in between pickup and delivery routes is relaxed. In particular, a request can be either directly delivered from its origin to its destination in one vehicle or, it can be picked up by one vehicle, transported to the cross-dock and delivered to its destination by another vehicle. The first study on a practical PDPCD application with time windows is presented by Petersen and Ropke (2011). A Large Neighborhood Search metaheuristic algorithm is developed to solve real instances from a transportation company with 500-1000 requests, aiming to minimize the total transportation costs. More recently, Santos et al. (2013) have studied the PDPCD without time window constraints. The objective in this work is the minimization of the transportation costs plus the sum of the costs of changing loads at the cross-dock. A Branch-and-Price method is developed for evaluating the contribution of direct routes in the total transportation costs for smaller size instances with up to 30 requests.

While the PDP and VRPCD have been studied in depth, a comparative analysis between the direct shipping and the cross-docking strategies does not appear in the literature. Furthermore, to the best of our knowledge, the effect of distribution network characteristics on the routing costs incurred by each strategy has not yet been investigated. To fill this gap, the present work aims at examining the relative suitability of each distribution strategy in terms of transportation costs for different network configurations. We address this issue through an analysis that examines the impact of several spatial and temporal parameters on the total travelling distance.

**Problem description and notation**

For both the PDP and the VRPCD, the aim is to satisfy a set of transportation requests *n* from a set of suppliers to a set of customers. Let a directed graph *G* = (*V*, *A*), where *V* = {0, 1, …, 2*n*} is the set of nodes and *A* is the set of arcs. Each arc (*i*, *j*) ∈ *A* is associated with a non-negative travel cost *dij*. Node subsets *P* = {1, …, *n*} and *D* = {*n* + 1, …, 2*n*} refer to the sets of pickup and delivery nodes, respectively. Node 0 represents the depot. In the remainder of the paper, the term depot will be used interchangeably and it will refer to the central vehicle station in the case of the PDP or to the cross-dock in the case of the VRPCD. The aim is to design the minimum cost set of routes in order to satisfy a set of transportation requests *R* = {1, …, *n*}, where *n* is the total number of requests. Each transportation request *r* ∈ *R* is associated with a pickup node *i* ∈ *P* and the corresponding delivery node (*n* + *i*) ∈ *D*. In addition, each request *r*∈ *R* is associated with a quantity *qr* that needs to be picked up from *i* ∈ *P* and be delivered to (*n* + *i*) ∈ *D*. For each node *i* ∈ *P* ∪ *D*, there exists a predetermined service time *ti*. Transportation requests can be served by a set of homogeneous vehicles located at the depot. Finally, a maximum vehicle capacity *Q* and a maximum route duration *T* are imposed.

For the PDP, the routes must satisfy the following requirements: (a) each route starts and terminates at the depot; (b) every pickup or delivery node is visited exactly once; (c) the pickup and delivery nodes associated with a single request must be assigned to the same route with the pickup node preceding the delivery one; (d) at no point of any route may the carrying load exceed *Q*; and (e) the total traveling time required by each vehicle to return to the depot cannot exceed *T*.

For the VRPCD, one needs to distinguish between pickup and delivery routes. A pickup route starts from the cross-dock, visits a number of suppliers at most once, and returns to the cross-dock, where the products are consolidated and loaded on the vehicles performing the delivery routes. Along these routes, vehicles visit customers to deliver the requested products. The product consolidation activities (unloading or loading) that take place at the cross-dock are assumed to require a constant amount of service time *s*, which is equal for both activities, regardless of the unloading and loading volumes. Thus, in the case where a vehicle performs both a pickup and a delivery route, the total service time at the cross-dock is equal to the sum of the unloading and loading times, i.e., 2*s*. On the contrary, if a vehicle performs only a pickup or a delivery route, then the total service time at the cross-dock is equal to the unloading or the loading time *s*, respectively. Regarding the synchronization of the pickup and the delivery routes, it is ensured that all products to be loaded on a vehicle for a delivery route are already available at the cross-dock. This practically means that the departure time of vehicle *k* is equal to the latest arrival time at the cross-dock of the requests that will be delivered by *k* plus the total service time at the cross-dock. The (pickup or delivery) routes are subject to the following constraints: (a) all routes start and end at the cross-dock; (b) every pickup node is visited exactly once by a pickup vehicle; (c) every delivery node is visited exactly once by a delivery vehicle, (d) the total product quantity assigned either to a pickup or a delivery route may not exceed the vehicle capacity *Q*; (e) the total travel time of all related pairs of both pickup and delivery routes may not exceed *T*.

The PDP and the VRPCD correspond to NP-hard combinatorial optimization problems (Savelsbergh and Sol, 1995, Wen et al. 2009). Therefore, substantial effort is needed to derive lower bounds and optimal solutions with complete and partial enumeration schemes even for small sized problems. For this reason, we resort to the development of local search metaheuristic algorithm capable of deriving high quality solutions in short computational times for medium and large scale problem instances. This is described in the next section, and can be applied directly for solving both the PDP and the VRPCD.

**Solution method**

A common optimization framework has been developed for solving the PDP and VRPCD. From the algorithmic viewpoint, the proposed framework is initiated by a construction heuristic algorithm that is used to generate an initial heuristic solution. This construction heuristic algorithm is described below. Subsequently, the generated solution is locally improved via a single point trajectory local search metaheuristic algorithm. In particular, a Tabu Search scheme is adopted that is equipped with edge-exchange neighborhood structures as well as with a static move descriptor mechanism for guiding the search process.

*Construction heuristic algorithm*

The proposed construction heuristic algorithm works as follows: Initially, a new empty route is generated that contains the depot at the beginning and at the end of the route. Then, the cost (detour) for inserting each unassigned node (pickup or delivery) to every feasible insertion position is determined. For both problem settings, a minimum cost insertion procedure is employed. More specifically, for the PDP at each iteration the pickup and delivery pair insertion that minimizes the additional travel cost is identified and applied. For the VRPCD, due to the vehicle synchronization constraints, the set of pickup routes is initially constructed by inserting un-routed pickup nodes iteratively at insertion positions that minimally increase the traveling distance. The delivery routes are then constructed by inserting un-routed delivery nodes at positions so that the additional travel cost is minimized. Whenever no feasible insertion positions are found, a new vehicle route is initiated to serve the un-routed set of pickup or delivery customers. To that end, an initial solution is obtained when all nodes are assigned to vehicle routes. Note that for both problems insertions must satisfy the corresponding problem constraints, and no limit is imposed for the total number of vehicle routes.

*Tabu Search*

Given an initial heuristic solution, a Tabu Search metaheuristic algorithm is applied for further improvement. The latter explores the solution search space by performing edge-exchange local moves from a current solution to another neighboring solution. These neighborhood structures are described below.

*Neighborhood structures*

Although the local search framework used for solving the PDP and the VRPCD is common, the neighborhood structures considered for each problem are differentiated with respect to the node pairing requirements. More specifically, for the PDP we make use of pair relocations, generalized pair exchanges and 2-opt local moves (Nanry and Barnes, 2000). On the other hand, the corresponding neighborhood structures employed for the VRPCD perform node relocations, node exchanges and 2-opt local moves (Zachariadis and Kiranoudis, 2010). In both cases, a lexicographic neighborhood evaluation scheme is adopted, while only feasible solutions are considered.

*Relocation neighborhood*

This local move relocates a transportation request from its current position. For the PDP, the pickup and delivery nodes of any request are removed from their current positions, and they are re-inserted into any combination of available positions. Similarly, for the VRPCD any supplier (or customer) node may be removed from the current pickup (or delivery) route, and can be reinserted into any feasible available position of another pickup (or delivery) route. Note that both intra- and inter-route relocation moves of nodes or pickup and delivery pairs are considered.

*Exchange neighborhood*

This local move exchanges two transportation requests on the same or different routes. For the PDP any two pairs of pickup and delivery requests that are served by the same or two different routes are selected and removed from their current positions. The pickup and delivery nodes of the first route can be inserted into any feasible position of the second route, while the pickup and delivery nodes of the second route can be inserted into any feasible position of the first route. For the VRPCD, we perform single node intra- and inter-route exchanges, and similarly we examine all feasible insertion positions.

*2-opt neighborhood*

The 2-opt neighborhood removes two edges from the same route or two different routes, and the remaining segments are reconnected so that two different routes are obtained.For the PDP, we need to ensure that both the pickup and delivery nodes of all transportation requests are contained in each route segment. For this reason, we employ 2-opt moves only between route pairs. For the VRPCD, we consider both intra- and inter-route 2-opt local moves.

Compared to the basic Tabu Search scheme, in the proposed implementation, the criteria for restricting and/or accepting a neighboring solution is based both on the local move attributes and the corresponding solution costs. This is controlled by monitoring the static move descriptors at each iteration described as follows.

*SMDs*

The local moves defined by each neighborhood structure are encoded using the Static Move Descriptor (SMD) mechanism (Zachariadis and Kiranoudis, 2010). The main principle of the latter is that local moves are mapped to static entities. Every SMD instance, apart from encoding a particular move, also contains the objective function change that this move would cause if applied to the candidate solution. Along the search process, when a structural modification is performed to the candidate solution, only the objective of the affected SMD instances has to be re-evaluated. Thus, redundant objective re-calculations are avoided.

Our Tabu Search framework operates according to the best admissible local move scheme. Specifically, all neighborhood structures of the current incumbent solution are exhaustively explored, and the highest quality feasible neighboring solution is selected at each iteration. To avoid an over-intensified search, the proposed framework is equipped with a diversification component based on the aspiration criteria of Tabu Search and the Attribute Hill Climber (Whittley and Smith, 2004). Each arc *(i, j) ∈ A* is associated with a threshold tag *ta*. Every time a local move *m* is applied to a solution *s* with objective value *z(s)*, the threshold tags of the eliminated arcs (*Em*) are set equal to *z(s)*, i.e. *ta* = *z(s)*, *∀ a ∈ Em*. Any local move *m* that forms a solution *s’* is considered admissible only if the cost tags of the generated arcs (*Cm*) exceed the objective value of the modified solution *s’*, i.e. *ta* > *z(s’)*, *∀ a ∈ Cm*. Note that the threshold tags are re-initialized to a large value after a number of iterations *w*, while a maximum number of iterations is imposed as termination condition.

**Computational study**

In this section, we present the results of our computational study. The computational experiments are organized in three subsections. In the first section, we solve well-known benchmark instances of the examined transportation problems so as to access the effectiveness of the proposed solution method with respect to the current state of the art approaches of the literature. In the next two sections, we generate and solve test cases with different characteristics for the direct shipping and the cross-docking strategies, namely Data Set I and Data Set II. The former considers uniquely defined supplier and customer nodes, whereas the latter considers groups of collocated supplier and customer nodes to reflect the scenario of a many-to-many relation between suppliers and customers (i.e. one supplier serving multiple customers, one customer being served by multiple suppliers). For both data sets, a comparative analysis on the relative performance of each distribution strategy as well as on the role of various key parameters is provided.

The proposed solution method was coded in Visual C#. All computational experiments are performed on a single core of a computer system equipped with an Intel(R) Xeon(R) CPU E5-2650 v2 (2.60 GHz) and 16 GB of RAM under Windows Server 2012. Unless otherwise stated, our algorithm is executed 10 times for each benchmark instance, and the best solution found is reported. The termination condition for each run was the completion of 100,000 Tabu Search iterations, and parameter *w* is set to 100. Detailed results are also available at xxx. All reported computational times are in seconds.

*Assessment of the proposed optimization method for the PDP and the VRPCD*

A series of computational experiments using well-known benchmark data sets for the PDP and the VRPCD has been conducted. The main effort was to evaluate the performance of the proposed local search metaheuristic algorithm. Due to the fact that most published solution approaches for the PDP and VRPCD have been applied to slightly differentiated problem variants, compared to the ones examined in this paper, we have appropriately modified our algorithm to deal with these differentiated configurations in order to ensure a secure assessment and a fair comparison.

The benchmark data set of Li and Lim (2001) with up to 100 customers is used for the PDP. This data set contains problem instances that are differentiated according to the geographic distribution of customers. In particular, the customers can be clustered (*LC*), randomly distributed (*LR*), or partially clustered and partially randomly distributed (*LRC*). Overall, two groups of problem instances are defined. The instances of the so-called *LC1*, *LR1* and *LRC1* classes have a short scheduling horizon, while the *LC2*, *LR2* and *LRC2* classes have a longer scheduling horizon.

Table 1 summarizes the results obtained for the PDP using the benchmark data set of Li and Lim (2001). The first part of the table reports the best known solutions (BKS) of the literature. The routing cost is denoted by *z,* while *k* refers to the number of vehicles. The corresponding references are also provided, i.e., LL stands for Li and Lim (2001), BH stands for Bent and Hentenryck (2006), SAM stands for the SINTEF heuristic, TS stands for the TetraSoft A/S heuristic. The second part of the table reports the results obtained by Ropke and Pisinger (2006), while the third part contains the best solutions found and the corresponding computational times over 10 simulation runs by our Tabu Search (TS) algorithm. Note that tsec refers to the average CPU time needed to perform one experiment, while %gap refers to the optimality gap between the best solutions produced by the TS and the BKS. Bold face is used to indicate the best overall solutions.

Overall, the proposed solution method seems to be very competitive. It matches the best known solution scores for most problem instances (49 out of 56), while for five out of seven instances, it produces solutions with less traveling distance in cases where the minimum number of vehicles is not obtained. It is worth to highlight that most of the existing approaches applied to this data set (including those of Li and Lim, 2001, Ropke and Pisinger, 2006, Bent and Hentenryck, 2006) are optimizing a hierarchical objective that primarily calls for the minimization of the number of vehicles, and secondarily the minimization of the traveled distance. Although our algorithm seeks solely to minimize the total routing cost, it is reasonable to expect that for some problems the minimization of the routing cost indirectly minimizes the number of vehicles. For the two cases where TS gives a worse solution compared to the BKS, the gap is less than 0.96%.

For the VRPCD, the benchmark data set of Wen et al. (2009) is used. Compared to the basic VRPCD previously described, these instances consider variable times for the unloading and loading operations at the cross-dock and time windows for both suppliers and customers. We have extended our algorithm to take into account these operational realties. In an effort to provide a fair basis for comparisons, we consider only the so-called scenario CS1 as presented by Tarantilis (2013). This scenario assumes that the same vehicle fleet is used for both pickup and delivery routes. Thus, synchronization constraints are imposed on vehicle routes to ensure that all products to be loaded into delivery vehicles are already available at the cross-dock.

**Table 1. Detailed results on Li and Lim (2001) Data Set for the PDP – 100 Customers**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **BKS** | | |  | **Ropke and Pisinger (2006)** | | |  | **TS** | | | |  |
| **Instance** |  | **k** | **z** | **Ref** |  | **k** | **z** | **tsec** |  | **k** | **z** | | **tsec** | **% gap** |
| LR101 |  | **19** | **1650.80** | LL |  | 19 | 1650.80 | 40 |  | **19** | **1650.80** | 30 | | 0.00 |
| LR102 |  | **17** | **1487.57** | LL |  | 17 | 1487.57 | 47 |  | **17** | **1487.57** | 71 | | 0.00 |
| LR103 |  | **13** | **1292.68** | LL |  | 13 | 1292.68 | 45 |  | **13** | **1292.68** | 92 | | 0.00 |
| LR104 |  | **9** | **1013.39** | LL |  | 9 | 1013.39 | 26 |  | **9** | **1013.39** | 286 | | 0.00 |
| LR105 |  | **14** | **1377.11** | LL |  | 14 | 1377.11 | 40 |  | **14** | **1377.11** | 44 | | 0.00 |
| LR106 |  | **12** | **1252.62** | LL |  | 12 | 1252.62 | 41 |  | **12** | **1252.62** | 61 | | 0.00 |
| LR107 |  | **10** | **1111.31** | LL |  | 10 | 1111.31 | 44 |  | **10** | **1111.31** | 153 | | 0.00 |
| LR108 |  | **9** | **968.97** | LL |  | 9 | 968.97 | 25 |  | **9** | **968.97** | 348 | | 0.00 |
| LR109 |  | **11** | **1208.96** | SAM |  | 11 | 1208.96 | 41 |  | **11** | **1208.96** | 78 | | 0.00 |
| LR110 |  | **10** | **1159.35** | LL |  | 10 | 1159.35 | 35 |  | 11 | 1165.83 | 415 | | 0.56 |
| LR111 |  | **10** | **1108.90** | LL |  | 10 | 1108.90 | 44 |  | **10** | **1108.90** | 144 | | 0.00 |
| LR112 |  | **9** | **1003.77** | LL |  | 9 | 1003.77 | 27 |  | **9** | **1003.77** | 747 | | 0.00 |
| LC101 |  | **10** | **828.94** | LL |  | 10 | 828.94 | 43 |  | **10** | **828.94** | | 190 | 0.00 |
| LC102 |  | **10** | **828.94** | LL |  | 10 | 828.94 | 44 |  | **10** | **828.94** | | 306 | 0.00 |
| LC103 |  | **9** | **1035.35** | BH |  | 9 | 1035.35 | 49 |  | 10 | 827.86 | | 86 | -20.04 |
| LC104 |  | **9** | **860.01** | SAM |  | 9 | 860.01 | 63 |  | 10 | 818.60 | | 772 | -4.82 |
| LC105 |  | **10** | **828.94** | LL |  | 10 | 828.94 | 41 |  | **10** | **828.94** | | 78 | 0.00 |
| LC106 |  | **10** | **828.94** | LL |  | 10 | 828.94 | 42 |  | **10** | **828.94** | | 75 | 0.00 |
| LC107 |  | **10** | **828.94** | LL |  | 10 | 828.94 | 43 |  | **10** | **828.94** | | 102 | 0.00 |
| LC108 |  | **10** | **826.44** | LL |  | 10 | 826.44 | 46 |  | **10** | **826.44** | | 170 | 0.00 |
| LC109 |  | **9** | **1000.60** | BH |  | 9 | 1000.60 | 35 |  | 10 | 827.82 | | 60 | -17.27 |
| LRC101 |  | **14** | **1708.80** | LL |  | 14 | 1708.80 | 38 |  | 15 | 1703.21 | | 45 | -0.33 |
| LRC102 |  | **12** | **1558.07** | SAM |  | 12 | 1558.07 | 41 |  | **12** | **1558.07** | | 170 | 0.00 |
| LRC103 |  | **11** | **1258.74** | LL |  | 11 | 1258.74 | 43 |  | **11** | **1258.74** | | 100 | 0.00 |
| LRC104 |  | **10** | **1128.40** | LL |  | 10 | 1128.40 | 52 |  | **10** | **1128.40** | | 235 | 0.00 |
| LRC105 |  | **13** | **1637.62** | LL |  | 13 | 1637.62 | 42 |  | **13** | **1637.62** | | 58 | 0.00 |
| LRC106 |  | **11** | **1424.73** | SAM |  | 11 | 1424.73 | 42 |  | **11** | **1424.73** | | 64 | 0.00 |
| LRC107 |  | **11** | **1230.14** | LL |  | 11 | 1230.14 | 43 |  | **11** | **1230.14** | | 101 | 0.00 |
| LRC108 |  | **10** | **1147.43** | SAM |  | 10 | 1147.43 | 25 |  | **10** | **1147.43** | | 161 | 0.00 |
| LR201 |  | **4** | **1253.23** | SAM |  | 4 | 1253.23 | 69 |  | **4** | **1253.23** | | 490 | 0.00 |
| LR202 |  | **3** | **1197.67** | LL |  | 3 | 1197.67 | 60 |  | **3** | **1197.67** | | 2293 | 0.00 |
| LR203 |  | **3** | **949.40** | LL |  | 3 | 949.40 | 98 |  | **3** | **949.40** | | 1783 | 0.00 |
| LR204 |  | **2** | **849.05** | LL |  | 2 | 849.05 | 181 |  | **2** | **849.05** | | 910 | 0.00 |
| LR205 |  | **3** | **1054.02** | LL |  | 3 | 1054.02 | 58 |  | **3** | **1054.02** | | 933 | 0.00 |
| LR206 |  | **3** | **931.63** | LL |  | 3 | 931.63 | 86 |  | **3** | **931.63** | | 1166 | 0.00 |
| LR207 |  | **2** | **903.06** | LL |  | 2 | 903.06 | 187 |  | **2** | **903.06** | | 1399 | 0.00 |
| LR208 |  | **2** | **734.85** | LL |  | 2 | 734.85 | 285 |  | **2** | **734.85** | | 6049 | 0.00 |
| LR209 |  | **3** | **930.59** | SAM |  | 3 | 930.59 | 73 |  | **3** | **930.59** | | 4118 | 0.00 |
| LR210 |  | **3** | **964.22** | LL |  | 3 | 964.22 | 77 |  | **3** | **964.22** | | 7544 | 0.00 |
| LR211 |  | **2** | **911.52** | SAM |  | 2 | 911.52 | 126 |  | 3 | 884.29 | | 5203 | -2.96 |
| LC201 |  | **3** | **591.56** | LL |  | 3 | 591.56 | 36 |  | **3** | **591.56** | | 280 | 0.00 |
| LC202 |  | **3** | **591.56** | LL |  | 3 | 591.56 | 59 |  | **3** | **591.56** | | 2161 | 0.00 |
| LC203 |  | **3** | **585.56** | LL |  | 3 | 591.17 | 81 |  | 3 | 591.17 | | 78 | 0.96 |
| LC204 |  | **3** | **590.60** | SAM |  | 3 | 590.60 | 141 |  | **3** | **590.60** | | 1151 | 0.00 |
| LC205 |  | **3** | **588.88** | LL |  | 3 | 588.88 | 48 |  | **3** | **588.88** | | 158 | 0.00 |
| LC206 |  | **3** | **588.49** | LL |  | 3 | 588.49 | 60 |  | **3** | **588.49** | | 584 | 0.00 |
| LC207 |  | **3** | **588.29** | LL |  | 3 | 588.29 | 62 |  | **3** | **588.29** | | 1067 | 0.00 |
| LC208 |  | **3** | **588.32** | LL |  | 3 | 588.32 | 69 |  | **3** | **588.32** | | 924 | 0.00 |
| LRC201 |  | **4** | **1406.94** | SAM |  | 4 | 1406.94 | 38 |  | **4** | **1406.94** | | 96 | 0.00 |
| LRC202 |  | **3** | **1374.27** | LL |  | 3 | 1374.79 | 82 |  | **3** | **1374.27** | | 1751 | 0.00 |
| LRC203 |  | **3** | **1089.07** | SAM |  | 3 | 1089.07 | 69 |  | 3 | **1089.07** | | 19495 | 0.00 |
| LRC204 |  | **3** | **818.66** | SAM |  | 3 | 818.66 | 173 |  | **3** | **818.66** | | 3802 | 0.00 |
| LRC205 |  | **4** | **1302.20** | LL |  | 4 | 1302.2 | 75 |  | **4** | **1302.20** | | 1368 | 0.00 |
| LRC206 |  | **3** | **1159.03** | SAM |  | 3 | 1159.03 | 48 |  | **3** | **1159.03** | | 2574 | 0.00 |
| LRC207 |  | **3** | **1062.05** | SAM |  | 3 | 1062.05 | 66 |  | **3** | **1062.05** | | 3072 | 0.00 |
| LRC208 |  | **3** | **852.76** | LL |  | 3 | 852.76 | 88 |  | **3** | **852.76** | | 2263 | 0.00 |
| **Total** |  | 402 | 58054 |  |  | 402 | 58060.07 | 3682 |  | 408 | 57611.53 | | 77954 |  |
| **Avg** |  |  | 1036.68 |  |  |  | 1036.787 | 65.75 |  |  | 1028.78 | | 1392.04 | -0.78 |

The computational results obtained for the benchmark data set of Wen et al. (2009) are summarized in Table 2. The first part provides the best known solutions (BKS) of the literature together with the corresponding reference (Ref). Next, the results reported by (W) Wen et.al (2009), (T) Tarantilis (2013), and (M) Morais et al. (2014) are provided. The last part of the table contains the best solutions found our Tabu Search (TS) algorithm after 10 simulation runs. Note that tsec refers to the computational time elapsed to obtain the best solution from a single simulation run. Morais et al. report only the best solution scores obtained for every instance. Finally, %gap refers to the optimality gap between the best solutions produced by the TS and the BKS. The best overall solutions are highlighted in bold.

**Table 2. Detailed results on Wen et al. (2009) Data Set for the VRPCD**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **BKS** | |  | **Wen et.al (2009)** | | **Tarantilis (2013)** | | **Morais et al. (2014)** |  | **TS** | | |
| **Instance** | **z** | **Ref** |  | **z** | **tsec** | **z** | **tsec** | **z** |  | **z** | **tsec** | **%**  **gap** |
| 50a | **6450.28** | T |  | 6471.90 | 3865 | **6450.28** | 28 | 6453.08 |  | **6450.28** | 68 | 0.00 |
| 50b | **7410.60** | W |  | **7410.60** | 3185 | 7428.54 | 15 | 7434.90 |  | 7428.54 | 8 | 0.24 |
| 50c | **7311.77** | T |  | 7330.60 | 3269 | **7311.77** | 13 | 7317.35 |  | **7311.77** | 39 | 0.00 |
| 50d | **7021.39** | T |  | 7050.30 | 3658 | **7021.39** | 117 | 7035.50 |  | 7028.22 | 22 | 0.10 |
| 50e | **7451.42** | T |  | 7516.80 | 3159 | **7451.42** | 20 | 7482.01 |  | **7451.42** | 32 | 0.00 |
| 100b | 14405.52 | T |  | 14526.10 | 9967 | 14405.52 | 987 | 12765.16 |  | **14398.17** | 444 | -0.05 |
| 100c | 13889.22 | T |  | 13967.80 | 10677 | 13889.22 | 904 | 14441.01 |  | **13869.80** | 227 | -0.14 |
| 100d | **13564.23** | T |  | 13763.30 | 11177 | **13564.23** | 866 | 13932.78 |  | 13603.03 | 169 | 0.29 |
| 100e | **14059.62** | T |  | 14212.70 | 10643 | **14059.62** | 922 | 13708.81 |  | 14063.29 | 417 | 0.03 |
| 150a | 19532.28 | M |  | 19537.30 | 24326 | 19638.04 | 1302 | 19532.28 |  | **19391.16** | 350 | -0.72 |
| 150b | 20823.40 | T |  | 20974.80 | 24461 | 20922.27 | 1172 | 20823.40 |  | **20764.50** | 203 | -0.28 |
| 150c | 19964.59 | T |  | 20126.50 | 23754 | 20019.50 | 1004 | 19964.59 |  | **19864.86** | 326 | -0.50 |
| 150d | 20509.97 | M |  | 20549.40 | 24468 | 20600.33 | 673 | 20509.97 |  | **20355.27** | 366 | -0.75 |
| 150e | 19716.87 | T |  | 19848.50 | 23400 | 19782.00 | 877 | 19716.87 |  | **19634.47** | 269 | -0.42 |
| 200a | 27112.48 | M |  | 27324.40 | 46586 | 27397.31 | 1891 | 27112.48 |  | **27073.57** | 751 | -0.14 |
| 200b | 27509.08 | T |  | 27637.70 | 43653 | 27582.87 | 1665 | 27509.08 |  | **27337.49** | 602 | -0.62 |
| 200c | 26320.39 | M |  | 26358.60 | 46389 | 26425.29 | 1904 | 26320.39 |  | **26181.73** | 1024 | -0.53 |
| 200d | 27686.75 | M |  | 27749.70 | 46615 | 27818.77 | 1789 | 27686.75 |  | **27439.50** | 1631 | -0.90 |
| 200e | 26443.29 | M |  | 26620.60 | 45649 | 26704.81 | 1102 | 26443.29 |  | **26305.30** | 2033 | -0.52 |
| **Total** | 319563.79 |  |  | 328978 | 408901.00 | 328473 | 17251 | 340312 |  | 222971.66 | 8981 |  |
| **Avg** | 31956.38 |  |  | 17314.61 | 21521.11 | 17288.06 | 907.95 | 17015.60 |  | 17151.662 | 472.68 | -0.26 |

Overall, our algorithm exhibits a stable and reliable performance. In particular, it outperforms existing approaches and 12 new best solutions are obtained. Cost reductions up to 0.9% are reached, while the worst gap from the BKS for the three instances where the proposed algorithm generated a worse solution is less than 0.29%. On average, the improvement on the BKS is 0.26%. On the other hand, the multiple simulation runs revealed no significant variations with respect to the solution cost. Finally, it is worth to highlight that the proposed method scales very well in terms of solution quality and computational times with respect to the problem size.

*Assessment of Direct Shipping and Cross-Docking strategies on Data Set I*

*Experimental setup and characteristics of Data Set I*

To analyze the relative effectiveness of the direct shipping and cross-docking strategies, we have introduced a set of benchmark instances (Data Set I) with unique pickup and delivery nodes (i.e. one-to-one relation of pickup and delivery locations). Data Set I consists of graphs where the total number of transportation requests n was taken from the set R = {12, 28, 36, 48}. Each request is associated with a supplier-customer pair leading to graphs with 24, 56, 72 and 96 nodes in total. These nodes are positioned in a [0,100]2 grid. The quantity qr of each transportation request randomly takes integer values within the range [1, 30]. Every node in the graph is assigned to a fixed service time of 2 time units. The distance matrix is generated by calculating the Euclidean distances between every node pair. We consider a homogeneous vehicle fleet and no bound is imposed on the number of vehicles. Finally, different distribution network classes are examined with respect to the geographical distribution of the nodes and the travel distances between pickup and delivery locations, as explained below.

*Geographic distribution of nodes*

We distinguish among three geographic distribution classes: i) random (*R*), ii) clustered (*C*), and iii) semi-clustered (*RC*). Regarding the R class, all nodes are uniformly distributed in the [0,100]2 grid. In C class, nodes are distributed in four node clusters of predefined dimensions located near the corners of the grid. For the RC class, approximately half of the nodes are distributed within the aforementioned four node clusters, while the rest of them are randomly distributed in the area between the clusters.

To quantify the distribution of node locations for the generated graphs, the Nearest Neighbor Index (*NNI*) is used (Clark and Evans, 1954). The *NNI* is defined as the ratio /, where the observed distance represents the average of the distance observed between each point and its nearest neighbor, and the random distance is the expected average distance that would occur, if the distribution was random. The former is given by , where . The latter can obtained as , where *A* is the area of the grid. *NNI* values less than 1, indicate some degree of clustering. In Figure 1 we provide the 56-node graphs for the three distribution classes.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Random distribution class  (*R*) | Semi-clustered distribution class (*RC*) | Clustered distribution class  (*C*) |
| **Figure 1.** Illustration of the geographic distribution classes (*R, RC* and *C*) of size *2n* = 56. | | |

*Proximity of supplier-customer pairs*

Three different classes of supplier-customer pair distances are examined. We consider large and small travel distances between supplier-customer pairs and the corresponding classes are denoted as *DS* and *CL*, respectively. A third class (*DCL*) is also considered with mixed large and small travel distances among supplier-customer pairs. We introduce the Pairing Index factor *PI* as a measure of the proximity of the supplier-customer pairs. *PI* is calculated as the ratio /, where *dpair* is the average distance of all supplier-customer pairs and is given by (note that *dpij* is the distance between a node *i* and its corresponding pair *j)*, and *davg* is the average distance of a node to all others. The latter is calculated as . Given the above *PI* definition, values of *PI* greater than 1, indicate large travel cost distances between pairs, while values of *PI* less than 1, indicate small travel cost distances between pairs. We consider all combinations of the graph size, the geographic distribution classes and the supplier–customer proximity classes, and we defined a total of 36 graphs. In Table 3, the corresponding *NNI* and *PI* values are provided.

**Table 3.** Nearest Neighbor Index and Paring Index of the generated graphs in Data Set I

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Instance Size |  | NNI | | |  | PI | | |
| 2·*n* |  | R | RC | C |  | DS | DCL | CL |
| 24 |  | 1.09 | 1.15 | 0.95 |  | 1.27 | 1.04 | 0.66 |
| 56 |  | 1.04 | 1.02 | 0.82 |  | 1.32 | 1.02 | 0.42 |
| 72 |  | 1.04 | 0.97 | 0.78 |  | 1.38 | 0.93 | 0.36 |
| 96 |  | 1.04 | 1.00 | 0.79 |  | 1.37 | 0.94 | 0.31 |

Furthermore, the impact of four key problem parameters is examined. In particular, focus is given on the vehicle capacity *Q*, the maximum route duration *T*, the depot location *D* and the loading/unloading time at the cross-dock *s* (this parameter is considered only for the VRPCD). Let *P =* {*Q, T, D, s*} be the set of the examined parameters. For each parameter *p ∈ P*, let *L*= {1, 2, 3} to denote the parameter level. Thus, in total 12 parameter level values are used and reported in Table 4. Note that the depot location parameter consists of a (x,y) coordinate tuple. The y-coordinate is varied, while the x-coordinate is fixed at 50. An illustration of the depot locations levels (*D1*, *D2* and *D3*) for an instance of the *RC* class with 56 nodes is depicted in Figure 2.

**Table 4.** Parameter level values used for Data Set I

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter level | Vehicle capacity  (*Q*) | Max route duration  (*T*) | Depot location  (*D*) | Loading/Unloading time  (*s*) |
| 1 | 70 | 250 | [50,50] | 5 |
| 2 | 100 | 265 | [50,60] | 7 |
| 3 | 120 | 280 | [50,70] | 9 |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| D1 | D2 | D3 |
| **Figure 2.** Illustration of the depot location levels (*D1, D2* and *D3*) for a problem instance of the *RC* class (2⋅*n*=56). The depot is illustrated as a square. | | |

For each generated graph all combinations of parameter levels are examined, and a full factorial experiment is performed. For each of the generated graphs, we examine 33 combinations to account for all parameters and levels for the PDP setting, and 34 combinations for the VRPCD setting, leading to a total of 972 (36·33) problem instances for the PDP and 2916 (36·34) problem instances for the VRPCD.

We define *λ*as a measure of the relative performance of the two strategies examined. Let *λ* be the ratio of the PDP objective value over the VRPCD objective value. In the remainder of the paper, we will consider *λ* as the average ratio of the PDP objective value () over the VRPCD objective value (), over specific subsets of PDP (and VRPCD) instances. Apparently, values greater than 1, imply that the direct shipping strategy (as depicted by the PDP) performs worse compared to the cross-docking strategy (as depicted by the VRPCD).

*Computational results for Data Set I*

Table 5 presents the relative performance *λ* between the direct shipping and the cross-docking strategies for the instances of Data Set I. In particular, the relative performance refers to the average results obtained over all instances belonging to different subsets with respect to the geographic distribution classes (*R, C* and *RC*) and the supplier-customer travel distances (*DS, DCL* and *CL*), respectively. Table 5 is divided into four sections (one for each parameter). Each of these sections contains three rows (one for each parameter level). Each row provides the relative performance *λ* of the two strategies by fixing each parameter *p ∈ P* to a specific level *l ∈ L* and varying all other parameters. For example, 0.68 in the first row of the first column corresponds to the average relative performance / for all 108 problem instances of R class with *Q1* level. (Note that the cross-dock service time *s* is not considered under the PDP model, thus the values of for the *s1,* *s2* and *s3* levels are always identical). Analytic solution scores for all Data Set I instances are provided in Tables 1, 2, 3 and 4 of the Appendix A.

**Table 5.** Relative performance *λ* between the direct shipping and the cross-docking strategies for Data Set I.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Level/Class** |  | **R** | **RC** | **C** |  | **DS** | **DC** | **CL** |
| **Q1** |  | 0.68 | 0.79 | 0.76 |  | 0.90 | 0.87 | 0.46 |
| **Q2** |  | 0.74 | 0.84 | 0.83 |  | 0.95 | 0.92 | 0.54 |
| **Q3** |  | 0.77 | 0.86 | 0.84 |  | 0.96 | 0.94 | 0.57 |
| **T1** |  | 0.73 | 0.83 | 0.82 |  | 0.95 | 0.91 | 0.51 |
| **T2** |  | 0.73 | 0.83 | 0.81 |  | 0.94 | 0.91 | 0.52 |
| **T3** |  | 0.73 | 0.83 | 0.81 |  | 0.93 | 0.90 | 0.53 |
| **D1** |  | 0.77 | 0.87 | 0.83 |  | 0.97 | 0.95 | 0.56 |
| **D2** |  | 0.75 | 0.84 | 0.82 |  | 0.95 | 0.92 | 0.54 |
| **D3** |  | 0.67 | 0.77 | 0.78 |  | 0.90 | 0.86 | 0.48 |
| **s1** |  | 0.73 | 0.83 | 0.81 |  | 0.94 | 0.91 | 0.53 |
| **s2** |  | 0.73 | 0.83 | 0.81 |  | 0.94 | 0.91 | 0.52 |
| **s3** |  | 0.72 | 0.83 | 0.81 |  | 0.93 | 0.91 | 0.52 |

In the following, we take a close look at the obtained results, in pursuit of useful conclusions on the impact of the tested parameters on the relative suitability of the VRPCD and PDP models with respect to the different geographic distribution classes (*R, RC, C*) and the supplier-customer pair distances classes (*DS, DCL, CL*).

*Vehicle capacity (Q)*

The relative performance of the two strategies is expected to be sensitive to the vehicle capacity. Figure 3 graphically depicts the results provided in the first sections of Table 5. It is clear that as the vehicle capacity increases, both problems are relaxed, and thus the transportation costs decrease. This behavior is observed for all geographic distribution classes, as well as all supplier-customer proximity classes. However, it is also apparent that the increase of the vehicle capacity has a stronger positive effect on the cross-docking strategy. This is attributed to the fact that vehicles are allowed to serve more suppliers/customers in the same route before returning to the cross-dock. From a different perspective, the PDP is not so sensitive to the capacity constraints; vehicles are unloaded through the routes, thus even if their capacity is limited, they are not forced to return to the depot. To empirically support this claim, we compare the decrease of the aggregate routing costs for both strategies when moving from the *Q1* to the *Q3* capacity level for the *D1* and *D3* classes of depot locations, respectively. For the VRPCD, moving from *Q1* to *Q3* yields a 26.35% (from 1014.10 to 746.93) and 23.37% (from 1159.79 to 888.78) improvement in the objective function values for *D1* and *D3*, respectively. The corresponding improvement percentages for the PDP are 18.51% (from 753.53 to 614.01) and 18.41% (from 770.24 to 628.42). These findings are in line with what was expected; heavy duty large capacity vehicles (i.e., *Q3* level) tend to promote the consolidation strategy, especially when the depot is centrally located (i.e., *D1* level).

Another observation from Figure 3 is that the increase of the vehicle capacity has a stronger positive effect for the cross-docking strategy for the RC, *C* and *DS* classes. In fact, for the direct shipping strategy, moving from *Q1* to *Q3* for the different geographic distribution classes, we observe 10.21% improvement on transportation costs (from 677.20 to 608.03) for the *R* class, while the corresponding improvement on transportation costs for the *C* class is 22.43 % (from 819.50 to 635.69). For the cross-docking strategy, the improvement on transportation costs is 32.52% (from 1013.25 to 793.02) and 39.82% (from 1033.84 to 746.73) for the *R* and *C* class, respectively. Regarding the different proximity to supplier-customer pair classes, moving from *Q1* to *Q3* for the direct shipping strategy we observe 24.05% improvement on transportation costs (from 912.49 to 693.06) for the DS class, while the corresponding improvement on transportation costs for the *CL* class is only 1.48 % (from 484.62 to 477.44). For the VRPCD, transportations costs for the *DS* class are reduced from an average value of 1016.24 to 727.72, yielding an improvement of 28.40%, whereas the corresponding improvement for the *CL* class is 22.59% (from 1121.67 to 868.25).

|  |  |
| --- | --- |
|  |  |

**Figure 3.** Impact of the vehicle capacity on the relative performance for Data Set I

*Vehicle route duration (T)*

The impact of loosening the route duration constraint is shown graphically in Figure 4. We observe that the average *λ* values seem to remain stable for the three levels of parameter *T*. A closer analysis of the solution costs for specific problem instances reveals that this stability comes from the interplay of the capacity and route duration constraints. There are cases where the increase of *T* causes *λ* to increase, meaning that the cross-docking yields more significant cost savings than the direct-shipping strategy. In principle, this behavior is observed when the initial *T* value is very low and the route shape strongly depends on *T*. However, there are other cases where the increase of *T* reduces the relative effectiveness of the cross-docking strategy. These cases involve high base values of *T*, which cause the shape of the VRPCD routes to be mainly defined by the capacity constraints, so that further *T* increase has no impact on the VRPCD solution cost. On the contrary, since the direct shipping strategy is not so sensitive to the capacity constraints (vehicles are unloaded along their trips), it exploits cost savings opportunities provided by the increase of the maximum route duration.

|  |  |
| --- | --- |
|  |  |

**Figure 4.** Impact of the route duration on the relative performance for Data Set I

*Location of the* *depot (D)*

Figure 5 shows the relative performance of the distribution strategies for the different depot levels. Overall, it appears that locating the depot at the center of the grid (*D1*) is more beneficial for both strategies, than the case where the depot is located further away from the center of the grid. An unfavorable position of the depot away from the center of the supplier-customer area (see *D2 and D3*), can substantially increase routing costs for both the PDP and VRPCD; however, the performance of the cross-docking strategy is more affected compared to the direct shipping strategy. In fact, moving from *D1* to *D3* we observe 2.31% increase on transportation costs for the PDP (i.e., from 672.78 to 688.32), while the corresponding increase on transportation costs for the VRPCD is 15.62% (i.e., from 841.21 to 972.57). This behavior stems from the fact that under the cross-docking strategy, we have to travel depot-adjacent arcs for both levels of transportation (pickup and delivery vehicle trips). In particular, the *λ*ratio is reduced by 10% when moving from the *D1* to *D3* configuration (*D1*: *λ* = 0.82*, D3*: *λ*= 0.74).

|  |  |
| --- | --- |
|  |  |

**Figure 5.** Impact of the depot location on the relative performance for Data Set I

*Unloading/loading times at the cross-dock (s)*

The role of the service time at the CD is straightforward to understand and is shown in Figure 6. The higher the time necessary for handling the consolidation operations in the cross-dock, the higher the VRPCD solution cost.

|  |  |
| --- | --- |
|  |  |

**Figure 6.** Impact of the unloading/loading time at the cross-dock on the relative performance for Data Set I

We would also like to comment on the relative effectiveness of the cross-docking and direct shipping strategies for the different types of network characteristics. Regarding the distribution of the service locations, we observe that in principle when the graph is clustered the cross-dock seems to perform better. More specifically, the average PDP solution scores divided by the average VRPCD solution scores over the *R* class instances is 0.73, whereas for the *C* class of test problems the aforementioned ratio goes up to 0.81. Regarding the proximity of the pickup and delivery pairs, we have a much clearer picture. For the closed configuration of the pickup and delivery pairs (class *CL*), the average solution score for the PDP over the VRPCD one is 0.52. Moving to the more distant configurations *DCL* and *DS*, the corresponding ratios are significantly increased to 0.91 and 0.94, respectively.

Lastly, another observation is that direct shipping incurs lower objective function values than the cross-docking strategy for the majority of test cases. However, there are several instances for which this is not the case. In the following, we report some indicative cases where the VRPCD appears to be more cost effective, in pursuit of useful conclusions regarding the relative suitability of the two strategies. The VRPCD is a winner for instance subsets belonging to the *DS* and *DCL* class of pickup and delivery pair proximity, i.e., cases where the pick-up and delivery nodes of a pair are far from each other. These instance subsets mainly involve a centrally positioned depot (level *D1*). For example, the instances contained within the Q1T1s1D1 and Q3T3s3D1 rows of Tables 1, 2, 3 and 4 of Appendix A (i.e, the instances with all possible combinations of levels for parameters *Q*, *T* and *s*), which involve the depot placed in the centre of the grid, is associated with better VRPCD solution scores, for all *RC, C, DS,* and *DCL* classes. This is also observed for some high capacity instances (capacity level *Q3*), where the depot is not placed in the centre of the grid (levels *D2, D3*). More specifically, for many problem instances of the instances Q2T1s1D2 to Q3T3s3D2 and Q2T1s1D3 to Q3T3s3D3, and for all *RC*, *C*, *DS*, and *DC*L classes, the cross-docking strategy exhibits a better performance than the direct shipping strategy.

*Assessment of Direct Shipping and Cross-Docking strategies on Data Set II*

*Experimental setup and characteristics of Data Set II*

The results of the previous section involved test problems (Data Set I) with unique pick-up and delivery nodes (i.e. suppliers are linked to customers with a one-to-one relation). In this section, we present the results of Data Set II that contains problem instances with a many-to-many mapping between suppliers and customers and thus the various requests may often involve co-located nodes.

For the construction of problem instances for the Data Set II we have used the following rationale. Firstly, we consider two instance sets *P’* and *D’* that correspond to the particular supplier and customer locations, respectively. To generate the set of requests *R,* and thus, define the *P* and *D* node sets, we construct a demand matrix *qij* for *i ∈ P’* and *j* *∈ D’*. The *qij* value corresponds to the quantity of products that have to be moved from a supplier *i ∈ P’* to a customer *j ∈ D’*. Obviously, if all *qij* > 0 (*i ∈ P’*, *j ∈ D’*), the cardinality of the transportation request set is |*R*| = |*P*’|·|*D*’|. This means that every non-zero entry of the demand matrix corresponds to a flow of products between a supplier and a customer. Therefore, the set of transportation requests *R* are determined as follows: for every non-zero *qij* entry of the demand matrix a request *r* is generated.

For the Data Set II problem instances, we have also considered three density classes *DM1, DM2,* and *DM3* for the demand matrix. Specifically, each *qij* value has a probability 25%, 50%, and 100% of being non-zero, for classes *DM1, DM2*, and *DM3* respectively, where every non-zero entry takes a random integer value within [5, 30]. The density classes are considered in order to assess the examined strategies under scenarios that involve supplier nodes (e.g. production plants, warehouses, etc.) serving multiple customers (e.g. retailers) and are frequently met in practice. The pickup node for this request is

a copy of the original pickup node *i* *∈ P’*, the delivery node of *r* is a copy of the pickup node *j* *∈ D*’, and the demand is *qr* = *qij*.

Data Set II problem instances are constructed considering |*P’*| values taken from {5, 6, 7} and for simplicity we assume that *P’* = *D’*. The original *P* and *D* nodes are placed within the [0, 100] grid. Two geographic distribution classes are examined, namely random (*R*) and clustered (*C*). For the *R* class, all nodes are randomly located in the [0,100]2 grid, whereas for the *C* class, the *P’* nodes and *D’* nodes are grouped into two separate clusters positioned in the upper and lower parts of the [0,100]2 grid. In order to broaden the scope of our analysis, we consider three different classes of assignment of supplier-customer pairs (*AA, AB,* and *AC*). In total, 18 graphs of *P’* and *R’* nodes are constructed (3 sizes x 2 distribution classes x 3 supplier-customer pair assignment). As mentioned earlier, three demand matrix classes are examined, and for each of this class three demand matrices are introduced, defining in total 54 request sets (6 *P’* and *R’* graphs x 3 demand matrix classes x 3 matrices per demand matrix class). An illustration of the geographic distribution classes (*R, C*) for a problem instance with |*P’*| = 7 of the density class *DM3*, is shown in Figure 7.

|  |  |
| --- | --- |
|  |  |
| Random distribution class  (*R*) | Clustered distribution class  (*C*) |

**Figure 7**. Illustration of the geographic distribution classes (R, C) for a problem instance with |P’| = 7 of the density class DM3.

To finalize the Data Set II problem instances, we have to set the four model parameters *P* = {*Q, T, D, s*} discussed in 5.1. To analyze their role, for each of these parameters, we test three different levels. Compared to the Data Set I, the values for the different levels of parameters *Q* and *T* are not the same for all problem instances, but instead they are selected based on the characteristics of each specific problem instance. An illustration of the depot location levels (*D1, D2* and *D3*) for a problem instance with |*P’*| = 7 of the *R* and *DM3* class, is shown in Figure 8. A total of 4374 VRPCD instances (54 request sets x 34 parameter combinations) and a total of 1458 PDP instances are defined in Data Set II. As discussed earlier, the distance matrix is obtained as the Euclidean distance between node pairs. Finally, each node is associated with a fixed service time of two time units.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| D1 | D2 | D3 |
| **Figure 8.** Illustration of the depot location levels (*D1, D2* and *D3*) for a problem instance for a problem instance with |P’| = 7 of the *R* and *DM3* classes. The depot is illustrated as a square. | | |

**Computational results for Data Set II**

Table 6 reports the computational results obtained for all Data Set II problem instances. In particular, it provides the average results over instances belonging to the node distribution classes (*R* and *C)* and demand density classes (*DM1, DM2*, and *DM3*). As discussed earlier in Table 5, each row corresponds to the average relative performance, which is calculated as the PDP objective function values divided by the VRPCD objective function values, for specific subsets of problem instances obtained by fixing a given parameter to a specific level and varying all other parameters.

**Table 6.** Relative performance *λ* between the direct shipping and the cross-docking strategies for Data Set II

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Level/Class** |  | **R** | **C** |  | **DS** | **DC** | **CL** |
| **Q1** |  | 0.60 | 0.78 |  | 0.64 | 0.69 | 0.73 |
| **Q2** |  | 0.65 | 0.80 |  | 0.68 | 0.71 | 0.78 |
| **Q3** |  | 0.66 | 0.80 |  | 0.70 | 0.72 | 0.78 |
| **T1** |  | 0.64 | 0.79 |  | 0.68 | 0.71 | 0.75 |
| **T2** |  | 0.65 | 0.80 |  | 0.68 | 0.72 | 0.78 |
| **T3** |  | 0.62 | 0.79 |  | 0.66 | 0.70 | 0.76 |
| **D1** |  | 0.88 | 1.00 |  | 0.86 | 0.93 | 1.02 |
| **D2** |  | 0.61 | 0.81 |  | 0.67 | 0.71 | 0.74 |
| **D3** |  | 0.42 | 0.58 |  | 0.49 | 0.50 | 0.52 |
| **s1** |  | 0.65 | 0.81 |  | 0.67 | 0.72 | 0.79 |
| **s2** |  | 0.63 | 0.80 |  | 0.67 | 0.71 | 0.76 |
| **s3** |  | 0.62 | 0.78 |  | 0.67 | 0.70 | 0.74 |

The obtained results confirm the remarks drawn and discussed in 5.2.2 concerning the impact of the parameters examined on the relative suitability of the cross-docking and direct shipping strategies. This is illustrated in Figures 9, 10, 11 and 12. Figure 9 shows that the higher the capacity, the more significant is the improvement of the VRPCD solution scores. Specifically, for all problem instances (both distribution classes and all three demand density classes) the *λ* value increments from 0.69 to 0.73, when moving from *Q1* to *Q3*. Figure 10 illustrates a non-monotonic impact of the route duration limit on the relative performance ratio *λ*. This behavior is attributed on the interplay of the capacity and route duration constraints analyzed in 5.2.2. In the first phase (moving from *T1* to *T2*), the *λ* ratio increases, implying that VRPCD is capable of gaining larger cost savings. In the second phase, this picture is reversed; additional increase of the *T* value (from *T2* to *T3*) has an almost insignificant impact on the VRPCD solution objective, because the capacity constraints are binding and define the shape of the VRPCD routes. On the contrary, the direct shipping strategy is capable of taking full advantage of the *T* increase, because the carrying load fluctuates along the PDP routes, so that the impact of the capacity constraints on the PDP solution shape is weaker. The depot location impact is shown in Figure 11. We observe that *λ* strongly depends on the depot positioning. In fact, locating the depot in central positions of the examined region favors the cross-docking strategy, whereas installing the depot in more remote positions has a positive effect on the direct shipping strategy. Note, for the *C* and *DM3* classes and for the *D1* level, the performance of the cross-docking strategy is better than the direct-shipping. Finally, as graphically shown in Figure 12, the more time required for the consolidation operations, the lower the competitiveness of the cross-docking strategy.

|  |  |
| --- | --- |
|  |  |

**Figure 9.** Impact of the vehicle capacity on the relative performance for Data Set II

|  |  |
| --- | --- |
|  |  |

**Figure 10**. Impact of the route duration on the relative performance for Data Set II

|  |  |
| --- | --- |
|  |  |

**Figure 11.** Impact of the depot location on the relative performance for Data Set II

|  |  |
| --- | --- |
|  |  |

**Figure 12.** Impact of the unloading/loading time at the cross-dock on the relative performance for Data Set II

Ιn the following, we will concentrate on the role of the spatial characteristics of the problem instances of Data Set II, namely the degree of connectivity between supplier and customer locations and the distribution of nodes.

The connectivity between supplier and customer locations is determined by the demand density class introduced in the previous section. The results indicate a strong positive correlation of the density class and *λ* values as shown in Figure 13. The higher the demand density class, the more competitive is the cross-docking strategy. In particular, the *λ* for all problem instances in Data Set II is augmented by a significant 13.25% when moving from *DM1* to *DM3*.

**Figure 13.** Impact of the DM on the on the relative performance for Data Set II

Regarding the distribution of the nodes, we observe that the cross-docking strategy seems to perform better for the C class, where supplier and customer nodes are located into separate geographic regions. The *λ* is 0.65 for the *R* class and increases up to 0.81 for the *C* class. In specific, for the C class and when the depot is located between the supplier and customer nodes (*D1* configuration), the cross-docking strategy is better.

Finally, we can see that the joint impact of the supplier and customer node connectivity, the geographic distribution of customers, and the depot location on the relative suitability of the model is significant. For cases where supplier and customer nodes are densely connected (*DM3* class), suppliers and customers are positioned in separated regions (*C* class) and the cross-dock is centrally located (*D1 level*), cross-docking appears to be more suitable. For problem instances of this subset, *λ* is equal to 1.02 (see Figure 12), indicating that with the cross-docking strategy generates on average solutions of lower costs. The cross-docking suitability is strengthened when the capacity of the vehicles is large. Indeed, for problem instances with capacity level *Q1* (low capacities) *λ* is 1.07, whereas for *Q3* capacity level *λ* increases to 1.12.

**Conclusions**

This paper is motivated by the importance of evaluating the relative performance of the direct shipping and cross-docking strategies. For this purpose, a Tabu Search metaheuristic algorithm has been developed, and used as a common optimization framework for addressing both the PDP and the VRPCD. The proposed local search algorithm employs edge-exchange neighborhood structures, and uses a novel mechanism based on static move descriptors to guide the search. Benchmark data sets of the literature were used initially to evaluate the effectiveness and efficiency of the proposed optimization method, while various computational experiments on new problem instances were performed to analyze the impact of several spatial and temporal network characteristics.

In particular, experiments on well-studied PDP and VRPCD benchmark instances with time window constraints indicate that the proposed Tabu Search metaheuristic algorithm is very competitive compared to the current state-of-the-art. More specifically, for the VRPCD several new best solutions have been identified, whereas for the PDP a very stable performance is observed, and for most problem instances the best known solutions are matched. Overall, the quality of the results obtained is a strong indication that the proposed optimization method can be adopted for practical real life applications.

Various computational experiments have been also performed to analyze the relative effectiveness of the direct shipping and cross-docking strategies. Several cases where each strategy appears to be more suitable and more cost effective are discussed. Furthermore, the results obtained indicated that the effectiveness of each strategy heavily depends on various parameters and most notably the geographic distribution of the customers, the supplier-customer pair proximity, and the node connectivity (or demand density).

Overall, the direct shipping strategy incurs lower objective function values than the cross-docking strategy for the majority of test cases. More specifically, the direct shipping yields better results in cases where suppliers and customers are positioned into the same geographic area and suppliers (and their associated customers) are in close proximity to each other. On the other hand, for large supplier-customer pair proximity and clustered distributions, the cross-docking strategy performs better. For problems with co-located nodes the cross-docking outperforms the direct shipping strategy when the suppliers and customers are densely connected. Finally, the cross-docking strategy seems more suitable when the transshipment facility is a centrally located.

From a managerial perspective we can draw the following conclusion. In practical cases where warehouses and industrial plants located in isolated industrial zones of the cities that serve multiple delivery locations placed in the central urban regions, the installation of cross-dock facilities in intermediate points can bring substantial routing cost savings compared to a direct shipping strategy. In setting, the selection of heavy duty large capacity vehicles can further increase the cost savings for the cross-docking strategy.

**Acknowledgements**

This work was supported by the ΙΚΥ fellowships of excellence for postgraduate studies in Greece – Siemens program.

**References**

Bent, R. and Van Hentenryck, P. (2006), “A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows”, *Computers and Operations Research*, Vol. 33, pp 875–893.

Cordeau, J.F., Laporte, G. and Ropke, S. (2008), “Recent models and algorithms for one-to-one pickup and delivery problems”. In: Golden, B., Raghavan, R., and Wasil, E., (Ed.), *The Vehicle Routing Problem: Latest Advances and New Challenges*, Springer, Heidelberg, pp. 327–357.

Cortes, C.E., Matamala, M. and Contardo, C. (2010), “The pickup and delivery problem with transfers: Formulation and a branch-and-cut solution method”, *European Journal of Operational Research*, Vol. 200 No. 3, pp. 711-724.

Dumas, Y., Desrosiers, J. and Soumis, F. (1991), “The Pickup and Delivery Problem with Time Windows” *European Journal of Operational Research*, Vol. 54, pp. 7–22.

Lee, Y.H., Jung, J.W. and Lee, K.M. (2006), ”Vehicle routing scheduling for cross-docking in the supply chain”, *Computers and Industrial Engineering*, Vol. 51 No. 2, pp. 247–256.

Li, H. and Lim., A., (2001), “A metaheuristic for the pickup and delivery problem with time windows”, in *Proceedings of the International Conference on Tools with Artificial Intelligence*, pp. 160–167, IEEE.

Masson, R., Lehuede, F. and Peton, O. (2013), “An adaptive large neighborhood search for the pickup and delivery problem with transfers”, *Transportation Science*, Vol. 47 No. 3, pp. 344–355.

Miao, Z., Yang, F., Fu, K. and Xu, D. (2010), “[Transshipment service through cross-docks with both soft and hard time windows](http://www.mendeley.com/research/transshipment-service-through-crossdocks-both-soft-hard-time-windows/), *Annals of Operations Research*, Vol. 192 No. 1, pp. 21-47.

Mitrovic-Minic, S. and Laporte, G. (2006), “The pickup and delivery problem with time windows and transshipment, *Information Systems and Operational Research*, Vol. 44 No. 3, pp. 217–228.

Morais, V.W.C., Mateus, G.R. and Noronha, T.F. (2014), “Iterated local search heuristics for the Vehicle Routing Problem with cross-docking”, *Expert Systems with Applications*, Vol. 41 No. 16, pp. 7495-7506.

Mues, C., and Pickl, S. (2005), “Transshipment and Time Windows in Vehicle Routing”, in proceedings of *The 8th International Symposium on Parallel Architectures*, Algorithms and Networks IEEE, Piscataway, 2005, pp. 113–119.

Nagata, Y. and Kobayashi, S. (2010), “Guided Ejection Search for the Pickup and Delivery Problem with Time Windows”, in: Cowling, P., Merz, P. (eds.), [*Evolutionary Computation in Combinatorial Optimization*](http://link.springer.com/book/10.1007/978-3-642-12139-5), [Lecture Notes in Computer Science](http://link.springer.com/bookseries/558), Vol. 6022, Springer, Heidelberg,  pp 202-213.

Nanry, W.P. and Barnes, J.W. (2000), “Solving the pickup and delivery problem with time windows using reactive tabu search”, *Transportation Research Part B*, Vol. 34 No. 107, pp. 107–121.

Parragh, S., Doerner, K. and Hartl, R. (2008), “A survey on pickup and delivery problems. part II: Transportation between pickup and delivery locations”, *Journal fur Betriebswirtschaft*, Vol. 58 No. 2, pp. 81–117.

Petersen, H.L. and Ropke, S. (2011), “The Pickup and Delivery Problem with cross-docking opportunity”, in Proceedings of *International Conference on Computational Logistics*, Hamburg, Germany, pp. 101-113.

Ropke, S. and Pisinger, D. (2006), “An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows”, *Transportation Science*, Vol. 40 No. 4, pp. 455–472.

Santos, F.A., Mateus, G.R., da Cunha, A.S. (2013), “The pickup and Delivery problem with cross-docking”, *Computers and Operations Research*, Vol. 40, pp. 1085–1093.

Savelsbergh, M.W.P. and Sol., M. (1995), “The general pickup and delivery problem”, *Transportation Science*, Vol. 29 No. 1, pp. 17-29.

Tarantilis., C.D. (2013), “Adaptive multi-restart Tabu search algorithm for the vehicle routing problem with cross-docking”, *Optimization Letters*, Vol. 7 No. 7, pp. 1583-1596.

Wen, M., Larsen, J., Clausen, J., Cordeau, J.F. and Laporte, G. (2009), “Vehicle routing with cross-docking”, *Journal of the Operational Research Society*, Vol. 60 No. 12, pp. 1708-1718.

Whittley, I.M. and Smith, G.D. (2004), “The attribute based hill climber”, *Journal of Mathematical Modelling and Algorithms*, Vol. 3 No. 2, pp. 167–178.

Zachariadis., E.E. and Kiranoudis, C.T. (2010), “A strategy for reducing the computational complexity of local search-based methods for the vehicle routing problem”, *Computers and Operations Research*, Vol. 37 No. 12, pp. 2089–2105.

**Corresponding author**

Amalia Nikolopoulou can be contacted at: anikolop@aueb.gr

**Appendix A**

Appendix A containing Tables A.1 – A.13 is provided in the Online Supplementary Material of the present paper.