Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 19 (12): 13710–13731.
DOI: 10.3934/mbe.2022639
Received: 08 July 2022
Revised: 28 August 2022
Accepted: 04 September 2022
Published: 19 September 2022

Research article

Mathematical estimation for maximum flow of goods within a cross-dock to reduce inventory

Taniya Mukherjee[1,2], Isha Sangal[1], Biswajit Sarkar[3,4,*] and Tamer M. Alkadash[2]

[1] Department of Mathematics &Statistics, Banasthali Vidyapith, Banasthali, Rajasthan-304022, India
[2] Department of Administrative Sciences, Gulf University, Sanad, Kingdom of Bahrain
[3] Department of Industrial Engineering, Yonsei University, 50 Yonsei-ro, Sinchon-dong, Seodaemun-gu, Seoul, 03722, South Korea.
[4] Center for Transdisciplinary Research (CFTR), Saveetha Dental College, Saveetha Institute of Medical and Technical Sciences, Saveetha University, 162, Poonamallee High Road, Velappanchavadi, Chennai, Tamil Nadu 600077, India

* **Correspondence:** Email: bsbiswajitsarkar@gmail.com; Tel: +82-10-7498-1981.

Abstract: Supply chain management has recently renovated its strategy by implementing a cross-docking scheme. Cross-docking is a calculated logistics strategy where freight emptied from inbound vehicles is handled straightforwardly onto outbound vehicles, eliminating the intermediate storage process. The cross-docking approach thrives on the minimum storage time of goods in the inventory. Most of the cross-docks avail temporary storage docks where items can be stored for up to 24 hours before being packed up for transportation. The storage capacity of the cross-dock varies depending on the nature of ownership. In the rented cross-docks center, the temporary storage docks are considered of infinite capacity. This study believes that the temporary storage facilities owned by the cross-dock center are of finite capacity, which subsequently affects the waiting time of the goods. The flow rate of goods within the cross-docks is expected to be maximum to avoid long waiting for goods in the queue. This paper uses a series of max-flow algorithms, namely Ford Fulkerson, Edmond Karp, and Dinic's, to optimize the flow of goods between the inbound port and the outbound dock and present a logical explanation to reduce the waiting time of the trucks. A numerical example is analyzed to prove the efficacity of the algorithm in finding maximum flow. The result demonstrates that Dinic's algorithm performs better than the Ford Fulkerson and Edmond Karp algorithm at addressing the problem of maximum flow at the cross-dock. The algorithm effectively provided the best result regarding iteration and time complexity. In addition, it also suggested the bottleneck paths of the network in determining the maximum flow.

Keywords: cross-dock; supply chain management; transportation; maximum flow; algorithm

1. Introduction

In the recent era, organizations are looking for innovations to survive in the business [1]. Following the trend, inventory management has also refurbished its look with new innovative strategies [2] for storing, ordering, forecasting inventory apart from maintaining product quality [3] and optimizing total inventory cost [4]. Even developing green products considering environmental benefits is a part of this innovatory strategy [5]. It is interesting to know that to synchronize supply with product demand [6], supply chain models [7] are dependent heavily on operation management [8] and logistics [9] for their success.Thus, the hunt for new logistic management plans to support successful supply chain models is always in demand. Cross-docking is one such widely popular logistic management mechanism Cross-docking includes integrating different middle forms like accepting the inbound items, deconsolidation, sorting, union, and stacking the solidified items in outbound shipments to reach the clients [10]. Thus, it focuses on exchanging the stream of things straightforwardly to the outgoing vehicles ignoring the alternatives of storing the products [11], which aids in consolidating shipments, diminishing conveyance time, and decreasing storage costs. Thus, because of this faster movement of goods and minimum storage options, more and more companies are diverting towards cross-docking than warehousing. As the trucks arrive at the door, the arrival is informed. This initiates the unloading of the products at the concerned docks. Inside the cross docks, a central location for sorting items is assigned. Duly notified, the pallets or cartons are scanned and examined to raise exceptions like inaccurate descriptions. After that, similar products are grouped in various locations using the most profitable and quickest transportation strategy. Thus, several smaller product loads are combined into one transportation mode, saving transportation costs. Also, these methods of breaking down the larger product into smaller units for transportation facilitate a more effortless and faster delivery process. Thus, because of this quicker movement of goods and minimum storage options, more and more companies are diverting towards cross-docking than warehousing.

It is no surprise that expenses related to transportation have tremendous potential for cross-dock savings. As per a technical report published by Saddle Creek Corp. [12] in 2011, about 69% of the companies already used the strategy in their supply chain, Walmart being the pioneer. Due to the adoption of this strategy, Walmart's sales cost dropped by 2% -3% on 85% of their product, making them a profitable enterprise [13].

Within a fully automated cross-dock, the focus is on five stages of internal activities [14]. The first is unloading goods at a strip door, where the packaged items are carried to the organizing region or the staging area, also known as receiving or inbound dock. It is to be noted that one inbound port is relegated for each inbound door at a certain distance from the inbound door. The second operation in the internal activities is picking the goods from the inbound dock by Automated Guided Vehicles (AGV). Each package from the inbound port is scanned and heads toward the pre-assigned location in the shipping dock, depending on the final delivery location. It is widely assumed that each AGV can transport only a single package. Travelling from the inbound to the outbound yard is the third operation at the cross docks. The travel rate at the incoming ports is directly dependent on the vehicle speed but is inversely related to the path length covered by the vehicle. The fourth cross-dock operation is unlocking the package from AGV onto the outbound dock. The final step is to stack containers from the outbound dock into outbound trucks stationed at specific doors. Each of these operations at the cross-dock has a service rate given by Unloading rate (UR), Picking rate (PR), Travel Rate (TR), and Delivery rate (DR). The unloading rate (UR) has been defined as the rate of the product's unloading from the inward-bound vehicle to the receiving dock. The picking rate, denoted as (PR), is an operation of high importance and depends on the nature of the handled goods. The rate of product delivery from the inward storage area to the outward dock is defined as DR. At the same time, the loading of the packages to the respective outbound trucks is LR, and the travel rate is depicted as the rate at which cargo is relocated from the inbound to outbound dock. The packing and unpacking of goods are treated as a part of the unloading and stacking activities, respectively, and are not treated as independent processes. All these operations' service rate is assumed to be constant and measured in packages/hr. A floating dock assignment is taken where the quantity of the inbound and the outbound items are known beforehand. The flow rate of the goods is initially estimated in pounds per second by dividing the AGV's weight capacity by the calculated journey duration. However, the number of Automated Guided Vehicles allocated to move packages from the inward docking area to the departing dock has a noticeable impact on the flow rate.

To sum up, in a nutshell, this paper focuses on:

a) Cross-dock of finite storage capacity where storage space is a fixed parameter affects theinventory's storing time.
b) The flow of goods within the cross-dock from the inbound dock to the outbound dock area should be maximum to avoid congestion. Faster movement of goods within cross docks initiates' a shorter waiting time of trucks in a queue, which benefits the quality of the stored items.
c) The maximum flow of goods is determined by using max-flow algorithms. The best result is determined by comparing the algorithm's execution time.

The remainder of this paper has been arranged in the given manner: A literature review is addressed in Section 2. Section 3 contains the definitions of the associated terminologies, Section 4 discusses the mathematical model's formulation along with notation and assumptions, and Section 6 discusses the numerical example. In contrast, Section 7 discusses the result, and Section 8 presents the research's discussion. The paper concludes with conclusions and recommendations for future research in Section 9.

2. Literature review

The Literature review has been designed following the keywords of the research.

2.1. Cross-dock

The competitiveness in the global environment has compelled companies to search for better options to cope with complicated demands within a short response time. Cross-docking, which moves products straightforwardly from the inward receiving dock to the outward shipping dock, has proven to be a beneficial logistic approach. Cross-docking productivity has impressively affected the lead time, stock levels, and customer response times [15]. There are various factors within the cross-dock [16,17] such as shape [18], size, number of available dock doors [18] number of available handling devices, congestion, the flow rate of goods within the cross-dock, appropriate truck assignment to the door and proper sequencing of incoming and outgoing truck [15] that have affected the functionality of the cross docks and the material flow within the cross docks. It has compelled the researchers to invest their time and effort in proposing some of the best models to complement strategic, tactical, and operational decisions resulting in minimizing waiting time, travel time, travel distance, makespan, and improving the system's throughput.

2.1.1. Handling operations within a cross-dock

It is evident from the previous research conducted that handling freight from the incoming vehicles to the outgoing vehicles is labor-intensive and demands proper operational decision-making. A well-planned material flow can influence the waiting time, travel time, travel distance [19], and the makespan. Bartholdi and Gue [20] developed a non-linear model that considers outbound trucks' mid-term assignments and a dock supervisor's short-term assignment right before the truck arrives. They emphasized that the actual travel time depends on the freight nature, material handling system, and congestion, hence aimed at minimizing labor expenses, travel costs, and congestion costs. To reduce total freight transfer time, an article was presented by Wang and Regan [21], which explains how to assign incoming trailers to receive doors in a cross-dock operation. They looked at some inbound vehicles waiting to be emptied when a dock became accessible. As a general rule, the following truck to arrive should follow a first-in, first-out approach to ensure that all trucks have a reasonable wait time. However, in a cross-dock center, this policy bears suboptimal outcomes. As a result, two algorithms were offered by the authors: the first one is based on the time associated with a new inbound truck's impact on the whole system (from when a product arrived on the inbound truck to when it left on an outbound truck) and the other based on the total transfer time of products. It is also noteworthy that good coordination between truck scheduling and loading and unloading processes must be maintained for the timely processing of goods at the cross-dock terminals. Tadumadze et al. [22] in their study, presented mixed integer models fusing workforce planning and truck schedule. The research results established that integrated planning can significantly improve truck scheduling's efficiency in terms of overall flow time and punctuality. Resat et al. [23] proposed a novel bi-objective solution methodology to optimize the sustainable material handling processes in practice in cross-docking areas. They presented a mixed-integer linear programming model considering the decline of lead times in the reception process, reducing unwanted material handling movements within the cross-dock, and fastening the sorting and transfer processes in the packaging sector to reduce the waiting time of pallets.

2.1.2. Effect of cross-dock capacity on the cross-docking network

The limitations on the physical space in a consolidation node and the limited availability of equipment (such as forklifts) constrain the number of vehicles that may be processed at once. Zachariadis et al. [24] investigated the impact of cross-dock processing capacity on the overall transportation costs. They employed a heuristic mechanism and a constraint programming optimizer as alternate methods for effectively assessing the viability of potential solutions to deal with the increased complexity. Madani-Isfahani et al. [25] presented a model based on mixed-integer programming considering numerous cross docks with limited capacity. Their focus was to reduce the total operational time and facilitate the throughput of the cross-docking system to its maximum level. However, they have considered multiple cross docks with one inbound and outbound door in their work.

2.2. *Supply chain management*

One of the finest ways for companies to provide excellent customer service [26] is through focusing on efficient supply chain management plans. This has facilitated the development of cost effective, adaptable as well as eco-friendly supply chain models [27] not only in manufacturing systems [28] but also in sectors like biodiesel [29] waste management [30] herbal medicines [31] and others. SC models invent synchronized organizing and standardizing forms to facilitate and manage efficient communication between the supply chain participants. A supply chain model witnesses the forward flow of materials from the producer to the customer, so a well-balanced information flow in a reverse direction becomes mandatory to incubate the impact of disturbances in the supply chain. This aspect was addressed by Teresa Wu and Jennifer Blackhurst [32] in their research work. They used Petri nets with incidence matrices to model the flow of materials within a supply chain and inverse incidence matrices for information flow analysis. The restricted number of supply chain operators and the inability to properly evaluate and quantify supply chain performance metrics are both limitations of the Petri net model in this chapter. Ponte et al. [33] emphasized that a reduced mean lead time can positively influence the performance of the internal operation of the production and distribution system and inflates customer satisfaction levels in a four-echelon supply chain model. They presented their analysis by merging agent-based modelling along with the Taguchi method. However, their study was restricted to a linear cost model.

2.2.1. Application of cross-docking in supply chain

Cross-docking is a crucial strategy for enhancing the effectiveness of distribution networks, particularly for streamlining supply chain activities [34]. Even in a cold supply chain, cross-docking terminals (CDTs) have been extensively employed to distribute products. If the relevant cross-dock is selected and then developed, staffed, and worked for optimal performance, a supply chain adopting the cross-dock strategy can expedite items from upstream suppliers to downstream customers rapidly and cost-effectively, with advantages to the entire chain. In his research study, Vogt [35] proposed nine important success indicators for cross-dock-based supply chain operations based on fieldwork in the industry and a literature review. Theophilus et al. [36] developed a mixed integer mathematical model considering the deterioration of perishable products during the service of arrival trucks at the cross-docks. Their study evaluated the presence of temperature-controlled storage spaces reserved exclusively for perishable goods. They aimed to keep the entire cost of the truck service as low as possible. Galbreth et al. [37] utilized a simulation model to evaluate the success of a cross-docking operation for worldwide manufacturing firms' specialized buyer item supply chains. They optimize the cross-docking executions within the setup constraints. Vanajakumari et al. [38] investigated the logistical and inventory management challenges faced by an oilfield service provider in their study. They considered a supply chain that includes suppliers, a Hybrid Cross Dock (HCD) facility, and numerous production facilities. The HCD allows the business to store merchandise without paying for inventory holding expenses temporarily. This study seeks managerial insights while offering strategies to reduce overall logistics and inventory-related expenditures.

2.3. Transportation

The increased competitiveness in today's business environment and customers' expectations for faster delivery of goods have magnified manifold, compelling researchers to look for approaches that would improve profitability and satisfy the customer [39]. The transportation facility depends on fulfilling these expectations of quicker product delivery in a supply chain system and improved lead time. Optimizing distribution and delay costs for the movement of products in open networks with numerous cross-docks [40] or single cross-docks have earned the attention of researchers. Travel time is another significant category of transportation expenditures, and time reductions are frequently the most anticipated benefit of transportation enhancement initiatives [41] apart from minimizing the overall transportation cost [42].

2.3.1. Scheduling in transportation

As cross-docking involves faster movement of goods, a synchronized movement pattern (scheduling) between the inbound vehicles and outbound vehicles[43] becomes mandatory to reduce travel time and transportation costs. Dulebenets contributed immensely in developing the evolutionary algorithms useful in scheduling the trucks at the cross-docks [44]. He further extended his study on evolutionary algorithms presenting a thorough comparison of strong and weak mutation mechanisms [45]. Contunuing his research on scheduling of trucks, Dulebenets proposed adaptive polypoid memetic algorithm to properly plan CDT operations [46]. Issi et al. [47] presented a mixed-integer linear programming model that analyzed the scheduling of inbound and outbound vehicles in a mixed service mode cross-dock. The proposed model minimizes the time between the first inbound and last outbound trucks. Also, the model aimed to reduce the trucks' waiting time at the platforms by forecasting a supply and dispatch schedule. However, the model did not consider the internal operational activities within the warehouse. In order to maximize two competing goals, Goodarzi et al. [48] handled vehicle routing problems with cross-docking taking into account truck scheduling, separating pick-up and delivery services with time windows at supplier and store locations. They suggested a new bi-objective mixed-integer linear programming model alongside a multi-objective meta-heuristic evolutionary algorithm to reduce overall operational costs and the total of maximum early arrivals and late arrivals. Heidari et al.[49] considered uncertainty in vehicle arrival time in their work and used a cost-stable scheduling technique to address the issue of scheduling arriving and departing trucks at a cross-dock facility. Wisittipanich et al. [50] tackled the significance of synchronizing the truck schedule across a network of cross docks and proposed a novel mathematical model of the truck scheduling problem in the network. The goal of their model, which is presented as mixed integer programming (MIP), was to determine the best truck schedule and product transhipment to reduce the makespan. In a similar approach considering multiple cross-docks, Shahmardan and Sajadieh [51] examined a truck scheduling issue in a cross-docking facility where inbound trucks are simultaneously employed as outbound vehicles. Additionally, arriving trucks can be partially unloaded rather than having to unload and reload at the designated destination completely. The issue is modelled as a mixed integer program to identify the ideal dock-door and destination assignments and the truck scheduling to reduce makespan. Sayed et al. [52] addressed an integrated cross-dock door assignment and truck scheduling problem to diminish the total process time of all trucks by simultaneously determining the selection and scheduling of inbound vehicles to inbound doors and outbound trucks to outbound doors. Handling times were determined by truckload and door and involved unloading, transferring, and loading items. They presented two mathematical programming formulations and two-hybrid metaheuristics approach to overcome the problem. In a forward/reverse logistics network, Khorshidian et al. [53] created a bi-objective mathematical model to combine truck scheduling and transportation planning in a cross-docking system. The problem was executed using a hybrid of the enhanced version of the augmented e-constraint approach and TOPSIS.

2.3.2. Door assignment

The performance of the cross-dock is heavily impacted by the number of dock doors and the open doors dedicated to outbound trucks [54]. Werners and Wulfing [55] modelled a linear assignment problem to determine the placement of goods storage at the endpoints and the vehicles designated to the docking doors. Their purpose was to shorten the distance between the ends and the docking doors and reduce manual transportation effort, significantly influencing runtime quality and bringing a noticeable cost drop. Essghaier et al. [56], in their study, considered uncertain transfer time to optimize truck-to-door assignment in a cross-dock. They addressed the approximate arrival time of trucks, uncertain breakdown of facilities, and variation of workload by triangular fuzzy numbers and solved the problem by fuzzy chance programming. Gelareh et al. [57] proposed eight new mixed-integer programming models for the cross-dock door assignment problem and demonstrated the mathematical equivalence of all the models. Their proposed model assigned origins to strip doors and destination to stack doors to minimize total cost within the cross-dock.

2.4. Maximum flow

The maximum flow of goods across a cross-dock can be conceptualized as a network flow problem between the receiving and shipping docks. Practitioners of the graph theory have identified network flow problems as the most studied combinatorial optimization problem, with a vast area of application in the sector of operational research, computer science, and engineering. With the evolution of cross-dock as the newest logistic strategy in practice in the business world, the lookout for performance analysis parameters has intensified. Thus, to determine the feasibility and performance parameters in a cross-dock under specified conditions, Nikita Ankem [14], in her study, proposed a computational model taking into consideration the shape size of the cross-dock and AGV specifications. She applied a max flow algorithm to determine the best cross-dock shape, which supports maximum throughput under specific criteria, and Mean value analysis models to determine the queue lengths and waiting time of the AGV at the queuing node. However, the mean value analysis model was not very successful in isolating the cross dock's performance based on its shape. It is noteworthy that the max flow approach is not restricted to the concept of the only flow of goods. Flow network models have shown their successful implementation in other real-world scenarios like the flow of current through electrical networks [58], the flow of fluids through pipes [59], the flow of information through a digital network or regulatory network [60] and so on.

2.5. Algorithms

Most of the research in the cross-docking sector has been focused on the scheduling of trucks, assignment of trucks to doors or optimizing the operations within a cross-docking system [61] , presenting a vivid variety of algorithms and meta-heuristic approaches related to this area. Daquin et al. [62] employed a variable neighborhood search-based algorithm to distribute trucks to doors when the truck count exceeds available doors. Their goal was to reduce the cost of transferring goods across cross-docks and eliminate delivery delays caused by a lack of open doors. But not many algorithms or research were oriented towards finding the maximum flow of goods within a cross-dock. The max flow through a flow network was initially calculated through the Ford Fulkerson algorithm [63], the general algorithm. However, academics have established newly improved and more efficient max-flow algorithms [64] to tackle problems on maximum flow, presenting a logical comparison based on the running time. In their study, Jain and Garg [65] mentioned that in most cases, the performance of the modified Edmond –Karp algorithm is better than Edmond–Karp algorithm. Mallick et al. [66] discussed the maximum flow from a source node to a sink node following the modified version of the Edmond Karp algorithm in their research paper.

3. Definition

The following section briefly explains the terminologies used in this paper.

3.1. Automated guided vehicles (AGV)

It is an autonomous movable robot that carries out repetitive and dangerous operations within the narrow aisles of warehouses. It can lift and haul huge loads and deliver them to their intended location. (Shown in Figure 1a and 1b). It uses the idea of line followers, whereby lasers, long lines, or wires on the floor help them navigate through their destination. Magnetic sensors with magnetic strip guidance are also used in automated guided vehicles, referred to as tracked AGVs. The AGV controller, which serves as the AGV's brain, and the embedded controller, which manages the AGV, are the two essential components of the AGV robot. Also, the software embedded within the AGV offers the users a desired control over the AGV, allowing them to alter the settings to comply with their needs. It is needless to mention that AGVs have various applications in various industries. AGV robots can provide a good return on investment, particularly for businesses with large facilities and warehouses. AGV is unquestionably one of the most important elements in ensuring that any company's operations run smoothly and efficiently. Hence, over the recent period, the material handling at the cross docks has gradually transcended the territory of automated vehicles, not being confined to only conveyor belts.

3.2. Flow network

A transportation network is a directed graph commonly used to depict material flow, people, or information flow. In this network, vertices represent nodes, and edges represent arcs. Every arc in a flow network has a defined capacity and receives a flow. There can be many arcs or edges leaving or entering a node. On the other hand, the source node has all outer edges, whereas the sink node has all inward edges. Flow networks are typically constructed with one particular source and one sink, or at the very least, approximated with a single dummy source and sink. Various system processes or travel in a transportation network are duly represented by the arcs or the edges in a material flow or transportation network. The delivery and receipt of goods are strictly done at the particular points known as nodes. Every directed edge is considered a channel for conveying materials and information within a flow network. Fixed weight bounds for each channel, expressed as the maximum material flow rate through the conduit, for instance, 400 gallons of liquid per hour flowing through a pipe or 30 amperes of electrical current passing through a wire. Vertices or the nodes in this system are conduit junctions, ensuring a smooth material flow through the vertices without collecting them. A flow conservation policy is maintained in a network. Thereby, the units of flow entering a node correspond to the divisions of flow exiting it. However, the same cannot be considered for the source and the sink node. Only outgoing flow is allowed from the start, and only incoming flow is permitted from the sink. It is also equally important to mention that the maximum flow in an arc cannot exceed its capacity.

Figure 1. (a) Forklift[1]. It is used for lifting and transporting products within short distances. (b) Electric Walkie Pallet Jack[2]. It makes pulling the products through the warehouse very handy. It is not only serviceable for mid-distance runs but also for loading and unloading trailers.

3.3. Max flow

Max-flow problem is a traditional optimization problem. It has been described as the greatest possible flow via a network when just one source node and one sink node are given. It comprises three important ideas: the residual network, augmenting path, and cuts. Thus, restrained within the arc's capacity, the flow network maximizes the flow of materials or information from the starting node to the finishing node. Various algorithms can determine the maximum flow through the web. The Ford–Fulkerson algorithm was designed by Lester R. Ford, the first known max flow algorithm. Jr and Delbert R. Fulkerson [63] back in the year 1955. Over time, many other algorithms with improved solutions for maximum flow have been discovered, like the Edmond–Karp algorithm [64], focussing on the shortest augmenting path , the blocking flow algorithm of Dinitz or Dinic and the new parallel algorithm [67].

4. Mathematical model

As mentioned above, there are assigned docking zones for staging, sorting, delivery, etc. The nodes of the graph represent these. While the arcs showcase the operations of unloading, picking, and travel with a finite capacity. In this system, the service rates for the concerned operations are represented by the capabilities of the arcs. Thus, the cross-dock has been transformed into a network

[1] Forklifts, Kingdom of Bahrain, accessed 24 June, 2022,22.30, pictures of forklifts yale – Bin;g images,

[2] Electric walkie pallet jack, Kingdom of Bahrain, accessed 24 June, 2022,22.35, Low Profile Electric Walkie Pallet Jack - Bing images

flow graph *G (V, E)*, where the various docks are represented by *V*, and the operations performed at the ports are noted by arcs *E*. The in-gate is considered the source node, and the out–gate is regarded as the sink node (Shown in Figure 2). It is important to note that the flow rate of materials from each strip door to the stacking door depends on the values of AGV speed, number of AGVs deployed for the operation, capacity, and service rate of all functions.

4.1. Notation

Unloading speed	U	packages/hr
Picking rate	P	packages/hr
Travel rate	Tpq	packages/hr
Delivery rate	D	packages/hr
Loading speed	L	packages/hr
Number of AGV allocated to each door	n	
Max flow	$\sum \min(U, nP, nT, nD, L)$	

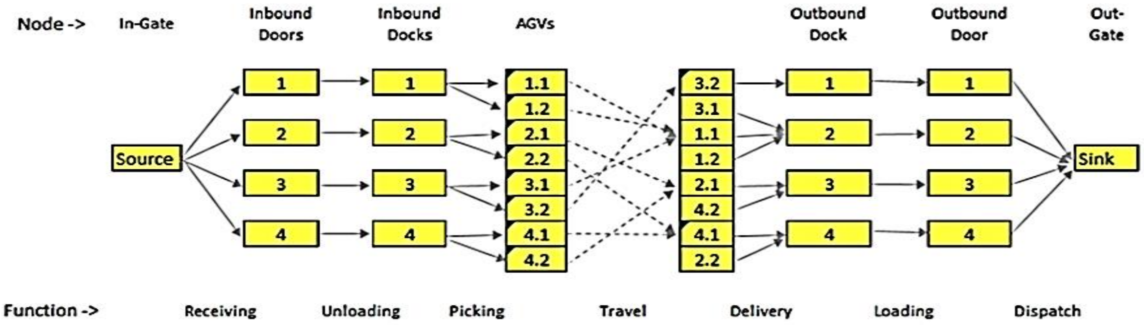

Figure 2. Different operations of unloading, picking, travel, delivery, loading, and dispatch within the cross-dock [14].

4.2. Assumption

a) To ensure faster movement of goods, the strip and stack doors are randomly assigned to the doors on the separate sides of the cross-dock and inbound trucks.

b) An equal number of doors are assigned to both the incoming and leaving docks to maintain a balanced flow of products. Also, the quantity of the goods in and outbound trucks is known beforehand.

c) An inbound package can be considered from any truck stationed at the inbound doors. There is an equal probability of the box being directed to any random outbound door [14].

d) A fixed number of AGVs are allocated for each inbound and outbound door where the automated guided vehicle move is a rectilinear path picking one package at a time [14]. The service rate of the operations performed by a particular AGV is similar, but it may vary depending on the type of AGV deployed for the process.

e) The cross docks have a limited storage capacity and are congestion-free.

4.3. Constraints

The following constraints govern the flow network.

a) Capacity constraints

A non-negative flow *f (p,q)* via an edge or arc $(p,q)\in E$ should always be present. Any edge's capacity *c (p, q)* is its upper bound. The capacity *c (p, q)* for a non-existent edge is 0.

$$0 \le f(p,q) \le c(p,q), \tag{1}$$

b) Flow conservation

$$\sum f(p,q) = \sum f(q,r), \quad \forall(p,q) \text{ and } \forall(q,r)\in E. \tag{2}$$

Every vertex *p* in the set *V* must satisfy the continuity criterion asserting that the flow does not stagnate. i.e., the total units of flow entering into a node *(p, q)* should be equivalent to the total outgoing flows from the node *(q,r).* As a result, the flow conservation property assures that a discharge is only produced or used up at the network's source and sink vertex.

c) Skew symmetric

$$f(p, q) = -f(q, p). \tag{3}$$

The flow from the vertex *p* to *q* is the same as the flow from the vertex *q* to *p* in the reverse direction.

5. Solution methodology

The proposed model's solution methodology necessitates a method that can handle a large extent of nodes and edges. However, managing many nodes with defined operations using a traditional approach is time-intensive. To get the best result, a variety of strategies can be applied. This study uses three sets of algorithms to maximize the flow of goods between the inward and outward docks. A more profound look into the concerned algorithms has been prioritized.

5.1. Ford Fulkerson algorithm

Ford-Fulkerson algorithm is a greedy approach for computing the maximum conceivable flow in a network or a graph [68]. The graph consists of a source vertex sink vertex connected by edges of specified weight. Ford Fulkerson does not specify a particular technique to find the augmenting path, either (BFS) or (DFS). An augmenting course is often summarised as a simple path in the residual network Gf from the source to the sink along edges where the current flow is less than the capacity and does not contain cycles. It is a redundant path created by continuously locating a positive capacity path from a starting node to an ending node and then adding it to the flow. The algorithm runs in an exponential time given by *O(nV) or O(n+m)V*, where the number of nodes is marked by *n*, the number of edges by *m*, and *V* denotes the maximum capacity of the graph. The algorithm fails to terminate in case of irrational abilities, and it has been the foundation of other heuristic algorithms [69].

The algorithm runs in the following manner (shown in Figure 3).

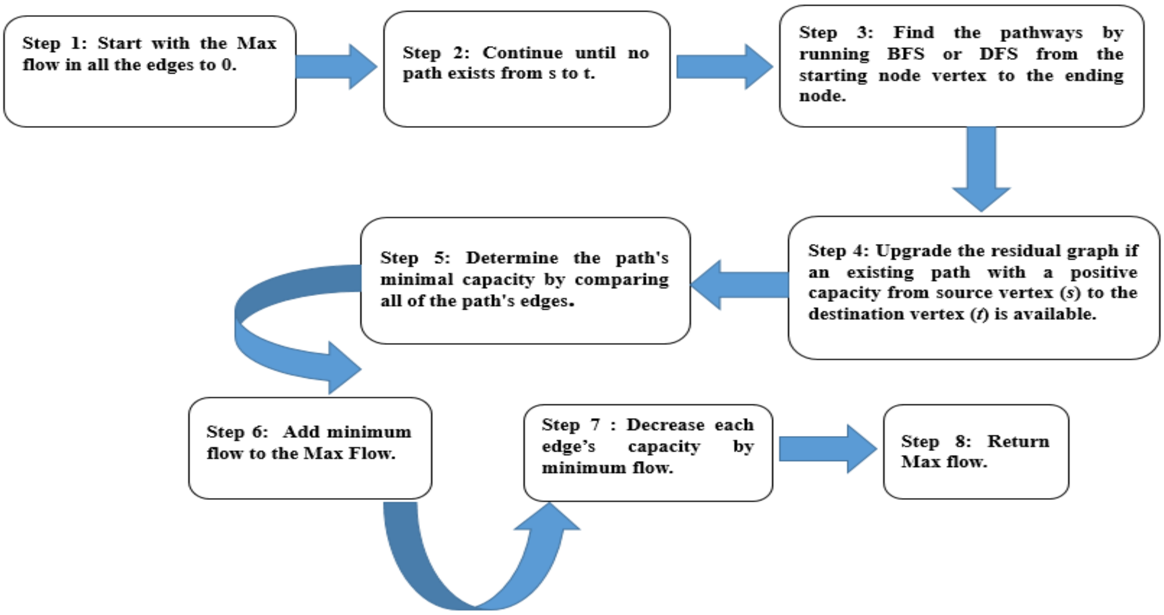
Step 1: Start with the Max flow in all the edges to 0.
Step 2: Continue until no path exists from s to t.
Step 3: Find the pathways by running BFS or DFS from the starting node vertex to the ending node.
Step 4: Upgrade the residual graph if an existing path with a positive capacity from source vertex (*s*) to the destination vertex (*t*) is available.
Step 5: Determine the path's minimal capacity by comparing all of the path's edges.
Step 6: Add minimum flow to the Max Flow.
Step 7 : Decrease each edge's capacity by minimum flow.
Step 8: Return Max flow.

Figure 3. Ford Fulkerson Algorithm.

5.2. *Edmond –Karp algorithm*

Jack Edmonds and Richard Karp introduced Edmond-Karp algorithm to the world in 1972 [64]. It follows the technique of the Breadth-first search method. The algorithm terminates when no more augmented paths can be found from the source to the sink. It searches through all possible ways *n* edges long and guarantees to be complete. Once a course is found, the minimum bounding capacity on that path is calculated, and much flow flows through the way. Studies related to the Edmond Karp algorithm have certified that all vertices in the residual graph increase monotonically. The algorithm is an iterative process that performs a maximum number of iterations of the order $O(nm)$ and outrages Ford Fulkerson's algorithm based on time complexity. Unlike Ford-Fulkerson's algorithm, the Edmond –Karp algorithm is independent of any edge's capacity and establishes itself with the total running time equal to $O(nm^2)$. Edmond-Karp algorithm has been the inspiration behind the exploration of other algorithms in either determining the minimum cost in a network flow problem [70], maximum flow in a data movement pattern [71], or multi-way trading [72].

The Edmond-Karp algorithm runs in the following manner. (Shown in Figure 4).

5.3. *Dinic algorithm*

Computer scientist Yefim (Chaim) A. Dinitz introduced Dinic's algorithm in 1970. Dinic's strongly polynomial algorithm uses the concept of the level graph and blocking flow to evaluate the maximum flow through the network. Time complexity is one of the important factors in determining this algorithm's efficiency. It runs in $O(n^2m)$. Dinic's algorithm also focuses on the shortest augmenting path where the length of the augmenting course is non-decreasing, and eventually, after some iterations, the algorithm terminates. Waissi [73] discussed the algorithm's worst-case bound and established that maximum value flow in a network can be obtained after $(n-1)$ network generations.

The Dinic algorithm executes in the following manner (Shown in Figure 5.)

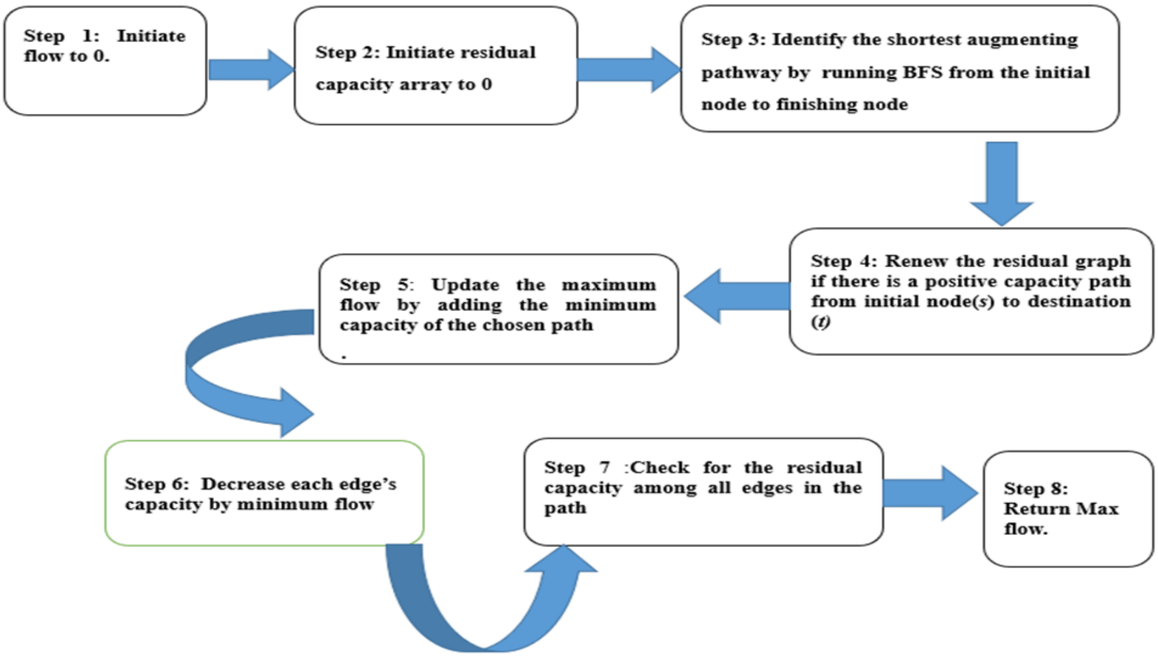
Step 1: Initiate flow to 0.
Step 2: Initiate residual capacity array to 0
Step 3: Identify the shortest augmenting pathway by running BFS from the initial node to finishing node
Step 4: Renew the residual graph if there is a positive capacity path from initial node(s) to destination (t)
Step 5: Update the maximum flow by adding the minimum capacity of the chosen path
Step 6: Decrease each edge's capacity by minimum flow
Step 7 :Check for the residual capacity among all edges in the path
Step 8: Return Max flow.

Figure 4. Edmond-Karp algorithm.

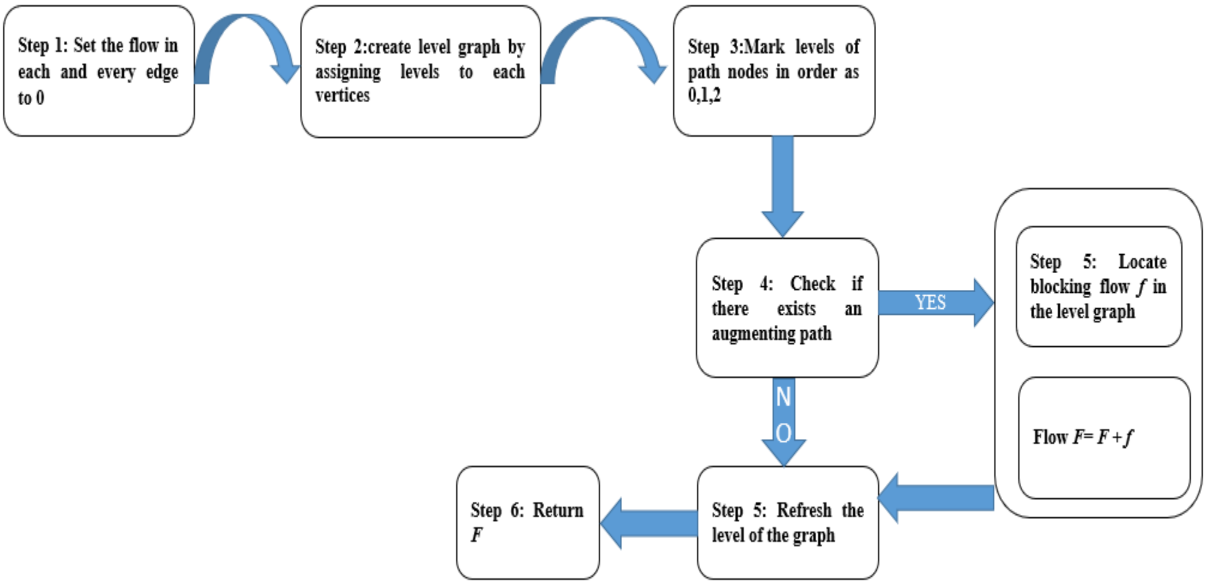
Step 1: Set the flow in each and every edge to 0
Step 2:create level graph by assigning levels to each vertices
Step 3:Mark levels of path nodes in order as 0,1,2
Step 4: Check if there exists an augmenting path
YES
NO
Step 5: Locate blocking flow f in the level graph
Flow F= F + f
Step 5: Refresh the level of the graph
Step 6: Return F

Figure 5. Dinic's algorithm.

6. Numerical example

This part represents the mathematical model using a numerical example, considering small-scale data. The data considered have been referred from [14,65] with some modifications. The following are the details of the standards, along with their explanations.

Table 1. Arc's capacity between two areas.

Source area	Destination area	Capacity (packages/Hr)	Source area	Destination area	Capacity (packages/Hr)
S	*A*	1000	*H*	*J*	550
S	*B*	2000	*H*	*K*	550
S	*C*	1000	*I*	*M*	550
S	*D*	3000	*J*	*N*	550
A	*E*	550	*K*	*0*	550
B	*F*	550	*L*	*P*	550
C	*G*	550	*M*	*Q*	550
D	*H*	550	*N*	*R*	550
E	*J*	550	*O*	*S*	550
E	*L*	550	*P*	*T*	550
F	*I*	550	*Q*	*U*	550
F	*O*	550	*R*	*U*	550
G	*I*	550	*S*	*U*	550
G	*L*	550	*T*	*U*	550

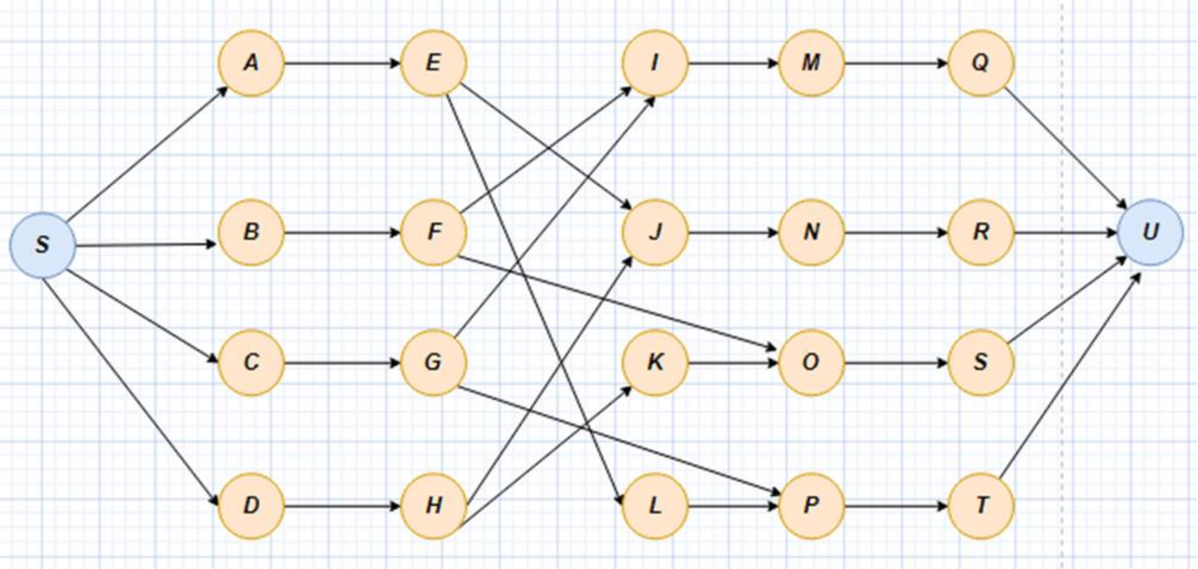

Figure 6. Directed graph connecting the vertices and the edges.

Let us consider a numerical example where goods from source area *S* are transferred to sink area *U*. Before reaching the sink. The products travel through 20 dock zones with authorized operations. These 20 areas are *A*, *B*, *C*, *D*, and *E*, ..., *T*. The arc's capacity between any two dock areas is determined. Table 1 below illustrates the arc's defined ability which are the service rates of AGVs performing the various operations between two docking areas within the cross-dock.

The problem addressed in the example has now been turned into a directed graph (Figure 6), with dock areas representing the graph's vertices and the AGV service rate between any two regions representing the graph's edges. Each area is mapped to a vertex of the directed graph, shown in Table 2. The amount of flow of goods through each operation in the dock is represented in Figure 7.

Table 2. Mapping of Area and Vertex.

Area	Vertex	Area	Vertex
S	0	*K*	11
A	1	*L*	12
B	2	*M*	13
C	3	*N*	14
D	4	*O*	15
E	5	*P*	16
F	6	*Q*	17
G	7	*R*	18
H	8	*S*	19
I	9	*T*	20
J	10	*U*	21

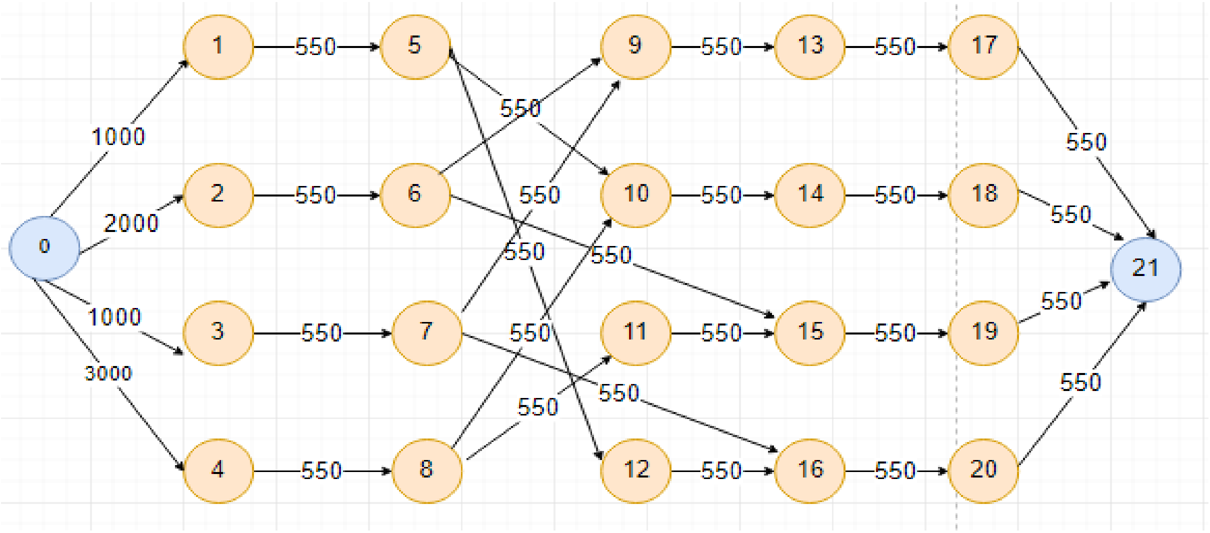

Figure 7. The weighted graph represents the service rate of each operation.

7. Result

In this part, the performance of the three algorithms is analyzed on the parameters of several iterations' execution time. The same number of vertices and maximum capacities are considered for all three algorithms.

Table 3. Algorithms versus Time complexity.

Name of the algorithm	Number of edges	Number of vertices	Maximum capacity	Time complexity (Sec)
Ford Fulkerson	28	22	3000	15000
Edmond Karp algorithm	28	22	3000	17248
Dinic algorithm	28	22	3000	13552

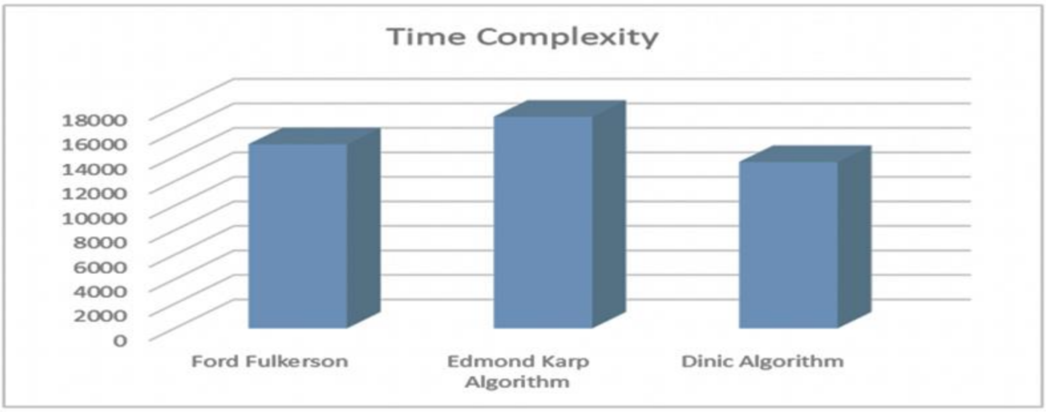

Figure 8. Graphical representation of algorithms and their time complexities.

Table 4. Algorithms versus Execution time.

Name of the algorithm	**Number of iteration**	**Execution time (seconds)**	**Maximum flow**
Ford Fulkerson	4	0.000205399 sec	2200
Edmond Karp algorithm	4	294.578 sec	2200
Dinic algorithm	**3**	**5.9629e-05 sec**	**2200***

* Dinic algorithm generates the best result.

8. Discussions

One of the most significant characteristics of an algorithm is its speed. Depending on the environment, where the algorithm runs, and the exact features of its implementation, the actual rate of the algorithm may differ. The runtime of the algorithm is dependent on input size as well. However, considering a small-scale data set has addressed the run time dependency on the input size.

In very specific situations, Ford-Fulkerson performs wonderfully, most notably when all edge capacities are equal to 1. However, its time complexity is based on the ability of the edge. Hence, the time complexity for Ford-Fulkerson is given by 15000 seconds following the formula $O(n+m)V$ (where n stands for several nodes, m for several edges and V denotes the maximum capacity of the graph). The execution time of the algorithm in a C++ compiler is 0.000205399 sec. After four iterations, the full flow through the network is 2200 units.

The Edmond -Karp algorithm is the same as the Ford-Fulkerson algorithm, except that the search order for finding the augmenting path is set. The algorithm pushes along an augmenting way every time it discovers one. And particularly, the edge with the lowest capacity gets saturated. To sum up, 28 advantages of this flow network get saturated 22 times, taking $O(22)$ for each breadth search. Thus, giving a time complexity of $O(nm^2)$, which is equal to 17248 seconds, while the execution time of the algorithm is 294.578 secs (Refer to Table 3 and Table 4).

The Dinic algorithm follows the concept of blocking flow in a path case of a saturated edge. In Dinic's technique, BFS is also applied to see whether more flow is available to build a level graph. However, once the assignment of the level graph is complete, multiple flows are sent through the network, unlike Edmond Karp, where the flow passes only through the path generated by BFS. This is why it is more effective than Edmond Karp. In the concerned flow network, in 3 iterations, four flows are sent following the path

$(S \longrightarrow 1 \longrightarrow 5 \longrightarrow 10 \longrightarrow 14 \longrightarrow 18 \longrightarrow U)$, $(S \longrightarrow 2 \longrightarrow 6 \longrightarrow 9 \longrightarrow 13 \longrightarrow 17 \longrightarrow U)$, $(S \longrightarrow 3 \longrightarrow 7 \longrightarrow 12 \longrightarrow 16 \longrightarrow 17 \longrightarrow U)$, $(S \longrightarrow 4 \longrightarrow 8 \longrightarrow 10 \longrightarrow 14 \longrightarrow 17 \longrightarrow U)$, giving a maximum flow of 2200. While the Ford Fulkerson and Edmond–Karp obtained the same flow in 4 iterations, allowing one flow in each iteration. *O(m)* time is consumed to perform a BFS to create a level graph, and *O(nm)* time is spent sending multiple more flows until a blocking flow is reached. Thus, the algorithm has an overall time complexity of $O(mn^2)$, which is equivalent to 13552 seconds and the algorithm's execution time is 5.9629e-05 seconds (refer to Table 3 and Table 4).

9. Conclusions and future scopes

To estimate the waiting time of the trucks carrying goods in a queue, the maximum flow of goods within the cross-docks in a definite time horizon has to be determined. As a floating assignment has been considered, the quantity of goods loaded in a truck is known beforehand. Hence, the numerical result of 2200 packages/Hr (Table 4) can be interpreted as the maximum flow rate of goods from the inbound dock to the outbound dock is 2200 packages in one hour. When a time horizon of 24 hours is considered, the total loads or number of packages in all the trucks passing through the in-gate is also known. Thus the unloading time of the total weight of all trucks can be abruptly calculated as the complete number of packages divided by the maximum flow rate. As the entry time of the trucks is recorded, the waiting time of the n^{th} sequence truck is given by adding the entry time of the n^{th} sequence truck with the unloading time of all the packages in $(n-1)^{th}$ trucks. It is equally important to note that the maximum flow within the cross-dock increases if the number of incoming doors increases. Also, an increased number of doors or nodes in the flow network influences the number of iterations and the algorithm's execution time. The comparative study of the algorithms shows that Dinic's algorithm provided the best result regarding iteration and time complexity.

Our study includes limitations that can be taken into account in subsequent research.

a) We considered small-scale data set. In future work, heuristic strategies can be developed to fathom issues with larger-scale occurrences and compare the outcomes with the results of this paper.

b) We considered only AGVs as material handling resources for the internal operations in our study. But the increased use of automation like rack moving mobile robot system (RMMR) or Robotic mobile fulfilment system (RMFS), electric pallet jacks, and automated cranes in the cross-docking area for material handling purposes within and outside the cross-dock has opened future research dimensions. It would certainly be interesting to explore the impact of these automated systems on the makespan, the travel time of the goods, material flow, and the waiting time of the trucks.

c) In this research, we considered limited storage capacity and congestion-free operations. However, congestion during internal operations is very common to occur. Future work can also explore research models considering congestion and other uncertainties.

d) In our research, we mentioned that faster movement of materials would reduce the trucks' waiting time and queueing time without taking a deep dive into the subject. Specific aspects of queuing, waiting, and loading management can be further studied in future work.

Acknowledgements

The authors would like to express their gratitude to the Gulf University in Bahrain for continuing to encourage academic excellence and scientific research.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. M. R. AlZgool, U. Ahmed, S. A. Shah, Q. AlMaamary, N. AlMahmoud, Examining the interplay of HR initiatives, knowledge management, technological capabilities and product innovation, *J. Secur. Sustain. Issues,* **10** (2020), 735–748. http://doi.org/10.9770/jssi.2020.10.2(29)
2. M. Tayyab, M. S. Habib, M. S. Jajja, B. Sarkar, Economic assessment of a serial production system with random imperfection and shortages: A step towards sustainability, *Comput. Ind. Eng.,* **171** (2022), 108398. https://doi.org/10.1016/j.cie.2022.108398
3. A. S. Mahapatra, M. S. Mahapatra, B. Sarkar, S. K. Majumder, Benefit of preservation technology with promotion and time-dependent deterioration under fuzzy learning, *Expert Syst. Appl.*, **201** (2022), 117169. https://doi.org/10.1016/j.eswa.2022.117169
4. S. Khalilpourazari, A. Mirzazadeh, G. W. Weber , S. H. R. Pasandideh, A robust fuzzy approach for constrained multi-product economic production quantity with imperfect items and rework process, *Optimization*, **69** (2020), 63–90. https://doi.org/10.1080/02331934.2019.1630625
5. B. Sarkar, M. Ullah, M. Sarkar, Environmental and economic sustainability through innovative green products by remanufacturing, *J. Clean. Prod.*, **332** (2022), 129813. https://doi.org/10.1016/j.jclepro.2021.129813
6. S. Kumar, M. Sigroha, K. Kumar, B. Sarkar, Manufacturing/remanufacturing based supply chain management under advertisements and carbon emission process, *RAIRO- Oper. Res.*, **56** (2022), 831–851. https://doi.org/10.1051/ro/2021189
7. R. Lotfi, Y. Z. Mehrjerdi, M. S. Pishvaee, A. Sadeghieh, G. W. Weber, A robust optimization model for sustainable and resilient closed-loop supply chain network design considering conditional value at risk, *Numer. Algebra, Control. Optim.*, **11** (2021), 221–253. https://doi.org/10.3934/naco.2020023
8. T. Paksoy, T. Bektaş, E. Özceylan, Operational and environmental performance measures in a multi-product closed-loop supply chain, *Transport Res E-Log,* **47** (2011), 532–546. https://doi.org/10.1016/j.tre.2010.12.001
9. A. Mondal, S. K. Roy, Multi-objective sustainable opened-and closed-loop supply chain under mixed uncertainty during COVID-19 pandemic situation, *Comput. Ind. Eng.,* **159** (2021), 107453. https://doi.org/10.1016/j.cie.2021.107453

10. O. Theophilus, M. A. Dulebenets, J. Pasha, O. F. Abioye, M. Kavoosi, Truck scheduling at cross-docking terminals: A follow-up state-of-the-art review, *Sustainability*, **11** (2019), 5245. https://doi.org/10.3390/su11195245
11. J. Van Belle, P. Valckenaers, D. Cattrysse, Cross-docking: State of the art, *Omega*, **40** (2012), 827–846. https://doi.org/10.1016/j.omega.2012.01.005
12. S. C. Corp, *Cross-docking Trend Report,* Whitepaper Series, Saddle Creek Corp: Lakeland, FL, USA, 2011. Available from: 070111DCMwe.pdf (distributiongroup.com)
13. G. Stalk, P. Evans, L. E. Shulman, Competing on capabilities: The new rules of corporate strategy, *Harv. Bus. Rev.*, **70** (1992), 57–69.
14. N. S. Ankem, *Models for performance analysis of a cross-dock*, Master Thesis, University Park, Pennsylvania: Pennsylvania State University, USA, 2017.
15. Y. Kuo, Optimizing truck sequencing and truck dock assignment in a cross docking system, *Expert Syst. Appl.*, **40** (2013), 5532–5541. https://doi.org/10.1016/j.eswa.2013.04.019
16. J. J. Bartholdi III, S. T. Hackman, *Warehouse and Distribution Science Release 0.94*, Supply Chain and Logistics Institute, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, 2011.
17. K. K. Yang, J. Balakrishnan, C. H. Cheng, An analysis of factors affecting cross docking operations, *J. Bus. Logist.*, **31** (2011), 121–148. https://doi.org/10.1002/j.2158-1592.2010.tb00131.x
18. J. J. Bartholdi III, K. R. Gue, Best shape of a cross-dock, *Transp. Sci.*, **38** (2004), 235–244. https://doi.org/10.1287/trsc.1030.0077
19. I. F. Vis, K. J. Roodbergen, Positioning of goods in a cross-docking environment, *Comput. Ind. Eng.*, **54** (2008), 677–689. https://doi.org/10.1016/j.cie.2007.10.004
20. J. J. Bartholdi III, K. R. Gue, Reducing labor costs in an LTL crossdocking terminal, *Oper. Res.*, **48** (2000), 823–832. https://doi.org/10.1287/opre.48.6.823.12397
21. J. F. Wang, A. Regan, Real-time trailer scheduling for cross dock operations, *Transp. J.*, **47** (2008), 5–20. https://doi.org/10.5325/transportationj.47.2.0005
22. G. Tadumadze, N. Boysen, S. Emde, F. Weidinger, Integrated truck and workforce scheduling to accelerate the unloading of trucks, *Eur. J. Oper. Res.*, **278** (2019), 343–362. https://doi.org/10.1016/j.ejor.2019.04.024
23. H. G. Resat, P. Berten, Z. Kilek, M. B. Kalay, Design and development of robust optimization model for sustainable cross-docking systems: A case study in electrical devices manufacturing company, In: Muthu, SS (eds) *Sustainable Packaging. Environmental Footprints and Eco-design of Products and Processes.* Springer, Singapore, (2021), 203–224. https://doi.org/10.1007/978-981-16-4609-6_8
24. E. E. Zachariadis, A. I. Nikolopoulou, E. G. Manousakis, P. P. Repoussis, C. D. Tarantilis, The vehicle routing problem with capacitated cross-docking, *Expert Syst. Appl.*, **196** (2022), 116620. https://doi.org/10.1016/j.eswa.2022.116620
25. M. Madani-Isfahani, R. Tavakkoli-Moghaddam, B. Naderi, Multiple cross-docks scheduling using two meta-heuristic algorithms, *Comput. Ind. Eng.,* **74** (2014), 129–138. https://doi.org/10.1016/j.cie.2014.05.009
26. S. B. Choi, B. K. Dey, S. J. Kim, B. Sarkar, Intelligent servicing strategy for an online-to-offline (O2O) supply chain under demand variability and controllable lead time, *RAIRO- Oper. Res.*, **56** (2022), 1623–1653. https://doi.org/10.1051/ro/2022026

27. B. Pal, A. Sarkar, B. Sarkar, Optimal decisions in a dual-channel competitive green supply chain management under promotional effort, *Expert Syst. Appl.*, (2023), 118315. https://doi.org/10.1016/j.eswa.2022.118315
28. B. Sarkar, S. Bhuniya, A sustainable flexible manufacturing–remanufacturing model with improved service and green investment under variable demand, *Expert Syst. Appl.*, **202** (2022), 117154. https://doi.org/10.1016/j.eswa.2022.117154
29. M. S. Habib, M. Omair, M. B. Ramzan, T. N. Chaudhary, M. Farooq, B. Sarkar, A robust possibilistic flexible programming approach toward a resilient and cost-efficient biodiesel supply chain network, *J. Clean. Prod.*, **366** (2022), 132752. https://doi.org/10.1016/j.jclepro.2022.132752
30. B. Sarkar, A. Debnath, A. S. Chiu, W. Ahmed, Circular economy-driven two-stage supply chain management for nullifying waste, *J. Clean. Prod.*, **339** (2022), 130513. https://doi.org/10.1016/j.jclepro.2022.130513
31. A. Garai, B. Sarkar, Economically independent reverse logistics of customer-centric closed-loop supply chain for herbal medicines and biofuel, *J. Clean. Prod.*, 334 (2022), 129977. https://doi.org/10.1016/j.jclepro.2021.129977
32. T. Wu, J. Blackhurst, Modelling supply chain information and material perturbations, in *Supply Chain Management and Knowledge Management* (A. Dwivedi and T. Butcher eds), Palgrave Macmillan, London, (2009), 107–123. https://doi.org/10.1057/9780230234956_6
33. B. Ponte, J. Costas, J. Puche, R. Pino, D. de la Fuente, The value of lead time reduction and stabilization: A comparison between traditional and collaborative supply chains, *Transport Res. E-Log.*, **111** (2018), 165–185. https://doi.org/10.1016/j.tre.2018.01.014
34. E. B. Tirkolaee, A. Goli, A. Faridnia, M. Soltani, G. W. Weber, Multi-objective optimization for the reliable pollution-routing problem with cross-dock selection using Pareto-based algorithms, *J. Clean. Prod.*, **276** (2020), 122927. https://doi.org/10.1016/j.jclepro.2020.122927
35. J. J. Vogt, The Successful cross-dock based supply chain, *J. Bus. Logist.*, **31** (2010), 99–119. https://doi.org/10.1002/j.2158-1592.2010.tb00130.x
36. O. Theophilus, M. A. Dulebenets, J. Pasha, Y. Y. Lau, A. M. Fathollahi-Fard, A. Mazaheri, Truck scheduling optimization at a cold-chain cross-docking terminal with product perishability considerations, *Comput. Ind. Eng.,* **156** (2021), 107240. https://doi.org/10.1016/j.cie.2021.107240
37. M. R. Galbreth, J. A. Hill, S. Handley, An investigation of the value of cross-docking for supply chain management, *J. Bus. Logist.*, **29** (2008), 225–239. https://doi.org/10.1002/j.2158-1592.2008.tb00076.x
38. M. Vanajakumari, H. Sun, A. Jones, C. Sriskandarajah, Supply chain planning: A case for Hybrid Cross-Docks, *Omega,* **108** (2022), 102585. https://doi.org/10.1016/j.omega.2021.102585
39. D. Mardanya, G. Maity, S. K. Roy, The multi-objective multi-item just-in-time transportation problem, *Optimization,* (2021), 1–32. https://doi.org/10.1080/02331934.2021.1963246
40. P. B. Castellucci, A. M. Costa, F. Toledo, Network scheduling problem with cross-docking and loading constraints, *Comput Oper Res*, **132** (2021), 105271. https://doi.org/10.1016/j.cor.2021.105271
41. B. K. Giri, S. K. Roy, Neutrosophic multi-objective green four-dimensional fixed-charge transportation problem, *Int. J. Mach. Learn. Cybern.*, (2022), 1–24. https://doi.org/10.1007/s13042-022-01582-y

42. S. K. Das, M. Pervin, S. K. Roy, G. W. Weber, Multi-objective solid transportation-location problem with variable carbon emission in inventory management: a hybrid approach, *Ann. Oper. Res.,* (2021), 1–27. https://doi.org/10.1007/s10479-020-03809-z
43. A. M. Fathollahi-Fard, M. Ranjbar-Bourani, N. Cheikhrouhou, M. Hajiaghaei-Keshteli, Novel modifications of social engineering optimizer to solve a truck scheduling problem in a cross-docking system, *Comput. Ind. Eng.*, **137** (2019), 106103, https://doi.org/10.1016/j.cie.2019.106103
44. M. A. Dulebenets, A diploid evolutionary algorithm for sustainable truck scheduling at a cross-docking facility, *Sustainability*, 10 (2018), 1333. https://doi.org/10.3390/su10051333
45. M. A. Dulebenets, A comprehensive evaluation of weak and strong mutation mechanisms in evolutionary algorithms for truck scheduling at cross-docking terminals, *IEEE Access*, **6** (2018), 65635–65650. https://doi.org/10.1109/ACCESS.2018.2874439
46. M. A. Dulebenets, An adaptive polyploid memetic algorithm for scheduling trucks at a cross-docking terminal, *Inf. Sci.*, **565** (2021), 390–421. https://doi.org/10.1016/j.ins.2021.02.039
47. G. C. Issi, R. Linfati, J. W. Escobar, Mathematical optimization model for truck scheduling in a distribution center with a mixed service-mode dock area, *J. Adv. Transp.*, **2020** (2020). https://doi.org/10.1155/2020/8813372
48. A. H. Goodarzi, R. Tavakkoli-Moghaddam, A. Amini, A new bi-objective vehicle routing-scheduling problem with cross-docking: mathematical model and algorithms., *Comput. Ind. Eng.*, **149** (2020), 106832. https://doi.org/10.1016/j.cie.2020.106832
49. F. Heidari, S. H. Zegordi, R. Tavakkoli-Moghaddam, Modeling truck scheduling problem at a cross-dock facility through a bi-objective bi-level optimization, *J. Intell. Manuf.*, **29** (2018), 1155–1170. https://doi.org/10.1007/s10845-015-1160-3
50. W. Wisittipanich, T. Irohara, P. Hengmeechai, Truck scheduling problems in the cross docking network, *Int. J. Logist. Syst. Manag.,* **33** (2019), 420–439. https://doi.org/10.1504/IJLSM.2019.101164
51. A. Shahmardan, M. S. Sajadieh, Truck scheduling in a multi-door cross-docking center with partial unloading—Reinforcement learning-based simulated annealing approaches, *Comput. Ind. Eng.*, **139** (2020), 106134. https://doi.org/10.1016/j.cie.2019.106134
52. S. I. Sayed, I. Contreras, J. A. Diaz, D. E. Luna, Integrated cross-dock door assignment and truck scheduling with handling times, *TOP*, **28** (2020), 705–727. https://doi.org/10.1007/s11750-020-00556-z
53. H. Khorshidian, M. A. Shirazi, S. M. F. Ghomi, An intelligent truck scheduling and transportation planning optimization model for product portfolio in a cross dock, *J. Intell. Manuf.*, **30** (2019), 163–184. https://doi.org/10.1007/s10845-016-1229-7
54. A. Motaghedi-Larijani, Solving the number of cross-dock open doors optimization problem by combination of NSGA-II and multi-objective simulated annealing, *Appl. Soft Comput.*, **128** (2022), 109448. https://doi.org/10.1016/j.asoc.2022.109448
55. B. Werners, T. Wülfing, Robust optimization of internal transports at a parcel sorting center operated by Deutsche Post World Net, *Eur. J. Oper. Res.*, **201** (2010), 419–426. https://doi.org/10.1016/j.ejor.2009.02.035
56. F. Essghaier, H. Allaoui, G. Goncalves, Truck to door assignment in a shared cross-dock under uncertainty, *Expert Syst. Appl.*, **182** (2021), 114889. https://doi.org/10.1016/j.eswa.2021.114889

57. S. Gelareh, F. Glover, O. Guemri, S. Hanafi, P. Nduwayo, R. Todosijević, A comparative study of formulations for a cross-dock door assignment problem, *Omega*, **91** (2020), 102015. https://doi.org/10.1016/j.omega.2018.12.004
58. M. T. Kyi, S. S. Maw, L. L. Naing, Mathematical estimation for maximum flow in electricity distribution network by Ford-Fulkerson iteration algorithm, *Int. J. Sci. Res. Publ.*, **9** (2019), https://doi.org/10.29322/IJSRP.9.08.2019.p9229
59. M. T. Kyi, L. L. Naing, Application of Ford-Fulkerson algorithm to maximum flow in water distribution pipeline network, *Int. J Sci. Res. Publ.*, **8** (2018), https://doi.org/10.29322/IJSRP.8.12.2018.p8441
60. A. Özmen, E. Kropat, G. W. Weber, Robust optimization in spline regression models for multi-model regulatory networks under polyhedral uncertainty, *Optimization*, **66** (2017), 2135–2155. https://doi.org/10.1080/02331934.2016.1209672
61. A. Bellanger, S. Hanafi, C. Wilbaut, Three-stage hybrid-flowshop model for cross-docking, *Comput Oper Res*, **40** (2013), 1109–1121. https://doi.org/10.1016/j.cor.2012.11.009
62. C. Daquin, H. Allaoui, G. Goncalves, T. Hsu, Variable neighborhood search based algorithms for crossdock truck assignment, *RAIRO Oper. Res.*, **55** (2021), S2291–S2323. https://doi.org/10.1051/ro/2020087
63. L. R. J. Ford, D. R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, NJ. 1962
64. J. Edmonds, R. M. Karp, Theoretical improvements in algorithmic efficiency for network flow problems, *J. ACM*, **19** (1972), 248–264. https://doi.org/10.1145/321694.321699
65. C. Jain, D. Garg, Improved Edmond Karp algorithm for network flow problem, *Int. J. Comput. Appl.*, **37** (2012), 48–53. https://doi.org/10.5120/4576-6624
66. K. K. Mallick, A. R. Khan, M. M. Ahmed, M. S. Arefin, M. S. Uddin, Modified Edmonds-Karp algorithm to solve maximum flow problem, *Open J. App. Sci.*, **6** (2016), 131–140. https://doi.org/10.4236/ojapps.2016.62014
67. Y. Peretz, Y. Fischler, A fast parallel max-flow algorithm, *J. Parallel Distrib. Comput.,* **169** (2022), 226–241. https://doi.org/10.1016/j.jpdc.2022.07.003
68. M. Bulut, E. Özcan, Optimization of electricity transmission by Ford–Fulkerson algorithm, *Sustain. Energy, Grids Netw.*, **28** (2021), 100544. https://doi.org/10.1016/j.segan.2021.100544
69. M. S. Sabbagh, H. Ghafari, S. R. Mousavi, A new hybrid algorithm for the balanced transportation problem, *Comput. Ind. Eng.*, **82** (2015), 115–126. https://doi.org/10.1016/j.cie.2015.01.018
70. D. Goldfarb, Z. Jin, A new scaling algorithm for the minimum cost network flow problem, *Oper. Res. Lett.*, **25** (1999), 205–211. https://doi.org/10.1016/S0167-6377(99)00047-4
71. H. Bui, E. S. Jung, V. Vishwanath, A. Johnson, J. Leigh, M. E. Papka, Improving sparse data movement performance using multiple paths on the Blue Gene/Q supercomputer, *Parallel Comput.*, **51** (2016), 3–16. https://doi.org/10.1016/j.parco.2015.09.002
72. R. M. Kaplan, An improved algorithm for multi-way trading for exchange and barter, *Electron. Commer. Res. Appl.*, **10** (2011), 67–74. https://doi.org/10.1016/j.elerap.2010.08.001
73. G. R. Waissi, Worst case behavior of the Dinic algorithm, *Appl. Math. Lett.*, **4** (1991), 57–60. https://doi.org/10.1016/0893-9659(91)90145-L