INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

WILEY

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 0 (2023) 1–34
DOI: 10.1111/itor.13319

A blockchain-based framework to optimize shipping container flows in the hinterland

Mariem Mhiri[a], Karim Al-Yafi[b], Benjamin Legros[c], Oualid Jouini[d] and Mohamed Haouari[a,*]

[a] *Department of Mechanical and Industrial Engineering, College of Engineering, Qatar University, 2713 Doha, Qatar*
[b] *Department of Accounting and Information Systems, College of Business and Economics, Qatar University, 2713 Doha, Qatar*
[c] *EM Normandie Business School, Métis Lab, 64 Rue du Ranelagh, Paris 75016, France*
[d] *Laboratoire Génie Industriel, CentraleSupélec, Université Paris-Saclay, 9 rue Joliot Curie, Gif-sur-Yvette91190, France*
E-mail: meriam.mhiri@qu.edu.qa [Mhiri]; karim.alyafi@qu.edu.qa [Al-Yafi]; benjamin.legros@centraliens.net [Legros]; oualid.jouini@centralesupelec.fr [Jouini]; mohamed.haouari@qu.edu.qa [Haouari]

Received 1 September 2022; received in revised form 9 March 2023; accepted 5 May 2023

Abstract

We address two interrelated issues affecting the hinterland portion of the maritime container supply chain: reducing the movement of empty containers and reducing empty trips by trucks carrying these containers. In this paper, we show that empty container flow optimization can be implemented via a blockchain based on the proof-of-useful-work concept where the proof of work requires the solution of an $\mathcal{NP}$-hard optimization problem whose solution benefits the blockchain participants. Accordingly, we propose that anonymous miners compete to solve the container truck routing problem, which seeks to find the most efficient routes for trucks. We show that this problem is $\mathcal{NP}$-hard. Miners must also solve the problem of optimally matching consignees and shippers, which will reduce transportation and storage costs for empty containers. In essence, the proposed framework turns blockchain into a massive optimization engine that directly benefits the hinterland container supply chain ecosystem.

Keywords: container supply chain; prize collecting vehicle routing problem; blockchain; proof of useful work

1. Introduction

Container logistics is a complex process that involves both the delivery of empty containers to exporting companies (shippers) and the delivery of loaded containers that have been preordered by importing companies (consignees). In this paper, we study the problem of managing the transportation of shipping containers in the hinterland. The hinterland refers to “the land area over which a port sells its services and interacts with its users, and regroups all the customers directly bounded

*Corresponding author.

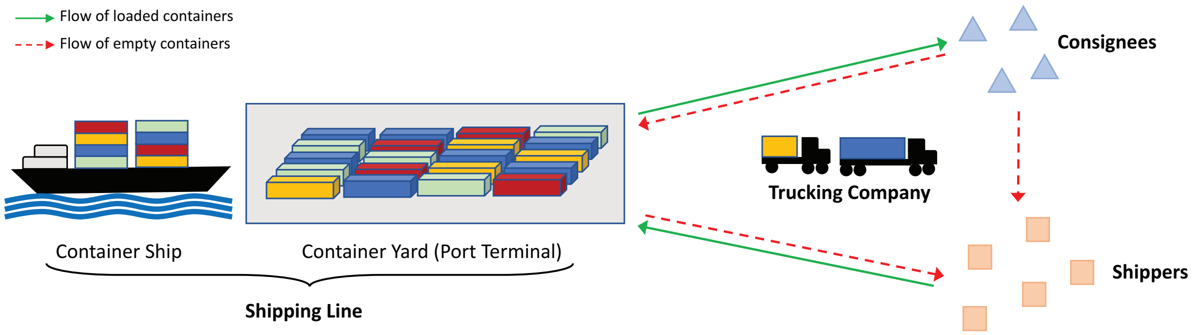


Fig. 1. Loaded and empty containers flows in hinterland supply chain.

to the port and the land areas from which it draws and distributes traffic" (Notteboom et al., 2022). The hinterland container supply chain plays a central role in global trade since its transportation cost represents 40%–80% of the total transportation costs (Bouchery et al., 2015).

Traditionally, container flow management has involved solving individual optimization problems, often from the perspective of either an importer or an exporter. In contrast to prior studies, we develop an approach enabling the actors of the hinterland supply chain to work together. This leads to improved performance by optimizing empty container movements and empty truck trips. The approach is based on a blockchain framework that avoids the need of a centralized coordination, while ensuring efficiency, security, and fairness. We provide the technical description of the blockchain solution and also solve the related optimization problems.

Blockchain is a distributed ledger technology that has been popularized by the phenomenal development of cryptocurrencies (Youssef et al., 2022). Since its inception, blockchain has been recognized as a powerful enabling technology to promote collaboration between stakeholders at all stages of the supply chain. We show that this technology is also useful to optimize the operations of the hinterland supply chain by allowing cooperation between independent agents that are neither known nor trusted. In doing so, the blockchain turns into a huge distributed optimization system. At this point, it is worth noting that, unlike collaboration, where different stakeholders work together to achieve a *common* (i.e., shared) goal, cooperation is a process that allows each stakeholder to achieve its *own* goal by working with the other stakeholders.

We propose to develop a public (or permissionless) blockchain that allows different stakeholders in the hinterland container supply chain, irrespective of the shipping lines that own the containers, to interact with the entries in the blockchain. Through this interaction, the proposed blockchain not only optimizes the flow of empty containers but also automates the cash flows thanks to smart contracts, which in turn automate the execution of an agreement between all the participants, when certain conditions are met without referring to a third party (IBM, 2017). The smart contracts concept is a smart invention with no smart component that helps in automating processes as well as reducing paperwork costs (Ganne, 2018). In Drewry Supply Chain Advisors (2018), the cost of paperwork process inefficiencies and the lack of trust (Cash Against Documents) is estimated at $34 billion per year.

Figure 1 illustrates the container flows in the hinterland surrounding a sea terminal. There are four main players directly involved in the hinterland supply chain, namely consignees, shippers, shipping lines, and trucking companies. The typical container movements are as follows. After being unloaded from the ships, the loaded containers are temporarily stored in the container yard (CY) before being loaded onto trucks and transported to their respective consignees. Once unloaded, the empty containers are loaded onto trucks and returned to CY where they are stored before being trucked to hinterland shippers that request them. Empty containers are transported from the CY to the hinterland shippers that have expressed their demand. To decrease expenses, consignees and shippers can choose to conduct *direct* shipments of empty containers, bypassing the need to go through the shipping line's CY. Once loaded, the containers are brought back by the shippers to the CY where they are temporarily stored before being loaded on the ships.

In what follows, we address the following research questions.

RQ1—Given a set of data including the locations of the CY, consignees and shippers, number and types of containers offered/requested, and release/due dates, *how to compute a minimum-cost container dispatching strategy?* In other words, how to find a feasible matching between consignees and shippers, which minimizes the sum of the cost of moving empty containers and the associated holding costs? This direct shipment strategy results in a net savings over the traditional strategy with no cooperation. A question then arises as how to share the savings between all participants while guaranteeing fairness and security?

RQ2—To maximize their profits, trucking companies seek to efficiently assign the fleets to the routes that are requested by consignees and shippers, while minimizing empty (i.e., unloaded) trips. In order to obtain the best route selection, a trucking company must be given the opportunity to select from the largest possible number of transportation requests available at any given time. Therefore, given a set of container transportation requests, how to model and solve the problem of selecting the most profitable set of truck routes?

We employ an analysis methodology that lies at the interface of Operations Research and Information Science. It consists of an *integrated* answer to the research questions based on combining discrete optimization models with blockchain technology. Specifically, we make the following contributions.

- We develop a matching model to optimize empty container flows between consignees and shippers, and show that it can be solved in polynomial time. We then introduce a new vehicle routing problem for selecting the optimal combination of empty and loaded trips, and show that it is $\mathcal{NP}$-hard.
- We show how the proposed optimization problems can be embedded as a useful proof of work (PoW) in a blockchain that will enable cooperation between participants through smart contracts. We also provide a technical description of the blockchain processes.
- We develop an estimate of the expected savings ratio as a function of the number of participants. A numerical study attests to the tractability of the matching problem, and show that larger, balanced instances are expected to yield higher savings.

The paper is organized as follows. In Section 2, we review the literature related to this study. In Section 3, we provide an overview of the proposed blockchain-based solution. In Section 4, we develop and analyze the related optimization problems. In Section 5, we provide the results of a computational study of the matching optimization procedure. Then, in Section 6, we present a description of the blockchain. Finally, Section 7 concludes the paper and describes some avenues for future research.

2. Literature review

In this literature review, we present prior studies that relate to the management of empty containers. Next, we provide contributions that highlight the role of the blockchain with potential applications in the logistic of containers repositioning. Finally, we emphasize the research gaps.

2.1. Empty containers management

Empty container management has deserved a lot of attention from the transport and maritime economics communities. We refer to Dejax and Crainic (1987) for a review of early works of the Operations Management/Transportation Science community on containerization and refer to Braekers et al. (2011), Lee and Meng (2014), and Lee and Song (2017) for more recent surveys.

Different problems arise in the management of containers like the loading (Hifi, 2002; Costa and Captivo, 2016; Deplano et al., 2021) or scheduling issue (Kozan and Preston, 1999; Choi et al., 2012). In this study, we focus on the containers repositioning problem where a container emptied at an importer can be reused by an exporter to transport goods to a sea terminal. Lee and Song (2017) classified the existing contributions in this field into two categories. The category investigates network flow models for the empty container repositioning issue (Li et al., 2007; Song and Dong, 2008; Dang et al., 2012, 2013; Hjortnaes et al., 2017). For instance, the study in Choong et al. (2002) focuses on a computational analysis of the effect of planning horizon length on empty container management for intermodal transportation networks. The problem is modeled as an integer program whose objective function consists in minimizing the total costs related to moving empty containers, subject to meeting requirements for moving loaded containers. Song and Carter (2009) identify critical factors affecting empty container movements and quantify the scale of empty container repositioning in major shipping routes. They employ a mathematical programming approach to evaluate and contrast the performance of four strategies for empty container repositioning in three major routes.

The second category considers the empty container repositioning problem from an inventory theory perspective. In these studies, empty containers movement is mainly analyzed between the sea terminal and consignee but not in a strategy involving several importers, exporters, and sea terminal as we do in this paper. We mention Li et al. (2004), Song and Zhang (2010), and Zhang et al. (2014) who considered a single empty depot located in a port and controlled by a shipping line, while Song (2005), Song (2007), Lam et al. (2007),Shi and Xu (2011), Ng et al. (2012), and Xie et al. (2017) focused on empty container (or equivalently vehicle) management for a two-depot system.

Specifically, in Li et al. (2004), the authors formulate the one-port containerization problem as an inventory problem with positive and negative demands. Considering general holding-penalty cost function and one-time period delay availability for full containers just arriving at the port are the two main assumptions. The investigation of the finite-horizon problem shows that a two critical point policy in each decision stage is optimal. Finally, Lam et al. (2007), Shi and Xu (2011), and Legros et al. (2019) employed Markov decision process approaches to determine optimal policies for repositioning management with different model assumptions than ours. However, in these studies, importers and exporters are analyzed as single entities. Our paper aims to provide a more general and realistic setting for the management of empty containers in the hinterland.

Related works dealing with transportation of empty containers do not introduce neither the cooperation approach nor the blockchain technology but rather use other traditional platforms, such as in Lin and Juan (2021), where the problem of empty container repositioning is studied according to a sharing strategy between shipping lines. The approach considers a matching platform company that provides empty container for shipping lines in the context of maritime transportation network. Cost reduction for shipping lines that adopt the container sharing strategy is highlighted through numerical experiments. We refer to the survey by Islam (2017) where the author highlights the relevance of collaboration in the container transportation industry in order to achieve sustainable transportation benefits at the port and its important surrounding areas. In Islam et al. (2019), the study reveals the potential truck-sharing constraints for container trucks traveling empty. The authors underline that empty truck trips lead to decrease transport capacity in the container distribution chain and increase the carbon emission, traffic congestion, fuel consumption, and environmental pollution.

2.2. Blockchain applications in the containers logistics

Research in the area of blockchain technology application in logistics and supply chain management continues to be extremely active. We refer to Pournader et al. (2020) for a comprehensive review of the literature, up to 2020, on the application of blockchain in supply chain management, logistics, and transportation. Another review study in Astarita et al. (2020) shows that blockchain application to the transportation sector is in an early phase of development and there are few implemented blockchain systems with real context. It is stressed that this technology can improve trust and data sharing among supply chain actors, reduce exhaust gas emissions, favor correct urban development, and improve life quality. Moreover, the study in Irannezhad (2020) clarifies how blockchain contributes in improving the inefficiencies in maritime supply chain and logistics. This technology helps in facilitating accessibility and readability, increasing knowledge and mutual information sharing between actors, providing transactions security, and enabling coordination via smart contracts. The study also reveals blockchain limitations and challenges such as scalability, interoperability, and lack of standards and regulations. Recently, two papers were published on the application of blockchain technology in the context of maritime supply chains. In the first article, Chen and Yang (2022) discussed the application of blockchain technology in the context of competition between shipping companies and freight forwarders, and analyzed the impact of blockchain technology on changes in market structure. On the other hand, Xin et al. (2022) investigated blockchain technology investment in the shipping supply chain and analyzed its impact on the consumer surplus as well as on the social welfare. Furthermore, many recently published articles discuss the contribution of blockchain technology and smart contracts to improving supply chain resilience. The latter concept refers to the ability of the supply chain to provide sustained continuity in the presence of disruptions. In this regard, Li et al. (2022) show that by supporting transparency, traceability, and confidence in the use of data, blockchain technology proves to be a powerful catalyst for resilience. Also, Omar et al. (2021) investigated the contribution of blockchain technology and smart contracts for managing the healthcare supply chain during major disruptions (such as COVID-19). They proposed a solution that integrates blockchain and decentralized storage technologies to promote collaboration, transparency, data integrity among supply chain participants. In Bekrar et al. (2021), blockchain advantages are presented for various aspects of reverse logistics and transportation activities. The evaluation of this technology integration is based on its various characteristics namely: immutability, traceability, smart contract, marketplace support, tokenization, and incentivization. Alacam and Sencer (2021) present a system architecture for the transportation control tower concept using smart contracts on a blockchain network. The proposed system favors collaboration in the trucking industry. The approach is based on integration with privacy-preserving off-chain computation and storage solutions in order to provide scalability and privacy of trucking operations. The proposed system is evaluated by blockchain experts and trucking industry professionals. The study in Ahmad et al. (2021) shows how blockchain improves port logistic operations and services. Two private blockchain-based architectures are considered to map and digitize port operations and logistics management services. The authors conclude that blockchain can provide trust, security, traceability, and transparency, eliminate fraud chances and reduce time and carbon emission.

Despite some notable exceptions, shipping companies have generally been reluctant to explore the potential benefits of blockchain technology for managing and tracking containers at ports. However, one of these exceptions is COSCO, a major shipping line that has partnered with CargoSmart, Shanghai International Port Group, and Tesla to successfully conduct a pilot project that involved exchanging real-time shipping data with a terminal operator via blockchain (GlobeNewswire, 2020; Crider, 2020).

2.3. Research gaps

Most studies consider inventory optimization or flow management optimization from the viewpoint of a single entity, which can be either an importer or an exporter (Lee and Song, 2017). This approach is realistic in situations where participants do not collaborate or only have little interaction. Furthermore, it avoids the problem of high dimensionality for mathematical resolution. In contrast, we analyze the complete system, including all participants, to find solutions that are socially optimal and profitable for all participants using a tractable mathematical formulation. Such solutions are feasible thanks to the blockchain technology that does not require participants to know or trust each other.

Since its inception and use as the underlying technology of cryptocurrencies, blockchain technology has attracted the attention of researchers because of its properties of traceability, immutability, confidentiality, which qualify it to operate in trustless environments. While these characteristics are the starring features when it comes to blockchain adoption in supply chain contexts, little is acknowledged to its decentralized architecture for creating new forms of governance among parties of an ecosystem that may not have common interests, as is the case with stakeholders involved in the operations of container transportation. In fact, such decentralization of authority does not only eliminate control or economical centrality but also incentivize self-interested parties to actively take part in their respective ecosystem where fairness is provided despite trust uncertainty.

This promotes cooperation among parties as opposed to coordination, which is a crucial ingredient of centralized platforms that also necessitates minimum levels of trust. Another blockchain feature that is frequently overlooked is the consensus mechanism to validate transactions and create blocks. While the technical aspect of a consensus protocol may harness miners' computation power to work on something useful, such as the proof of useful work (PoUW) (Haouari et al., 2022), the rewarding model may considerably affect the incentive structure of the nodes maintaining the blockchain network and that of its users. This is notably true in public blockchains where the value of the blockchain network is driven by its tokenomics.

To address these gaps, this paper proposes a comprehensive blockchain solution, backed by smart contracts, where the consensus mechanism is exploited to solve the problem of optimizing the flows of empty containers and where the mining rewarding system fuels a tokenomics model that fosters cooperation among parties in a market that is driven by supply and demand. To our knowledge, this is the first work that suggests using blockchain technology to optimize empty containers flows through a fully autonomous cooperative mechanism.

3. Overview of the blockchain-based solution

To address the aforementioned research questions, we propose to implement a blockchain-based integrated solution. The proposed blockchain is used to store transactions related to the physical flows of the containers. In addition to storing container flows, the blockchain also stores all the monetary transactions that are associated with the physical flows. These monetary flows involve the payment of trucking fees, container holding fees, and the sharing of savings from optimized logistics costs between consignees and shippers. It should be noted that all these cash flows are achieved through the use of tokens.

While it might be advocated that conventional systems backed by centralized databases have been selected by default to address such optimization problems, we suggest that, in the context of the container-related application studied in this paper, blockchain technology is arguably the only technology to enable cooperation among distinct participants, which can handle the interrelated processes involved in this application.

We propose to implement a solution based on a custom blockchain. The motivation for this infrastructure is threefold and is all relevant to the unique features of blockchain. First, from a technical perspective, blockchains have proven effective when it comes to execute anonymous (or pseudonymous) information transactions and transfers of value among different parties in trustless environments (Hewa et al., 2021). Furthermore, having a single source of truth may not only promise data integrity but may also ensure transparency and visibility among different stakeholders. A requirement that is not always evidently provided by individual centralized systems operating in dynamic environments with a large number of actors players. In line with this, the decentralized nature of a blockchain ecosystem may provide a governance scheme that is easier to be accepted by parties that may not necessarily coordinate or collaborate under a centralized model (Berdika et al., 2021). Indeed, the problem at hand includes several parties that are competitive in nature and driven by self-oriented goals. A well-designed blockchain solution therefore can provide an incentive mechanism that, while respecting individual competitive advantages, may lead to global optimization.

Second, recent features in blockchains to recycle the computational power to validate transactions and mine new blocks can be exploited to solve the underlying optimization problems. This concept, referred to as PoUW, is a variant of the traditional PoW mechanism used by other blockchains such as Bitcoin and Ethereum (at the time of this writing). Unlike the PoW, the PoUW joins hardness to usefulness by solving $\mathcal{NP}$-hard problems of practical interest without waste of energy (Ball et al., 2017; Haouari et al., 2022). The blockchain infrastructure we exploit harnesses the PoUW by using transactions validated in previous blocks as input to optimization problems. These are related to the management of container flows in the hinterland supply chain, and are described next in Section 4. Best found solutions, along with the hash of the previous block, are used when mining new blocks.

Third, while the proposed blockchain is primarily used to store transactions related to the physical transportation of containers, it employs a smart contract to handle all the monetary transactions that are associated with the transportation flows. These monetary flows involve the payment of trucking fees, container-holding fees, and the sharing of savings from optimized logistics costs between consignees and shippers. It should be noted that all these cash flows are achieved using tokens or a cryptocurrency generated exclusively for this application. Further technical processes of the blockchain solution will be described in Section 6.

4. Optimization problems for containers movements in hinterland

The proposed blockchain solution operates within two interrelated processes. The first process involves managing the movement of empty containers, while the second process focuses on routing container trucks to ensure that the resulting solution of the first process is executed efficiently. The analysis of the two processes allows to address the two research questions RQ1 and RQ2, respectively.

For the first process, we formulate an optimization model for matching consignees and shippers with the objective of minimizing the total cost of transportation and holding the empty containers while satisfying certain constraints including their number, their size, and their temporal compatibility. Since the cooperative scheme generates savings compared to the traditional strategy, we develop a profit sharing mechanism allowing fairness and security between all participants of this first process. The mathematical approach is detailed in Section 4.1.

In the second process, we aim to further optimize the efficiency of trucking companies by also considering loaded containers. This approach enables us to develop solutions that are more effective for the trucking companies. Specifically, we develop an optimization model for the container truck routing problem (CTRP), which maximizes the profit margin over a specified time interval while accommodating various constraints related to container numbers and types. The proposed model determines the set of truck routes to be achieved by alternating empty and loaded trips. The mathematical approach is given in Section 4.2.

4.1. An optimization model for matching consignees and shippers

We assume that the blockchain is used to store all container movements. It therefore includes up-to-date data on the supply and demand of empty containers. Specifically, each consignee has an associated supply of empty containers of specific sizes and their corresponding availability (i.e., release) dates. Similarly, each shipper has an associated demand for empty containers of specific sizes and their corresponding due dates.

Each consignee can either return the empty containers directly to the shipping line's CY or send them to a shipper that has expressed a need for them. Of course, this option is available only if the consignee's release date is compatible with the shipper's due date. Thus, a shipper's request can be fulfilled either directly by the CY or by a consignee. It is worth reminding that holding a container beyond a specific number of days will result in penalties that will be paid to the shipping line.

To discourage customers from holding empty containers for too long, and thus causing potential disruptions (dos Santos and Borenstein, 2022), shipping lines often charge a unit daily holding penalty that is a staircase function with strictly increasing steps. Consequently, the unit holding cost is piecewise convex function of the holding time.

The problem amounts to finding a time-feasible matching between consignees and shippers with the objective of minimizing the overall transportation and holding costs. In the sequel, we will present the notation and the proposed formulation. We highlight that the model is defined per shipping line and per type of container.

4.1.1. Formulation of the matching problem

Notation. *We denote by T the planning horizon (in days) and $G = (V, A)$ the underlying graph, which is defined on a set of nodes V that is the union of the following three sets:*

$\mathcal{S}$ *set of consignee nodes. Each node $i \in \mathcal{S}$ is characterized by*
- a_i *number of containers supplied by node i;*
- r_i *release date for containers in node i.*

$\mathcal{D}$ *set of shipper nodes. Each node $j \in \mathcal{D}$ is characterized by*
- b_j *number of containers requested by node j;*
- d_j *due date for containers in node j.*

$\mathcal{P}$ *set of T nodes that are associated to the CY. Each node $t \in \mathcal{P}$ corresponds to the inventory of empty containers at the CY at a given date t.*

Furthermore, four types of arcs are accounted in our formulation as illustrated in Fig. 2. Clearly, an arc is defined only and only if the time conditions are met. Hence, the set of arcs $\mathbf{A}$ is defined as the union of the four sets of arcs $\mathbf{A_1}$, $\mathbf{A_2}$, $\mathbf{A_3}$, and $\mathbf{A_4}$ defined as follows:

Set $\mathbf{A_1}$: $(i, j) \in \mathcal{S} \times \mathcal{D}$ is an arc in A_1 if and only if

$$r_i + \delta_{ij} \leq d_j, \tag{1}$$

where δ_{ij} denotes the transportation time between a consignee i ($i \in \mathcal{S}$) and a shipper j ($j \in \mathcal{D}$).

Set $\mathbf{A_2}$: $(i, t) \in \mathcal{S} \times \mathcal{P}$ is an arc in A_2 if and only if

$$r_i + \delta_{i0} = t, \tag{2}$$

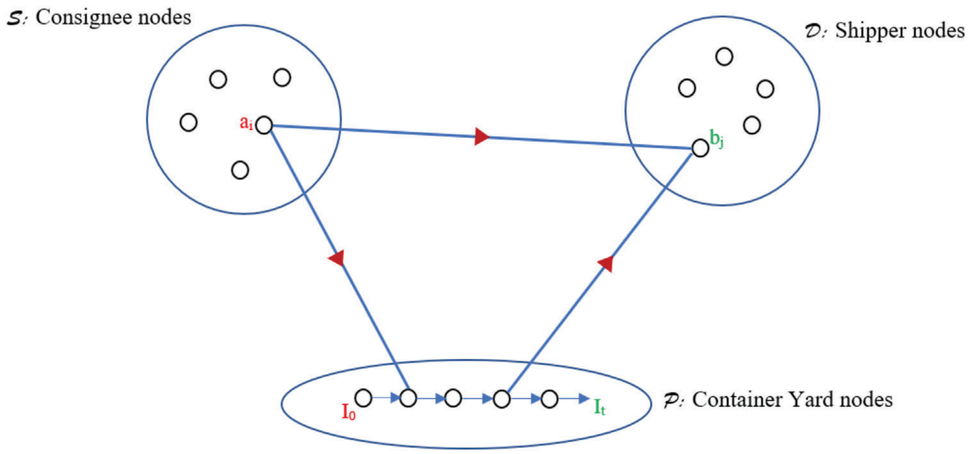


Fig. 2. Four types of flows when introducing cooperation between consignees and shippers.

Table 1
Problem parameters

Parameter	Description
I_0	Inventory level of empty containers at the CY at the beginning of the planning horizon
c_{i0}	Unit transportation cost from a consignee to the CY (unit cost for each arc in $\mathbf{A_2}$)
c_{0j}	Unit transportation cost from the CY to the shipper (unit cost for each arc in $\mathbf{A_3}$)
c_{ij}	Unit transportation and holding cost of a container that is supplied from consignee i to shipper j (unit cost for each arc in $\mathbf{A_1}$)
$g_i(.)$	Unit holding cost of an empty container at node i ($i \in \mathcal{S} \cup \mathcal{D}$). It is equal to the sum of the storage cost (which is node dependent) and the penalty paid to the shipping line. It is worth reminding that the holding cost is a convex piecewise linear function as detailed above.

where δ_{i0} is the transportation time between a consignee i ($i \in \mathcal{S}$) and the CY. Equation (2) means that the containers are transported on date r_i from consignee i to the CY where they are delivered on day t.

Set $\mathbf{A_3}$: $(t, j) \in \mathcal{P} \times \mathcal{D}$ is an arc in A_3 if and only if

$$t + \delta_{0j} = d_j, \tag{3}$$

where δ_{0j} defines the transportation time between the CY and a shipper j ($j \in \mathcal{D}$). Equation (3) means that the containers are transported from the CY to shipper j so that they made available on date d_j.

Set $\mathbf{A_4}$: This is defined by the inventory arcs: $(t, t+1) \in \mathcal{P} \times \mathcal{P}$, for $t = 1, \cdots, T-1$.

The problem parameters are presented in Table 1.

Throughout this paper, all data and processes are assumed to be deterministic. Furthermore, we assume that the shipping company is the owner of the containers and assumes full responsibility for (possibly) damaged containers.

The unit cost for each inventory arc in $\mathbf{A_4}$ is zero since neither transportation fees nor holding fees are charged within the port.

The unit cost of a flow between consignee $i \in \mathcal{S}$ and shipper $j \in \mathcal{D}$ includes (a) the transportation cost denoted as γ_{ij} and (b) the holding cost denoted as s_{ij}. Thus, the cost of (i, j) in $\mathbf{A_1}$ is given by

$$c_{ij} = \gamma_{ij} + s_{ij}, \quad (i, j) \in \mathcal{S} \times \mathcal{D}. \tag{4}$$

Let $\tau_{ij} = d_j - \delta_{ij} - r_i$ denote the holding time of a container between a consignee i and a shipper j. If θ defines the holding time of the empty container at consignee i, the remaining time during which it will be held by shipper j is then $\tau_{ij} - \theta$. Thus, the holding cost s_{ij} is defined by

$$s_{ij} = \min_{\theta \in [0, \tau_{ij}]} \quad g_{ij}(\theta), \tag{5}$$

where $g_{ij}(\theta) = g_i(\theta) + g_j(\tau_{ij} - \theta)$.

By observing that

$g_i(\theta)$ is an increasing piecewise linear function and the slopes are strictly increasing, and thereby is convex;

$g_j(\tau_{ij} - \theta)$ is a decreasing piecewise linear function and the slopes are strictly increasing, and thereby is convex.

We infer that $g_{ij}(\cdot)$ is convex.

Proposition 1. *The holding cost is a convex function of θ.*

Consequently, a global optimum θ^* can be easily computed using any unidimensional optimization algorithm such as the Golden search optimization method. Interestingly, in the current special case, where $g(\cdot)$ is a piecewise linear function, the optimal holding duration can be trivially computed using the following result.

Proposition 2. *The optimal duration for holding a container at a consignee is a breakpoint of the holding function g.*

Proof. See Appendix A. □

Decision variables.

x_{ij} flow between node i and node j
I_t inventory level of empty containers at the CY on date t

We propose the following mathematical programming model to find the optimal matching between consignees and shippers:

$$\text{(M): Minimize} \sum_{(i,j)\in A} c_{ij} x_{ij}, \tag{6}$$

$$\text{subject to} \sum_{j:(i,j)\in A_1\cup A_2} x_{ij} = a_i, \quad \forall i \in \mathcal{S}, \tag{7}$$

$$\sum_{i:(i,j)\in A_1\cup A_3} x_{ij} = b_j, \quad \forall j \in \mathcal{D}, \tag{8}$$

$$I_{t-1} + \sum_{i:(i,t)\in A_2} x_{it} - \sum_{j:(t,j)\in A_3} x_{tj} - I_t = 0, \quad t = 1, \cdots, T, \tag{9}$$

$$I_t, x_{ij} \geq 0. \tag{10}$$

The objective function (6) amounts to minimizing the total transportation and holding costs. Constraints (7) and (8) are the supply and demand constraints, respectively. Constraint (9) enforces the flow conservation of the CY's inventory.

We denote $\bar{x}$ and $\bar{I}$ the resulting optimal solution of the linear program (M) and $\bar{C}$ the global cost after cooperation given by

$$\bar{C} = \sum_{(i,j)\in A} c_{ij}\bar{x}_{ij}. \tag{11}$$

Proposition 3. *Problem (M) can be restated as a minimum-cost flow problem and therefore an (integer) optimal solution can be obtained in polynomial-time.*

Proof. See Appendix B. □

Remark. For the sake of simplicity and clarity, we have so far assumed that the consignees' and shippers' warehouses have sufficient capacity to hold all their empty containers. In case where a holding capacity is imposed, the model is adjusted as follows. We assume that each consignee i and each shipper j have a holding capacity denoted by u_i and u_j, respectively. In this case, we observe that there are at most $\min(u_i, u_j)$ containers that might be held either by i or j. In addition, there are at most $|u_i - u_j|$ containers that can be held by the participant with the largest capacity. Accordingly, each flow x_{ij} can be split in two flows as follows:

$$x_{ij} = x_{ij}^1 + x_{ij}^2, \quad (i, j) \in \mathcal{S} \times \mathcal{D}, \tag{12}$$

where x_{ij}^1 represents the number of containers that could be held at either location, and x_{ij}^2 represents the additional number of containers that could be held only at the participant with the largest capacity.

Hence, the capacitated variant of Model (M) requires substituting for each arc $(i, j) \in \mathcal{S} \times \mathcal{D}$, variable x_{ij} by variables x_{ij}^1 and x_{ij}^2 such that

$$0 \leq x_{ij}^1 \leq \min(u_i, u_j), \quad (i, j) \in \mathcal{S} \times \mathcal{D}, \tag{13}$$

$$0 \le x_{ij}^2 \le |u_i - u_j|, \quad (i, j) \in \mathcal{S} \times \mathcal{D}. \tag{14}$$

The corresponding costs for x_{ij}^1 and x_{ij}^2 are as follows:

$$c_{ij}^1 = \gamma_{ij} + s_{ij}, \quad (i, j) \in \mathcal{S} \times \mathcal{D}, \tag{15}$$

$$c_{ij}^2 = \gamma_{ij} + g_k(\tau_{ij}), \quad (i, j) \in \mathcal{S} \times \mathcal{D}, \text{ with } k = i \text{ if } u_i \ge u_j, \text{ and } j \text{ otherwise.} \tag{16}$$

4.1.2. Profit sharing

We denote by Δ the savings that results from cooperation. Thus, it is given by

$$\Delta = \frac{C^* - \bar{C}}{C^*}, \tag{17}$$

where C^* is the global cost before cooperation and whose expression is given by

$$C^* = \sum_{i \in \mathcal{S}} C_i^* + \sum_{j \in \mathcal{D}} C_j^*, \tag{18}$$

with C_i^* and C_j^* correspond to the transportation cost of consignee $i \in \mathcal{S}$ and shipper $j \in \mathcal{D}$, respectively. They are determined as follows:

$$C_i^* = c_{i0} a_i, \quad i \in \mathcal{S}, \tag{19}$$

$$C_j^* = c_{0j} b_j, \quad j \in \mathcal{D}. \tag{20}$$

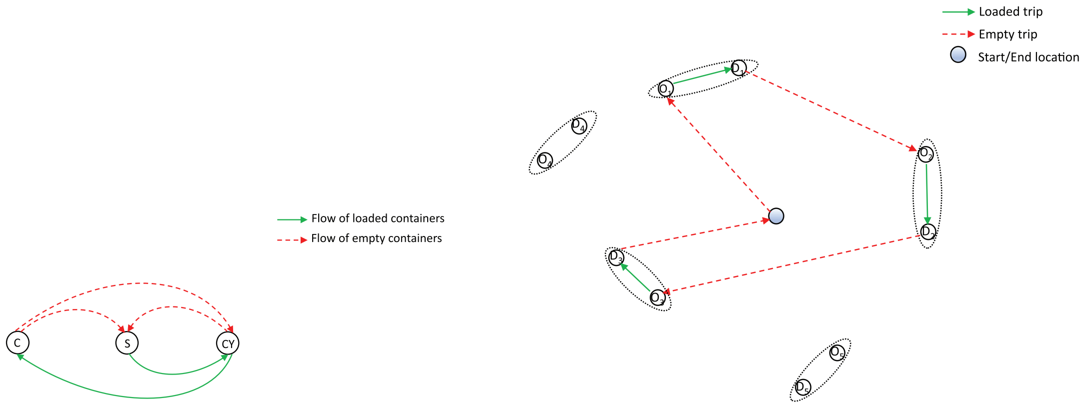
For the sake of fairness, each consignee and each shipper will be granted the same discount of $100\Delta\%$ on its logistics cost. That is, the actual cost that is paid by participating company k is $\bar{C}_k = (1 - \Delta) C_k^*$, for $k \in \mathcal{S} \cup \mathcal{D}$.

The individual holding cost per unit of container for consignee i (resp., shipper j) is $g_i(\theta^*)$ (resp., $g_j(\tau_{ij} - \theta^*)$). Thus, the holding cost for each participant is given by

$$h_i = \sum_{j:(i,j) \in A_1} g_i(\theta^*) \bar{x}_{ij}, \quad \forall i \in \mathcal{S}, \tag{21}$$

$$h_j = \sum_{i:(i,j) \in A_1} g_j(\tau_{ij} - \theta^*) \bar{x}_{ij}, \quad \forall j \in \mathcal{D}. \tag{22}$$

For each participating company $k \in \mathcal{S} \cup \mathcal{D}$, if $\bar{C}_k > h_k$ then it will pay a net amount that is equal to the difference $(\bar{C}_k - h_k)$. Otherwise, k's account will be credited a net amount that is equal to the excess $(h_k - \bar{C}_k)$. In so doing, companies that pay an excessive holding cost resulting from the



(a) The five types of trips. (b) Example of a cycle of trips from the start to the end location.

Fig. 3. Types of trips and a feasible solution of trips.

cooperation strategy will be compensated. As a final result, each shipper and consignee will take advantage of a discount of 100Δ%.

In conclusion, this section highlights that the proposed optimization model enables the computation of a minimum-cost empty container allocation strategy. Additionally, the profit sharing scheme ensures the fair distribution of profits among all participants. Together, these findings provide a comprehensive solution to RQ1.

4.2. *Optimal container truck routing*

After solving model (M), we refer to the resulting flows between consignees, shippers and CY, as TRs of empty containers. Therefore, the set of TRs include not only trips with loaded containers, but also those with empty containers. In this section, we will focus on the resulting CTRP.

Trucking companies, as blockchain participants, will perform trips to transport containers from different origins to their respective destinations, both loaded and empty (e.g., after transporting empty containers to a shipper, the latter needs to move loaded containers to the CY for exportation). Therefore, the truck routing problem that arises in this context includes not only the empty container flows as ascribed by the solution of model (M) but also all associated loaded container flows. Accordingly, we consider five types of TRs as detailed in Fig. 3a (C: consignee; S: shipper; P: CY).

All requests (empty/loaded containers) are then stored in details (origin, destination, time, number and type of containers, etc.) in the blockchain. Trucking companies participating in the blockchain can use the information stored in the blockchain to select the most profitable set of routes (using the optimization procedure developed below) and complete them in the allotted time (Fig. 3b). We are considering the two most frequently used container types by shipping lines: the 40-ft container (Type 1) and two 20-ft containers (Type 2). The trucking companies operate long trucks capable of carrying either one 40-ft container or two 20-ft containers, as well as short trucks that can only transport one 20-ft container. It is noteworthy that the proposed model will not only

Table 2
Trips requests characteristics

Characteristic	*Description*
O_i	*Origin of trip i*
D_i	*Destination of trip i*
ν_i	*Scheduled departure time of trip i*
μ_i	*Total trip i duration from O_i to D_i*
p_i	*Prize (i.e., revenue) of trip i*
q_i^1	*Number of Type 1 containers of trip i*
q_i^2	*Number of Type 2 containers of trip i*
c_{ij}^{α}	*Cost of the loaded trip i and the empty trip from i to j. The first term is zero if $i = 0$ (i.e., $c_{0j}^{\alpha} = 0$).*

improve the profitability of trucking companies but will also lead to a reduction in CO_2 emissions by decreasing the number of empty trips, which has become a major objective of vehicle route planners (Erdoğdu and Karabulut, 2022).

In this section, we propose to develop an optimization approach to find a set of truck routes that alternate empty and loaded trips, with the objective of maximizing the profit margin over a specified time interval.

Notation. *Let us denote the trips requests (TRs) by $i = 1, \cdots, n$ (with n is the total number of TRs) where each TR_i has a set of characteristics which are presented in Table 2.*

Furthermore, we denote by m_1 and m_2 the available number of long and short trucks, respectively. We build a graph $\bar{G} = (\bar{V}, \bar{A})$ where $\bar{V} = \{0, 1, \cdots, n\}$ is the set of nodes, 0 represents the start/end location of the vehicles, and $\bar{A}$ is the set of arcs. An arc (i, j) is defined if and only if

$$\nu_i + \mu_i + \nu_{ij} \leq \nu_j, \quad (i, j) \in \bar{A}, \tag{23}$$

where ν_{ij} is the duration of trip from D_i to O_j. The arc cost is the cost of the corresponding empty trip.

Decision variables.

z_{ij}^{α} number of vehicles of type α ($\alpha = 1, 2$) that achieve trip j immediately after trip i (integer)
y_i binary variable which equals 1 if trip i is selected, and 0 otherwise

The CTRP consists of selecting a set of loaded trips to be made so as to maximize the profit margin, which provides a measure of the amount of profit generated by a firm's revenues. Therefore, it is formulated as follows:

$$\text{(CTRP): Maximize} \quad \frac{\sum_{i=1}^{n} p_i y_i - \sum_{\alpha=1}^{2} \sum_{(i,j)\in\bar{A}} c_{ij}^{\alpha} z_{ij}^{\alpha}}{\sum_{i=1}^{n} p_i y_i}, \tag{24}$$

$$\text{subject to} \quad \sum_{(0,j)\in\bar{A}} z_{0j}^{\alpha} \le m_{\alpha}, \qquad \alpha = 1, 2 \tag{25}$$

$$\sum_{j:(i,j)\in\bar{A}} z_{ij}^{\alpha} - \sum_{k:(k,i)\in\bar{A}} z_{ki}^{\alpha} = 0, \qquad \alpha = 1, 2, i = 0, 1, \cdots, n, \tag{26}$$

$$\sum_{j:(j,i)\in\bar{A}} z_{ji}^{2} \le q_i^2 y_i, \qquad i = 1, \cdots, n, \tag{27}$$

$$2\sum_{j:(j,i)\in\bar{A}} z_{ji}^{1} + \sum_{j:(j,i)\in\bar{A}} z_{ji}^{2} \ge \left(2q_i^1 + q_i^2\right) y_i, \quad i = 1, \cdots, n, \tag{28}$$

$$2\sum_{j:(j,i)\in\bar{A}} z_{ji}^{1} + \sum_{j:(j,i)\in\bar{A}} z_{ji}^{2} \le \left(2q_i^1 + q_i^2 + 1\right) y_i, \qquad i = 1, \cdots, n, \tag{29}$$

$$\sum_{i=1}^{n} y_i \ge 1, \tag{30}$$

$$z_{ij}^{\alpha} \in \mathbb{N}, \qquad \alpha = 1, 2, (i, j) \in \bar{A}, \tag{31}$$

$$y_i = \{0, 1\}, \quad i = 1, \cdots, n. \tag{32}$$

The objective function (24) amounts to maximizing the profit margin, which is defined as the ratio of net worth to total revenue. The net worth is equal to the sum of the profits associated with the selected trips minus the cost of the empty trips. Constraint (25) enforces that the number of trucks of each type leaving the trucking company's depot must not exceed the number of available trucks. Constraint (26) specifies that the flow conservation constraint is satisfied at each node of the graph. Constraint (27) enforces that if trip i is selected, then the number of short trucks assigned to i should not exceed the number of Type 2 containers. Constraints (28) and (29) require that the number and mix of trucks assigned to route i meet the demand for each container type. Constraint (30) is appended in order to prevent the null solution ($y_i = 0$ for all $i = 1, \cdots, n$) that would otherwise lead to an indefinite objective. Finally, constraints (31) and (32) identify the type of the decision variables. A formal proof of the validity of constraints (28) and (29) is provided in Appendix C.

4.2.1. Complexity of the CTRP

Interestingly, the CTRP can be viewed as a variant of the prize collecting vehicle routing problem (PCVRP). The latter problem involves finding a set of routes that starts and ends at the depot, and visiting an (unknown) subset of nodes. A specific prize is collected at each node visit, and a cost is incurred for each arc traversed. The objective is to find the most profitable set of routes. The PCVRP, which is known to be $\mathcal{NP}$-hard, has been intensely investigated in the literature (Tang and Wang, 2006; Stenger et al., 2013).

Unlike the PCVRP, the CTRP has, to our knowledge, never been studied in the literature. The following proposition shows that the CTRP cannot be solved by a polynomial algorithm unless $\mathcal{P} = \mathcal{NP}$.

Proposition 4. *The CTRP is $\mathcal{NP}$-hard.*

Proof. See Appendix D. □

An important consequence of the $\mathcal{NP}$-hardness of the CTRP is that it can be used as PoUW to validate transactions and maintain highly secure immutability of the blockchain. Indeed, finding a container truck routing solution with an objective greater than a threshold value $\kappa \geq 0$ would require an intensive computation. On the other hand, since it is in Class $\mathcal{NP}$, it is then possible to check in polynomial time for a given solution whether the corresponding objective is greater than a threshold value.

In conclusion, the CTRP solution enables the selection of the most profitable set of truck routes, providing an answer to RQ2.

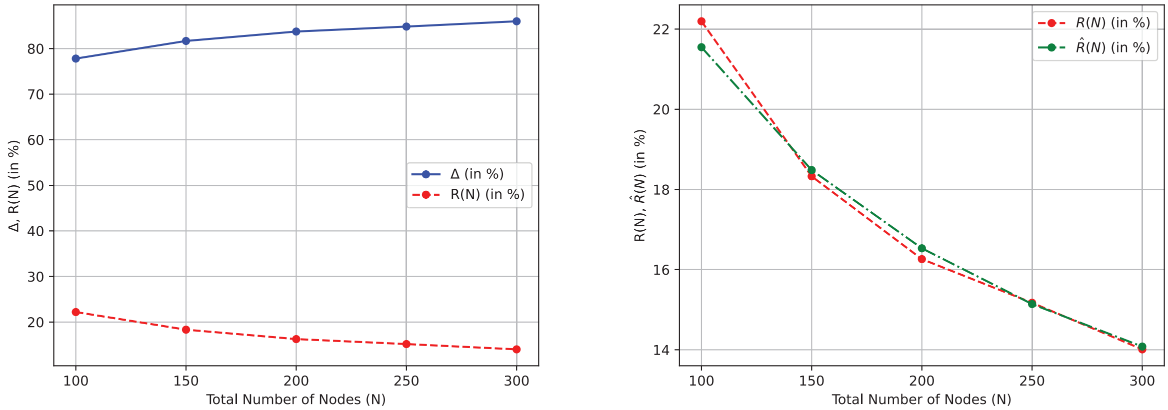
5. Computational experiments

We carried out a computational study to assess the empirical performance of the matching procedure. All simulations were implemented with CPLEX Python API (CPLEX Version 20.10 and Python Version 3.8.7) and run on a machine with a processor 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80 and 1.69 GHz, and 16 GB of RAM.

We generated random instances using the following settings.

- $T = 15$ days;
- a number of N nodes (including both shippers and consignees) are uniformly generated in the square $[0, 100]^2$; and the CY is located in (0,0);
- a node is assumed to be a consignee with probability p;
- the transportation costs are equal to the Euclidean distances;
- the supply and demand levels are integers randomly drawn from $U[1, 10]$;
- the transportation times are integers randomly drawn from $U[1, 2]$;
- the consignee release dates are integers randomly drawn from $U[1, T-2]$;
- the shipper due dates are integers randomly drawn from $U[3, T]$;
- the unit holding cost function includes four increasing steps that are integers randomly drawn from $U[1, 13]$;
- the initial inventory at the CY is equal to the total shippers' demand.

For each generated instance, we solved the corresponding matching problem, and computed the realized saving rate Δ. In this regard, we carried out two experiments to analyze the sensitivity of Δ to the number of nodes N, and the probability p, respectively. It should be noted that we have observed that the computation times are very short (even for large instances). Therefore, we will not report computation times in this section.



(a) The profit Δ is an increasing function of the total number of nodes (N) (the cooperation ratio is a decreasing function of N).

(b) The ratio estimate converges to the observed ratio.

Fig. 4. The profit and the ratio estimate as a function of N.

5.1. Sensitivity to N

We set $p = 0.5$, and randomly generated instances for different values of N ranging from 100 to 300. For each value of N, we generated 50 instances and computed the resulting average. The results are displayed in Fig. 4a. This figure shows that the savings ratio increases as the number of participants increases. This could be explained by the fact that the larger the number of participants and the more the opportunities for finding time-compatible pairs of consignees and shippers, the more are opportunities to achieve profitable matchings. Interestingly, using mild assumptions, it is possible to derive an estimate of the expected saving ratio as a function of the number of participants. Indeed, define $R(N)$ as the ratio of the total cost after cooperation between N participants to the cost before cooperation. An estimate of the expected value of $R(N)$ is given by

$$\widehat{R}(N) = \lambda\sqrt{\frac{\ln\frac{N}{2}}{N}}, \tag{33}$$

where $\lambda = 1.089$ is a constant that is empirically estimated by minimizing the sum of quadratic errors. The justification of (33) is provided in Appendix E. In our experiments, we found that the estimate value is very accurate as it displays an average deviation of $5.22 \times 10^{-3}\%$. In Fig. 4b, we display the observed curve of $R(N)$ along with $\widehat{R}(N)$.

5.2. Sensitivity to p

We set $N = 50$, and randomly generated instances for different values of p ranging from 0.1 to 0.9. The results are displayed in Fig. 5. From this figure, we see that the savings ratio function is

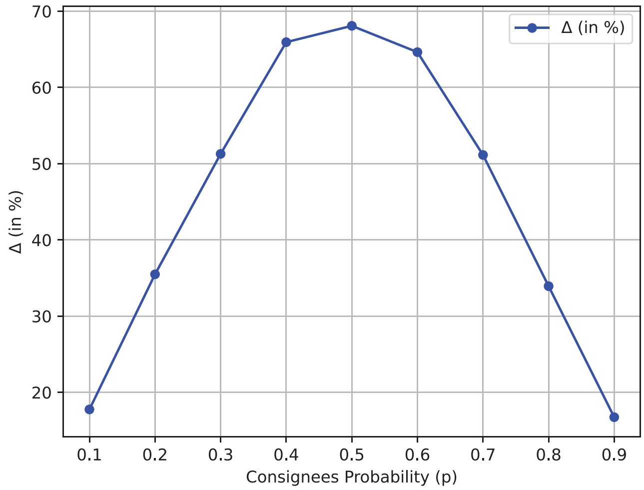


Fig. 5. The profit Δ is a concave function of the consignees probability.

bell-shaped with a maximum value for $p = 0.5$. This behavior could be explained by the fact that balanced instances (i.e., having similar numbers of consignees and shippers) offer the largest opportunities for finding mutually beneficial matching. In contrast, when the considered hinterland mostly include importing (exporting) companies, that is, when the number of consignees (shippers) is much larger than the number of shippers (consignees), then it is expected that most empty containers will be returned to (supplied from) the CY.

5.3. Practical implications

Our computational experiments prove two important implications. The first one is regarding the saving rate that increases with the number of participants. From a practical standpoint, when a consignee or a shipper becomes a new blockchain participant, his involvement will enhance each individual saving rate. The second implication is with regard to the most profitable instances, which happens when the numbers of consignees and shippers are balanced. From a practical perspective, the best beneficial matching occurs when there are as many consignees as shippers.

6. Description of the blockchain

As stated in Section 3, the two proposed optimization models are integrated in the proposed blockchain to define accordingly the PoUW. Hence, the technical description of the employed blockchain along with the overall processes for planning, executing, and payment settlement is presented in this section. At this point, it is important to clarify that in order to keep the article to a reasonable length, we will not delve into some very technical aspects such as the tokenomics of the proposed blockchain and the security issues related to forking, network latency.

6.1. Proof of useful work

The PoUW concept is first developed in Ball et al. (2017) where the hardness of computational problems is turned into useful solutions of practical interest while saving energy. Instead of having the mining nodes competing to find a random nonce that satisfies the value of a certain hash (i.e., number of leading zeros of a hash value), miners in PoUW work on exploring the solution space of a given $\mathcal{NP}$-hard problem. In fact, blockchains using PoUW (Haouari et al., 2022) are not only considered tamper-proof distributed ledgers; they transform the blockchain network into a highly distributed computer that increases in performance with network size. The more nodes join the network the more secure it becomes and the more the solution space is explored yielding better solutions. Given its relevance, PoUW recently captivates the attention of researchers. In this context, Loe and Quaglia (2018) developed a novel PoUW by constructing an instance of the $\mathcal{NP}$-hard travelling salesman problem. The proposed approach provides a fiscally incentivized platform for algorithm research whose purpose is to optimize an $\mathcal{NP}$-hard computational problem. In Baldominos and Saez (2019), a PoUW scheme is proposed to support a blockchain cryptocurrency network. The mining process is based on artificial intelligence (AI) and deep learning models by doing neural architecture search for a given AI problem requiring a machine learning model for training and evaluation. Recently, several studies focus on the PoUW and develop various approaches. Haouari et al. (2022) propose a novel PoUW which requires solving an $\mathcal{NP}$-hard optimization problem related to the context of maritime transportation. The objective is to minimize the total cost of the transportation requests between same origin port and same destination port. Todorović et al. (2022) propose a new consensus protocol based on the PoUW concept and which assumes solving instances of real-life combinatorial optimization problems. Various heuristic methods are then developed to deal with the problems complexity and to be used in the proposed consensus protocol. Males˘ et al. (2023) develop a general framework for difficulty estimation of the useful work. The proposed approach allows controlling and balancing the work of miners while ensuring the reward fairness and the block insertion time stability. Tong et al. (2023) design a block structure for the storage of historical data in the context of cloud manufacturing. The purpose of the study is to enhance the quality of solutions for the demand of dynamic service numbers and constraints. The proposed PoUW considers a mechanism based on multiobjective service composition and optimal selection for the mining mechanism.

6.2. Mining and block validation process

To properly address the problem described in this paper in a decentralized manner and to consider its extent repercussions on the traditional mode of transportation and container flow, the proposed blockchain infrastructure, on which a holistic solution could be developed, is assumed to benefit from some mandatory features. First, the blockchain in question is to operate its own native coin as a vehicle of value to express monetary transactions between the relevant stakeholders and as a way to implement decentralized governance on the solution. For instance, the wealth of coins held by a trucking company may grant it some privileges when it comes to the allocation of transportation requests. Second, the support for smart contracts is necessary to implement some crucial parts of the problem logic. Furthermore, the ability to define and execute multiple types of transactions and an adapted consensus mechanism deem necessary too. Therefore, we propose in this paper a blockchain powered by PoUW as the consensus mechanism to create new blocks containing transaction data. In the context of the proposed blockchain, miners must successively solve the matching problem (which is easy) and then the CTRP, which is $\mathcal{NP}$-hard. Regardless of the optimization heuristic employed by the miners for solving the CTRP, each miner starts its search for solutions with a different random seed that is calculated as a function of its unique identifier and the hash of the previous block. To ensure all nodes are working on the same problem instance, the blockchain ensures all nodes are synchronized and that the operations they execute are organized in timely phases (mining, block broadcasting, block validation, and block addition phases). Accordingly, miners must first solve the matching problem, then the routing problems that are defined for all participating trucking companies in a specific order that could be determined by a certain governance model such as the amount of coins held by each trucking company. The higher the amount, the higher the order of priority to be considered first in the solution. It should be noted that the aforementioned problems are always feasible. This satisfies the solvability condition for a given problem to be used as PoUW. The second necessary condition is problem measurability. That is, the difficulty of the problem is adjustable and measurable. In our case, we guarantee measurability by adopting a strategy that is the same vein as those suggested by Baldominos and Saez (2019) and Haouari et al. (2022). After solving the matching problem and obtaining the resulting cooperation ratio $\rho_0 = \frac{\bar{C}}{C^*}$, the routing problem is solved for each trucking company sequentially. We assume that there are K participating trucking companies ($K \geq 1$). The different instances of the CTRP are solved iteratively. At iteration k, the instance that corresponds to the kth trucking company is solved taking into account all TRs that have not yet been assigned to other trucking companies in previous iterations as well as the available fleet of company k. Of course, the trucking company whose instance is solved first has the greatest opportunity to make a high profit and so on. We define ρ_k ($k = 1, \cdots, K$) as the ratio of total trips costs to total trips revenues for each company and which is given by

$$\rho_k = \frac{\sum_{\alpha=1}^{2} \sum_{(i,j)\in \bar{A}} c_{ij}^{\alpha} z_{ij}^{\alpha}}{\sum_{i=1}^{n} p_i y_i}, \quad \forall k = 1, \cdots, K. \tag{34}$$

The platform sets an initial threshold value ρ between 0 and 1. This parameter is used for controlling the practical hardness of the PoW. Miners compete to find solutions to the matching and routing problems that satisfy

$$\rho_0 \rho_1 \cdots \rho_K < \rho. \tag{35}$$

The ratios ρ_k ($k = 1, \cdots, K$) are initialized to 1. Accordingly, when all the TRs are assigned for the M first trucking companies ($M < K$), we have $\rho_0\rho_1 \cdots \rho_K = \rho_0\rho_1 \cdots \rho_M$. Clearly, minimizing ρ_k

(for $k \geq 1$) requires solving a CTRP. Hence, the hardness of the PoW can be mitigated by increasing the value of ρ. As long as there is no set of feasible solutions that satisfy the set threshold, miners can submit a request to increase the value of ρ by an amount predetermined by the consensus protocol. This request, which could happen periodically, should be signed at least by 51% of the nodes so that the value of parameter ρ is slightly increased by a constant factor. The first miner to submit a complete set of feasible solutions that satisfies the threshold condition is designated as the winner.

When a solution is found and validated by the majority of nodes on the network (i.e., by being the best solution found), it is included in the block header and hashed along with the transactions in that block before adding it to the blockchain. PoUW ensures the blockchain is tamper-proof since changing any block would require solving all optimization problems of subsequent blocks. The mining reward for successfully mining a new block may be a fixed amount (such as Bitcoin and Ethereum) that is paid to the miners using the blockchain's native cryptocurrency.

In the context of our CTRP problem, miners would apply PoUW on transactions reporting transportation requests validated in blocks over the past 30 days and that are to be fulfilled within the next 48–72 hours. The solution resulting from PoUW to mine a new block corresponds to the CTRP problem and defines an optimized route for a single trucker. Blocks are mined at a frequency that ensures all transportation requests to be executed in the following 48–72 hours are cleared and assigned to the optimal trucker.

In order to fulfill the logic of the application problem, we define four types of transactions as following:

- *trucker_details_tx*: This is submitted by truckers only. It is to specify their public address, fleet details, and constraints. This information will be part of the optimization process when assigning routes to truckers. This information is stored in the truckers map data structure of the smart contract of the genesis block.
- *transport_request_tx*: This is submitted by shippers to request trucks to transport (a) empty containers or (b) loaded containers to destination. This type of transaction is also available to consignees to request trucks (a) to transport loaded containers from a source to their location or (b) to ship away empty containers. Besides all the necessary details of the transportation request (e.g., source, destination, dates for delivery/pickup, type and number of truck(s), and/or container(s)), transactions of this type require the shipper/consignee to submit payments for the required service. Since savings on these transactions are unknown at the time of request and will only be calculated at a future block creation, full regular payment are submitted with the transaction. Such transactions should be done at least 48 hours before the service time and no earlier than 30 days. It is important to stress that since the only detail that discloses the participant identity is the provided locations (source/destination of shippers/consignees), we consider that these latter are provided to the network by region zone/sector to ensure the blockchain confidentiality.
- *trucker_payment_tx*: This is issued by shippers/consignees as a confirmation to release funds to truckers. They should be done after being served by the transportation. These transactions could be initiated either manually or through internet of things (IoT) devices. These transactions, once validated in blocks, trigger the smart contract to move the amount transferred by the shipper/consignee into the smart contract at the time of *transport_request_tx* to the trucker account, after applying the discount obtained from optimization.

Table 3
Smart contract map example

Customer	Tx ID	Cost	Payable	Approve service
C1	1	1000	800	✓
C2	2	1500	1200	✓
C3	3	800	640	✓
C4	4	500		
C5	5	800		
C6	6	1000		

- *value_transfer_tx*: This is used to move the blockchain's coin from one account to another. For instance, moving coins from a miner's account to the one of a shipper as a result of trading over a cryptocurrency exchange, for example.

The blockchain is initiated with a smart contract in its genesis block (i.e., the first block). The role of this smart contract is to keep truckers' information and to handle all payments for transportation services and the distribution of savings achieved from optimization. That smart contract keeps track of payment transactions in a map data structure as shown in Table 3. When a *transport_request_tx* submitted by a shipper or consignee is added to a block, its unique identifier and that of the submitter (i.e., public key) along with the paid amount is recorded in the payment map (first three columns). The payable amount (fourth column in light gray) is populated when the request has been included in a route as a result of PoUW optimization during the creation of a future block. The last column highlighted in dark gray denotes the delivery of service and is updated as a result of executing a *trucker_payment_tx*. At this point, funds are moved from the smart contract account to that of the respective trucker.

At 30-day intervals, the smart contract calculates the savings rate achieved from optimizing route costs and transfers the corresponding amount back to customers' accounts (shippers/consignees) before it clears the map to start a new 30-day round.

A detailed description of the blockchain process is provided in Appendix F.

7. Conclusion and future research

In this paper, we addressed the topic of optimizing hinterland maritime container flows. To this end, we proposed an integrated blockchain-based framework to optimize container flows in the hinterland. The ultimate benefit of the proposed blockchain is that it enables consignees and shippers in the hinterland to make direct shipments of empty containers from consignees to shippers without going through the shipping line's CY, thereby optimizing their container-related logistics costs. In addition, it enables trucking companies to efficiently allocate their fleets to routes requested by consignees and shippers. The proposed cooperation strategy was made possible by implementing the PoUW concept where $\mathcal{NP}$-hard problems, directly beneficial to the optimization of participants' operations, are solved by anonymous miners for the validation of transaction blocks. The proposed blockchain manages not only container flows but also all monetary flows between participants. In particular, it ensures the fair sharing of savings among all participants as well as payments from trucking companies.

We have presented the results of computer experiments that prove that the savings are significant if the number of participants is large and if, in the hinterland under consideration, the supply of empty containers is balanced with the demand for them. In this case, the implementation of the proposed blockchain-based solution should transform the hinterland container supply chain ecosystem.

It is noteworthy that the proposed blockchain-based solution exhibits two limitations that can be considered in future studies. First, the processes were assumed to be deterministic and fully reliable, whereas in practice they can be subject to various random disruptions, such as trucks failing to collect or deliver containers on time, unpredictable travel times, and shippers not making containers available on the promised dates. Second, it was assumed that the blockchain is entirely secure. However, in reality, several security issues may arise when implementing such a platform, particularly in a public blockchain context. These issues include the confidentiality of transaction data belonging to stakeholders, cyberattacks targeting the platform, and the identification of malicious nodes or actors engaging in adversary behavior. These concerns are all critical areas that require thorough investigation.

Furthermore, we recommend two interesting research avenues that are worth exploring for future studies. The first relates to the tokenomics structure of the blockchain coin. This paper highlights the various uses of the coin that can leverage its value. While it primarily serves as a mode of payment among stakeholders in the network, it can also grant its holder some extra advantages (e.g., the order of inclusion of a trucking company in a solution worked by miners). It can also act as a governance token, leading the overall platform to follow a decentralized autonomous organization model. This creates a certain demand for the coin for different purposes that may differ from one holder to another. Therefore, issues around the economics of the coin, such as initial pricing, rate of creation by miners, demand and supply, and fluctuation are all essential factors worth investigating.

Second, implementing and testing an actual prototype of the concepts and design presented in this paper and exploring possibilities for applying such cooperative blockchains to other contexts is a valuable area for further research.

Acknowledgments

This research was made possible by NPRPC grant no. NPRP11C-1229-170007 from the Qatar National Research Fund (a member of The Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

Ahmad, R.W., Hasan, H., Jayaraman, R., Salah, K., Omar, M., 2021. Blockchain applications and architectures for port operations and logistics management. *Research in Transportation Business & Management* 41, 100620.

Ajtai, M., Komlós, J., Tusnády, G., 1984. On optimal matchings. *Combinatorica* 4, 259–264.

Alacam, S., Sencer, A., 2021. Using blockchain technology to foster collaboration among shippers and carriers in the trucking industry: a design science research approach. *Logistics* 5, 2, 37.

Astarita, V., Giofrè, V.P., Mirabelli, G., Solina, V., 2020. A review of blockchain-based systems in transportation. *Information* 11, 1, 21.

Baldominos, A., Saez, Y., 2019. Coin.AI: a proof-of-useful-work scheme for blockchain-based distributed deep learning. *Entropy* 21, 8, 1–17.

Ball, M., Rosen, A., Sabin, M., Vasudevan, P.N., 2017. Proofs of useful work. *International Association for Cryptologic Research* 1–28. https://eprint.iacr.org/2017/203.pdf.

Bekrar, A., Cadi, A.A.E., Todosijevic, R., Sarkis, J., 2021. Digitalizing the closing-of-the-loop for supply chains: a transportation and blockchain perspective. *Sustainability* 13, 5, 2895.

Berdika, D., Otoum, S., Schmidta, N., Portera, D., Jararweh, Y., 2021. A survey on blockchain for information systems management and security. *Information Processing & Management* 58, 1, 102397.

Bouchery, Y., Fazi, S., Fransoo, J.C., 2015. Hinterland transportation in container supply chains. In Lee, C.Y., Meng, Q. (eds) *Handbook of Ocean Container Transport Logistics: Making Global Supply Chains Effective*. Springer, Berlin, pp. 497–520.

Braekers, K., Janssens, G., Caris, A., 2011. Challenges in managing empty container movements at multiple planning levels. *Transport Reviews* 31, 6, 681–708.

Chen, Y., Yang, B., 2022. Analysis on the evolution of shipping logistics service supply chain market structure under the application of blockchain technology. *Advanced Engineering Informatics* 53, 101714.

Choi, B.C., Lee, K., Leung, J.Y.T., Pinedo, M.L., Briskorn, D., 2012. Container scheduling: complexity and algorithms. *Production and Operations Management* 21, 1, 115–128.

Choong, S.T., Cole, M.H., Kutanoglu, E., 2002. Empty container management for intermodal transportation networks. *Transportation Research Part E: Logistics and Transportation Review* 38, 6, 423–438.

Costa, M.D.G., Captivo, M.E., 2016. Weight distribution in container loading: a case study. *International Transactions in Operational Research* 23, 1-2, 239–263.

Crider, J., 2020. Tesla participated in a pilot using blockchain technology to speed up China import process. Available at https://cleantechnica.com.

Dang, Q., Nielsen, I., Yun, W., 2013. Replenishment policies for empty containers in an inland multi-depot system. *Maritime Economics & Logistics* 15, 1, 120–149.

Dang, Q., Yun, W., Kopfer, H., 2012. Positioning empty containers under dependent demand process. *Computers & Industrial Engineering* 62, 3, 708–715.

Dejax, P., Crainic, T., 1987. Survey paper—a review of empty flows and fleet management models in freight transportation. *Transportation Science* 21, 4, 227–248.

Deplano, I., Lersteau, C., Nguyen, T.T., 2021. A mixed-integer linear model for the multiple heterogeneous knapsack problem with realistic container loading constraints and bins' priority. *International Transactions in Operational Research* 28, 6, 3244–3275.

Drewry Supply Chain Advisors, 2018. Invoicing and payment processes in global container shipping: ready for disruption? Technical Report. Available at https://www.cadenadesuministro.es/wp-content/uploads/2018/05/Drewry-Facturacion-y-Procesos-de-Pago.pdf.

Erdoğdu, K., Karabulut, K., 2022. Bi-objective green vehicle routing problem. *International Transactions in Operational Research* 29, 3, 1602–1626.

Ganne, E., 2018. Can blockchain revolutionize international trade? Technical Report. World Trade Organization, Geneva.

GlobeNewswire, 2020. *GSBN shareholders pilot innovative cargo release application in Shanghai*. CargoSmart Ltd., Hong Kong.

Haouari, M., Mhiri, M., El-Masri, M., Al-Yafi, K., 2022. A novel proof of useful work for a blockchain storing transportation transactions. *Information Processing & Management* 59, 1, 102749.

Hewa, T., Ylianttila, M., Liyanage, M., 2021. Survey on blockchain based smart contracts: applications, opportunities and challenges. *Journal of Network and Computer Applications* 177, 102857.

Hifi, M., 2002. Approximate algorithms for the container loading problem. *International Transactions in Operational Research* 9, 6, 747–774.

Hjortnaes, T., Wiegmans, B., Negenborn, R., Zuidwijk, R., Klijnhout, R., 2017. Minimizing cost of empty container repositioning in port hinterlands, while taking repair operations into account. *Journal of Transport Geography* 58, 209–219.

IBM, 2023. What are smart contracts on blockchain?, Available at https://www.ibm.com/topics/smart-contracts (accessed May 2023).

Irannezhad, E., 2020. Is blockchain a solution for logistics and freight transportation problems? *Transportation Research Procedia* 48, 290–306.

Islam, S., 2017. Empty truck trips problem at container terminals: a review of causes, benefits, constraints and solution approaches. *Business Process Management Journal* 23, 2, 248–274.

Islam, S., Shi, Y., Ahmed, J., Uddin, M., 2019. Minimization of empty container truck trips: insights into truck-sharing constraints. *The International Journal of Logistics Management* 30, 2, 641–662.

Kozan, E., Preston, P., 1999. Genetic algorithms to schedule container transfers at multimodal terminals. *International Transactions in Operational Research* 6, 3, 311–329.

Lam, S., Lee, L., Tang, L., 2007. An approximate dynamic programming approach for the empty container allocation problem. *Transportation Research Part C: Emerging Technologies* 15, 4, 265–277.

Lee, C., Meng, Q., 2014. *Handbook of Ocean Container Transport Logistics: Making Global Supply Chains Effective*, Vol. 220. Springer, Berlin.

Lee, C., Song, D., 2017. Ocean container transport in global supply chains: overview and research opportunities. *Transportation Research Part B: Methodological* 95, 442–474.

Legros, B., Bouchery, Y., Fransoo, J., 2019. A time-based policy for empty container management by consignees. *Production and Operations Management* 28, 6, 1503–1527.

Li, G., Xue, J., Li, N., Ivanov, D., 2022. Blockchain-supported business model design, supply chain resilience, and firm performance. *Transportation Research Part E: Logistics and Transportation Review* 163, 102773.

Li, J., Leung, S., Wu, Y., Liu, K., 2007. Allocation of empty containers between multi-ports. *European Journal of Operational Research* 182, 1, 400–412.

Li, J., Liu, K., Leung, S., Lai, K., 2004. Empty container management in a port with long-run average criterion. *Mathematical and Computer Modelling* 40, 1-2, 85–100.

Lin, D.Y., Juan, C.J., 2021. A bilevel mathematical approach for the empty container repositioning problem with a sharing and exchanging strategy in liner shipping. *Journal of Marine Science and Technology* 29, 3, 13.

Loe, A.F., Quaglia, E.A., 2018. Conquering Generals: An NP-Hard Proof of Useful Work. CryBlock'18: Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems, pp. 54–59. https://doi.org/10.1145/3211933.3211943.

Males˘, U., Ramljak, D., Jaks˘ić Krüger, T., Davidović, T., Ostojić, D., Haridas, A., 2023. Controlling the difficulty of combinatorial optimization problems for fair proof-of-useful-work-based blockchain consensus protocol. *Symmetry* 15, 1, 140.

Ng, C., Song, D., Cheng, T., 2012. Optimal policy for inventory transfer between two depots with backlogging. *IEEE Transactions on Automatic Control* 57, 12, 3247–3252.

Notteboom, T., Pallis, A., Rodrigue, J.P., 2022. *Port Economics, Management and Policy*. Routledge, Milton Park.

Omar, I.A., Jayaraman, R., Debe, M.S., Salah, K., Yaqoob, I., Omar, M., 2021. Automating procurement contracts in the healthcare supply chain using blockchain smart contracts. *IEEE Access* 9, 37397–37409. doi: https://doi.org/10.1109/ACCESS.2021.3062471.

Pournader, M., Shi, Y., Seuring, S., Koh, S.L., 2020. Blockchain applications in supply chains, transport and logistics: a systematic review of the literature. *International Journal of Production Research* 58, 7, 2063–2081.

dos Santos, P.T.G., Borenstein, D., 2022. Multi-objective optimization of the maritime cargo routing and scheduling problem. *International Transactions in Operational Research* 1–25. https://doi.org/10.1111/itor.13147.

Shi, N., Xu, D., 2011. A Markov decision process model for an online empty container repositioning problem in a two-port fixed route. *International Journal of Operations Research* 8, 2, 8–17.

Song, D., 2005. Optimal threshold control of empty vehicle redistribution in two depot service systems. *IEEE Transactions on Automatic Control* 50, 1, 87–90.

Song, D., 2007. Characterizing optimal empty container reposition policy in periodic-review shuttle service systems. *Journal of the Operational Research Society* 58, 1, 122–133.

Song, D., Dong, J., 2008. Empty container management in cyclic shipping routes. *Maritime Economics & Logistics* 10, 4, 335–361.

Song, D., Zhang, Q., 2010. A fluid flow model for empty container repositioning policy with a single port and stochastic demand. *SIAM Journal on Control and Optimization* 48, 5, 3623–3642.

Song, D.P., Carter, J., 2009. Empty container repositioning in linear shipping. *Maritime Policy & Management* 36, 4, 291–307.

Stenger, A., Schneider, M., Goeke, D., 2013. The prize-collecting vehicle routing problem with single and multiple depots and non-linear cost. *EURO Journal on Transportation and Logistics* 2, 1, 57–87.

Tahami, H., Rabadi, G., Haouari, M., 2020. Exact approaches for routing capacitated electric vehicles. *Transportation Research Part E: Logistics and Transportation Review* 144, 102126.

Tang, L., Wang, X., 2006. Iterated local search algorithm based on very large-scale neighborhood for prize-collecting vehicle routing problem. *The International Journal of Advanced Manufacturing Technology* 29, 1246-1258.

Todorović, M., Matijević, L., Ramljak, D., Davidović, T., Uroševíć, D., Jakšić Krüger, T., Jovanović, D., 2022. Proof-of-useful-work: blockchain mining by solving real-life optimization problems. *Symmetry* 14, 9, 1831.

Tong, J., Zhao, B., An, Y., 2023. A novel multi-objective service composition architecture for blockchain-based cloud manufacturing. *Journal of Computational Design and Engineering* 10, 1, 185–203.

Xie, Y., Liang, X., Ma, L., Yan, H., 2017. Empty container management and coordination in intermodal transport. *European Journal of Operational Research* 257, 1, 223–232.

Xin, X., Liu, M., Wang, X., Chen, H., Chen, K., 2022. Investment strategy for blockchain technology in a shipping supply chain. *Ocean & Coastal Management* 226, 106263.

Youssef, M., Naoua, B.B., Abdelaziz, F.B., Chibane, M., 2022. Portfolio selection: should investors include crypto-assets? A multiobjective approach. *International Transactions in Operational Research* 1–20. https://doi.org/10.1111/itor.13203.

Zhang, B., Ng, C., Cheng, T., 2014. Multi-period empty container repositioning with stochastic demand and lost sales. *Journal of the Operational Research Society* 65, 2, 302–319.

Appendix A: Proof of Proposition 2

We assume that for each $k \in \mathcal{S} \cup \mathcal{D}$, we have $g_k(\cdot)$ continuous, convex, and strictly increasing. If consignee i delivers a container to shipper j, then the optimal incurred holding cost is

$$s_{ij} = \min_{\theta \in [0, \tau_{ij}]} \quad g_{ij}(\theta), \tag{A1}$$

where $g_{ij}(\theta) = g_i(\theta) + g_j(\tau_{ij} - \theta)$ is a convex function of θ (Proposition 1).

Consequently, g_{ij} has a global optimum in $[0, \tau_{ij}]$. We can use any unidimensional optimization algorithm for solving Equation (A1).

A special case is when $g_i(.)$ is a piecewise linear function with $K + 1$ breakpoints $u_0 = 0, u_1, u_2, \cdots, u_K = \tau_{ij}$. Thus, the optimum solution minimizing g_{ij} is in $S = \{u_k, \tau_{ij} - u_k, k = 1, \cdots, K\}$. The set S includes at most $N = 2K$ distinct values. We denote $S = \{v_1, \cdots, v_N\}$. For $\theta \in [v_{p-1}, v_p]$, we have

$$g_i(\theta) = \alpha_{ip} + \beta_{ip}\theta, \tag{A2}$$

$$g_j(\theta) = \alpha_{jp} + \beta_{jp}\theta. \tag{A3}$$

Thus,

$$g_{ij}(\theta) = g_i(\theta) + g_j(\tau_{ij} - \theta) \tag{A4}$$

$$= \alpha_{ip} + \alpha_{jp} + \beta_{jp}\tau_{ij} + (\beta_{ip} - \beta_{jp})\theta. \tag{A5}$$

Table B1
Arcs capacities

	Consignee i/shipper j	CY (first node/last node)
s	$(0, a_i)$	$(0, I_0)$
t	(b_j, ∞)	$(0, \infty)$

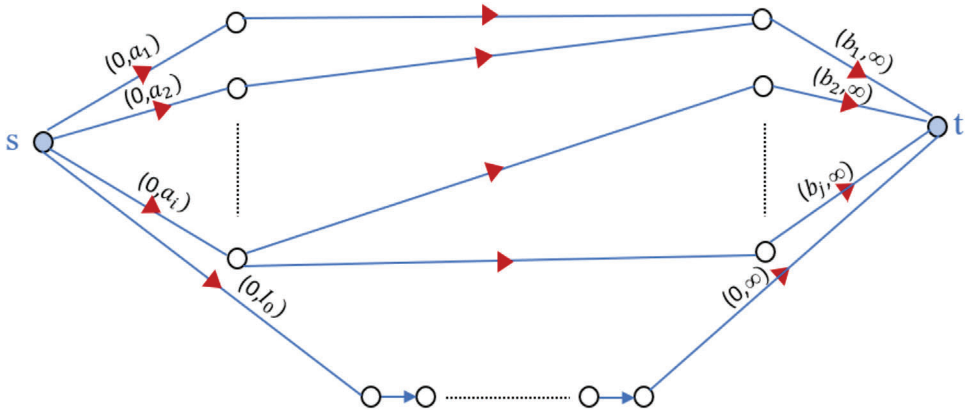


Fig. A1. Reformulation of problem (M) as a minimum-cost flow problem.

Therefore,

$$\text{If} \quad \beta_{ip} \geq \beta_{jp}, \quad \text{then} \quad \theta^* = v_{p-1}, \tag{A6}$$

$$\text{Otherwise,} \quad \theta^* = v_p. \tag{A7}$$

Hence, the optimum is always reached at $\theta \in S$.

Appendix B: Proof of Proposition 3

The problem can be reformulated as minimum-cost flow problem in the following way. We add two dummy nodes, a source s and a sink t and we define the following arcs:

- an arc between s and each consignee node i;
- an arc between s and the first node of the CY;
- an arc between each shipper j and t;
- an arc between the last node of CY and t.

The arcs capacities are bounded as per Table B1 (see Fig. A1).

Clearly, there is a one-to-one mapping between feasible solution to matching problem and st-flows of value $v = \sum_i a_i + I_0$. Therefore, the optimal solution corresponds to the minimum-cost flow between s and t.

Appendix C: Proof of the validity of constraints (28) and (29)

If $y_i = 0$, then $\sum_j : (j, i) \in \bar{A}x^2_{ji} = 0$ and according to the above constraints, we have $\sum_j : (j, i) \in \bar{A}x^1_{ji} = 0$ which is true as trip i is not selected ($y_i = 0$). If $y_i = 1$, then the constraints become

$$2 \sum_{j:(j,i)\in\bar{A}} x^1_{ji} + \sum_{\substack{j:(j,i)\in\bar{A}\\ x^2_{ji}}} \geq \left(2q^1_i + q^2_i\right), \quad \forall i = 1, \cdots, n, \tag{C1}$$

$$2 \sum_{j:(j,i)\in\bar{A}} x^1_{ji} + \sum_{j:(j,i)\in\bar{A}} x^2_{ji} \leq \left(2q^1_i + q^2_i + 1\right), \quad \forall i = 1, \cdots, n. \tag{C2}$$

We have

$q^2_i - \sum_{j:(j,i)\in\bar{A}} x^2_{ji}$ is the number of containers of Type 2 that are not loaded on a vehicle of Type 2,
$\sum_{j:(j,i)\in\bar{A}} x^1_{ji} - q^1_i$ is the excess of vehicles of Type 1 that are not loaded on any container of Type 1.

Each of these vehicles can accommodate two containers of Type 2.

So, we have

$$\sum_j : (j, i) \in \bar{A}x^1_{ji} - q^1_i = \left\lceil \frac{q^2_i - \sum_j : (j, i) \in \bar{A}x^2_{ji}}{2} \right\rceil. \tag{C3}$$

On the one hand, we get

$$\frac{q^2_i - \sum_j : (j, i) \in \bar{A}x^2_{ji}}{2} \leq \sum_j : (j, i) \in \bar{A}x^1_{ji} - q^1_i, \tag{C4}$$

which leads to constraint (28):

$$2\sum_j : (j, i) \in \bar{A}x^1_{ji} + \sum_j : (j, i) \in \bar{A}x^2_{ji} \geq 2q^1_i + q^2_i. \tag{C5}$$

Using the fact that $\lceil \frac{a}{b} \rceil = \lfloor \frac{a}{b} + 1 - \frac{1}{b} \rfloor$ (Tahami et al., 2020), we obtain

$$\left\lceil \frac{q^2_i - \sum_j : (j, i) \in \bar{A}x^2_{ji}}{2} \right\rceil = \left\lfloor \frac{q^2_i - \sum_j : (j, i) \in \bar{A}x^2_{ji} - 1}{2} \right\rfloor + 1. \tag{C6}$$

Therefore,

$$\sum_j : (j, i) \in \bar{A}x^1_{ji} - q^1_i - 1 = \left\lfloor \frac{q^2_i - \sum_j : (j, i) \in \bar{A}x^2_{ji} - 1}{2} \right\rfloor, \tag{C7}$$

which means that

$$\frac{q_i^2 - \sum_{j:(j,i)\in\bar{A}} x_{ji}^2 - 1}{2} \geq \sum_{j:(j,i)\in\bar{A}} x_{ji}^1 - q_i^1 - 1, \tag{C8}$$

and thus, we get constraint (29):

$$2\sum_{j:(j,i)\in\bar{A}} x_{ji}^1 + \sum_{j:(j,i)\in\bar{A}} x_{ji}^2 \leq 2q_i^1 + q_i^2 + 1. \tag{C9}$$

Appendix D: Proof of Proposition 4

The proof is based upon reduction from the (binary) knapsack problem. Given an instance of the knapsack problem where we are given a knapsack having capacity m, and a set S of n items, where for each item $i \in S$, are defined an integer nonnegative profit w_i and an integer nonnegative weight q_i. We build an instance of CTRP-PM as follows.

- The number of trips is n.
- The instance is only with Type 1 trucks and containers.
- The fleet size of trucks of Type 1 is m.
- For Trip 1, the demand of Type 1 containers is m and the revenue is $\mathcal{R}$.
- For each trip $i \in S \setminus \{1\}$, the demand of Type 1 containers is q_i and the revenue is w_i. We assume that $q_i < m$, $\sum_{i\in S\setminus\{1\}} q_i > m$, and $w_i < \frac{\mathcal{R}q_i}{m}$ for all $i \in S \setminus \{1\}$.
- We assume that the origin of TR_1 (O_1) is located immediately adjacent to the depot (i.e., the cost of the trip from the depot to O_1 is 0), and that the cost of reaching the destination of this trip (D_1) is c and this cost is the same for the reverse trip.
- We assume that for each $i \in S \setminus \{1\}$, the origin of TR_i is located immediately adjacent to the destination of TR_1 (i.e., $O_i = D_1$). Also, we assume that the destination of TR_i is located immediately adjacent to the depot. The cost of a trip from O_i to D_i is c.
- A truck can perform Trip 1 before any trip $i \in S \setminus \{1\}$.
- All trips in $S \setminus \{1\}$ are mutually time incompatible.

There can only be three possible types of solutions:

(S1) solution that covers only node 1;
(S2) solutions that covers only a subset of nodes $S' \subseteq S \setminus \{1\}$;
(S3) solutions that covers node 1 as well as a subset of nodes $S' \subseteq S \setminus \{1\}$.

For a solution of type (S1), the profit margin ratio is $1 - \frac{2mc}{\mathcal{R}}$. For a solution of type (S2) that covers a subset S' of transportation requests, the corresponding profit margin ratio is $1 - \frac{\sum_{i\in S'} 2q_i c}{\sum_{i\in S'} w_i}$. Since $\frac{m}{\mathcal{R}} < \frac{q_i}{w_i}$ for all $i \in S'$, then it implies that $\frac{2mc}{\mathcal{R}} < \frac{\sum_{i\in S'} 2q_i c}{\sum_{i\in S'} w_i}$. Therefore, (S2) is strictly outperformed by (S1). For a solution of type (S3) that covers node 1 as well as nonempty subset $S' \subseteq S \setminus \{1\}$ of transportation requests, the corresponding profit margin ratio is $1 - \frac{2mc}{\mathcal{R}+\sum_{i\in S'} w_i}$. Thus, (S3) strictly dominates (S1). Consequently, the optimum profit margin ratio can be obtained by finding $S' \subseteq S \setminus \{1\}$ such that $\sum_{i \in S'} q_i \leq m$ and $\sum_{i \in S'} w_i$ is maximum. Therefore, the CPTR-PM reduces to the binary knapsack problem.

Appendix E: Estimate of $\widehat{R}(N)$ (33)

To derive a simple estimate of the savings that result from cooperation between consignees and shippers, we make the following simplifying assumptions:

A1—The shippers and the consignees are uniformly distributed in $[0, 1]^2$ and their numbers are equal. The container yard is located at (0,0).
A2—All the demands and supplies are unitary.
A3—The transportation cost is proportional to the Euclidean distance. The holding costs are negligible.

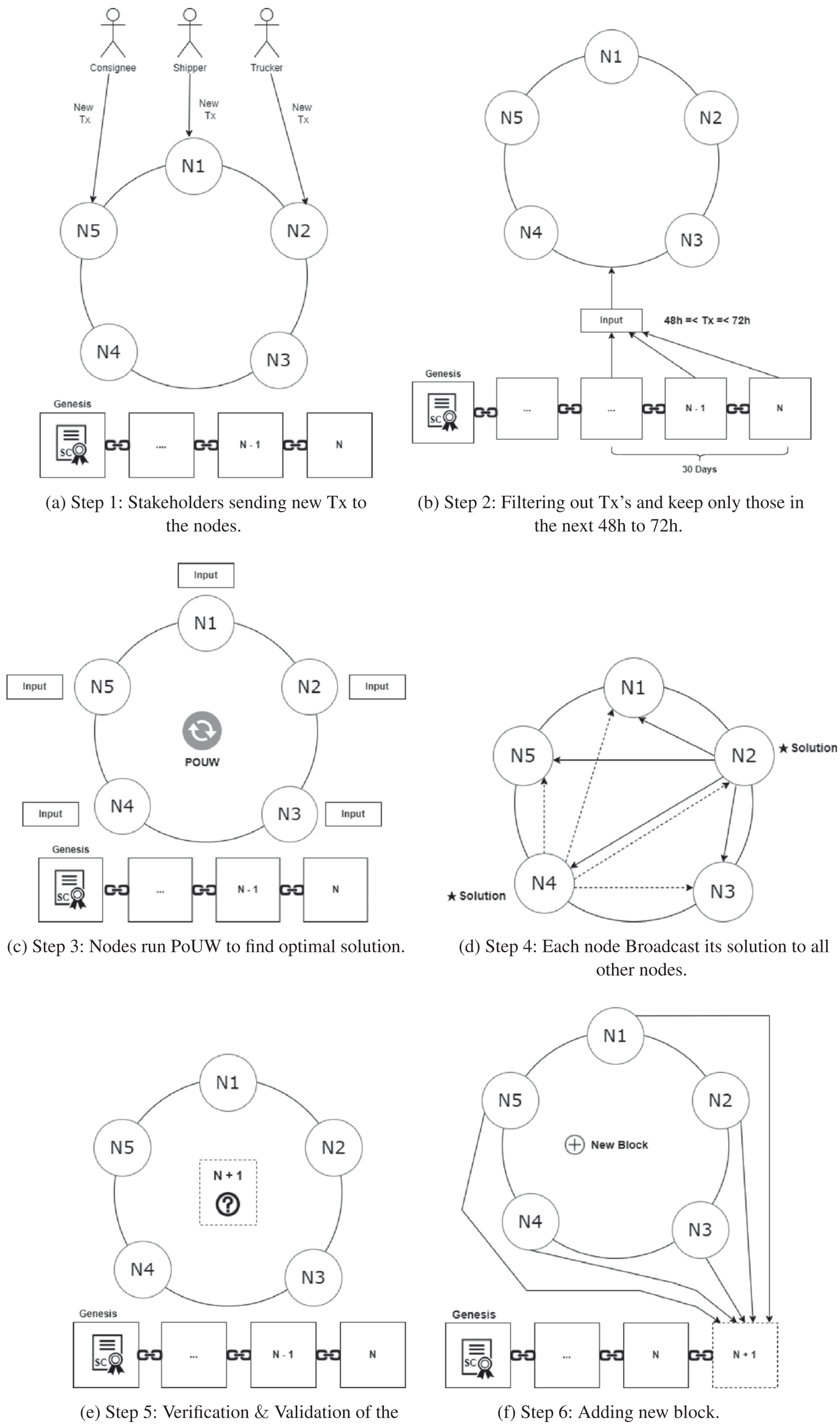
Using these assumptions, we infer that the optimal matching amounts to solving an Euclidean bipartite matching problem. Ajtai et al. (1984) show that the optimal cost of Euclidean matchings of large samples converges asymptomatically to a limit function that is proportional to $\sqrt{\frac{N}{2} \ln(\frac{N}{2})}$, where N is the total number of nodes. On the other hand, if there is no cooperation, each consignee must return the empty containers to the container yard, and each shipper must be supplied only from the container yard. Hence, the expected total transportation cost is proportional to

$$N \int_0^1 \int_0^1 \sqrt{x^2 + y^2} dx dy. \tag{E1}$$

Hence, it is proportional to N. Therefore, the cost reduction is proportional to $\sqrt{\frac{\ln(\frac{N}{2})}{N}}$.

Appendix F: Detailed description of the blockchain process

Assuming the blockchain is created with the aforementioned smart contract in its genesis block and that it is maintained by a network of active nodes, the blockchain process starts with the submission of transactions. These transactions can be of any of the four types described earlier and could be initiated through mobile or web-based decentralized application (dApp). As soon as a transaction reaches the blockchain network, it is broadcasted to all participating nodes, as shown in Fig. F1a. It is worth mentioning that a node might be run and maintained by parties that do not necessarily belong to the transportation optimization ecosystem. Since the blockchain is designed to run on as many nodes as possible, the blockchain network is availed publicly, subject to consider all the required security measures. When the minimum number of transactions pending validation is reached, nodes start the PoUW mining process. This starts by filtering all transactions of type *transport_request_tx* from the blocks mined over the past 30 days and retaining only those that need to be fulfilled in the next 48–72 hours. This will form the input to the optimization problem addressed by the current round of PoUW as shown in Fig. F1b and c. We consider this time window appropriate since it gives to truckers, shippers, and consignees sufficient time to prepare while having a decent likelihood of finding consignee–shipper matches on a route that would maximize the profit for a trucker by minimizing the cost of holding and/or transporting empty containers.



(a) Step 1: Stakeholders sending new Tx to the nodes.

(b) Step 2: Filtering out Tx's and keep only those in the next 48h to 72h.

(c) Step 3: Nodes run PoUW to find optimal solution.

(d) Step 4: Each node Broadcast its solution to all other nodes.

(e) Step 5: Verification & Validation of the new block.

(f) Step 6: Adding new block.

Fig. F1. Blockchain steps.

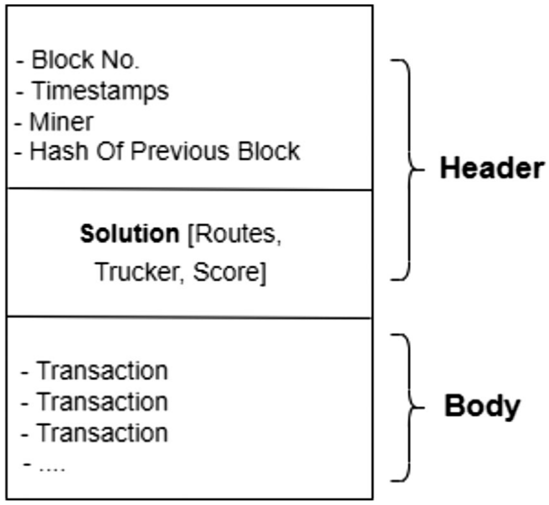


Fig. F2. Block structure.

When the mining phase is over due to elapsed mining time or having reached the maximum number of attempts, peers having found a solution to the CTRP optimization problem satisfying its constraints broadcast their tentative blocks (Fig. F1d). Each tentative block includes the transactions pending validation in its body, and the problem solution and the block data (block number, timestamp, miner identifier, etc.) in its header. Figure F2 depicts the block structure of our blockchain accordingly. When a node receives tentative blocks, transactions are verified and the solution is validated by testing it against the problem constraints and the resulting score (Fig. F1e). The block holding the best solution is retained for addition to the blockchain while other blocks are discarded. Subsequently, each node adds the new block to its local copy of the blockchain for execution (Fig. F1f). Figure F3 depicts the overall blockchain process in an activity diagram.

When a block is added to the blockchain, each node executes that block independently. This refers to running all transactions, updating the genesis smart contract state and, accordingly, updating the overall blockchain state. When a block is executed, the smart contract is typically updated in fours ways. First, new *transport_request_tx* are added in the payment map. Second, entries in the payment map referring to previous *transport_request_tx* that are now on an optimized route are updated with the payable amount. Third, *trucker_payment_tx* are reflected by transferring the payable amount to the truckers account. Finally, if any *trucker_details_tx* is found, the truckers map is updated accordingly.

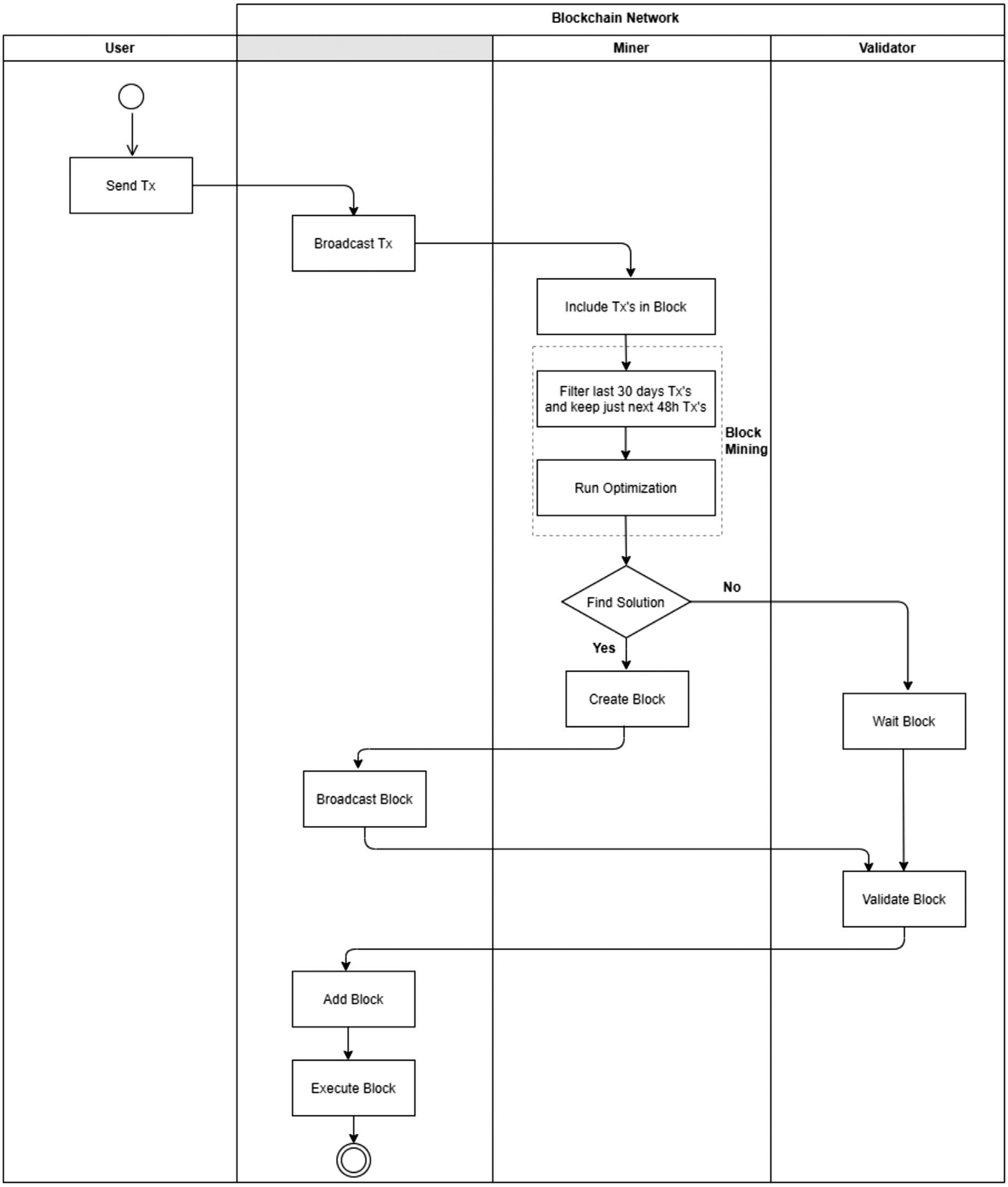


Fig. F3. Overall blockchain process to validate a transaction and adding a new block.