Transportation Research Part C 137 (2022) 103423

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Estimating economic losses from cyber-attacks on shipping ports: An optimization-based approach

Gabriel A. Weaver [a], Brett Feddersen [a], Lavanya Marla [a,*], Dan Wei [b], Adam Rose [b], Mark Van Moer [a]

[a] University of Illinois at Urbana-Champaign, United States of America
[b] University of Southern California, United States of America

ARTICLE INFO

Keywords:
Optimization
Cybersecurity
Critical infrastructure
Transportation
Ports

ABSTRACT

The *Maritime Transportation System (MTS)* accounts for more than 80% of global merchandise trade in volume and roughly one-sixth of the Total Gross Output of the United States. Given that national and global economies depend upon efficient supply chains, port stakeholders must develop security plans to respond to all hazards, natural and manmade. Given recent cyber-attacks affecting shipping ports, along with the multi-billion dollar cyber insurance gap, ports need to understand the tradeoffs between increased competitiveness and higher risk through investment in automation and advanced logistics technologies. This article addresses the need to understand the economic impact of cyber-attacks that affect shipping port operations and thereby enable risk assessments that holistically evaluate interactions among port *Information Technology (IT)* and *Operational Technology (OT)* systems. Using a *Nearly-Orthogonal Latin Hypercube (NOLH)* experimental design, we construct transportation disruption profiles based on actual cyber-attacks that specify the range of operational effects of IT/OT dependencies on stakeholder transportation assets. To capture the costs of the physical disruption, we extend Boland et al's *Dynamic Discretization Discovery (DDD)* algorithm to capture capacity constraints and enable delay modeling to accommodate commodities arriving late due to disruption. Economic loss functions for seven commodity categories based on the willingness to pay literature are used to compute delay costs so that stakeholders can estimate the range of economic and operational impacts within a disruption profile. Results based on data for cyber-attacks on landlord port and terminal operator assets provided by Port Everglades, FL illustrate impacts at $80,000 and $1.2M on average during one week in October 2017 and at $141,000 and $2.8M for May 2017 respectively. The runtime performance of our enhanced DDD algorithm improves on the state of the art by an order of magnitude and on larger problem sizes based on real-world port networks.

1. Introduction

The *Maritime Transportation System (MTS)* accounts for more than 80% of global merchandise trade in volume and roughly one-sixth of the Total Gross Output of the United States (Hoffman and Sirimanne, 2017). Increasing vessel sizes drive the need for land-side transportation systems that can handle larger container volumes. The off-loading and movement of these larger container volumes must be handled efficiently in order to support just-in-time supply chains upon which local and regional economies depend.

* Corresponding author.
E-mail address: lavanyam@illinois.edu (L. Marla).

https://doi.org/10.1016/j.trc.2021.103423
Received 7 April 2021; Received in revised form 1 October 2021; Accepted 2 October 2021
Available online 23 February 2022

These efficiencies are increasingly enabled via complex logistics automation systems that coordinate business processes across diverse stakeholders including *terminal operators (TOs)*, trucking/rail companies, port owners, and many others.

Ports connect ocean-side or sea-side transport conducted by ships, to land-side movements connected by road, rail and sometimes, air. Thus ports are facilities with a combination of multi-modal transfers and movements. A variety of commodities are transported in shipping containers, including non-perishables like apparel and cars as well as perishables such as produce and food items. To facilitate movement of commodities from the port to other downstream destination in the commodity supply chain, ports themselves contain multi-modal transportation terminals, storage terminals, several container yards, storage areas and a network of roads to facilitate movements within the port to external areas. Each of these assets inside the port is operated by various stakeholders. For example, a port itself is operated by the county or an owning company, its storage terminals are operated by contracting companies, the transportation terminals are operated by transportation companies or public organizations, and trucks to move containers are operated by drayage companies. All these organizations are stakeholders who have a keen interest in ensuring efficient port operations and minimizing the impact of disruptions. Because ports often represent the first point of entry into a country, the Coast Guard and Customs and Border Protection are other stakeholders who aim to ensure port security. Moreover, because ports are often relatively compact, roadways, gates, yards and cranes in the port have limited capacities and processing rates. Thus ports have traditionally focused on improving throughput to move more containers to downstream destinations. As global trade has increased, ports everywhere have adopted higher levels of automation to store their data, as well as to ensure more efficient operations. For example, gantry cranes that unload containers and gates that allow trucks to move across regions of the port, are all operated using processors connected to software components. Similarly, containers themselves typically include GPS trackers to be able to identify their locations in a ship or in a container yard at a terminal. A cyber-attack that spoofs or disables GPS tracking, or causes cranes or gates to fail, can thus significantly affect operations both for the duration of the attack as well as require time for the system to get back to normal operations.

The various stakeholders at ports need approaches for holistic risk assessments that account for dependencies between *Information Technology (IT)* and *Operational Technology (OT)* systems. A consequence of cross-organizational, inter-infrastructure dependencies is that the threats faced by individual stakeholders, often considered separately due to specialization and organizational siloing, can affect others within the global supply chain. Events such as the Port of San Diego ransomware attack (Anon., 2018), the Port of Barcelona cyber attack (Tsonchev, 2018), the Maersk NotPetya incident (Greenberg, 2018), as well as other historical events like the Port of Antwerp hack from 2011–2013 (Bateman, 2013), motivate the need for stakeholders to understand the economic losses from cyber-attacks via secondary, operational effects on transportation networks. The remainder of this section describes this article's four intended contributions.

Gaps within the academic literature on optimization models applied to cyberinfrastructure, motivate this work from a modeling and algorithmic perspective. A recent survey of 68 relevant peer-reviewed articles on optimization models and methods identified that much of the existing literature focuses on the Energy sector and that other Critical Infrastructure (CI) sectors should be explored (Enayaty-Ahangar et al., 2020). The same survey concludes that more research needs to be done on how to respond to cyber-originating disruptions. The results of our article in Section 7, present a holistic risk assessment approach, based on the nearly-orthogonal Latin Hypercube experimental design, to identify high-impact disruptions enabled by a cyber–physical threat model within the MTS. This approach enables stakeholders to estimate how risk exposure within cyberinfrastructure translates to operational measures of performance and changes with seasonal variations in trade.

In addition to investments in the physical transportation infrastructure within shipping ports, investment in automation technologies and advanced logistics systems may create what some estimate is another trillion dollar market (Coren, 2017). Although increasingly necessary for ports to be competitive with more efficient, transparent, and integrated with regional supply chains, automation and other infrastructure dependencies also increase risks faced by shipping ports. In recent news, cyber attacks on critical infrastructure systems are becoming increasingly frequent and have significant economic impact. For example, the Maersk NotPetya incident was a 10 day outage that cost roughly $200M. Recent attacks on the Colonial Pipeline resulted in a $4M ransom that resulted in temporary gas shortages in several states. Such disruptions are likely to increase in the future given gray-zone warfare techniques and increased nation-state competition for trade through 2050 as well as the increased adoption of emerging technologies within the sector and surrounding ecosystem (eg. smart cities).

1.1. Problem description

Cyber disruptions are unique in their own right and should be studied as a different type of phenomenon than physical disruptions via weather or kinetic weapons. Traditional 'Orange Book' security defines security in terms of Confidentiality, Integrity, and Availability. This paper focuses on cyber disruptions that affect availability of infrastructure and resources needed to move commodities through the MTS. However, cyber attacks may also compromise the integrity of data to alter commodity movements (e.g. Port of Antwerp hack from 2011–2013 to smuggle drugs) or be used to observe potentially sensitive movements. In this sense, cyber allows for a greater variety of effects than other types of disruption categories. Briefly, cyber disruptions are different from regular kinetic disruptions because of (a) spatially and temporally targeted nature of cyber-threats, (b) cyber-attacks being coordinated across multiple locations simultaneously, and (c) state-dependent nature of the cyber-threat. We elaborate on these features further, and the modeling implications, in Section 3.

In ports and other facilities of interest, cyber-attacks or disruptions translate almost instantaneously into attacks on physical assets, causing disruptions on the transportation network – examples are loss of instructions from sensors directing the assets, stalling of current physical resources (gantry cranes, automated vehicles) – resulting in cascading effects of such disruptions through the

port. Our work aims to estimate the physical effects of such cyber-disruptions by assuming that, following a disruption event, actions for *recovery* of movements (using unaffected vehicles or other resources) are undertaken. Such recovery actions involve dynamically re-configuring the previous plan of resource allocations and movements for vehicles, people and shipments. In particular, we aim to find an optimal or near-optimal reconfiguration, that will estimate a lower bound on the possible costs of the disruption, because action is often taken to reduce the impacts of the disruption. The resulting problem is a network design problem, which we model as a Disrupted Capacitated Continuous Time Network Design Problem (DC-CTSNDP), as we describe in the following section. While in practice, *optimal* recovery actions are not always taken, solving this problem helps estimate the minimum costs that will have to be incurred to recover the physical system from the cyber-disruption.

Our work and approach are motivated by large, real-world problem instances at ports in the US, which are larger in size in both space and time, than common network design optimization benchmarks and prior case studies. Because cyber-disruptions can affect operations at a high granularity, we aim to capture resource allocations at similar granularity, i.e., the movements of individual containers or TEUs. Our case study draws from fieldwork at Southport container operations at Port Everglades, FL and required a larger network, more commodities, and a longer optimization time window than previous studies. For example, economic losses for many imported commodity categories do not occur until past the 5 day mark. Therefore, we aim to study recovery windows that last at least a full week, consistent with historically-attested cyber-originating disruptions (e.g. NotPetya) described in Section 3.

1.2. Overview of methodology

Our approach to estimating costs due to cyber–physical disruptions includes three types of modeling components. First, we create a modeling framework for cyber–physical dependencies in shipping ports — in particular, to map cyber disruptions to physical disruptions using adjacencies between cyber and physical assets. This helps generate a physical disruption profile from the cyber disruption, which fundamentally influences the movements of assets in the port and causes delays in the operations of the physical layer.

Our second modeling component estimates the cost of recovering from the mapped disruption in the physical (transportation) layer. This component estimates the costs of dynamically re-configuring the network, consisting of the physical assets such as infrastructure, vehicles and shipments and moving them across the network to minimize costs from the disruption. This problem is called a service network design problem (SNDP) — it studies cost-minimizing ways of moving shipments over space and time within specified time-windows. These movements should occur via vehicles whose routes and schedules are also to be determined in the SNDP, while respecting the vehicle capacities. For cost-minimization, the movements of shipments or commodities should be consolidated on the vehicles; and thus the SNDP is in fact, a problem of consolidating commodity movements in space and time. Routing vehicles and commodities on a network is a problem of large-scale network design, which adds on a packing element to the multi-commodity flow problem over time, which itself is weakly NP-hard (Hall et al., 2007). This network design problem is often studied using discretized time–space networks, which model movements on the network in both space and time, using a priori specified time discretization to model decisions (Powell et al., 1995). However, discretization approaches often face issues of scalability in solving the SNDP at large scales. To address this, Boland et al. (2017) and Vu et al. (2020) study the continuous-time service network design problem (CTSNDP), and show that the SNDP can be solved without discretizing every point in time, and instead discretizing dynamically, only as necessary. Their approach is called the Dynamic Discretization Discovery (DDD) algorithm. However, the CTSNDP and its associated DDD algorithm do not capture two aspects — allowing commodities or shipments to arrive late, and minimizing such lateness; as well as modeling the rate of flows of commodities through nodes or arcs (e.g. road capacity, crane unloading rates). We therefore introduce the DC-CTSNDP (Disrupted Capacitated Continuous Time Service Network Design Problem), in which we accommodate container movements arriving late. Furthermore, the rate of flow of commodities through a port network is limited (e.g. road capacity, crane rate) and so we introduce capacities to enforce flow rates of vehicles through the nodes and arcs of the network. Specifically, in our enhanced DDD algorithm for the DC-CTSNDP, we introduce delay nodes and arcs to capture delays and minimize delay costs, and enforce flow rates, both in a continuous-time framework. This algorithm allows us to estimate the costs due to optimal reconfiguration of the routes and schedules of vehicles and shipments following a disruption.

The third modeling component estimates the objective function (cost) parameters to be input to the optimizer in the second component. The objective function measures direct economic losses due to import delays. Specifically, the cost penalties for import delays differ across commodity categories, a classification defined to aggregate commodity groups found within Port Everglades' economic data on imported commodities. Our analysis and valuation of import delay cost penalties is related to the work of Hummels et al. (2007) and Minor (2013). This paper extends these penalties for 1–2 day delays from on-loading/off-loading inefficiencies to medium and longer-term disruptions such as those cataloged in Section 3. The intent of this approach is to address the need for scientific studies to determine economic loss due to on-site port disruptions (Acosta, 2020).

1.3. Paper outline

We discuss related literature and the contributions of our paper in Section 2. In Section 3, we discuss how we model cyber network services and physical (transportation network assets) dependencies to create a disruption profile. In Section 4, we present an algorithm for the DC-CTSNDP, for optimized recovery from the mapped disruption profile, to measure the best possible actions that could be taken to mitigate the disruption costs. We discuss the details of how costs are computed to be input into the recovery algorithm in Section 5. Section 6 discusses data analysis and fusion for real instances used in our paper, and Section 7 presents the results. We conclude in Section 8.

2. Related work and contributions

Within the optimization literature, a recent, 68-page survey on using optimization to enhance and improve cyberinfrastructure security (Enayaty-Ahangar et al., 2020) concluded with both the need to study CI sectors other than energy and information—such as transportation and communications as well as to address the vulnerabilities introduced by CI sector interactions. The problem of routing vehicles and commodities is a large-scale network design problem, which adds on a packing element to the multi-commodity flow problem over time which itself is weakly NP-hard (Hall et al., 2007). Existing methods based on time–space network models model delays and capacities at a large computational expense (Jarrah et al., 2009; Erera et al., 2013; Crainic et al., 2016). The *Continuous Time Service Network Design Problem (CTSNDP)* and *Dynamic Discretization Discovery (DDD)* algorithm proposed by Boland et al. (2017) and Vu et al. (2020), use continuous time to reduce the size of the time–space network and improve scalability to larger problem sizes. The CTSNDP and associated algorithm however, do not consider transportation network delays and rate-based capacities. Furthermore, there appears to be a gap between the size of flat networks used to benchmark and evaluate algorithms within the optimization literature and empirical observations on transportation networks such as those surveyed by Lin and Ban (2013). For example, Boland's algorithm was benchmarked on flat networks with up to 30 nodes, 700 arcs, and 400 commodities whereas intermodal networks surveyed by Lin ranged from hundreds to thousands of nodes. Within the operations research community Sanchez et al. (2018) emphasize the need for studies backed by a strong experimental design and tout the benefits of a *Nearly Orthogonal Latin Hypercube (NOLH)* approach. Such benefits include the ability to identify combinations of factors leading to high impact scenarios and the ability to visualize the extent to which runs have covered an experimental design space.

The importance of understanding CI interactions is echoed by a recent report by RAND (Engstrom, 2018), which emphasizes the need for a system-of-systems perspective when thinking of disruption models. Within the critical infrastructure security literature, DiRenzo et al. (2017) provide an overview of cyber-based threats to the MTS. In addition, the modeling and simulation community has conducted several studies looking at inter-infrastructure impacts to shipping networks (Pant et al., 2011; Bou-Harb et al., 2017; Cimino et al., 2017; Beyeler et al., 2004). The economics literature also estimates the impact of port disruptions. *Input–Output (I–O)* models have been used to relate goods coming out of a port to dependent industries (Danielis and Gregori, 2013) as well as estimate the impact of disruptions encompassing all hazards ranging from terrorist attacks (Park, 2008; Rose, 2009), cyber attacks (Rose, 2009), and port shutdowns (Wei et al., 2020; MacKenzie et al., 2011; Rose and Wei, 2013). In fact, a decision-support system that estimates the economic consequence of maritime cyber threats, fits into several of the Coast Guard's strategic priorities and this is discussed further by Rose et al. @in Rose et al. (2017). Pant et al. (2011) coupled the output of a simulation of the movement of imported goods through a port with an I–O model to estimate the economic impact of disruptions. In addition, work done by Hummels et al. (2007) and Minor (2013) focuses on losses from short-term (1–2 day) delays due to on-loading/off-loading inefficiencies.

This article intends to integrate and extend state-of-the-art techniques in modeling the economic impact of cyber-originating disruptions to shipping ports with applicability to intermodal transportation systems in general. We now describe how each of our four contributions extends previous research discussed above.

First, the results of our article contribute to a holistic risk assessment approach based on sound experimental design to identify high economic impact scenarios from cross-infrastructure, inter-organizational disruptions. In addition, our adjacency matrix, of cyber–physical interactions—with real-world disruption attestations shown in Table 4—explicitly encodes general information dependencies between the Communications/IT and Transportation sectors. These general information dependencies are translated to factor ranges for assets in a specific transportation network. These factor ranges specify a disruption profile used to generate NOLH experimental design matrices.

Second, with respect to solving multi-commodity flow problems on large-scale networks, our extensions to the state-of-the-art DDD optimization algorithm improve the scalability and applicability of existing approaches. In order to accommodate disruptions, motivated by the need to study *what-if* scenarios for risk assessments, we added the ability to model late-arriving commodities. Moreover, we add the modeling of per-unit-rate capacities on nodes and arcs, to the existing DDD algorithm.

Third, we conduct extensive empirical and computational analysis on instances larger than those in recent literature. Specifically the size of the transportation networks, number of commodities, and planning horizon for the optimizer are larger. These problem instances are motivated by cyber-originating disruptions on transportation networks such as those surveyed by Lin and Ban (2013). The network size in our Port Everglades study is comparable to the local transit networks surveyed by Lin (using an L-space representation Lin and Ban, 2013), but larger than those used for network optimization benchmarks and Boland's case study (Boland et al., 2017). Moreover, we adapt the Pant et al.'s simulation model of commodity movements through a terminal operator and use it as a template to represent the four terminal operators in our Port Everglades' transportation network. In addition, the number of commodities we consider, derived from vessel schedules and cargo manifests provided by Port Everglades, is an order of magnitude larger. Finally, in order to see significant economic losses to commodity values as well as model the duration of real-world cyber disruptions, we considered a longer planning horizon (one week) versus the prior work considering 5 days.

In order to understand the economic impact of cyber-originating disruptions, we estimate direct business losses due to cargo delays. Specifically, we translate physical cargo volumes into dollar values and use our DDD algorithm to get a conservative estimate on losses from delay costs. Given that the duration of disruptions due to cyberattacks can be much longer than short-term on-loading/off-loading inefficiencies—it took ten days for Maersk to recover from NotPetya (Mathews, 2017)—we extended and adapted work by Hummels and Minor to extrapolate loss functions for commodities delayed over longer periods of time than the previous literature.

3. Modeling cyber–physical dependencies in shipping ports

The *characteristics* of cyber-originating disruptions are different from other types of disruptions to intermodal transportation systems such as extreme weather conditions or other rare events. First, intelligent adversaries can target specific infrastructure assets at particular times to maximize the impact (or secondary impact) of the event. In contrast, extreme weather conditions or other rare events cannot target specific infrastructure assets or flows at a particular time. As such, adversaries may even choose to launch a cyber attack opportunistically in combination with another event such as a natural disaster. For example, after Hurricane Harvey, there was an increase in spam posing as the Red Cross asking for donations. Cyber attacks are harder to predict than weather and as a result, the time to prepare and respond is shorter, motivating the need for optimal response. Second, depending upon the target, cyber attacks can simultaneously affect geographically distant locations. For example, the Maersk NotPetya incident simultaneously affected several of Maersk's global terminal operations. In contrast, kinetic disruptions are limited to the geographic area in which they occur. Third, the trigger for cyber attacks may be logical, based on the overall state of the transportation system rather than at a particular time. For example, a disruption to a particular type of cargo being unloaded by a specific company may result in larger economic losses than a physical disruption of longer duration. Finally, the time scale at which cyber attacks can occur and their impacts persist is different than that of physical disruptions. For example, electrons may move through copper communications networks at the speed of light. In contrast, vehicle movements on a transportation network occur on the scale of minutes. The fact that the timelines for events on the transportation network versus that of a communication network are at very different levels of precision motivates us to explore continuous-time methods. A discrete time step within the transportation domain timeline may undersample cyber events occurring in a short timeframe. In contrast, a discrete time step within the communications/IT domain timeline may oversample given a lack of events occurring within the transportation domain when no events are occurring and instead increase problem size leading to intractability. Thus we present a continuous-time algorithm for the recovery of the network from these disruptions in Section 4.

Consequently, we explicitly represent dependencies of different types of intermodal transportation assets on information provided by network services as an adjacency matrix. Table 1 encodes a template for an adjacency matrix of information dependencies between communications/IT network assets (blue rows) and transportation network assets (orange columns). Specifically, the template is expressed in terms of the infrastructure networks' *semantic* attribute values. These dependencies, if present, affect values of transportation network queueing parameters (see Table 4) depending on the threat profile. Disruptions to such dependencies can affect the flow of commodities through a port by affecting gate service times, routes taken, traffic congestion, crane rates, and other transportation network performance factors.Entries in the corresponding table document attestations of real-world disruptions for a given dependency and may be used to inform the construction of realistic disruption profiles based on the unique characteristics of cyber-originating disruptions. For example, a disruption profile could be constructed to target a specific terminal, type of transportation asset, or communication service within a port network. Since our disruption profile is based on logical dependencies, the affected transportation assets may or may not fall within a contiguous geographic region, consistent with the second unique characteristic of cyber-originating disruptions versus other hazards. Finally, this approach sets the stage for modeling cyber attacks based upon the state of the transportation system, by integrating our optimization algorithm with a discrete event simulation (Weaver and Marla, 2019; Weaver et al., 2019). We now provide a brief overview of the different types of technologies upon which shipping port operations depend and associated potential disruptions.

Table 2 encodes a template for an adjacency matrix of information dependencies between communications/IT network assets and transportation network assets. Specifically, the template is expressed in terms of the infrastructure networks' *semantic* attribute values. These dependencies, if present, affect values of transportation network queueing parameters (see Table 4) depending on the threat profile. As such, transportation network parameters are instantiated relative to dependency between a cyber asset and transportation network asset involved. An example of such a network is illustrated in Fig. 1. For example, Broward County, as landlord of Port Everglades, is responsible for gate service times and crane rates for transportation network assets, depend upon Gate Kiosk and Crane HMI availability and performance within the Broward County comms/IT network. In contrast, Crowley, as terminal operator, has their own gate kiosks to ensure efficient movements to/from the Container Yard. These cross-layer dependencies determine how experimental transportation network factors (queuing parameter values) are sampled relative to a threat profile as described in Section 7 of our paper.

Both port harbormaster and vessel pilots depend upon the *Automatic Identification System (AIS)* and *Global Positioning System (GPS)*. AIS connects new real-time information sharing among ships, via base stations and satellites, to supplement radar to prevent collisions at sea (DiRenzo et al., 2017). GPS, an essential positioning, navigation, and timing service, has been jammed and spoofed in the past, rendering vessels unable to navigate correctly (Burgess, 2019; Anon., 2013; Newman, 2017) and interfering with the ability to track cargo (Anon., 2014). In addition, vessels may depend upon *Programmable Logic Controllers (PLCs)* for their engines and/or ballast (Muccin, 2016). Any of these cyber-originating disruptions have the potential to disrupt the ability to navigate shipping port channels by reducing capacity or completely blocking movement or degrading on-board systems (Cimpanu, 2019b). Based on fieldwork as part of the Jack Voltaic v 3.0 exercises (Vavra, 2020), such a disruption could take from two days to two weeks to clear, depending upon whether hazardous cargo was involved as well as the size of the vessel. More recently the mega-container vessel Ever Given blocked the Suez Canal for six days; this disruption resulted in an estimated loss of $400M USD per hour (LaRocco, 2021).

Gantry cranes increasingly depend upon technologies to rapidly identify and locate shipping containers while loading and unloading containers from vessels once docked. For example, both *Optical Character Recognition (OCR)* and *Radio-Frequency Identification (RFID)* tags can be used by crane systems (Anon., 2017). Such systems, if the communications network is degraded or

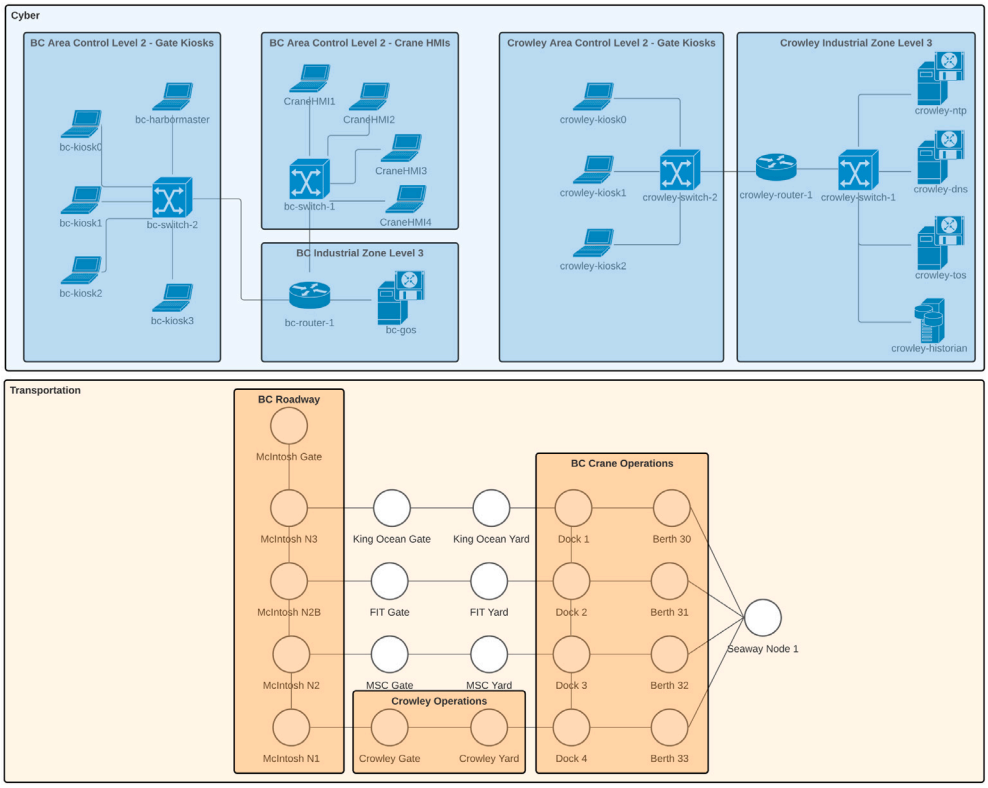


Fig. 1. An example cyber–physical network for stakeholders based upon fieldwork at Port Everglades used in Section 7. The transportation network G_{Trans} (the flat network D in Section 4) is illustrated in orange while the corresponding, per-stakeholder communications/IT networks are illustrated in blue. Potential dependencies and associated disruptions between types of assets in the two layers are encoded by an adjacency matrix shown in Table 1. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

connectivity lost, may result in slower crane rates. Typical crane rates at ports are 22–26 TEU moves per hour, but this may be as large as 30 TEU moves per hour. These faster rates are driven by the need for more efficiency to accommodate larger vessel sizes due to increased trade volumes and the formation of alliances among major ocean liners.

Within a container yard, terminal operators depend upon systems to manage vessel stow plans, optimize container storage in the yard, and coordinate container arrivals and departures across multiple modes of transportation. For example, the *Terminal Operating System (TOS)* is responsible for several of these functions within the port and if disrupted, can have a large impact on container dwell times, gate service time, and effective road capacity due to traffic congestion. In 2017, a modified version of the Petya ransomware, NotPetya, infected systems within Maersk. Maersk's operations were affected at several ports for days, resulting in inoperable TOS and *Gate Operating Systems (GOS)* (Greenberg, 2018). This ransomware incident forced Maersk to reinstall a reported 4000 servers and 40,000 PCs in a period of 10 days (Osborne, 2018) and cost Maersk and estimated $200M (Mathews, 2017). A study of the impact of a ransomware attack on 15 Asia-Pacific ports estimated a $101 Bn insurance gap Anon. (2019). Additional ransomware attacks have been experienced by the Port of Long Beach Cimpanu (2019a), the Port of San Diego (Anon., 2018), the Port of Barcelona (Tsonchev, 2018), the Port of Kennewick (Anon., 2020c), and the impacts of the Ryuk ransomware in a maritime organization are documented by DHS CISA alert AA20-049 (Anon., 2020a). Other systems that could be impacted by network latency or loss of connectivity due to unauthorized access (Mohindru, 2017) include customs inspection databases, resulting in longer container yard dwell times.

Larger vessel sizes also drive demand for access roads and intermodal rail connections, demands which if not met, increase congestion of land-side operations and costs due to delays. Movements of containers depend upon a variety of systems to process incoming/outgoing trucks (e.g. TWIC, VOIP, OCR, RFID) and coordinate and document movements across transportation modes (e.g. EDI). For example, when a truck arrives/leaves a port through a gate, the gate kiosk used to confirm identity and booking number may depend upon *Voice Over IP (VOIP)* for communications with a gate clerk, as well as OCR or RFID and video. An attack on the network's quality of service (e.g. the VOIP attack in WI Anon., 2016), could result in longer gate service times. Similarly, a disruption to trucks' *Electronic Logging Devices (ELD)*, could affect their operation and result in severe financial damage to companies within the trucking industry Simpson (2018). Furthermore, under the *Maritime Transportation Safety Act (MTSA)* of 2002, employees must have a *Transportation Worker Identification Credential (TWIC)* in order to get into US port facilities. Some ports use these TWIC cards to determine physical access to port facilities by looking up scanned cards in an access control database. Hacking the TWIC

Table 1
Cyber–physical dependencies within shipping ports. Each entry documents functions provided by a network service (blue rows) upon which transportation network assets (orange columns) depend. Where appropriate, references identify historically-attested disruptions to the corresponding cyber–physical dependency.

Cyber–physical dependencies and disruptions within shipping port container operations								
	Shipper, Trucking Co. Distribution Center	Landlord Port, Terminal Operator Gate	Terminal Operator Container Yard	Landlord Port Gantry Crane	Shipping Co. Vessel	Shipping Co. Landlord Port Channel	Shipping Co. Sea	Other
AIS					Harbormaster: Vessel Positioning and Metadata	Vessel Pilot: Navigation		
GPS			Jamming GPS Cargo Trackers (Anon., 2014)		Harbormaster: Positioning, Navigation, Timing (Burgess, 2019; Anon., 2013; Newman, 2017)	Vessel Pilot: Positioning, Navigation, Timing		
OCR		Container Identification, Gate kiosk video		Container Identification				
RFID		Container Identification	Straddle: Container Identification	Container Identification				
TOS		GOS: Booking confirmation, Container location (Greenberg, 2018)	Bobcart/Straddle Container location	Container location in stow plan and yard (Bateman, 2013).	Container location in vessel stow plan			Shen Virus, Lloyds (Anon., 2019), COSCO Terminal Ryuk (Cimpanu, 2019a); Barcelona (Tsonchev, 2018), San Diego (Anon., 2018), Kennewick (Anon., 2020c)
TWIC		GOS: Access Control to port facilities						TWIC Database Hack (McGlone, 2014)
VOIP		Gate kiosk phone	Corporate: TO voice communications					WI Law Enforcement (Anon., 2016)
EDI	Container release outgate messages from TO.	GOS: Container release outgate to Shipper, Booking number from Trucking Co to TO.	Container arrival message to Shipper/Customers, Container location to Trucking Co.	Send container discharge to Shipping Co.	Receive cargo manifest from Shipper.	Receive container discharge from TO.	Vessel request berth to TO	
Other	Contractor (Mohindru, 2017), Electronic Logging Device (ELD) (Simpson, 2018)				Engine PLCs, (Muccin, 2016), (Cimpanu, 2019b)		Vessel Navigation (Newman, 2017)	DC Police, RDP (Green, 2017; Cimpanu, 2019a)

database, as attested to in McGlone (2014), could result in longer gate service times, traffic congestion, and inability to access port facilities. Similarly, compromising video surveillance systems (Green, 2017). Finally, *Electronic Data Interchange (EDI)* uses

standard messaging formats among shipping companies, *Terminal Operators (TOs)*, and drayage companies to coordinate container movements. EDI compromises could potentially slow down the transfer of containers across stakeholders' organizational boundaries.

4. Optimized disruption recovery: Model and algorithm for network design and commodity flows

Given a mapping of the cyber disruption to the physical transportation system as described in Section 3, our algorithm for the DC-CTSNDP may be used to optimize and recover disrupted commodity routes across a variety of intermodal transportation networks at different geographic scales (Weaver, 2021a,b). However, as described in Section 3, it is especially motivated by historically observed cyber–physical disruptions and their characteristics; in particular, the ability to have flexible time-scales and to solve problems with large number of commodities scalably.

Before describing the details of our Dynamic Discretization Discovery (DDD) algorithm for the DC-CTSNDP, we first discuss the basic CTSNDP problem and provide a sketch of the DDD for that setting. The DDD algorithm for CTSNDP begins with a minimal discretization consisting only of points where events occur, such as the earliest available times of commodities and the latest delivery times, and similarly for vehicles. This low or partial discretization captures all travel times between points on the static network as equal or lower than the true travel times. Such discretization also allows for the maximum consolidation of shipments or commodities, and a solution of the network design problem on this partially expanded time–space network provides an *infeasible* lower bound. Repairing this solution involves checking if the movements prescribed by the solution can be executed without any of the travel times being shorter than they truly are; and if so, generating a feasible set of movements. If not, arcs that need to lengthened to their correct length are identified, which also indicates some required points of discretization to create a feasible solution. In the process, nodes and arcs that are required to be added to the partially time–space network are identified. This repaired solution provides a feasible upper bound. If the upper bound is optimal, the algorithm terminates. Otherwise, new arcs to be lengthened and points of discretization to be added are identified iteratively, and the algorithm is again solved to find a new lower bound and upper bound. The algorithm continues until the upper bound is provably optimal or near-optimal.

In this paper, we are interested in solving the DC-CTSNDP, in which commodities are allowed to be delayed and where the nodes and edges have further time-based capacities. That is, the first extension is to model commodities being delayed at the destination. Because our goal is to capture re-routing and re-scheduling *under disruptions*, the consequent delays and their associated costs; it is essential to allow commodities to potentially arrive later than the scheduled delivery time. The CTSNDP does not allow the capture of such delays. The modeling challenge in allowing delayed arrivals is that the amount of delay required to allow successful delivery of the commodity to minimize the combination of vehicle costs and delay costs is unknown a priori. The second extension is to model per-unit-time capacities on nodes and arcs. That is, on each physical link (roads) in the network, we wish to allow a certain flow rate of vehicles that transport commodities in unit time (e.g., vehicles through a gate per unit time or vehicles in a road per unit time). Because the time discretization is not fixed a priori, it is not straightforward to capture these capacity constraints. To solve the more complex DC-CTSNDP, we now expand on the DDD approach of Boland et al. (2017) and Vu et al. (2020), to incorporate features not present in the original models.

The port's land-side transportation network, represented by static network D in our model, consists of several terminals connected by roadways as well as entry and exit gates to and from the port. Specifically, we employ a modified version of the transportation network used in our simulation study in Weaver et al. (2019), which consisted of the following stakeholders: (1) landlord port (e.g. Broward County) — crane operations and port gate operations, (2) terminal operators (e.g. Crowley, MSC, FIT, King Ocean) — movement from shore to land via Container Yards and Terminal Gates, (3) Drayage Companies — responsible for TEU movement from Container Yard out of Port, and (4) shipping companies: responsible for vessel movement from Seaway to Cranes. Each of the terminals consists of a subgraph that reflects the queueing network topology of Pant's simulation model (Pant et al., 2011). Specifically, a terminal subgraph consists of a berth where commodity shipments arrive, gantry cranes that offload shipments, a container yard where commodities are stored, and terminal gates where containers move from the terminal to intermodal port roads. From the complex transportation networks perspective, our port network is an L-space representation in which "steps or states are vertices and two vertices are connected if they are consecutive on an arbitrary route" (Lin and Ban, 2013). Port transportation network components have limited ability to process containers per unit time. For example, roadways have a limited vehicle capacity and crane rates specify the number of TEU per hour that can be moved. Therefore, transportation network nodes in the static network D are initialized with queueing attributes as shown in Table 4 and expressed as optimization constraints to enforce rates in a continuous-time framework.

Containers moving through the port network are represented as commodities K. A given commodity $k \in K$ represents a single twenty-foot equivalent container (TEU) and contains a single commodity type represented by a 2-digit HS code. The commodity type affects the loss function should that TEU arrive to its destination after its latest delivery time (LDT) . Finally, given that multiple terminals are represented in D, other domain-specific considerations, such as that all containers offloaded from a given liner must be routed through that liners' terminal, are captured by side constraints that can be turned on depending upon the scenario.

In order to model *disrupted* commodity movements, in which TEU may arrive to their destination well after their LDT, we developed delay nodes and arcs. For cyber-originating disruptions, the location and magnitude of a disruption, expressed as changes in a transportation component's queueing attributes, depends upon its dependencies on information provided by services accessed via computer networks. These information dependencies between network services and types of transportation network assets (e.g. gates, cranes), and how they are used to create a disruption profile, have been described in Section 3. Finally, port stakeholders need the ability to relate disruptions to communications network services to economic losses, Section 5 describes our approach which required extending loss functions in the willingness to pay literature from short 1–2 day delays to longer 6–10 day disruptions as seen in recent ransomware attacks.

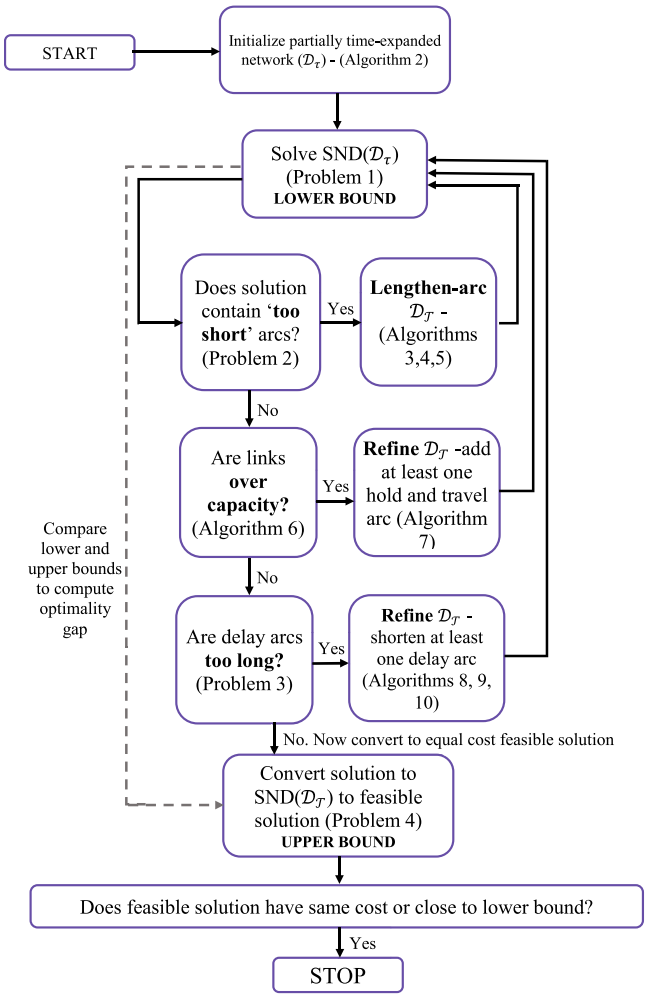


Fig. 2. Flow chart describing our proposed algorithm (Algorithm 1). Bold lines describe the main algorithm and dashed lines the comparison with the upper bound.

4.1. Sketch of algorithm

The flowchart in Fig. 2 presents an overview of our novel algorithmic approach to solving the CTSNDP including delay and per-unit-time vehicle flow-related capacity constraints. The steps that check for capacity violations "are links over capacity?" and delays "are delay arcs too long?", and the associated 'Refine' and 'convert' algorithms to address constraint violations are novel additions to the techniques proposed in Boland et al. (2017) and Vu et al. (2020).

The model proposed by Boland et al. (2017) and Vu et al. (2020) begins with a 'flat' or static network or graph $D = (\mathcal{N}, \mathcal{A})$, which represents the physical connections from ships, to and within the port, and to delivery points in the supply chain. $\mathcal{N}$ represent the physical locations in the network and $\mathcal{A}$ represent the existing directed connections between these nodes. For each arc $a = (i, j) \in \mathcal{A}$, there exist parameters $\tau_{ij} \in \mathbb{N}_{>0}$ that represent the travel time, capacity $u_{ij} \in \mathbb{N}_{>0}$, per unit flow costs $c_{ij} \in \mathbb{R}_{>0}$, and fixed costs $f_{ij} \in \mathbb{R}_{>0}$. K denotes a set of commodities that needs to be moved on this network, in space and time. Specifically, each commodity $k \in K$, of quantity q_k, should be moved from its origin $o_k \in \mathcal{N}$ to its destination $d_k \in \mathcal{N}$. $e_k > 0$ is the time that commodity k becomes available at its source and should be delivered to its destination by time l_k. The basic service network design problem is to design the underlying network of vehicle movements that facilitate commodity flows from their sources to sinks within the specified time windows while minimizing the sum of fixed costs and flow costs.

These problems are typically solved using a time–space network representation, which consists of the flat network expanded in time, where time is discretized in small increments (usually 5 or 10 min) that represent possible action points. However, we run into issues of intractability when the number of commodities in these networks are large. Boland et al. (2017) address the considerable scalability challenges using a *continuous time* solution approach. In this approach, discretization by time is introduced dynamically as needed (Vu et al., 2020), starting from a *partially* expanded or discretized network where only time points of interest are captured (each point in space are captured, at some selected points in time). The (partial or full) time-expanded network is referred to as $\mathcal{D}_\mathcal{T}$,

where $\mathcal{D}_\mathcal{T} = (\mathcal{N}_\mathcal{T}, \mathcal{H}_\mathcal{T} \cup \mathcal{A}_\mathcal{T})$. The initialization of this network without modeling delays is described in lines 1–13 of Algorithm 2. Note that because only some nodes are included, the arcs in the partially expanded time–space network, by construction, are 'short' compared to the true travel times on the corresponding arcs on the flat network. The initial construction of the network allows for the highest possible consolidation of the commodities with least cost, even if infeasibility can occur.

The solution to $SND(\mathcal{D}_\mathcal{T})$ on the partially expanded time–space network is thus a dual-type lower bound. The solution when repaired (using the *Construct Feasible Solution Problem*) creates a feasible solution which is an upper bound. If the upper bound solution can be shown to be optimal or the gap is small, then the algorithm terminates. Else, the current solution cannot be converted to an equivalent feasible solution, and the algorithm discovers new time points to be added to the partially expanded time–space network that allow for further discretization of the partially-time expanded network. The authors demonstrate that because these partially expanded time–space networks also conform to some basic principles that are also satisfied in fully-discretized time–space networks, infeasibilities are detected in each iteration and the network is appropriately expanded, the algorithm arrives at the optimal solution.

In this paper, we expand the above algorithm to include the ability to capture delayed delivery of commodities (when delays are unknown a priori), and approximately capture per-unit-time vehicle-flow capacities corresponding to the number of vehicle flows per unit time. To capture these aspects, we introduce new aspects into the network construction in the form of delay arcs, and hold arcs. We describe these new network construction aspects in Sections 4.2.1 and 4.2.2 .

Algorithm 1 (SOLVE-CTSNDP)

Require: Flat network $\mathcal{D} = (\mathcal{N}, \mathcal{A})$, commodity set $\mathcal{K}$
1: Create a partially time-expanded network $\mathcal{D}_\mathcal{T}$ satisfying Properties 1–4
2: **while** not solved **do**
3: Solve SND($\mathcal{D}_\mathcal{T}$)
4: Determine whether the solution to SND($\mathcal{D}_\mathcal{T}$) contains any arcs that are "too short"
5: **if** arcs are "too short" **then**
6: Refine the partially time-expanded network $\mathcal{D}_\mathcal{T}$ by correcting the length of at least one such arc
7: **else**
8: **if** links are "over capacity" **then**
9: Refine the partially time-expanded network $\mathcal{D}_\mathcal{T}$ by adding at least one such hold and travel arc
10: **else**
11: **if** delay arcs are "too long" **then**
12: Refine the partially time-expanded network $\mathcal{D}_\mathcal{T}$ by correcting the length of at least one delay arc
13: **else**
14: The solution to SND($\mathcal{D}_\mathcal{T}$) can be converted to a feasible solution to CTSNDP with the same cost
15: Stop. The converted solution is optimal for CTSNDP.
16: **end if**
17: **end if**
18: **end if**
19: **end while**

We introduce additional notation for the problem:

Sets:

- $\mathcal{N}$: set of nodes (locations) in the flat network;
- $\mathcal{A}$: set of arcs (links) in the flat network;
- $\mathcal{K}$: set of commodities to be transported on the service network, indexed by k;
- $\mathcal{L}$ be the subset of commodities in $\mathcal{K}$ that are allowed to arrive late;
- $\mathcal{T}$: set of time points at which the time–space network is expanded, with $\mathcal{T} = \cup_{i \in \mathcal{N}} \mathcal{T}_i$;
- $\mathcal{T}_i$: set of time points $\{t^i_1, t^i_2, \ldots, t^i_{n_i}\}, t^i_1 < t^i_2 < \cdots < t^i_{n_i}$, at which node $i \in \mathcal{N}$ is expanded;
- $\mathcal{N}_\mathcal{T}$: set of nodes in the partially expanded time–space network, described as (i,t) where $i \in \mathcal{N}$ and $t \in \mathcal{T}_i$;
- $\mathcal{A}_\mathcal{T}$: set of travel arcs in the partially expanded time–space network that connect some (i,t) and $(i,\bar{t})$, where $(i,j) \in A$;
- $\mathcal{H}_\mathcal{T}$: set of holdover arcs in the partially expanded time–space network that connect some (i,t) and $(i,\bar{t})$, $i = j, t \in \mathcal{T}_i$;
- $\mathcal{A}'_\mathcal{T}$ be a subset of $\mathcal{A}_\mathcal{T}$ where the travel time for an arc is positive (travel arcs);
- $\mathcal{A}''_\mathcal{T}$ be a subset of $\mathcal{A}_\mathcal{T}$ where the travel time for an arc is negative (delay arcs);
- $\mathcal{A}'''_\mathcal{T}$ be a subset of $\mathcal{A}_\mathcal{T}$ where the source or destination node is a delay node

Parameters:

- o_k, d_k: origin location of commodity k and destination location of commodity k
- e_k, l_k: earliest available time at origin o_k and latest delivery time at destination d_k for commodity k
- q_k: quantity or demand of arc (i,j) to be transported from origin to destination
- f_{ij}: fixed cost of 'installing' arc (i,j) in the static network, incurred based on the number of times the arc (i,j) must be installed to allow movements from location i to location j between any times t to $\bar{t}$,
- c_{ij}: variable (or handling) costs of using arc (i,j) in the static network,

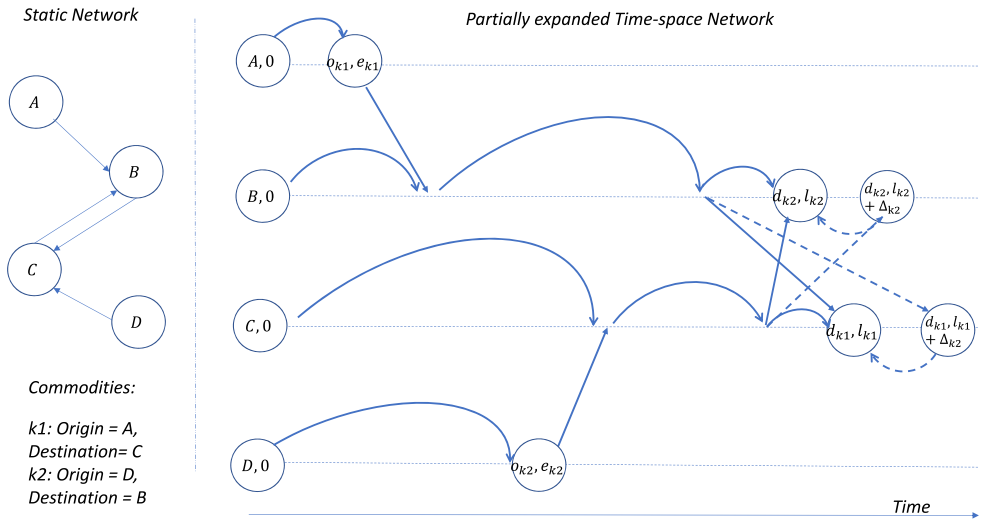


Fig. 3. Network initialization.

- u_{ij}: capacity of arc (i, j) in the static network, for each time the arc is used,
- $delay\_cost_k$: fixed delay charge cost charged to the shipper for any amount of delay, for commodity k,
- Δ_k: the maximum time commodity k is allowed to be delayed by,
- $delay\_penalty\_rate_k$: rate at which a late arriving commodity is penalized per unit of time,
- $delay\_penalty\_limit_k$: upper bound to the total delay penalty for commodity k

We will define variables for each problem as we describe the steps of the algorithm.

The following Properties 1–4 should hold for the consistent construction of the partially-expended time–space network at each iteration of the algorithm, as described in Boland et al. (2017). Our additions in the construction of delay nodes, delay arcs, and per-unit-time vehicle-flow capacities will also obey these properties. Moreover our enhanced algorithm maintains these properties at each iteration and modification of the partially expanded time–space network to ensure that repeated re-solving of the $SND(\mathcal{D}_\tau)$ always yield a lower bound to the true problem.

Property 1. *For all commodities $k \in K$, the nodes (o_k, e_k) and (d_k, l_k) are in $\mathcal{N}_\tau$.*

Property 2. *Every arc $((i,t),(j,\bar{t})) \in \mathcal{A}_\tau$ has $\bar{t} \leq t + \tau_{ij}$.*

Property 3. *For every arc $a = (i,j) \in \mathcal{A}$ in the flat network, $\mathcal{D}$, and for every node (i,t) in the partially expanded network, $\mathcal{D}_\tau = (\mathcal{N}_\tau, \mathcal{A}_\tau \cup \mathcal{H}_\tau)$, there is a timed copy of a in $\mathcal{A}_\tau$ starting at (i,t).*

Property 4. *If arc $((i,t),(j,t')) \in \mathcal{A}_\tau$, then there does not exist a node $(j,t'') \in \mathcal{N}_\tau$ with $t' < t'' \leq t + \tau_{ij}$.*

4.2. Network construction

Algorithm 2 describes the initialization of the network. At this step, nodes that capture both time and space, in particular, (o_k, e_k) and (d_k, l_k), which are the origin location and time of commodity k and the destination location and delivery time of commodity k respectively, are added. For all locations, the nodes $(u, 0)$ are added to represent that movements can begin at each location after time 0. For every time–space node thus created in this network, arcs that are shorter than the true travel time to each connected location are added. We also include holdover arcs that connect two nodes at the same location but successive times for which nodes exist — representing the storage of a commodity at a given location for a time spanning the difference between the head and tail nodes of the arc. Thus, the commodity can arrive early at the location and depart at a later time. Fig. 3 illustrates such initialization with an example.

4.2.1. Late commodities

Let the maximum allowed delay for commodity k be Δ_k. Each commodity k arriving late has a delay penalty per unit time of p_k associated with it. Moreover, we allow the ability to impose a maximum delay penalty cost that can upper bound the total penalty cost. To model these aspects, we introduce the concept of delay nodes and delay arcs.

Delay node: A delay node is a node in the partially time expanded network at the commodity's destination location, d_k, with a time, $late_k (l_k < late_k \leq \Delta_k)$, for commodity k. Each delay node has a commodity associated with the node because we restrict

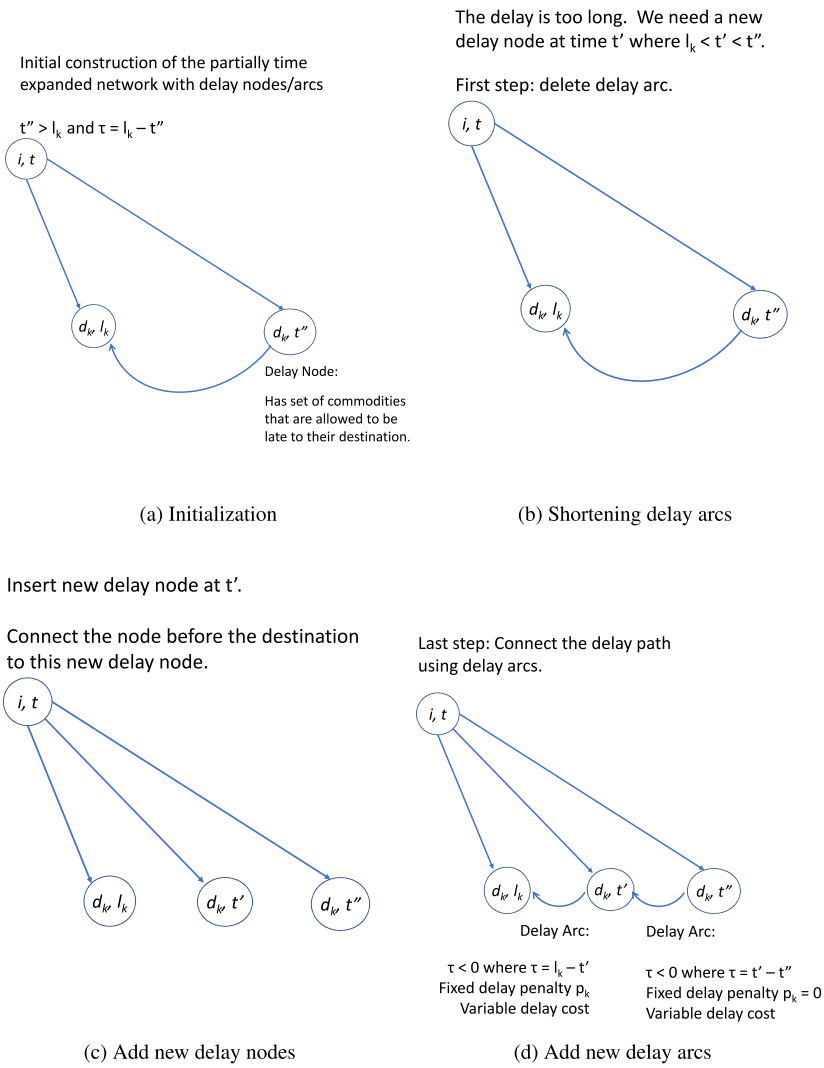


Fig. 4. Construction of delay nodes and arcs and repair-and-restore of the network to capture shorter delays as required.

this portion of the partially expanded network only to the delayed commodity. Note that there could be multiple delay nodes per commodity k.

Delay arc: A delay arc connects the delay node in the partially time expanded network to the delay node that is closest in time *before* it, or to the destination node (d_k, l_k), for the commodity. The "travel time" assigned to the delay arc is equal to the negative of the time difference between the nodes it connects. Travel times on delay arcs will always be negative, and allows constraint (4) of the minimum cost problem $SND(D_{\mathcal{T}})$ to be satisfied. To include delay arcs in the initialized network, we introduce operations 14 through 25 into Algorithm 2 to allow the initialization of delay arcs. Fig. 4 illustrates the initial construction of delay arcs, and refine and restore steps in Algorithms 9, 10, and 11.

The existence of both holdover and delay arcs indicates that there are times when it would be beneficial for a commodity to arrive late to its destination because the consolidation of commodities could result in an overall lower cost in terms of capacity availability.

4.2.2. Modeling per-unit-time vehicle-flow capacities

We define three new terms to help capture per-unit-time vehicle-flow capacities.

Cycle Time (CT_{ij}): To capture the rate at which vehicles, each with a capacity for commodities can enter a link $(i,j) \in \mathcal{A}$ in the network, we define the amount of time needed before a new vehicle can utilize an established service link. For example, a shipping lane and dock might only have space for a single ship. The ship needs to enter the shipping land, dock, unload and clear the shipping lane before another ship can utilize the shipping lane and dock. Similarly, trucks might be separated by a couple of

minutes before entering a roadway to avoid congestion on the roadway. In the former example the cycle time for the link might be quite large, and short in the latter example. The cycle time is defined per link in the service network.

Consolidation Time ($\bar{C}$): When the cycle time is short, we will be adding a large number of arcs to the partially time-expanded network $\mathcal{D}_{\mathcal{T}}$. To avoid increasing size of $\mathcal{D}_{\mathcal{T}}$, we define consolidation time. Consolidation time groups arcs with a cycle time less than the consolidation time into a single arc with a corresponding capacity increase. For example, if CT_{ij}=10 and $\bar{C}$ =10, we create a single arc at the current time with a capacity of $\frac{\bar{C}}{CT_{ij}} = 1$ (consolidation time divided by the cycle time). Another arc will be added at the current time plus the consolidation time. Fig. 5 demonstrates the addition of travel arcs to capture per-unit-time vehicle-flow capacities.

Hold Arc: Like holdover arcs, we introduce the concept of hold arcs that represent the storage of a commodity at a given node, but *must* stay at that node for a given amount of time, $\tau_{ii} > 0$. Hold arcs are added to $\mathcal{A}_{\mathcal{T}}$ as they are part of the total time it takes for the commodity to travel through the service network, and are used for accounting purposes.

4.3. Algorithm components

Algorithm 2 (CREATE-INITIAL)

Require: Directed network $\mathcal{D} = (\mathcal{N}, \mathcal{A})$, commodity set $\mathcal{K}$

for all $k \in \mathcal{K}$ **do**
  Add node(o_k, e_k) to $\mathcal{N}_{\mathcal{T}}$
  Add node(d_k, l_k) to $\mathcal{N}_{\mathcal{T}}$
end for
for all $u \in \mathcal{N}$ **do**
  Add node$(u, 0)$ to $\mathcal{N}_{\mathcal{T}}$
end for
for all $(i,t) \in \mathcal{N}_{\mathcal{T}}$ **do**
  for all $(i,j) \in \mathcal{A}$ **do**
    Find largest t' such that $(j,t') \in \mathcal{N}_{\mathcal{T}}$ and $t' \leqslant t + \tau_{ij}$ and add arc$((i,t),(j,t'))$ to $\mathcal{A}_{\mathcal{T}}$
  end for
  Find smallest t' such that $(i,t') \in \mathcal{N}_{\mathcal{T}}$ and $t' > t$ and add arc$((i,t),(i,t'))$ to $\mathcal{H}_{\mathcal{T}}$
end for
for all $k \in \mathcal{L}$ **do**
  if k has maximum delay time **then**
    $t_{max} = l_k + \Delta_k$
  else
    t_{max} = MAX_ALLOWED_DELAY ▷ Largest allowed value in optimizer
  end if
  Add delay node$(d_k, l_k + t_{max}, k)$ to $\mathcal{N}_{\mathcal{T}}$
  Add delay arc$((d_k, l_k + t_{max}),(d_k, l_k))$ to $\mathcal{A}_{\mathcal{T}}$
  for all arc$((i,t),(d_k,l_k)) \in \mathcal{A}$ **do**
    Add arc$((i,t),(d_k, l_k + t_{max}))$ to $\mathcal{A}_{\mathcal{T}}$
  end for
end for

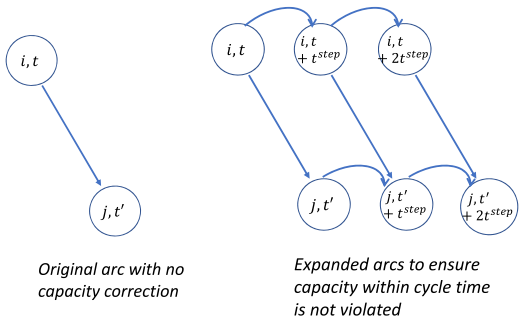
Problem 1. Problem $SND(\mathcal{D}_{\mathcal{T}})$ that solves for minimum cost flow on a partially discretized network

$$z(\mathcal{D}_{\mathcal{T}}) = min\left\{ \sum_{((i,t),(j,\bar{t}))\in\mathcal{A}_{\mathcal{T}}} f_{ij} y_{ij}^{t\bar{t}} + \sum_{k\in\mathcal{K}} \sum_{((i,t),(j,\bar{t}))\in\mathcal{A}_{\mathcal{T}}} c_{ij} q_k x_{ij}^{kt\bar{t}} \right\} \tag{1}$$

$$\text{s.t.} \sum_{((i,t),(j,\bar{t}))\in\mathcal{A}_{\mathcal{T}}\cup\mathcal{H}_{\mathcal{T}}} x_{ij}^{kt\bar{t}} - \sum_{((j,\bar{t}),(i,t))\in\mathcal{A}_{\mathcal{T}}\cup\mathcal{H}_{\mathcal{T}}} x_{ji}^{k\bar{t}t} = \begin{cases} 1 & (i,t) = (o_k, e_k), \\ -1 & (i,t) = (d_k, l_k), \\ 0 & otherwise \end{cases} \quad \forall\, k \in \mathcal{K}, (i,t) \in \mathcal{N}_{\mathcal{T}} \tag{2}$$

$$\sum_{k\in\mathcal{K}} q_k x_{ij}^{kt\bar{t}} \leqslant u_{ij} y_{ij}^{t\bar{t}} \qquad \forall\, ((i,t),(j,\bar{t})) \in \mathcal{A}_{\mathcal{T}} \tag{3}$$

$$\sum_{((i,t),(j,t'))\in\mathcal{A}_{\mathcal{T}}} \tau_{ij} x_{ij}^{kt\bar{t}} \leqslant l_k - e_k; \qquad \forall\, k \in K \tag{4}$$



(a) Initialization (b) Expanding arcs to capture per-unit-time vehicle-flow capacities

Fig. 5. Creating copies of travel arcs to capture flows per unit time.

Table 2
Seven aggregate commodity categories and their delay cost functions. Given a commodity category and the length of delay, a percentage of TEU value lost can be used to compute delay cost. The objective function of our DDD algorithm uses these loss functions to minimize delay costs when computing routes and corresponding schedules.

Import commodity category loss parameters								
Category	Subcategory	Inventory holdings (days)	Initial loss value	Loss threshold (days)	Loss adjustment	Trajectory	Eventual loss	Loss Pct
Perishables	Short-term	n.a.	1.93%	3	n.a.	Exponential	100% end of day 3	100%
	Long-term	n.a.	1.51%	5	n.a.	Exponential	100% end of day 5	100%
Just-in-time inputs		n.a.	1.11%	1	n.a.	Exponential	100% end of day 10	22.33%
Raw materials		15–30	1.75%	22	Double	Linear	100% end of 3 months	2.25%
Non-durables		4–8	0.95%	6	Double	Linear	100% end of 3 months	3.07%
Durables	Producer	4–8	0.77%	6	n.a.	Exponential	100% end of 1 month	2.10%
	Consumer	8–15	0.89%	15	Double	Linear	100% end of 3 months	1.28%

$$x_{ij}^{kt\bar{t}} \in \{0,1\} \qquad \forall\, k \in \mathcal{K}, ((i,t),(j,\bar{t})) \in \mathcal{A}_{\mathcal{T}} \cup \mathcal{H}_{\mathcal{T}} \tag{5}$$

$$y_{ij}^{t\bar{t}} \in \mathbb{N}_{\geqslant 0} \qquad \forall\, ((i,t),(j,\bar{t})) \in \mathcal{A}_{\mathcal{T}} \tag{6}$$

Problem 1 minimizes the sum of fixed costs due to the establishment of links for vehicle movements and the variable costs due to the movement of commodities. (2) describes commodity flow balance constraints on the partially expanded time-expanded network. Constraints (3) provide sufficient capacity for commodity travel between locations i and j, between time t and time t'. Other constraints include the variable ranges and potentially constraining some commodities k to not travel on a link (i,j), we set $x_{ij}^{kt\bar{t}} = 0$ for all $k \in \mathcal{K}$. Constraint (4) is added particularly for commodities that can be delayed, and thus use one or more delay arcs, appended to the end of the path, for travel. The constraint indicates that the path chosen for each commodity by $SND(\mathcal{D}_{\mathcal{T}})$, inclusive of the negative travel time delay arcs used by the commodity, should not exceed the difference between the earliest available and latest delivery times for the commodity. The output from Problem 1 is a path P_k for each commodity, consisting of both travel arcs and holdover arcs, and also includes delay arcs for commodities that require additional time to be delivered at their destination.

Problem 2. Identify Arcs to Lengthen Problem

Problem 2 identifies arcs that are 'too short', that is, the movements and associated vehicle and commodity consolidations resulting from solving $SND(\mathcal{D}_{\mathcal{T}})$ which provide a dual infeasible solution cannot be converted into a corresponding feasible solution of the same cost. Then, there exists an optimality gap between the (infeasible) dual solution and the feasible primal solution after conversion.

$$Z = min \sum_{k \in \mathcal{K}} \sum_{j=1}^{|P_k|-1} \sigma^k_{i_j,i_{j+1}} \tag{7}$$

$$\theta^k_{i_j,i_{j+1}} \geq \tau_{i_j,i_{j+1}}(1 - \sigma^k_{i_j,i_{j+1}}) \qquad \forall (i_j,i_{j+1}) \in \mathcal{A}'_{\mathcal{T}} \tag{8}$$

$$\theta^k_{i_j,i_{j+1}} \leq \tau_{i_j,i_{j+1}}(1 - \sigma^k_{i_j,i_{j+1}}) \qquad \forall (i_j,i_{j+1}) \in \mathcal{A}''_{\mathcal{T}} \tag{9}$$

$$\gamma^k_{i_j} + \theta^k_{i_j,i_{j+1}} \leqslant \gamma^k_{i_{j+1}}, \qquad \forall\, k \in \mathcal{K},\ j = 1,\ldots,|P_k|-1, \tag{10}$$

Table 3
List of data sources used to calculate the dollar value of international imports through Port Everglades on a monthly basis.

Data sources for analysis				
Id	Name	Time period	Description	Source
DS-1	Vessel Schedules	FY2017–2018	Provide the time of arrival/departure of vessels to/from PEV	PEV Harbormaster
DS-2	Per-Vessel Commodities	FY2017–2018	Provide the contents of vessels in terms of commodity type	PEV PIERS Data
DS-3	Commodity Origins	FY2017–2018	Provides the country of origin for each of the commodity types	PIERS Data
DS-4	Commodity Values	FY2018	Provides the dollar value of imports/exports for monthly imported commodity types	Census
DS-5	Port Map	2017-Current	Map of shipping port intermodal transportation networks.	Shipping Port
DS-6	Satellite Imagery	Current	Imagery of port	Google Earth
DS-7	Port Seaway GIS	2018	GIS data of seaways at the port	USACE Waterways
DS-8	Port/Operator Contracts	2019	Include demurrage rates	PEV
DS-9	Port Economic Reports	FY2017–2018	Provides monthly import/export volumes for empty/loaded TEU	PEV

$$e_k \leqslant \gamma^k_{o_k} \qquad \forall\, k \in \mathcal{K} \tag{11}$$

$$\gamma^k_{|P_k|-1} + \theta^k_{i_{|P_k|-1},d_k} \leqslant l_k \qquad \forall\, k \in \mathcal{K}, \tag{12}$$

$$\gamma_i^{k_1} = \gamma_i^{k_2}, \qquad \forall\, (k_1,k_2) \in J_{((i,t),(j,t'))}, \quad \forall\, ((i,t),(j,t')) \in \bar{\mathcal{J}} \tag{13}$$

$$\gamma^k_{i_j} \geqslant 0, \qquad \forall\, k \in \mathcal{K}, j = 1,\dots,|P_k|-1, \tag{14}$$

$$\theta^k_{i_j,i_{j+1}} \geqslant \bar{\tau}^k_{i_j,i_{j+1}}, \qquad \sigma^k_{i_j,i_{j+1}} \in \{0,1\}, \quad \forall\, k \in \mathcal{K}, j = 1,\dots,|P_k|-1 \tag{15}$$

Let P_k be the path of commodity k found from $SND(\mathcal{D}_\mathcal{T})$, represented by $P_k = \{(o_k,e_k) = i^k_1, i^k_2, i^k_3, \dots, i^k_{|P_k|} = (d_k,l_k)\}$ that is the sequence of p_k nodes in the path. Note that this path is inclusive of delay arcs if some commodities are delivered late; in which case, the last arc $(i^k_{|P_k|-1}, i_{|P_k|^k})$ (and possibly the penultimate arcs of length zero) is a negative length delay arc. Let $\bar{\tau}_{i_j i_{j+1}}$ be the travel time modeled in the current partially expanded time–space network $\mathcal{D}_\mathcal{T}$ for arc (i_j, i_{j+1}). Let J_a be the set of all pairs of commodities k that use arc $a \in \mathcal{A}_\mathcal{T}$. Also, let $\bar{J}$ be the set of arcs that are shared by multiple commodities during travel, that is, they belong to multiple paths P_k. We define variables $\gamma^k_{i_j}$ as the dispatch time for commodity k from node i_j, $\theta^k_{i_j i_{j+1}}$ as the maximum travel time on (i_j, i_{j+1}) for commodity k's path to be feasible, and $\sigma^k_{i_j i_{j+1}}$ as a binary variable that is 1 if arc (i_j, i_{j+1}) is allowed to be too short when taken by commodity k.

The formulation (7)–(15) takes the solutions from $SND(\mathcal{D}_\mathcal{T})$ and computes if these solutions are only feasible when some travel or holdover arc (but not delay arc) is 'too short', that is, the travel time has to necessarily be lower than the true travel time. Boland et al. (2017) formulate this problem for arcs with non-negative travel times (set $\mathcal{A}'_\mathcal{T}$). Thus, constraint (8) ensures that sigma takes on value 1 only if $\theta^k_{i_j,i_{j+1}}$ is forced to take a value less then the true travel time $\tau_{i_j,i_{j+1}}$ for all arcs $a \in \mathcal{A}'_\mathcal{T}$ with non-negative travel times. We extend this to delay arcs, which have non-positive travel times (set $\mathcal{A}''_\mathcal{T}$), by adding a corresponding constraint (9) for the negative length delay arcs. Constraints (10) ensure that the dispatch times γ for each commodity on each arc differ by at least the travel time θ modeled for that arc. Constraints (11) and (12) ensure that the dispatch times lie between the earliest available times and latest delivery times of each commodity, with respect to the travel times modeled in $\mathcal{D}_\mathcal{T}$. Constraints (13) ensures that commodities traveling together have the same dispatch times, and (14) and (15) specify the variable ranges. The objective (7) requires that the least number of arcs in the network are 'too short' and are required to be lengthened in the next step (Algorithm 3) for feasibility.

Algorithm 3 LENGTHEN-ARC($(i,t),(j,t')$)

Require: Arc$((i,t),(j,t')) \in \mathcal{A}_\mathcal{T}$
1: REFINE$(j, t+\tau_{ij})$
2: RESTORE$(j, t+\tau_{ij})$.

Algorithm 4 REFINE((i, t^i_{new}))

Require: Node $i \in \mathcal{N}$; time point $t^i_{new} \in \mathcal{T}_i$ with $t^i_k < t^i_{new} < t^i_{k+1}$
1: Add node (i, t^i_{new}) to $\mathcal{N}_\mathcal{T}$;
2: Delete arc $((i,t^i_k),(i,t^i_{k+1}))$ from $\mathcal{A}_\mathcal{T}$
3: Add arcs $((i,t^i_k),(i,t^i_{new}))$ and $((i,t^i_{new}),(i,t^i_{k+1}))$ to $\mathcal{A}_\mathcal{T}$
4: **for all** $((i,t^i_k),(j,t)) \in \mathcal{A}_\mathcal{T}$ **do**
5: Add arc $((i,t^i_{new}),(j,t))$ to $\mathcal{A}_\mathcal{T}$
6: **end for**

Algorithm 5 RESTORE((i, t^i_{new}))

Require: Node $i \in \mathcal{N}$; time point $t^i_{new} \in \mathcal{T}_i$ with $t^i_k < t^i_{new} < t^i_{k+1}$
1: **for all** $((i, t^i_k), (j, t)) \in \mathcal{A}_\mathcal{T}$ **do**
2: Set $t' = \arg\max \{s \in \mathcal{T}_j | s \leqslant t^i_{new} + \tau_{ij}\}$
3: **if** $t' \neq t$ **then**
4: Delete arc $((i, t^i_{new}), (j, t))$ from $\mathcal{A}_\mathcal{T}$
5: Add arc $((i, t^i_{new}), (j, t'))$ to $\mathcal{A}_\mathcal{T}$
6: **end if**
7: **end for**
8: **for all** $((j, t), (i, t^i_k)) \in \mathcal{A}_\mathcal{T}$ such that $t + \tau_{ji} \geqslant t^i_{new}$ **do**
9: Delete arc $((j, t), (i, t^i_k))$ from $\mathcal{A}_\mathcal{T}$
10: Add arc $((j, t), (i, t^i_{new}))$ to $\mathcal{A}_\mathcal{T}$
11: **end for**

Algorithms 3, 4 and 5 are used in conjunction, once Problem 2 identifies arcs (i_j, i_{j+1}) that need to be lengthened. Because the existing solution forces these arcs to be shorter than the true travel times, it results in a solution that is in reality infeasible. Algorithm 3 lengthens these arcs to their true travel times $\tau_{i_j, i_{j+1}}$ in $\mathcal{D}_\mathcal{T}$. Note that a 'short' arc $((i, t), (j, t'))$, with $t' < t + \tau_{ij}$ may be lengthened at most once, to its true travel time. The two steps in Algorithm 3 consist of Algorithms 4 and 5. Algorithm 4 introduces a new time point for location j in $\mathcal{D}_\mathcal{T}$, specifically, at $t + \tau_{ij}$, while adding holdover arcs to preserve Properties 1–4. Algorithm 5 removes the too short travel arc, and replaces it with the arc $((i, t), (j, t+\tau_{ij}))$ that captures the true travel time, again while preserving Properties 1–4.

Algorithm 6 detects if per-unit-time vehicle-flow capacities on specific links $a \in A$ in the flat network are violated by the current solution to $SND(\mathcal{D}_\mathcal{T})$, based on the consolidation time and cycle times of the links. If yes, Algorithms 7 and 8 refine to partially expanded network $\mathcal{D}_\mathcal{T}$ further to ensure that the rate of flow on each link cannot exceed the per-unit-time vehicle-flow capacity.

Algorithm 6 DETECT-CAPACITY-VIOLATIONS

Require: Solution from $SND(\mathcal{D}_\mathcal{T})$, consisting of path P_k for all $k \in K$
1: **for all** $(i, j) \in \mathcal{A}$ **do**
2: $J'_{ij} = \{k : ((i, t), (j, \bar{t})) \in P_k \quad \forall t, \bar{t}\}$
3: **end for**
4: **for all** $(i, j) \in \mathcal{A}$ for which $\sum_{((i,t),(j,t'))\in\mathcal{A}_\mathcal{T}} \sum_{k\in J'_{ij}} q_k x^{kt\bar{t}}_{ij} > u_{ij}$ **do**
5: $t^{step}_{ij} = max\{CT_{ij}, \bar{C}\}$
6: $st_i = argmin_t\{\gamma^k_{(i,t)} : \gamma^k_{(i,t)} > 0 \quad \forall k \in \mathcal{K}\}$
7: Divide the timeline of $\mathcal{D}_\mathcal{T}$ from st_i into intervals of size l_{ij}
8: Compute utilization of $(i, j) \in \mathcal{A}$ in each interval $(t_1, t_2) = (t_1 + t^{step}_{ij})$ as:
9: $\sum_{((i,t'_1),(j,t'_2))\in\mathcal{A}_\mathcal{T}} \sum_{k\in J'_{ij}} q_k x^{kt\bar{t}}_{ij}$ for $t'_1 \geq t_1$ and $t'_2 \leq t_2$
10: **if** utilization exceeds per-unit-time vehicle-flow capacity **then**
11: ENSURE_CAPACITY $(((i, t_1), (j, t_2)), \bar{C})$
12: **end if**
13: **end for**

Algorithm 6 examines each arc $(i, j) \in \mathcal{A}$ and detects if the total flow on it, summed over all periods of time, exceeds the per-unit time capacity of the arc (line 4). If that occurs, the first time from which the arc $(i, j) \in \mathcal{A}$ is installed is divided into intervals equal to the interval length t^{step}_{ij} and checked if per-unit time capacity in each interval is violated. If it is violated, Algorithm 7 is invoked to further expand the partially expanded time–space network $\mathcal{D}_\mathcal{T}$ by adding multiple arcs $((i, t), (j, t + \tau_{ij}))$ for various values t in order to ensure that per-unit time capacities are not violated.

Algorithm 7 ENSURE-CAPACITY$(((i, t), (j, t')), \bar{C})$

Require: Arc$((i, t), (i, t')) \in \mathcal{A}_\mathcal{T}$)
1: Find t_{new} by finding the last outflow hold arc from (i, t)
2: Find $t^{step}_{ij} = max(\bar{C}, CT_{ij})$
3: **if** $(i, t_{new}) \notin \mathcal{N}_\mathcal{T}$ **then**
4: (i, t') = REFINE-CAPACITY$((i, t), t_{new}, t^{step}_{ij}, capacity)$
5: RESTORE$((i, t'), t_k, t_{new})$.
6: **end if**
7: **for** i = 1 to utilization of (i, j) **do**
8: $t_{new} = t' + t^{step}_{ij}$
9: (i, t') = REFINE-CAPACITY$((i, t'), t_{new}, t^{step}_{ij}, capacity)$
10: RESTORE$((i, t'), t_k, t_{new})$.
11: **end for**

Algorithm 8 REFINE-CAPACITY($(i,t), t_{new}, t_{ij}^{step}, capacity, (i,j)$)

Require: Node $i \in \mathcal{N}$; time point $t_{new}^i \in \mathcal{T}_i$ with $t_k^i < t_{new}^i$; $(i,j) \in \mathcal{A}$

if $t_{new}^i > t_{k+1}^i$ and $(t_{new}^i - t_{k+1}^i) < t_{ij}^{step}$ **then**
  for all $((i,t_{k+1}^i),(j,t)) \in \mathcal{A}_\mathcal{T}$ **do**
    if $((i,t_{k+1}^i),(j,t)) = (i,j)$ **then**
      Delete arc $((i,t_{k+1}^i),(j,t))$ from $\mathcal{A}_\mathcal{T}$
    end if
  end for
end if
Add node (i,t_{new}^i) to $\mathcal{N}_\mathcal{T}$;
if $((i,t_k^i),(i,t_{k+1}^i)) \in \mathcal{H}_\mathcal{T}$ **then**
  Delete arc $((i,t_k^i),(i,_{k+1}^i))$ from $\mathcal{H}_\mathcal{T}$
  Add arc $((i,t_{new}^i),(i,t_{k+1}^i))$ to $\mathcal{H}_\mathcal{T}$
else
  Delete arc $((i,t_k^i),(i,_{k+1}^i))$ from $\mathcal{A}_\mathcal{T}$
  Add hold arc $((i,t_{new}^i),(i,t_{k+1}^i),t_{k+1} - t_{new})$ to $\mathcal{A}_\mathcal{T}$
end if
Delete arc $((i,t_k^i),(i,t_{k+1}^i))$ from $\mathcal{A}_\mathcal{T}$
if $(t_{new}^i - t_k^i) <= t_{step}$ **then**
  Add hold arc $((i,t_k^i),(i,t_{new}^i),t_{new} - t_k)$ to $\mathcal{A}_\mathcal{T}$
else
  Add arc $((i,t_k^i),(i,t_{new}^i))$ to $\mathcal{H}_\mathcal{T}$
end if
for all $((i,t_k^i),(j,t)) \in \mathcal{A}_\mathcal{T}$ **do**
  Add arc $((i,t_{new}^i),(j,t))$ to $\mathcal{A}_\mathcal{T}$
  Set w_{ij} for arc $((i,t_{new}^i),(j,t))$
end for
return (i,t_{new}^i)

Table 4
Data attributes for G_{Trans} and data sources used to populate them.

Transportation network (G_{Trans}) attributes		
Attribute	Description	Sources
Semantic		
$\texttt{rdf:type}_i$	The type of network component $i \in V_{Trans} \cup E_{Trans}$ as defined by an critical infrastructure ontology.	DS-6, DS-7
Queueing		
$c_{v,e}$	Capacity, the number of entities (e.g. TEU on roadways, vessels on seaways) that can be simultaneously served at a vertex or edge.	DS-7
s_v	Number of minutes to process an entity at vertex.	SMEs
t_e	Travel time in minutes along an edge, computed using geodesic distance and stakeholder feedback	DS-7, SMEs
q_v	The maximum number of entities that can be stored while waiting for service at a vertex.	SMEs
Extended Queueing		
$\texttt{cost}_{\texttt{v,e}}$	The fixed cost of using a vertex or edge.	SMEs
$\texttt{holding\_cost}_{\texttt{v,e}}$	The fixed cost incurred by an entity held in storage	SMEs
$\texttt{holding\_time}_{\texttt{v,e}}$	The time an entity at a vertex or edge must spend in storage before being processed.	SMEs
Extended Optimization		
$\texttt{cycle\_time}_{\texttt{v,e}}$	The time to wait before trying to use a vertex or edge currently at full capacity	SMEs
Spatial		
lat_v, lon_v	Latitude and longitude of location $v \in V_{Trans}$.	DS-5, DS-6, DS-7

Problem 3. Minimize Delay Problem

$$Z = \min \sum_{k \in \mathcal{L}} (1 + delay\_penalty\_rate_k)(\gamma^k_{|P_k|-1} - \alpha^k_{|P_k|-1,|P_k|}) \tag{16}$$

$$\gamma^k_{i_{|P_k|-2}} \geqslant \alpha^k_{i_{|P_k|-2},i_{|P_k|-1}}, \qquad \forall\, k \in \mathcal{L} \tag{17}$$

$$\gamma^k_{|P_k|-2} + \tau_{i_{|P_k|-2},i_{|P_k|-1}} \leqslant l_k + \Delta_k \qquad \forall\, k \in \mathcal{L}' \tag{18}$$

For commodities $k \in \mathcal{L}$ that are allowed to be delayed, we define variables α^k_{ij} for delay arc (i, j), to capture the minimum arrival time of commodity k along the arc. Specifically, if γ^k_i is the dispatch time of commodity k from node i, we have $\alpha^k_{ij} = \tau_{ij} + \gamma^k_i$. Recollect that the path of the commodity is represented by $P_k = \{(o_k, e_k) = i^k_1, i^k_2, i^k_3, \ldots, i^k_{|P_k|} = (d_k, l_k)\}$ that is the sequence of $|P_k|$ nodes in the path. The last arc $(i^k_{|P_k|-1}, i_{|P_k|})$ is the delay arc. If the lengths of the delay arcs have been set too long, we minimize that length using (16), subject to constraints (17) that ensure that α is the minimum arrival time of commodity k along the arc, and constraints (18) that require the dispatch time to be such that the commodity cannot be delayed by more than the maximum allowed delay at the destination.

Algorithm 9 SHORTEN-DELAY-ARC$((i,t),(i,t')), t^i_{new})$

Require: Arc$((i,t),(i,t')) \in \mathcal{A}_{\mathcal{T}}$); time point $t^i_{new} \notin \mathcal{T}_i$

1: Find $((i,t^i_{k+1}),(i,t^i_k)) \in \mathcal{A}_{\mathcal{T}}$ such that $t^i_k < t^i_{new} < t^i_{k+1}$
2: REFINE_DELAY_ARC$((i,t',\mathcal{K}'),t_k,t_{new},t_{k+1})$
3: RESTORE_DELAY_ARC$((i,t_k,\mathcal{K}'),t_{new})$.

Algorithm 10 REFINE_DELAY_ARC$((i,t',\mathcal{K}'))$

Require: Delay Node $(i,t',\mathcal{K}') \in \mathcal{N}_{\mathcal{T}}$; time point $t^i_{new} \notin \mathcal{T}_i$ with $t^i_k < t^i_{new} < t^i_{k+1}$

1: Add delay node $(i,t^i_{new},\mathcal{K}')$ to $\mathcal{N}_{\mathcal{T}}$;
2: Delete delay arc $((i,t^i_{k+1}),(i,t^i_k))$ from $\mathcal{A}_{\mathcal{T}}$
3: Add delay arcs $((i,t^i_{k+1}),(i,t^i_{new}))$ and $((i,t^i_{new}),(i,t^i_k))$ to $\mathcal{A}_{\mathcal{T}}$
4: **for all** $((j,t),(i,t^i_{k+1})) \in \mathcal{A}_{\mathcal{T}}$ **do**
5: Add arc $((j,t),(i,t^i_{new}))$ to $\mathcal{A}_{\mathcal{T}}$
6: **end for**

Algorithm 11 RESTORE_DELAY_ARC$((i,t^i_k,\mathcal{K}'))$

Require: Delay Node $(i,t^i_k,\mathcal{K}') \in \mathcal{N}_{\mathcal{T}}$; time point $t^i_{new} \in \mathcal{T}_i$ with $t^i_k < t^i_{new}$

1: **for all** $((j,t),(i,t^i_k)) \in \mathcal{A}_{\mathcal{T}}$ such that $t + \tau_{ji} \geqslant t^i_{new}$ **do**
2: Delete arc $((j,t),(i,t^i_k))$ from $\mathcal{A}_{\mathcal{T}}$
3: **end for**

Problem 4. Construct Feasible Solution Problem

$$Z = \min \sum_{((i,t),(j,t')) \in \bar{\mathcal{J}}} \sum_{(k_1,k_2) \in J_{((i,t),(j,t'))}} \delta^{k_1k_2}_{ijt} \tag{19}$$

$$\gamma^k_{i_j} + \tau_{i_j,i_{j+1}} \leqslant \gamma^k_{i_{j+1}}, \qquad \forall\, k \in \mathcal{K}, j = 1, \ldots, |P_k| - 1, \tag{20}$$

$$e_k \leqslant \gamma^k_{o_k} \qquad \forall\, k \in \mathcal{K}, \tag{21}$$

$$\gamma^k_{i_{|P_k|-1}} + \tau_{i_{|P_k|-1},d_k} \leqslant l_k \qquad \forall\, k \in \mathcal{K}, \tag{22}$$

$$\delta^{k_1k_2}_{ijt} = \gamma^{k_1}_i - \gamma^{k_2}_i, \quad \forall\, (k_1,k_2) \in J_{((i,t),(j,t'))}, \quad \forall\, ((i,t),(j,t')) \in \bar{\mathcal{J}}, \tag{23}$$

$$\delta^{k_1k_2}_{ijt} = \gamma^{k_2}_i - \gamma^{k_1}_i, \quad \forall\, (k_1,k_2) \in J_{((i,t),(j,t'))}, \quad \forall\, ((i,t),(j,t')) \in \bar{\mathcal{J}}, \tag{24}$$

$$\gamma^k_{i_j} \geqslant 0, \qquad \forall\, k \in \mathcal{K}, j = 1, \ldots, |P_k| \tag{25}$$

If the 'Identify Arcs to Lengthen' problem does not have an objective of zero, it means that the solution cannot be converted to an equivalent feasible solution with the same cost (if it can be converted, we would have achieved the optimal solution). The 'Construct Feasible Solution Problem', (19)–(25) constructs a feasible solution of greater cost than the lower bound, as a linear program instead of an integer program. Similar to Problem 2, the 'Identify Arcs to Lengthen' problem, we define the variables $\gamma^k_{i_j}$

that represent dispatch times for each commodity k at each node in its path P_k with respect to the *true* travel times between locations. These constraints ensure that the commodity departs after its availability time e_k and arrives at the destination by the delivery time (note that delays are negative so the commodity will still arrive by l_k) and commodities' consolidations discovered in the optimal solution to $SND(D_{\mathcal{T}})$ are preserved to the extent possible. $\delta_{ijt}^{k_1 k_2}$ captures the difference in dispatch times between commodities k_1 and k_2 on arc connecting locations i and j. The objective (19) minimizes the total difference in dispatch times required to create a feasible solution. Also observe that this problem is always feasible because the length of the path of each commodity k is always less than the difference $l_k - e_k$. This solution, which forms an upper bound, also allows us to evaluate the optimality gap when compared to the solution of $SND(D_{\mathcal{T}})$.

Fig. 2 and Algorithm 1 summarize the working of our enhanced dynamic discretization discovery algorithm, which first captures if the partially discretized network requires arcs to be lengthened for a feasible schedule; next if it captures delays required for disruption management, followed by if per-unit-time vehicle-flow capacities are captured appropriately, and expands the network to enforce capacities if otherwise. Upon termination of the algorithm, we would have found an optimal solution, or a close-enough optimal solution as measured by a comparison between the upper and lower bounds.

5. Economic model

The objective of the optimization function is based on economic losses due to delays from disruptions that cause containers to arrive later than their *Latest Delivery Time (LDT)*. The economic cost of a TEU arriving late is a function of the length of the delay and the commodity category to which the TEU contents belong.

Our approach uses Vessel Schedules (DS-1), Per-Vessel Commodities (DS-2), and the Commodity Origins (DS-4) datasets to generate a schedule of imported loaded containers, *Twenty-foot Equivalent Units (TEU)*, that move from a berth, through the port terminal transportation network, onto a nearby highway. More details about the data fusion process and datasets may be found in Section 6. Each TEU is represented as a commodity ($k \in K$) in the optimization and assigned an origin and destination as well as an EAT and LDT as described in Section 4. In order to support the economic analysis, a single *Harmonized System (HS)* 2-digit code for traded products, along with country of origin is assigned to each commodity. In this manner, our intent is to use this formalism to align the senses of the word *commodity* from the optimization and economics domains respectively.

Our model may overestimate traffic congestion as it assumes that each TEU has a unique HS 2-digit code and vehicle. This is not the case in ports where a single TEU may be packed with multiple commodities and a Class 8 truck may carry multiple TEU. On the other hand, our model only considers traffic generated from imports and so may underestimate traffic congestion in this sense as we do not model exports. While we leave such details to future work, we still believe our approach represents a meaningful contribution to estimate economic losses from port disruptions.

The cost penalties for import delays differ across commodity categories. Table 2 presents a classification of import commodities and parameters associated with these penalties. Our analysis and valuation of import delay cost penalties is related to the work of Hummels et al. (2007) and Minor (2013), who estimated time delay costs for commodity groups across countries using extensive databases and sophisticated econometric methods. Note, however, the work of these researchers was oriented toward short-term (1–2 days) delays in cargo transfers by sea and air, relating to the normal course of business, such as inspections, paperwork, and on-loading/off-loading inefficiencies, as opposed to medium to longer-term disruptions analyzed in this paper. Thus, their estimates are taken as lower bounds applicable to the initial time stages of the disruptions analyzed here. Note also that the Hummels and Minor analyses also implicitly take inventories into account, but only for those shorter time periods. We view these estimates to be superior to using standard demurrage rates, primarily because the latter rates are relatively low and expressed as a standard per container value and hence ignore the economic value of the cargo itself and the specific characteristics of the commodities.

The adaptation of the Hummels/Minor estimates are as follows. Base estimates "Ad Valorem Value of (Cargo Delay) of One Day" are presented in Minor, Appendix A-1 (Minor, 2013). Those estimates are expressed in percentage terms of import values for a one day delay by major GTAP commodity group and are aggregated to 7 aggregate commodity categories presented in Column 3 of Table 2.[1] The values range from zero for Perishable and Raw Material categories to 2.93% of the value of the cargo for Vegetables/Fruits/Nuts. Other high values include Refined Petroleum (1.99%), Ferrous Metals (1.96%), Motor Vehicles and Parts (1.77%), and Food Products (1.63%) in the more detailed categories of the source material. The estimates in the Minor report are used as a starting point. Because our time periods of analysis are longer than that underlying the basic estimates, we insert consideration of Inventory Holdings in terms of number of days of supply in Column 2 of Table 2 (note that inventories are assumed to be essentially zero for the first three commodity categories). Raw material stockpiles are often large, and inventories for the other three categories range from 4 to 15 days. Note also that the category of consumer durables pertains not to the ultimate consumer but rather to wholesale and retail outlets; hence, the specification of inventory levels for this category as well. The Loss Threshold column reflects the date at which the perishables rot or the inventories run out. It is the end of Day 1 for JIT commodities, and the average of the Inventory Holdings values for the other commodity categories. This is the time at which the Loss Adjustment—a doubling of Initial Loss Values—affects the linear adjustment cases. The exponential trends serve the purpose of adjustments and apply for the Perishable categories, JIT inputs, and Producer Durables after Day 1. See columns 5, 6, 7 for the Loss Adjustment, Loss Trajectory, and Eventual Loss Value specifications, respectively. Loss functions for each of these commodity categories are shown in Fig. 6.

[1] GTAP stands for Global Trade Analysis Project, which consists of extensive databases on economic activity of all countries in the world with an emphasis on import and export linkages between them.

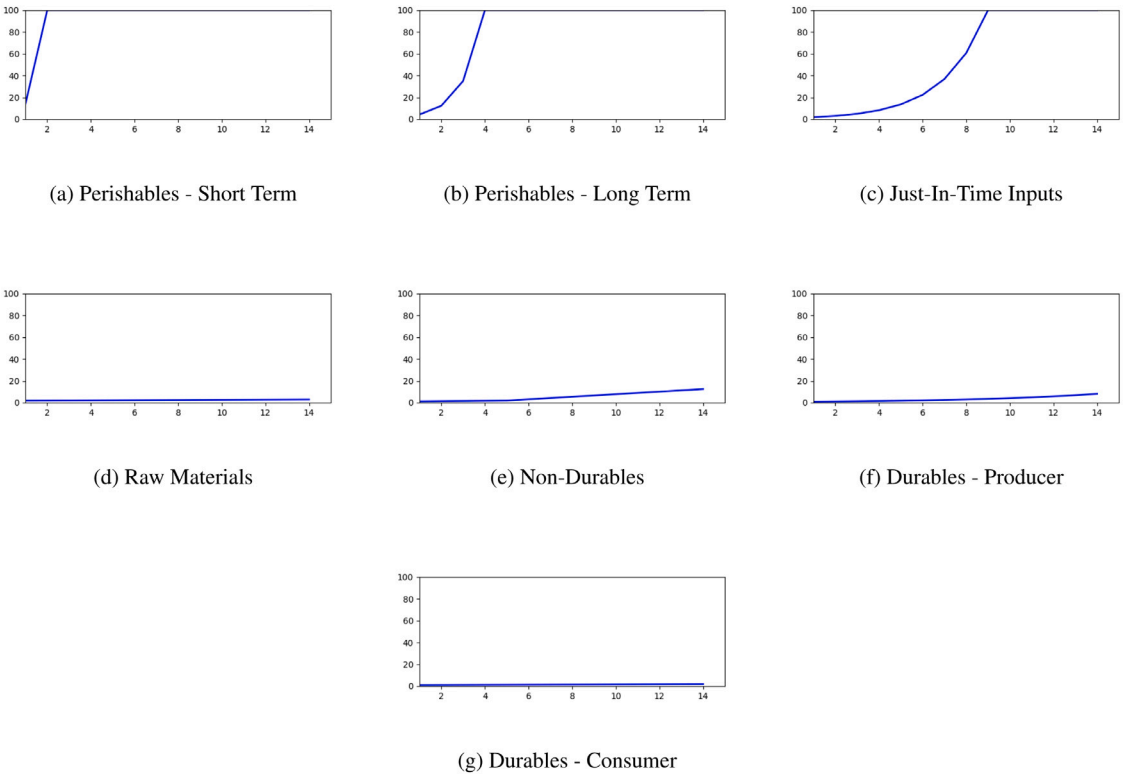


Fig. 6. Loss functions for the seven aggregate commodity categories used in the study. The plots illustrate percentage of total TEU value lost over a period of 15 days.

There are two categories of perishables, whose values plummet exponentially to zero, i.e., the total loss reaches the total cost of the cargo, after the threshold number of days. Just-in-Time inputs are needed by day 1, and the initial cost penalty at the threshold rises exponentially to complete loss after 10 days. The losses are exacerbated because customer inventory holdings of these three commodity categories are close to nil.

Raw material inventories are typically substantial, and we estimate the initial loss threshold at 22 days (halfway between the inventory holdings range). The cost penalties, however, are fairly linear after doubling them when the inventories run out, up to a total value at the end of 3 months. Non-durable inventories are typically only 4 to 8 days, and we estimate the initial loss threshold at the mean of 6 days. Again, the cost penalty is linear prior to and after the loss adjustment doubling, and we also assume total loss reaches total value of the cargo at the end of 3 months.

Durables are broken into two categories. Producer durable inventory holdings are typically only 4 to 8 days, and again we take the mean of six days for the loss threshold. The loss trajectory is assumed to be exponential. With this loss trajectory, the delay cost is 2.3 times of the Day 1 initial loss on Day 6. Retailers and wholesalers of consumer durables typically have between 8 and 15 days of inventory, but their initial loss threshold is actually past the inventory limit because they are not a production line operation that comes to a halt or requires expensive input substitution of the commodity not in stock. Moreover, consumer purchasing delays do not cause immediate significant losses, and thus we assume the loss trajectory is linear, and, in this case, not a complete loss until 3 months have passed.

Recent work by Rose et al. (2018) and Wei et al. (2020) indicates considerable customer resilience when imports are disrupted. This refers to a broad range of resilience tactics that can be utilized to cushion the negative impact of the disruption (e.g. input conservation, input substitution, relocation to branch plants where the commodity is available). Hence, we perform sensitivity tests in which we reduce the cost penalty by 50% for the last four categories in Table 2. We also perform a sensitivity analysis of a 50% increase in the cost penalties in consideration of our conservative assumptions underlying the adapted estimates presented in Table 2.

Translating delay costs into the optimization model. The loss functions are used by our optimization algorithm when a delay arc is added for a commodity $k \in L$ that is allowed to be delayed. Given the delay time, provided by the absolute value of the delay arc's travel time, and HS 2-digit code of commodity k, a percentage of value lost coefficient is calculated by the loss functions described in Table 2. Delay cost is given by the product of this coefficient and the dollar value for k's HS 2-digit code averaged over all countries of origin. Note that the value of commodities with the same HS 2-digit code may vary from month to month and our analysis framework takes this into account. For instances where a given HS 2-digit code maps to multiple categories in our

Table 5
Data sources for commodity attributes.

Commodity (K) attributes		
Attribute	Description	Sources
Optimization		
o_k	Origin of commodity k in $V[G_{Trans}]$.	DS-1, DS-5
d_k	Destination of commodity k.	DS-5
e_k	*Earliest Arrival Time (EAT)* of k at its origin.	DS-1
l_k	*Latest Departure Time (LDT)* of k at its destination.	SMEs
Economic		
`country`(k)	The country of origin for commodity k	DS-3
`hs2`(k)	The HS 2-digit code for commodity k	DS-2
`month`(k)	The month in which commodity k arrives at port	DS-1

classification, we interpolate the cost with a weighted sum of delay costs across the relevant classification categories. More details about how the dollar values for each HS-2 code are estimated may be found in Section 6.2.

6. Data analysis and fusion

This section provides a detailed discussion of data processing to estimate commodities imported through Port Everglades; the dollar value per TEU; and the calibration, verification, and validation of the optimizer relative to generated inputs. The data sources used to instantiate inputs to the optimizer—the transportation graph and commodity flows—are listed in Table 3.

6.1. Data fusion for optimizer inputs

The current implementation of our enhanced DDD algorithm takes two types of inputs: a transportation network and a set of commodities for which to compute an optimal schedule.

The graph G_{Trans} is an L-space representation in which "stops or stations are vertices and two vertices are connected if they are consecutive on an arbitrary route" (Sienkiewicz and Hołyst, 2005; Von Ferber et al., 2009). The attributes in Table 4 list transportation network attributes and data sources used to populate them. Attributes are categorized into semantic, queueing, and spatial categories. *Semantic attributes* allow one to conduct analyses relative to graph component types defined within an ontology. An ontology for the transportation network defines concepts and roles that are used to specify types for vertices and edges. A more in-depth discussion of this approach, including description logics, ontologies, and graph theory, may be found in Cheh et al. (2015). *Queueing attributes* allow stakeholders to interpret G_{Trans} as a queueing network and thereby simulate the movement of vessels and containers over time. By assigning attributes for capacity, service/travel time, queue length, and queuing discipline, a queueing network can be instantiated. More details about this approach may be found in Weaver et al. (2019). Finally, spatial attributes enable stakeholders to conduct risk assessments based on geographic regions of interest (e Silva et al., 2019). Through including latitude and longitude, we interpret and operate upon G_{Trans} as a *spatial network*, a network in which a metric is defined over the vertices (Barthélemy, 2011). These attributes are used to process G_{Trans} into the 'flat' network $D = (N, A)$ for our DDD algorithm.

Commodities K encode *Twenty-foot Equivalent Units (TEU)* with attributes for the origin (o_k), destination (d_k), *Earliest Arrival Time (EAT)* (e_k), and *Latest Departure Time (LDT)* (l_k). Table 5 lists commodity attributes and data sources used to populate them. The remainder of this subsection describes the data fusion tasks used to generate K. To compute o_k, e_k, and `hs2`(`k`) a schedule of vessel calls is constructed by extracting and aligning vessel arrivals from the Vessel Schedules (DS-1) and Per-Vessel Commodities (DS-2) datasets.

Commodity arrivals. Imported commodities, and the vessels upon which they arrive, are defined by the Per-Vessel Commodities dataset (DS-2). A commodity's origin, EAT, and LDT are a function of the arrival of the vessel containing that commodity. Specifically, a commodity's origin in the transportation network (o_k) must equal its vessel's berth in the Vessel Schedule (DS-1). A commodity's EAT must be the time at which that vessel arrives at the berth. Finally, a commodity's LDT is based on its EAT and expected dwell time informed by demurrage rates in shipping port contracts (DS-9).

Both DS-1 and DS-2 are used to instantiate a set of vessel arrivals (VA_{DS-1} and VA_{DS-2} respectively). In order to construct a set of vessel arrivals used as input for our analysis, we compute the intersection ($\text{VA}_I = \text{VA}_{DS-1} \cap \text{VA}_{DS-2}$). Equality for this intersection was defined in terms of normalized Vessel Name arrival/departure dates that overlapped within 1 day of each other. Table 6 provides the results of this fusion approach for FY2018 imports at Port Everglades.

Table 6 illustrates that between 83%–92% of vessel arrival events DS-1 are represented in intersection I. In addition, between 55%–71% of the vessel arrival events occurring within the DS-2 are represented in set I. However, when measured in terms of total TEU imported $|I|_{TEU}$ rather than in terms of vessel arrivals, the intersection of the two datasets captures 87%–97% of the TEUs in DS-2. This indicates that for a study focusing on TEU imported in FY2018 (e.g. Section 7), our data fusion approach is adequate. Further research into better data fusion algorithms, though outside the scope of this paper, could improve results to support higher fidelity modeling of port operations.

Table 6
Results of mapping `Vessel Arrivals` from Harbormaster Vessel Calls (DS-1) and PIERS vessel commodity data. When measured in terms of number of imported TEU, the intersection of the two datasets (I) captures 87%–97% of the imported container volumes in FY2018.

Analysis of FY2018 vessel arrival fusion: Datasets 1 & 2				
Month	VA_I	$\frac{\lvert\text{VA}_{DS-1}-VA_I\rvert}{\lvert\text{VA}_{DS-1}\rvert}$	$\frac{\lvert\text{VA}_{DS-2}-\text{VA}_I\rvert}{\lvert\text{VA}_{DS-2}\rvert}$	$\frac{\lvert\text{VA}_{DS-2}-\text{VA}_I\rvert_{TEU}}{\lvert\text{VA}_{DS-2}\rvert_{TEU}}$
10/17	136	0.13	0.32	0.1
11/17	148	0.12	0.29	0.09
12/17	141	0.14	0.37	0.13
1/18	130	0.13	0.33	0.09
2/18	126	0.15	0.45	0.1
3/18	133	0.14	0.41	0.08
4/18	127	0.15	0.4	0.12
5/18	137	0.11	0.33	0.05
6/18	145	0.1	0.36	0.03
7/18	134	0.13	0.38	0.04
8/18	153	0.08	0.39	0.1
9/18	114	0.17	0.42	0.07

6.2. Dollar value per TEU

The delay cost functions in Section 5 describe the percentage of total TEU value lost due to arriving at its destination after its LDT. We use Eq. (26) to compute USD(k), the estimated dollar value of a commodity $k \in K$. The term $\text{TotalImportsUSD}_{\text{hs2}(k)}^{\text{country}(k)}$ refers to the total value of goods with a given HS code from a given country of origin. This is given by the Commodity Values (DS-4) dataset provided by the US Census Trade Online Database. Similarly, the term $\text{TotalImportedTEU}_{\text{hs2}(k)}^{\text{country}(k)}$, refers to the total number of full (non-empty) TEU imported with a given HS code and country of origin. This is given by the Per-Vessel Commodities (DS-2) dataset.

$$\text{USD}(k) = \frac{\text{TotalImportsUSD}_{\text{hs2}(k)}^{\text{country}(k)}}{\text{TotalImportedTEU}_{\text{hs2}(k)}^{\text{country}(k)}}, k \in K \tag{26}$$

Eq. (26) requires that each commodity have a country of origin. This country is probabilistically assigned to a commodity based on DS-3. By sampling an empirical distribution for a given month and HS-2 code, a country of origin is assigned. The probabilities for different countries of origin for the same HS 2 code vary from month to month and thereby may affect the economic impact of a disruption.

Two types of errors were encountered while computing Eq. (26) due to mismatches between datasets. In the first case, the Per-Vessel Commodities (DS-2) dataset may have a value for $\text{TotalImportedTEU}_{\text{hs2}(k)}^{\text{country}(k)}$, but the Census data (DS-4), may lack an entry for that HS code, country pair. Conversely, the Census Trade Online Database (DS-4) may have a value for $\text{TotalImportsUSD}_{\text{hs2}(k)}^{\text{country}(k)}$, but imported TEU of a given commodity type are not found in the Per-Vessel Commodities dataset (DS-2).

A few possible causes of a mismatch between these data sources are briefly discussed. First, if the imports of a given commodity only come from one or two big US companies, the Census might not disclose such data to protect the confidentiality of company-specific information. Second, if the import amount is trivial for certain shipments, it might not be reflected in the data. Third, there can be data capture errors. As noted on the Census website on Foreign Trade Statistics: "The U.S. Census Bureau captures import and export information either from paper documents that are keyed manually or from automated collection programs. . . . Lost documents, errors in the on-line validations and edits of electronically reported data, and incorrectly keyed, coded, or recorded documents are examples of data capture errors that can impact the statistics" (Anon., 2020b). Fourth, it may be possible that different data sources are used for the two datasets. Census data comes from information collected by CBP which is largely based on the self-reporting by importers, exporters, and their agents.

Since our study is based on movements of commodities through the port as captured by DS-2, the study only handles the first type of error. When Census data was missing, the average price of that HS Code was calculated. This average was given by the total customs value of that HS code imported through the port (via DS-4) by the total number of TEUs with that HS code (via DS-2). Generally, this approach worked, but where the price data was missing from the Census data at the aggregated port level, or the calculated average price was too low to be possible (e.g. under $1000 per TEU), additional adjustments were made to deal with these cases. For example, when the price data for a certain commodity was lacking in one month, the average price that was available in the most adjacent month was used. In other cases, the average price was used for similar commodities in the same month.

6.3. Calibration, verification, and validation

Calibration of baseline scenarios was accomplished by comparing optimizer inputs/outputs to known data sources. Both the volume and duration of imported, loaded TEUs were used as calibration points. Looking at a period of 7 days in October, 28 vessel

calls delivered 4618 TEU for import. This is reasonable considering all loaded TEU imported in October 2017 were 25,000; therefore our inputs capture roughly 70% of total weekly TEU volumes assuming an even distribution of volumes per week (DS-9). Within the 17 baseline scenarios, TEU took 3 days to move through the port. This is consistent with stakeholders at Port Everglades, who pointed us to Port/Operator contracts (DS-8) as an indicator of container dwell times in container yards. As mentioned in the results, no delay costs were incurred across all baseline scenarios.

There is a good deal of literature on calibration of microscopic traffic models and simulation in general. The intent is to understand overall system behavior and estimate economic loss from disruption. The *measures of performance (MOP)* by which land-side import flows were calibrated include number of vessels, number of TEU, and duration of TEU in the system. The approach is briefly summarized below with results presented in Section 7. More details about calibration with respect to these measures may be found in Weaver et al. (2019).

A calibration procedure of the transportation network from the microsimulation literature was employed (Chu et al., 2003). Specifically, this procedure consists of estimating an Origin–Destination Demand Matrix for vessels and imported TEU, and defining a Route Choice Model. Calibration of the model with respect to number of TEU of a given HS code moving from a berth (o_k) to immediately outside of port (d_k) provides an approach to calibrate the number of type of TEU that move through a port under baseline conditions. We note that runs of our DDD algorithm presented in Section 7 do not require shipments to pass through a specific Terminal Operator and this may affect utilization of a given TO. In the future, experiments that use such a side constraint may be interesting to explore as a way of evaluating the impact of a disruption to a specific Terminal Operator.

Given the timing of container arrivals defined by the vessel arrivals (via DS-(1), and a good approximation of dwell time (via port SMEs and DS-8), this calibration may be good enough to understand the number of TEU in components of the transportation network close to TEU origin nodes. Within the microscopic simulation literature, some consider estimating the number of trips between an Origin–Destination pair during a given time period as the first (and potentially only necessary) step in model calibration (Chu et al., 2003). Estimated O-D pairs are provided by datasets DS-1 and DS-3. The Route Choice through the port is an output of our DDD algorithm in Section 4.

In order to gain credibility among stakeholders, a study should conduct verification and validation to determine the degree to which outputs align with the behavior of the system being modeled (Youngblood et al., 2000). A full *Verification, Validation, and Accreditation (VV&A)* process is beyond the scope of this article, however, steps have been taken to ensure results yield reasonable estimates of system behavior. For example, our extended DDD algorithm was built using the Gurobi optimization framework (Gurobi, 2020). The data used to generate NOLH experiment design points were obtained directly from the port. Moreover, the CTSNDP code, and supporting data processing pipeline, uses a unit testing framework to test for errors in implementation and is implemented using a build system to encourage repeatability.

Validation is necessary to determine the degree to which the outputs provide an accurate representation of the system. In order to validate results, estimated economic losses generated by the CTSNDP algorithm were compared to the economic losses generated by a discrete-event simulation using a shortest path algorithm to compute commodity routes. More details about calibration, verification, and validation of the simulation model are discussed in Weaver et al. (2019).

7. Results and discussion

In this section, we present the results of our holistic risk assessment approach, based on a *Nearly Orthogonal Latin Hypercube (NOLH)* design, to identify high-impact disruptions enabled by cyber–physical dependencies in the *Maritime Transportation System (MTS)*. The intent is to enable stakeholders to estimate the economic costs of how risk within IT systems affects efficient port operations.

Columns of our NOLH experimental design matrix define factors that specify the function of assets in a shipping port transportation network (e.g. gate service times, crane rates). The range of values for these factors specifies the degree to which cyber dependencies—such as those shown in Table 1—may affect transportation network operations. Ranges of possible values for factors may be defined by *Subject Matter Experts (SMEs)*, historical incidents, the academic literature, or even empirical data. These possible ranges of values may be used to define a general threat profile or a profile for a specific threat catalog item. Using the NOLH approach defined by Sanchez et al. these factors are sampled and used to generate an experimental design matrix whose rows correspond to scenario parameter settings for the optimizer.

There are several benefits to this approach. First NOLH allows one to explore the impact of varying several factors simultaneously, with the ability to identify factor interactions (Cioppa and Lucas, 2007). Second, NOLH provides a comprehensive approach to define the space of possible disruption scenarios and to understand the degree to which these scenarios have been explored (Kleijnen et al., 2005). Each row in the design matrix is used to set the parameter values of the transportation network and TEU shipments. The mapping from profile to network/shipment is based on asset type and so they enable the same disruption profile to be applied to different networks or shipment schedules. Baseline factor levels were calibrated based on a procedure from the microsimulation literature (Chu et al., 2003). More details about this calibration, as well as verification and validation, can be found in Section 6.3.

The results in this section apply two threat profiles (with multiple disruption scenarios in each threat profile) for cyber-originating disruptions to commodity flows through container operations in Port Everglades, FL. The first threat profile considers the impact of a cyberattack on the systems controlled by Broward County. As a landlord for the port, responsibilities include proper operation of gantry cranes and gates along the shipping port perimeter. As such, disruptions affecting crane rates, perimeter gate service times, and capacities for roads leading to those gates were generated using the NOLH approach. The second threat scenario considers the impact of a cyberattack on systems controlled by all four *Terminal Operators (TOs)* at the port. Each terminal operator is responsible

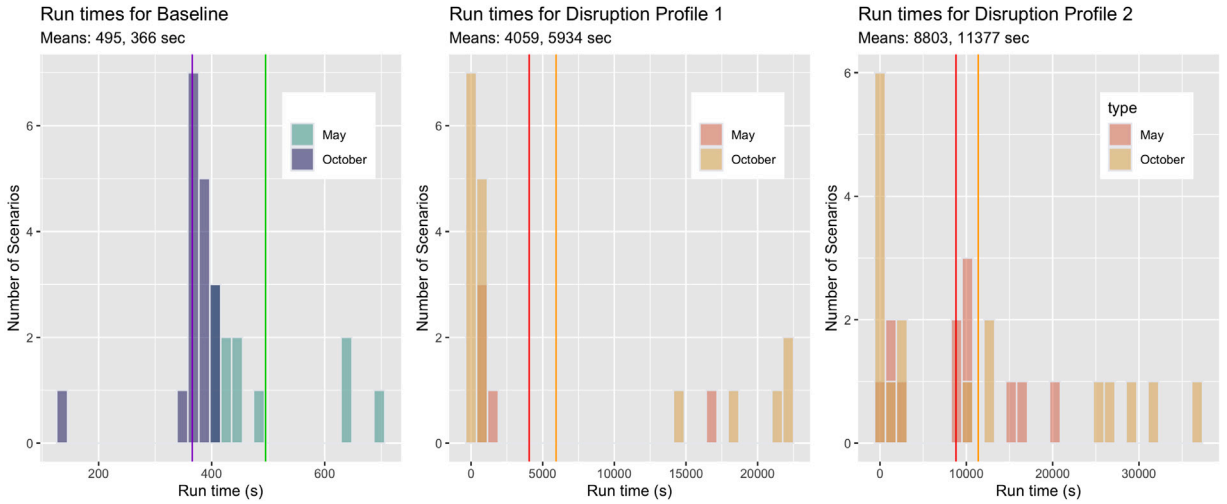


Fig. 7. Runtimes for baseline and disruption scenarios. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

for moving containers between the land and sea, with containers staying in the container yard for some period of time. As such, disruptions increase the time spent in the container yard, TO gate service times, and capacities for roads leading to those gates. We run our enhanced DDD algorithm to estimate costs occurring due to these disruptions. Observe that the estimates obtained will be lower bounds on the true costs, as these recovery actions are optimized whereas in practice, decision support tools may not be available and costs may be higher.

7.1. Algorithm performance

Fig. 7 illustrates the runtimes across the baseline and disrupted scenarios. Experiments were run on a VM with 4 CPUs and 64 GB of memory. Baseline scenario run times for the first week of October, with 4618 TEU, ranged from approximately 2 min to 7 min with an average of 6 min. In contrast, baseline run times for the first week of May, with 7214 TEU ranged from 6 min to 11 min with an average of 8 min. Run times for disruption profile 1, cyber-attacks on landlord port resources, took between 8 min to 6 h with a mean of 1.5 h to complete for October scenarios and between 10 min to 4.6 h with a mean around 1 h to complete for May data. In contrast, run times for disruption profile 2, cyber-attacks on terminal operator resources, took from 7 min to 10 h with an average runtime of 3 h for October scenarios. The May scenarios, however took from 9 min to 5.5 h with an average runtime of 2.4 h. The baseline runtimes for scenarios are faster and the runtimes for disrupted profiles are (on average) comparable to the 2 h runtimes (on 32 GB machines) presented by Boland et al. (2017) when applied to 5 days of freight movements in the Pacific Northwest.

Our algorithm's run times are based on a larger transportation network, more commodities, and a longer time window than in previous literature (Boland et al., 2017). The parameters in our experimental setup are an order of magnitude higher than in existing work — each of our scenarios (experiment design points) are based on a flat network with 138 nodes, 163 links, and 4618/7214 commodities in the October/May scenarios respectively. As described in 6.1, the number of commodities is derived from actual vessel manifests for loaded TEU imported into Port Everglades. Moreover, we increased the duration of the recovery window in our scenarios as the durations in existing work (5 days) were too short to capture economic losses resulting from inventory depletion in our commodity classification. Therefore, a minimum of one week was considered in our study. With respect to the number of nodes, our scenarios are comparable in size to several public transportation networks surveyed by Lin et al. whose node counts ranged from 124 to 46244 (Lin and Ban, 2013).

The number of nodes and arcs in the expanded network varied across the disruption profiles and number of commodities. As shown in the first row of Fig. 8, the number of delay nodes and storage arcs was close to zero for baseline scenarios in October and May, with May having more nodes and arcs overall in the baseline due to the larger number of commodities. In contrast, in a disrupted scenario, the number of nodes (including delay nodes) increases as do the number of storage arcs. Overall, the number of nodes and arcs from baseline to disruption profile increases by a few hundred between baseline and disrupted cases. Within the baseline and disrupted cases, there is no clear separation between counts in October and May scenarios, indicating that the size of the expanded networks depends more on the disruption profile than on a difference of a few thousand commodities.

7.2. Summary of economic impacts

Fig. 9 illustrates the coverage of scenarios run within the experimental design spaces for both threat profiles using the October scenarios. In all, we present results from 28 (17/11) baseline or no-disruption scenarios, 15 (10/5) landlord port disruption scenarios,

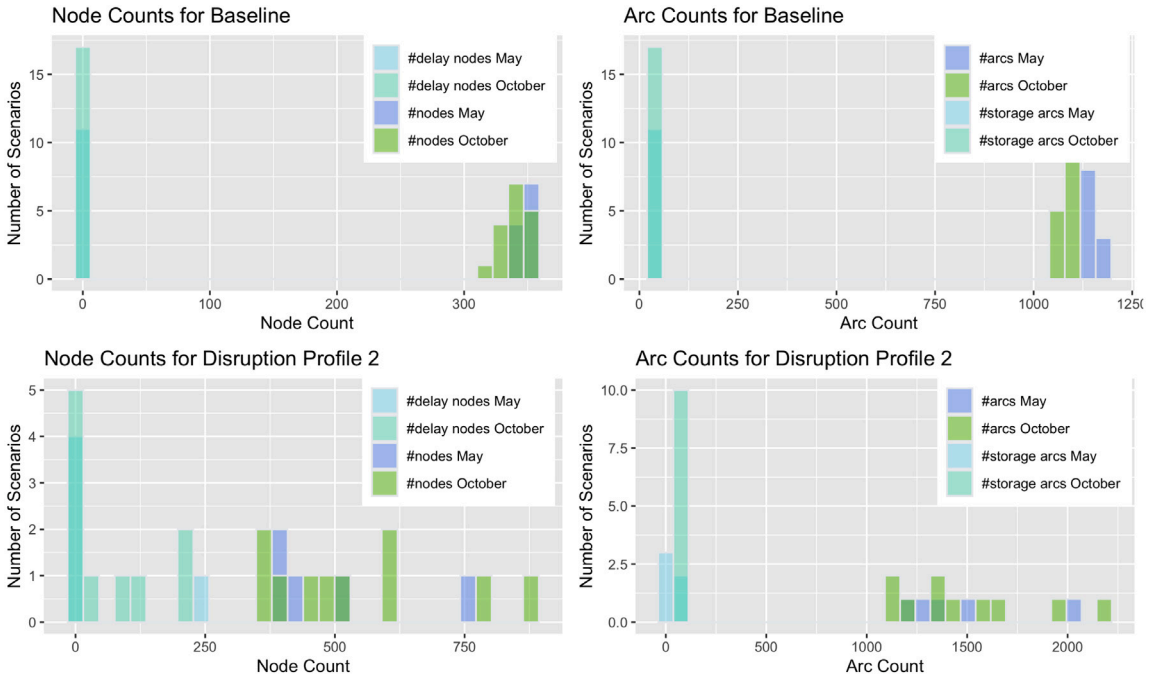


Fig. 8. Node and arc counts for baseline and disruption profile 2 scenarios. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and 27 (15/12) terminal operators' disruption scenarios in October/May respectively, using our enhanced DDD algorithm. Design points for which an optimal solution could be computed are colored from lowest economic loss to highest economic loss. The first disruption profile (threats to the Landlord Port) results in delay costs ranging from $0 to approximately $650,000 shown in colors ranging from blue to red respectively. The second disruption profile (threats to Terminal Operators) results in delay costs ranging from $0 to $4M shown in colors ranging from purple to orange. Although October scenarios are shown, scatterplots of the May shipments yielded the same relative distribution of economic loss across the same design points though with different magnitudes of impact. For example, the maximum delay cost for the Terminal Operator disruption scenarios was $4M in October but $7M in May.

Fig. 10 below illustrates estimated daily costs of commodity schedules output by our extended DDD algorithm for the baseline and two disruption scenarios. The delay costs in the baseline cases were $0 USD, indicating negligible costs if the port is functioning normally with no disruptions. The economic impact of disrupted scenarios, on average, were $82,224/$141,647 for cyber-attacks on the landlord port's cyber-controlled assets and $1.2M/$2.8M for cyber-attacks affecting all port terminal operators in October/May respectively.

7.3. Landlord port threat profile

In the first threat profile, a cyber-originating attack affects transportation assets controlled by the landlord port; and consists of 34 scenarios. 15 of 34 scenarios run in this profile's design space ran to completion using our algorithm. Of the scenarios that ran to completion, ten belonged to October and five to May. Among these, five scenarios from October and four from May did not result in any economic delay costs because of the re-routing and re-scheduling decisions provided by our enhanced DDD algorithm. This demonstrates that updated re-routing and re-scheduling decisions that respond to the disruption, computed using an efficient and highly tractable algorithm, can significantly recover system delay costs. Of the five remaining October scenarios that incurred a cost despite re-routing and re-scheduling, direct economic losses ranged from $3400 to $650,000 as shown in Fig. 10. In contrast, the May scenario that still incurred a cost despite re-routing and re-scheduling incurs a cost of $708,234 dollars. The same scenario, when run on October data has a cost of $243,154 dollars, indicating the importance of seasonality on the impact of disrupts at ports.

One scenario that resulted in losses of approximately $400,000 over the week-long period was of particular interest as the gate service time was 9 min, and the crane rate was 21 TEU per hour. These parameter settings are both close to baseline configurations, 1 to 3 min for gate service time and 23 TEU per hour respectively. This indicates that a slowdown in crane rate (e.g. due to a cyber disruption that affects container recognition) combined with a longer gate service time (e.g. due to MARSEC level increase) can result in economic losses over the course of a week. Fig. 11 (Scenario 11) illustrates the operational impact of this particular scenario on container lateness leaving the port. While in the baseline scenarios, TEU spent 3 days in the port system and none arrived late, under this disruption, TEU take nearly 5 days to move through the system.

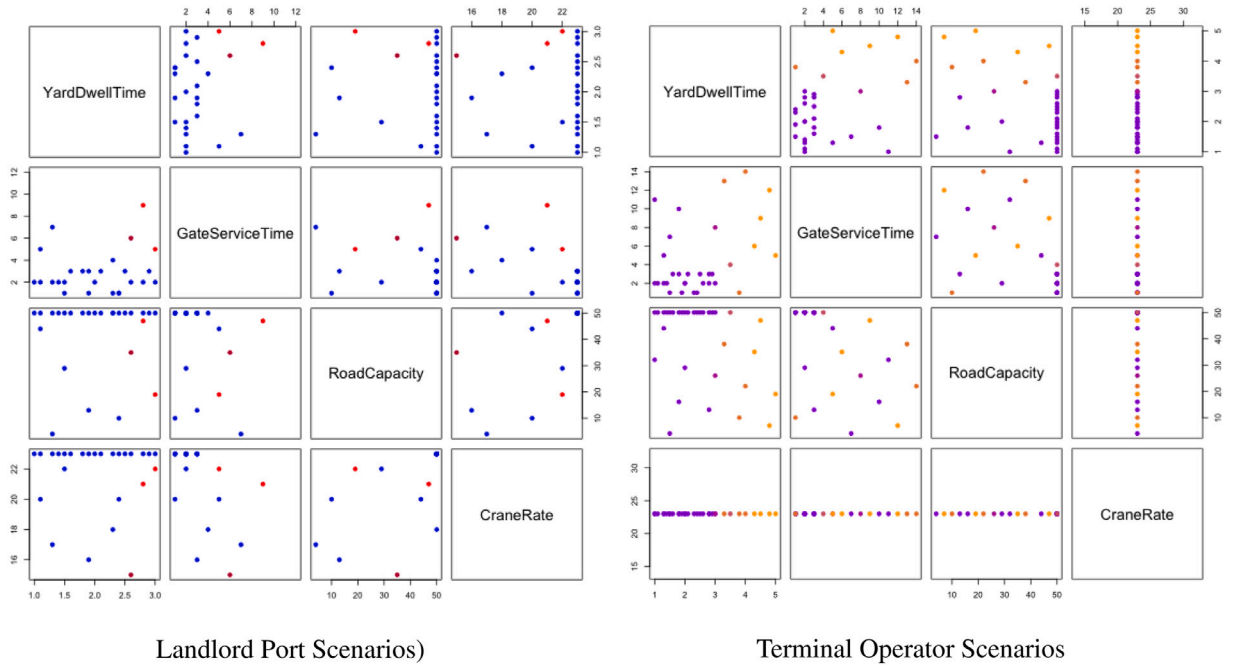


Fig. 9. The coverage of scenarios run within the experimental design space for both threat profiles in October. Economic losses from cyber-originating disruptions to the Landlord Port were up to $650,000 (red) in whereas those for the Terminal Operators were up to $4M (orange). The relative impact of the Terminal Operator disruption scenarios for May shipments looked much the same as those for October, but were up to $7M in delay costs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

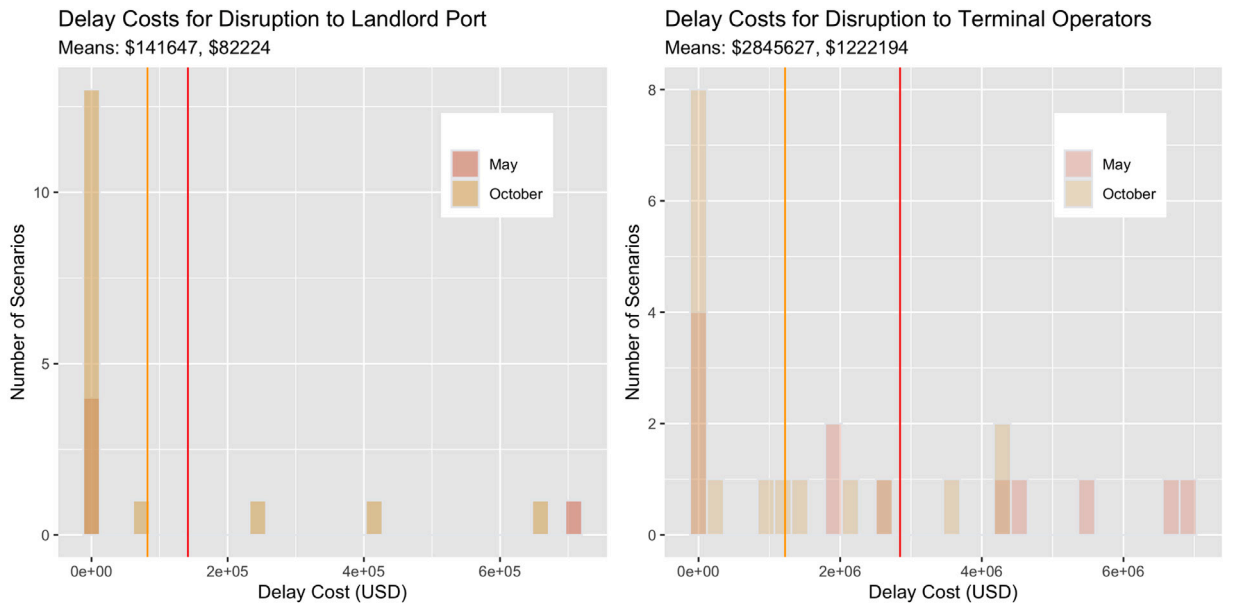


Fig. 10. The estimated economic impact of cyber attacks on a landlord port versus that of terminal operators. Notice that the in the latter case, disruptions are an order of magnitude more severe economically. This is for a week-long disruption to container imports in the first week of October. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

7.4. Terminal operator threat profile

The second threat profile looks at the impact of a cyber-originating disruption simultaneously affecting the four terminal operators' transportation assets, and consists of 34 scenarios with 17 each from October and May. Across both months, 27 of 34

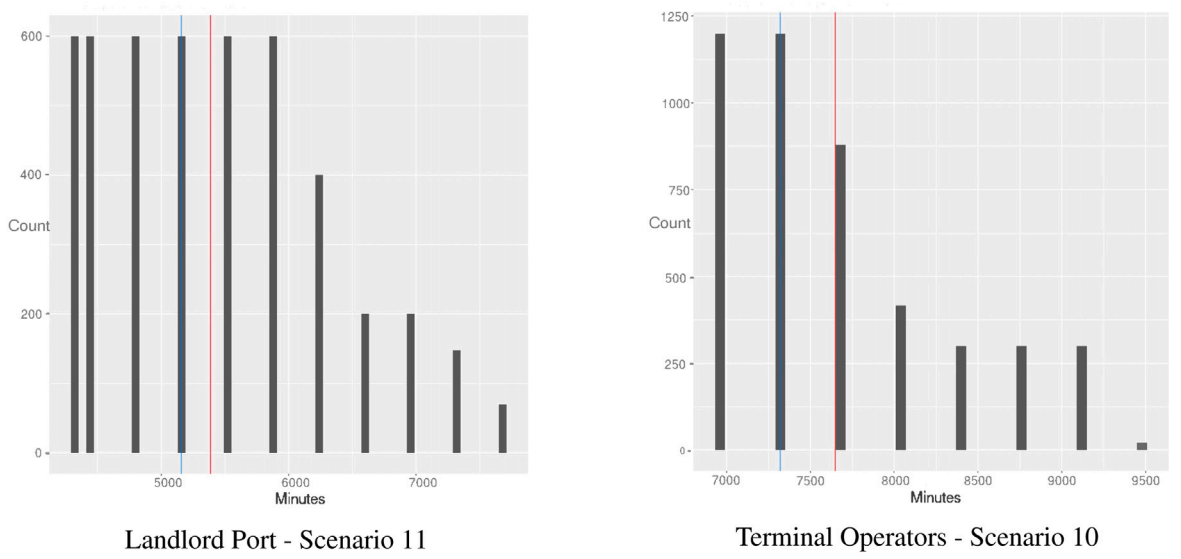


Fig. 11. The operational impact of cyber-attacks on landlord port assets versus those of terminal operators. In the former, imported TEU take up to 5 days to move through the port whereas in the latter, they take a little under a week. Mean and median durations are shown by red and blue lines respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

scenarios ran to completion: 15 of 17 for October, and 12 of 17 for May. A third of these scenarios resulted in no delay costs, as re-routing and re-scheduling solutions from our algorithm mitigated costs. Of the remaining ten October scenarios, economic losses ranged from $5000 to $4.3M as shown in Fig. 10, with an average of $1.22M. The remaining eight May scenarios had economic losses ranging from $50K to nearly $7M with an average of $2.8M.

The scenarios that incurred the most costs were those that affected container dwell times the most. Rather than 1–3 days, scenarios in this category that resulted in 4–5 day container yard dwell times saw the most economic losses. One scenario combined a slightly longer dwell time (3.3 days) with traffic congestion and a longer gate service time (13 min) resulted in a roughly $1M loss in October. Fig. 11 (Scenario 10) illustrates the operational impact of the scenario with the highest economic losses in October, approximately $4.4M. In this scenario, TEU take from half a week to nearly a week to move through the port system, resulting in the highest increases in costs. These cost estimates and magnitudes have been validated with the terminal operators and compared with similar historical scenarios.

8. Conclusions

This paper presents an optimization-based approach to estimate the functional and economic impacts of cyber-originating disruptions to a shipping port's transportation network. Disruption profiles, informed by a real-world cyber–physical threat catalog, are specified as experimental design matrices. We present an extended Dynamic Discretization Discovery algorithm building on the approach of Boland et al. (2017), by building novel methodology to capture the delays and disruptions and per-unit-time vehicle-flow capacities. The algorithm generates schedules and routes that optimize disrupted shipping container flows relative to seasonal commodity values — thus finding the lower bound on costs due to the disruptions. The economic cost, based on extending the willingness to pay literature, provides a way to understand the range of impacts for multiple scenarios across a threat profile. Informed by fieldwork at Port Everglades, FL, our work advances the state of the art. In terms of performance, our enhanced DDD algorithm runs disruption scenarios an order of magnitude faster than the current state of the art in the baseline case and in comparable time for disrupted scenarios. As a result, we hope to enable MTS stakeholders to continually evaluate the operational and economic risks introduced when adopting automation technologies in order to drive efficiencies in this highly competitive industry.

We also anticipate that in addition to optimizing for recovery actions, our enhanced algorithm can also be used for evaluating infrastructure investment decisions that mitigate future costs, under 'what-if' scenarios. In future work, we aim to utilize this algorithm to also generate infrastructure planning and routing and scheduling decisions that are robust to future disruptions.

CRediT authorship contribution statement

Gabriel A. Weaver: Data curation, Formal analysis, Funding acquisition, Investigation, Project administration, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Brett Feddersen:** Formal analysis, Investigation, Methodology, Software, Writing – original draft. **Lavanya Marla:** Conceptualization, Formal analysis, Funding

acquisition, Investigation, Methodology, Resources, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Dan Wei:** Formal analysis, Methodology, Validation, Writing – original draft. **Adam Rose:** Supervision, Methodology, Validation, Writing – original draft. **Mark Van Moer:** Formal analysis, Software, Visualization.

Acknowledgments

The material presented in this paper is based upon work supported in part by the U.S. Department of Homeland Security under Grant Award Number, 2015-ST-061-CIRC01 as well as the Herman M. Dieckamp Post-Doctoral Fellowship. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security. The authors would also like to thank Dr. Terrie Walmsley for her discussions and feedback on the economic analysis presented in this paper.

References

Acosta, L.G.M., 2020. China's one road, one belt grand strategy: Founded on the weaponization of the global supply chain. Def. Transp. J..

Anon., 2013. UT Austin researchers successfully spoof an $80 million yacht at sea. UTNews. https://news.utexas.edu/2013/07/29/ut-austin-researchers-successfully-spoof-an-80-million-yacht-at-sea. (Accessed 2 October 2018).

Anon., 2014. Cargo thieves use GPS jammers to mask GPS trackers, tLP: Green. https://info.publicintelligence.net/FBI-CargoThievesGPS.pdf. (Accessed 2 October 2018).

Anon., 2016. State of Wisconsin Hazard Mitigation Plan. Tech. Rep., amended 2017. Retrieved from REDACTED. (Accessed 2 October 2018).

Anon., 2017. Emerging Systems at Automated Container Terminals. Tech. Rep., Department of Homeland Security NPPD OCIA.

Anon., 2018. Port of San Diego hit by cyberattack. https://maritime-executive.com/article/port-of-san-diego-hit-by-cyberattack. (Accessed 2 October 2018).

Anon., 2019. Shen Attack, Cyber Risk in Asia Pacific Ports. Tech. Rep., https://risk-studies-viewpoint.blog.jbs.cam.ac.uk/2019/10/30/shen-attack-cyber-risk-scenario-up-to-110-billion-at-risk-from-maritime-malware-attack/. (Accessed 26 February 2021).

Anon., 2020a. Alert (AA20-049A): Ransomware impacting pipeline operations. https://us-cert.cisa.gov/ncas/alerts/aa20-049a. (Accessed 15 December 2020).

Anon., 2020b. Guide to foreign trade statistics. https://www.census.gov/foreign-trade/guide/sec2.html. (Accessed 26 February 2020).

Anon., 2020c. Washington's Port of Kennewick hit by cyberattack. https://www.professionalmariner.com/washingtons-port-of-kennewick-hit-by-cyberattack/. (Accessed 16 February 2020).

Barthélemy, M., 2011. Spatial networks. Phys. Rep. http://dx.doi.org/10.1016/j.physrep.2010.11.002.

Bateman, T., 2013. Police warning after drug traffickers' cyber-attack. BBC News.

Beyeler, W.E., Conrad, S.H., Corbet, T.F., O'Reilly, G.P., Picklesimer, D.D., 2004. Inter-infrastructure modeling — ports and telecommunications. Bell Labs Tech. J. 9 (2), 91–105.

Boland, N., Hewitt, M., Marshall, L., Savelsbergh, M., 2017. The continuous-time service network design problem. Oper. Res. 65 (5), 1303–1321.

Bou-Harb, E., Kaisar, E.I., Austin, M., 2017. On the impact of empirical attack models targeting marine transportation. In: Models and Technologies for Intelligent Transportation Systems, MT-ITS, 2017 5th IEEE International Conference on. Institute of Electrical and Electronics Engineers, pp. 200–205.

Burgess, M., 2019. To protect Putin, Russia is spoofing GPS signals on a massive scale. WIRED Mag..

Cheh, C., Weaver, G.A., Sanders, W.H., 2015. Cyber-physical topology language: Definition, operations, and application. In: Dependable Computing, PRDC, 2015 IEEE 21st Pacific Rim International Symposium on. IEEE, pp. 60–69.

Chu, L., Liu, H.X., Oh, J.-S., Recker, W., 2003. A calibration procedure for microscopic traffic simulation. In: Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems, vol. 2, Institute of Electrical and Electronics Engineers, Inc., pp. 1574–1579.

Cimino, M.G., Palumbo, F., Vaglini, G., Ferro, E., Celandroni, N., La Rosa, D., 2017. Evaluating the impact of smart technologies on harbor's logistics via BPMN modeling and simulation. Inf. Technol. Manag. 18 (3), 223–239.

Cimpanu, C., 2019a. US coast guard disclosed Ryuk ransomware infection at maritime facility. ZDnet. https://www.zdnet.com/article/us-coast-guard-discloses-ryuk-ransomware-infection-at-maritime-facility/. (Accessed 26 February 2021).

Cimpanu, C., 2019b. US coast guard warns about malware designed to disrupt ships' computer systems. ZDNet. https://www.zdnet.com/article/us-coast-guard-warns-about-malware-designed-to-disrupt-ships-computer-systems/. (Accessed 26 February 2021).

Cioppa, T.M., Lucas, T.W., 2007. Efficient nearly orthogonal and space-filling Latin hypercubes. Technometrics (ISSN: 00401706) http://dx.doi.org/10.1198/004017006000000453.

Coren, M.J., 2017. Venture capitalists have found another trillion dollar market to upend: Shipping. Quartz.

Crainic, T.G., Hewitt, M., Toulouse, M., Vu, D.M., 2016. Service network design with resource constraints. Transp. Sci. 50 (4), 1380–1393.

Danielis, R., Gregori, T., 2013. An input–output-based methodology to estimate the economic role of a port: The case of the port system of the Friuli Venezia Giulia region, Italy. Marit. Econ. Logist. 15 (2), 222–255.

DiRenzo, J., Drumhiller, N.K., Roberts, F.S., 2017. Issues in Maritime Cyber Security. Westphalia Press.

Enayaty-Ahangar, F., Albert, L.A., DuBois, E., 2020. A survey of optimization models and methods for cyberinfrastructure security. IISE Transactions 53 (2), 182–198

Engstrom, J., 2018. Systems Confrontation and System Destruction Warfare. RR1708, Rand Corporation.

Erera, A., Hewitt, M., Savelsbergh, M., Zhang, Y., 2013. Improved load plan design through integer programming based local search. Transp. Sci. 47 (3), 412–427.

Green, M., 2017. Romanian hackers infiltrated 65% of DC's outdoor surveillance cameras. CNN. https://www.cnn.com/2017/12/20/politics/romanian-hackers-dc-cameras/index.html. (Accessed 2 October 2018).

Greenberg, A., 2018. The untold story of NotPetya, the most devastating cyberattack in history. WIRED Mag..

Gurobi, L., 2020. Optimization. Gurobi Optimizer Reference Manual, URL http://www.gurobi.com.

Hall, A., Hippler, S., Skutella, M., 2007. Multicommodity flows over time: Efficient algorithms and complexity. Theoret. Comput. Sci. 379 (3), 387–404.

Hoffman, J., Sirimanne, S.N., 2017. Review of maritime transport. https://unctad.org/en/PublicationsLibrary/rmt2017_en.pdf. (Accessed 26 February 2021).

Hummels, D., Minor, P., Reisman, M., Endean, E., 2007. Calculating Tariff Equivalents for Time in Trade. USAID Report.

Jarrah, A.I., Johnson, E., Neubert, L.C., Large-scale, 2009. Less-than-truckload service network design. Oper. Res. 57 (3), 609–625.

Kleijnen, J.P., Sanchez, S.M., Lucas, T.W., Cioppa, T.M., 2005. A user's guide to the brave new world of designing simulation experiments. INFORMS J. Comput. http://dx.doi.org/10.1287/ijoc.1050.0136.

LaRocco, L.A., 2021. Suez canal blockage is delaying an estimated $400 million an hour in goods. https://www.cnbc.com/2021/03/25/suez-canal-blockage-is-delaying-an-estimated-400-million-an-hour-in-goods.html. (Accessed 5 April 2020).

Lin, J., Ban, Y., 2013. Complex network topology of transportation systems. Transp. Rev. 33 (6), 658–685.

MacKenzie, C.A., Barker, K., Grant, F.H., 2011. Evaluating the consequences of an inland waterway port closure with a dynamic multiregional interdependence model. IEEE Trans. Syst. Man Cybern. A 42 (2), 359–370.

Mathews, L., 2017. NotPetya ransomware attack cost shipping giant Maersk over $200 million. Forbes. https://www.forbes.com/sites/leemathews/2017/08/16/notpetya-ransomware-attack-cost-shipping-giant-maersk-over-200-million/#2c5274404f9a. (Accessed 14 November 2017).

McGlone, T., 2014. Cybersecurity options lag behind hackers' abilities. http://www.govtech.com/security/Cybersecurity-Options-Lag-Behind-Hackers-Abilities.html. (Accessed 2 October 2018).

Minor, P., 2013. Time as a trade barrier: A GTAP database of ad valorem trade time costs.

Mohindru, S., 2017. Shipping: BW Group's computer systems hacked; steps up cyber security. https://www.spglobal.com/platts/en/market-insights/latest-news/shipping/101317-shipping-bw-groups-computer-systems-hacked-steps-up-cyber-security. (Accessed 2 October 2018).

Muccin, E., 2016. Cyber security at sea. https://www.maritime-executive.com/blog/cyber-security-at-sea. (Accessed 2 October 2018).

Newman, L.H., 2017. A bug in a popular maritime platform left ships exposed. WIRED Mag. https://www.wired.com/story/bug-in-popular-maritime-platform-isnt-getting-fixed/. (Accessed 2 October 2018).

Osborne, C., 2018. Nonpetya ransomware forced Maersk to reinstall 4000 servers, 45000 PCs.

Pant, R., Barker, K., Grant, F.H., Landers, T.L., 2011. Interdependent impacts of inoperability at multi-modal transportation container terminals. Transp. Res. E 47 (5), 722–737.

Park, J., 2008. The economic impacts of dirty bomb attacks on the Los Angeles and Long Beach ports: Applying the supply-driven NIEMO (National Interstate Economic Model). J. Homel. Secur. Emerg. Manag. 5 (1).

Powell, W.B., Jaillet, P., Odoni, A., 1995. Stochastic and dynamic networks and routing. In: Handbooks in operations research and management science, vol. 8, pp. 141–295.

Rose, A.Z., 2009. A framework for analyzing the total economic impacts of terrorist attacks and natural disasters. J. Homel. Secur. Emerg. Manag. 6 (1).

Rose, A., DiRenzo, III, J., Drumhiller, N.K., Roberts, F.S., 2017. Economic consequence analysis of maritime cyber threats. Issues Marit. Cyber Secur. 321–356.

Rose, A., Wei, D., 2013. Estimating the economic consequences of a port shutdown: The special role of resilience. Econ. Syst. Res. 25 (2), 212–232.

Rose, A., Wei, D., Paul, D., 2018. Economic consequences of and resilience to a disruption of petroleum trade: The role of seaports in US energy security. Energy Policy 115, 584–615.

Sanchez, S.M., Sánchez, P.J., Wan, H., 2018. Work smarter, not harder: A tutorial on designing and conducting simulation experiments. In: 2018 Winter Simulation Conference. WSC, IEEE, pp. 237–251.

Sienkiewicz, J., Hołyst, J.A., 2005. Statistical analysis of 22 public transport networks in Poland. Phys. Rev. E (3) (ISSN: 15393755) http://dx.doi.org/10.1103/PhysRevE.72.046127.

e Silva, F.B., Forzieri, G., Herrera, M.A.M., Bianchi, A., Lavalle, C., Feyen, L., 2019. HARCI-EU, a harmonized gridded dataset of critical infrastructures in Europe for large-scale risk assessments. Sci. Data 6 (1), 1–11.

Simpson, B., 2018. Cyberattacks called a growing threat to trucking industry. https://www.ttnews.com/articles/cyberattacks-called-growing-threat-trucking-industry. (Accessed 2 October 2018).

Tsonchev, A., 2018. Troubled waters: Cyber-attacks on san diego and barcelona's ports. Darktrace Blog. https://www.darktrace.com/blog/troubled-waters-cyber-attacks-on-san-diego-and-barcelonas-ports/. (Accessed 2 October 2018).

Vavra, S., 2020. US army combines fake hacks, natural disaster simulation to test municipal responses. CyberScoop, https://www.cyberscoop.com/army-savannah-charleston-cyber-test/. (Accessed 26 February 2021).

Von Ferber, C., Holovatch, T., Holovatch, Y., Palchykov, V., 2009. Public transport networks: Empirical analysis and modeling. Eur. Phys. J. B 68 (2), 261–275.

Vu, D.M., Hewitt, M., Boland, N., Savelsbergh, M., 2020. Dynamic discretization discovery for solving the time-dependent traveling salesman problem with time windows. Transp. Sci. 54 (3), 703–720.

Weaver, G., 2021a. A data processing pipeline for cyber-physical risk assessments of municipal supply chains. In: Proceedings of the 2021 Winter Simulation Conference. WSC '21, Institute of Electrical and Electronics Engineers, Inc., Phoenix, AZ.

Weaver, G., 2021b. Scientific data management for interconnected critical infrastructure systems. In: Proceedings of the 2021 Joint Conference on Digital Libraries. JCDL '21, Association for Computing and Machinery/Institute of Electrical and Electronics Engineers, Inc., Champaign, IL.

Weaver, G., Marla, L., 2019. Cyber-physical simulation and optimal mitigation for shipping port operations. In: WSC 2018-2018 Winter Simulation Conference, Proceedings - Winter Simulation Conference. Institute of Electrical and Electronics Engineers Inc., United States, pp. 2747–2758. http://dx.doi.org/10.1109/WSC.2018.8632551, funding Information: The material presented in this paper is based upon work supported in part by the U.S. Department of Homeland Security under Grant Award Number, 2015-ST-061-CIRC01 as well as the Herman M. Dieckamp Post-Doctoral Fellowship. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security. Publisher Copyright: ©2018 IEEE; 2018 Winter Simulation Conference, WSC 2018; Conference date: 09-12-2018 Through 12-12-2018.

Weaver, G., Van Moer, M., Salo, G., 2019. Stakeholder-centric analyses of simulated shipping port disruptions. In: Proceedings of the 2019 Winter Simulation Conference. Institute of Electrical and Electronics Engineers, Inc., National Harbor, MD.

Wei, D., Chen, Z., Rose, A., 2020. Evaluating the role of resilience in reducing economic losses from disasters: A multi-regional analysis of a seaport disruption. Pap. Reg. Sci. 99 (6), 1691–1722.

Youngblood, S.M., Pace, D.K., Eirich, P.L., Gregg, D.M., Coolahan, J.E., 2000. Simulation verification, validation, and accreditation. Johns Hopkins APL Tech. Dig. 21 (3), 359–367.