Dynamic shipments of inventories in shared warehouse and transportation networks

van der Heide, G.; Buijs, P.; Roodbergen, K. J.; Vis, I. F. A.

Published in:
Transportation Research Part E: Logistics and Transportation Review

DOI:
10.1016/j.tre.2018.07.012

Dynamic Shipments of Inventories in Shared Warehouse and Transportation Networks

G. Van der Heide*, P. Buijs, K.J. Roodbergen, I.F.A. Vis

University of Groningen, P.O. Box 800, 9700 AV Groningen, The Netherlands

Abstract

In shared warehouse and transportation networks, dynamic shipments of inventories are carried out based on up-to-date inventory information. This paper studies the effect of network structures on optimal decision-making. We propose a discrete time modeling framework with stochastic demand, capturing a wide variety of network structures. Using Markov decision processes, we obtain optimal order and dynamic shipment decisions for small networks. We compare optimal solutions of different four-node network structures. Results indicate product characteristics significantly influence the effectiveness of network structures. Surprisingly, two-echelon networks are occasionally costlier than any other network. Moreover, dynamic shipments yield considerable gains over static shipments.

Keywords: Shared warehouse and transportation network, Dynamic shipments, Network comparisons, Inventory, Markov decision process

1. Introduction and Literature Review

In recent years, on-demand warehousing marketplaces, such as FLEXE and Stockspots, have been unlocking unused space in hundreds of warehouses throughout the United States and Europe. Companies can use these marketplaces to quickly rent available pallet spaces anywhere in a network of warehouses when needed. At the same time, companies with excess warehouse space can earn some additional revenue. By using networks of shared warehouses, companies avoid the fixed costs that typically come with operating or leasing a warehouse, enabling them to store inventories closer to their end-consumers. On top of that, modern information and communication technologies, such as the Internet of Things, enable real-time tracking of (in-transit) inventories. A product's end-location can be adjusted, possibly even during transportation, enabling dynamic shipments of products between warehouses based on the latest available information. Together, these trends have the potential to reshape the way future transportation and inventory control will be organized.

Inspired by these recent advances, we study the potential benefits of utilizing dynamic shipments in networks of shared warehouses. We will refer to such networks as Shared Warehouse and Transportation Networks (SWTN). In SWTNs, a company may push its products strategically into the network and relocate those products to where they are most needed on a just-in-time basis, providing significant flexibility in dealing with inventory imbalances. Trucks can periodically transport products between connected warehouses in the network. Upon arrival at a warehouse, products can be transported to another connected warehouse in a next period. Usually, not all pairs of warehouses in SWTNs are directly connected. A product may thus need to travel through multiple warehouses to reach its end-location.

As SWTNs may differ vastly in structure, this paper focuses on generating insights into the effects of different network structures on optimal decision-making for ordering and shipping inventory. In order to achieve this, we first propose a discrete time modeling framework capable of capturing a large variety of different network structures. We then apply that framework to compute optimal order and shipment decisions for several small-sized SWTNs using

*Corresponding author.

Email addresses: g.van.der.heide@rug.nl (G. Van der Heide), p.buijs@rug.nl (P. Buijs), k.j.roodbergen@rug.nl (K.J. Roodbergen), i.f.a.vis@rug.nl (I.F.A. Vis)

Markov decision processes (MDPs). The optimal solutions for SWTNs with different structures are compared through several numerical experiments, taking into account the effect of product characteristics such as cost parameters, stock-out behavior of customers, and variance of demand. These numerical experiments provide various scientific and managerial insights for designing effective SWTNs and the added value of dynamic shipments.

In SWTNs, transportation and inventory decisions are fully integrated. Therefore, in the remainder of this introduction, we discuss related contributions in both transportation and inventory research. The emphasis in our review of the literature is on how existing research incorporates the underlying network structure. To the best of our knowledge, we are the first to consider combined transportation and inventory control in SWTNs and we are the first to analyze the effect of network structure on optimal decision-making in such networks.

In inventory research, it is common to first select a specific network structure and then determine (near-) optimal inventory policies for that specific structure depending on, among others, assumptions for stock-out behavior, cost parameters, and demand distributions. This approach is reasonable when the network structure is fixed due to geographical or organizational reasons, as is the case in most warehouse and transportation networks that are dedicated to one or a few companies. Two-echelon networks are most frequently studied (Zhao et al., 2010; Madadi et al., 2010). Moreover, networks are typically assumed to be arborescent, implying a unique path exists from the source node to each end-node. This considerably simplifies analysis, even though it has been argued that arborescent networks are not always realistic in practice (Cattani et al., 2011). An important contribution of our paper, therefore, is the introduction of a modeling framework capable of capturing a wide variety of network structures, including non-arborescent networks. Our framework is inspired by the work of Karmarkar (1981), with the difference that we include positive lead times, as travel times are not negligible in SWTNs, and that we compare different network structures.

Our paper aims to address the lack of in-depth comparisons between different network structures in the literature. Only a small number of authors has made such comparisons, i.e., Muckstadt and Thomas (1980) compare single- and two-echelon policies and Özdemir et al. (2006) compare networks with different transshipment locations. Through our comparisons of small-sized networks, we generate understanding of effective decision-making in a much wider variety of network structures than considered before.

Our modeling framework includes multi-echelon models, for which optimal solutions can only be derived analytically in special cases (see, e.g., De Kok et al., 2018). Moreover, effective heuristics typically rely on exploiting the arborescent structure (Cattani et al., 2011). With this in mind, it is beyond the scope of our paper to develop exact or heuristic solution methodologies for general network structures. However, insights from our comparisons may assist in developing heuristics for general network structures in future research.

Several problems in transportation research feature adaptive decision-making based on stochastic information. For example, in dynamic vehicle routing problems routes can be adapted in real-time or periodically when stochastic demands arrive (Pillac et al., 2013; Ritzinger et al., 2016). Strategically placing goods in dynamically changing networks has been the subject of several studies in the domain of humanitarian logistics (e.g., Özdamar et al., 2004; Rennemo et al., 2014; Ahmadi et al., 2015; Alem et al., 2016), where relief goods must be transported through a network whose edges and nodes can become unavailable due to disasters. In SWTNs, the main issue is also strategic placement and transportation of products in the network in anticipation of and in response to random events. Prior studies in that regard have addressed problems with limited horizons, whereas we are interested in analyzing inventory and transportation policies for the long-run.

The integration of transportation and inventory decisions is mainly studied in regards to inventory-routing (Coelho et al., 2013) and vendor-managed inventory problems (Govindan, 2013; Marquès et al., 2010). In vendor-managed inventory (VMI), a vendor has the responsibility to manage its inventory at one or more retailers (Dong and Xu, 2002; Disney et al., 2003). Accordingly, a two-echelon network structure is natural when studying VMI (Yao and Dresner, 2008; Chen et al., 2010; Gürler et al., 2014). Typically, inventory at retailers is controlled through parametric policies such as periodic review order-up-to policies (Yao and Dresner, 2008). After the parameters of these policies are set heuristically, inventory-routing problems can be solved to obtain effective routes (Coelho et al., 2013). Such inventory-routing problems are classical in the sense that typically vehicles start from and return at the same warehouse, visiting several customers on a route. In SWTNs, the main concern is not routing vehicles, but routing products over available connections. The network structure in SWTNs can, therefore, differ substantially from a two-echelon structure. Moreover, the possibility to store products at and transport products between warehouses based on newly arriving inventory information enables new dynamic VMI strategies. In our numerical experiments, we consider a company responsible for inventory control at multiple warehouses, which can be regarded as the first application of

VMI in SWTNs.

SWTNs are also a fundamental element of the Physical Internet initiative (Montreuil, 2011), which is gaining attention from researchers and practitioners (Treiblmaier et al., 2016; Pan et al., 2017; Sternberg and Norrman, 2017). The Physical Internet initiative relies on the analogy with the digital internet to inspire innovation in logistics. In the digital internet, information packages are sent automatically from source to destination through a network of hubs with the use of routing protocols. Similarly, in the Physical Internet (PI), a company simply specifies a destination for a physical product, and then trusts the system to arrange its transportation and storage. Products potentially travel through multiple, "open" warehouses and can be combined — and re-combined — with other products at every warehouse along the way. This may help to address possible mismatches between inventory fitting in a truckload and the inventory required at an end-location, either giving rise to excess inventories (when shipping full truckloads) or poor vehicle utilization (when carrying out small-sized shipments). Early studies indicate that PI can increase consolidation of transport when products of multiple companies travel in the same direction. As a result, it enables considerable improvements in truck utilization and reductions in CO_2 emissions (Sarraj et al., 2014a). Accordingly, the European Commission strives to realize PI by 2050 (ETP-Alice, 2017; EC, 2017). Recently, several authors have studied inventory control in PI networks (Pan et al., 2015; Yang et al., 2017a,b). The common approach in this stream of research is to construct a PI network from an existing logistics network by introducing extra edges between warehouses of multiple companies. Subsequently, in order to demonstrate its potential cost-savings, the PI network is compared with the classical network in simulation experiments. Pan et al. (2015) and Yang et al. (2017a) show that, under the same service level constraints, the possibility of dynamic supplier selection in a PI network leads to significantly lower inventories than in the existing logistics network considered. Yang et al. (2017b) instead show that dynamic supplier selection can significantly reduce the cost associated with being resilient against supply chain disruptions. As the above literature makes no comparisons between different PI networks and all decisions are taken heuristically, it does not address the question what constitutes effective PI network configurations and inventory policies. We aim to bridge this gap by comparing different network configurations with each other under optimal decision-making.

The remainder of this paper is organized as follows. In Section 2, we introduce a modeling framework for dynamic shipments in warehouse networks and a Markov decision process for obtaining optimal inventory and transportation decisions. In addition, we analyze feasibility constraints for shipment decisions. In Section 3, we apply the modeling framework to a company responsible for inventory control at several warehouses in an SWTN. Through numerical experiments in Section 4, we compare optimal decisions and costs for different networks with each other. Finally, in Section 5 and 6 we discuss the main insights and conclude the paper.

2. Modeling Framework

This section introduces a modeling framework for dynamically transporting products through warehouse networks in discrete time under discrete stochastic demand. Since we consider networks, from now on we use the words warehouse and node interchangeably. We call transportation from external suppliers to nodes in the network *orders* and transportation between nodes in the network *shipments*. We first discuss the modeling framework and give examples of models fitting in the framework. Afterwards, we specify the cost structure and discuss some limitations of the model.

Let n be the number of nodes and let $N = \{i, i = 1, \dots, n\}$ be the set of nodes in the network. We consider a periodic review model, where periods are indexed by t. As is common in inventory control, the order of events during a period is as follows:

1. Incoming orders and shipments arrive at the nodes.
2. New order and shipment decisions are taken.
3. Random demand arrives and is fulfilled from on-hand stock. Depending on the application, unfulfilled demand is backordered and/or lost.
4. Inventory costs are incurred.

The framework has the following characteristics.

- Edges in the network are defined by an $n \times n$ binary matrix $A = (a_{ij})$, where $a_{ij} = 1$ indicates that it is possible to ship from node i to j. If $a_{ii} = 1$, node i is a *stock-keeping node* otherwise, if $a_{ii} = 0$, node i is a *transfer node*. Transfer nodes can be used to model warehouses without storage space such as cross-docking facilities. Without loss of generality, A must be such that the network is connected, because if a subset of nodes $\mathcal{S} \subset N$ and its complement $\mathcal{S}^c = N \backslash \mathcal{S}$ are disconnected, $\mathcal{S}$ and $\mathcal{S}^c$ can be analyzed in isolation. Therefore, we require

  $$\sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}^c} a_{ij} + a_{ji} \geq 1 \text{ for all } \mathcal{S} \subset N,$$

  i.e., each subset of nodes has an edge to another node in the network. At more appropriate points we discuss other restrictions on A.

- All lead times are integer and deterministic. Let $N_o \subseteq N$ be the subset of nodes that can place orders at and receive orders from an external supplier. An order at node $i \in N_o$ has a lead time of $L_i \geq 1$ periods, meaning that an order placed at the start of period t is delivered to node i at the start of period $t+L_i$. A shipment initiated at the start of period t arrives at its destination at the start of period $t+1$, hence all shipments have a lead time of one period. Longer shipment times can be considered by introducing dummy transfer nodes with $a_{ii} = 0$. Although integrality seems restrictive, in theory any combination of order and shipment lead times can be approximated by adding a number of intermediate transfer nodes and scaling costs and demand appropriately.

- Nodes $i \in N \backslash N_o$ can only receive stock through shipments from other nodes. We assume each node can receive stock, otherwise it can be removed from the network without loss of generality. Therefore, A must be such that for each node $i \in N$ there is a directed path from some node $j \in N_o$ to i.

- The vector $D = (D_i)_{i \in N}$ gives the random demand at all nodes, where D_i is the random demand at node i, $i \in N$. All D_i are non-negative, discrete, and finite. Since our goal is to determine average-cost optimal policies, we assume that all D_i are time-homogeneous. The joint cumulative distribution function of D is given by F, which in general allows demand of different nodes to be correlated with each other.

- Demand is met from on-hand stock to the extent possible. We assume that up to B units of demand per node can be backordered; additional units of demand are lost. This assumption is quite general since it has pure lost sales ($B = 0$) and pure backordering ($B = \infty$) as special cases.

- The demand distribution F and shipment matrix A need to satisfy several practical restrictions. Transfer nodes face no demand, so F should be such that $P(D_i = 0) = 1$ when $a_{ii} = 0$ for some $i \in N$. Nodes with positive demand need to be able to hold inventory, so we require $a_{ii} = 1$ when F is such that $P(D_i = 0) < 1$ for some $i \in N$. Note that the latter assumption is critical when $B = 0$; otherwise all demand at node i is lost. It is possible that stock-keeping nodes face no demand.

By being flexible in the choice of the shipment matrix A, the framework generalizes many well-known inventory models, including multi-echelon models (De Kok et al., 2018). Consider the following examples. A situation with n parallel warehouses can be modeled with $N_o = N$ and $A = I_n$, where I_n is the $n \times n$ identity matrix. Shipments between all warehouses can be enabled with $N_o = N$ and $A = J_n$, where J_n is the $n \times n$ matrix of ones. For $n = 3$, a directed two-echelon model follows from $N_o = \{1\}$ and

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Now we specify the cost structure, which consists of cost functions for orders, shipments, on-hand inventory, backorders, and lost sales. We assume that the cost functions are finite, non-negative, and non-decreasing in their function arguments. This is reasonable in practice, since typically costs are positive and higher quantities result in higher costs. For our goal of studying average cost optimal policies, we assume all costs are time-homogeneous.

Ordering costs are given by the function $k_i(.)$, which depends on the order quantity at node i, $i \in N_o$. Shipment costs for shipping a number of units from node i to j are given by the function $c_{ij}(.)$. Typical order and shipment cost functions contain a fixed setup cost and/or a variable cost depending on number of items ordered/shipped. Holding costs, given by the function $h_i(.)$, are incurred each period depending on the number of units on-hand at and shipped to node i. Units shipped to node i also incur holding costs, since otherwise unrealistic situations may arise where units are shipped constantly in order to avoid holding costs. Backorder costs, given by $b_i(.)$, are incurred depending on the number of backorders at node i. Similarly, lost sales costs, given by $\ell_i(.)$, are incurred depending the number of lost sales at node i.

In our modeling, we implicitly assume that there are always trucks available that can transport the products and that trucks have ample space. While availability and space are practical considerations, we prefer to study network flows in their purest form to determine to what extent edges would ideally be used. For the same reason, we do not consider batching of orders and shipments. Batching can be implemented by constraining the number of items shipped and ordered to be multiples of the batch size. If the batch sizes for ordering and shipments are the same, which seems reasonable for products transported in containers, it is also possible to regard units in the model as complete batches and demand as demand for complete batches. Since demand at a node is met when there is on-hand stock, we also do not consider rationing (Ha, 1997; Deshpande et al., 2003; Escalona et al., 2015). While rationing can reduce costs when nodes have different holding and penalty costs, the main interest of this article is pure ordering and shipment decisions.

2.1. Markov Decision Process

In this section, we formulate a Markov decision process (MDP) for obtaining optimal order and shipment decisions. MDPs have been successfully applied to other inventory control problems (Puterman, 2009; Zhao et al., 2010). For a two-echelon network, Zhao et al. (2010) solve, heuristically, a similar MDP as ours by decomposing the problem into several one-dimensional MDPs for each node in the network. We instead solve the full-dimensional MDP, as we aim for optimal solutions and as it is unclear whether the general problem can be decomposed. The size of the state space is subject to the curse of dimensionality, so that solutions can be obtained only with limited numbers of nodes, lead times, and demand sizes. However, the models that can be solved are large enough to generate valuable insights into the effect of network structure on optimal decision-making.

The following variables are used in the MDP. At time t, $I_i(t)$ is the on-hand inventory at node i and $S_i(t)$ is the number of shipments en route to node i, $i \in N$. $O_{ip}(t)$ is the number of items ordered p periods ago at node i, $i \in N_o$, $p = 0, \ldots, L_i - 1$. The state $x(t)$ of the MDP at time t is then

$$x(t) = \{I_i(t), S_i(t), O_{jp}(t) \mid i \in N, j \in N_o, p = 0, \ldots, L_j - 1\}.$$

When no subscript is included, $I(t)$ and $S(t)$ denote the n-vectors of inventory and shipments at all nodes. Furthermore, for an arbitrary scalar or vector z, we define $(z)^+ = \max\{z, 0\}$ and $(z)^- = \max\{-z, 0\}$, defined elementwise when z is a vector.

Now we discuss the inventory dynamics in our network. Recall that a period starts with arrivals of previous orders and shipments, then decisions are taken for new orders and shipments, and then demand arrives.

We start by explaining the decisions. We let $x(t)$ be the *pre-decision state*, i.e., the state just before decisions, and $x'(t)$ the *post-decision state*, i.e., the state just after decisions. Order decisions $o_i \in \mathbb{N}, i \in N_o$, specify the number of ordered items at node i. Shipment decisions $s_{ij} \in \mathbb{N}$, $i, j \in N$, specify the number of items shipped from node i to j. Shipping from a warehouse to itself is not possible, hence $s_{ii} = 0$ for all $i \in N$. The post-decision state resulting from a pre-decision state and order and shipment decisions is given by

$$\begin{aligned}
I'_i(t) =& I_i(t) - \sum_{j \in N} s_{ij} && i \in N, \\
O'_{i0}(t) =& o_i && i \in N_o, \\
O'_{ip}(t) =& O_{ip}(t) && p = 1, \ldots, L_i - 1, i \in N_o, \\
S'_i(t) =& \sum_{j \in N} s_{ji} && i \in N.
\end{aligned}$$

We believe the interpretation follows directly from the definitions of the variables. The shipment decisions must satisfy several intuitive constraints. If $a_{ij} = 0$ then $s_{ij} = 0$ (ship only over available edges), if $a_{ii} = 0$ then s_{ij} must be such that $I'_i(t) = 0$ (transfer nodes keep no inventory), if $I_i(t) \leq 0$ then $s_{ij} = 0$ (do not ship if there is no inventory), and if $I_i(t) > 0$ then $\sum_{j=1}^{n} s_{ij} \leq I_i(t)$ (ship no more than is available).

In the post-decision state, the demand becomes known. The inventory after demand, with demand beyond B being lost, is given by

$$I''_i(t) = \max\left(I'_i(t) - D_i(t), -B\right).$$

Accordingly, the number of lost sales is $(D_i(t) - I'_i(t) - B)^+$. After meeting demand the next period starts with the arrival of orders and shipments. For a given demand $D_i(t)$, the new pre-decision state in period $t+1$ becomes

$$I_i(t+1) = I''_i(t) + S'_i(t) \qquad i \in N \backslash N_o \tag{1}$$

$$I_i(t+1) = I''_i(t) + S'_i(t) + O'_{i,L_i-1}(t) \qquad i \in N_o \tag{2}$$

$$O_{ip}(t+1) = O'_{i,p-1}(t) \qquad p = 1, \ldots, L_i - 1, i \in N_o \tag{3}$$

$$O_{i0}(t+1) = 0 \qquad i \in N_o \tag{4}$$

$$S_i(t+1) = 0 \qquad i \in N \tag{5}$$

Eqs. (1) and (2) adds incoming shipments and orders to the inventory after demand. In (3) the orders in the pipeline are updated, which arrival is now one period closer. Finally, in (4) and (5) the just arrived orders and shipments are reset to 0.

Now let $d(t) = \{s_{ij}, o_k \mid i, j \in N, k \in N_o\}$ be shorthand for the decision in period t. The expected cost in the current period for taking decision $d(t)$ in state $x(t)$ is

$$\begin{aligned} C(x(t), d(t)) = &\sum_{i \in N_o} k_i(o_i) + \sum_{i \in N} \sum_{j \in N} c_{ij}(s_{ij}) \\ &+ E\left[\sum_{i \in N} h_i\left(S'_i(t) + (I''_i(t))^+\right) + b_i\left((I''_i(t))^-\right) + \ell_i\left((D_i(t) - I'_i(t) - B)^+\right)\right]. \end{aligned} \tag{6}$$

The costs include the direct costs for ordering and shipments and the expected holding, backorder, and lost sales cost at the end of the period. Recall that the term $S'_i(t)$ is included to charge holding costs for stock shipped to node i.

Let π be the policy, i.e., a function specifying for each state $x(t)$ the decision $d(t)$ to be taken. The average cost of starting in state x and following this policy is

$$g^\pi(x) = \lim_{T \to \infty} \frac{1}{T} E_x^\pi \left[\sum_{t=1}^{T} C(x(t), d(t))\right].$$

We are interested in the policy π^* that minimizes $g^\pi(x)$ for every possible x. When demand and costs are time-homogenous, it can be shown that a stationary optimal policy π^* exists if the unichain property holds and the cost function and state space are finite. The remark below discusses a restriction on the order decisions that is necessary to solve the MDP.

Remark 1. The number of possible order decisions is infinite if we allow for placing orders with an infinite number of items. However, this is suboptimal when the demand distributions have a finite support and all costs are positive. We ensure finiteness of the action space and state space by restricting the number of items in an order. Let $I_{\max}$ be the maximum number of items in the system, then orders should satisfy

$$\sum_{k \in N_o} o_k \leq I_{\max} - \sum_{i \in N} (I_i(t))^+.$$

This restriction has no effect when $I_{\max}$ is higher than the maximum number of items in the system under the optimal policy. The following procedure can be used to obtain a sufficiently high value for $I_{\max}$. First, solve the MDP with restricted orders for a given $I_{\max}$. Determine the set of reachable states under the optimal policy of this MDP, for example, by checking connected components of the Markov chain transition matrix induced by this policy. If all reachable states have lower total inventory than $I_{\max}$, then $I_{\max}$ is sufficiently high, otherwise increase $I_{\max}$ and repeat the same steps.

2.2. Analysis of Feasible Shipment Decisions

For any network structure, we derive constraints for feasible post-decision states that are reachable after shipments. Depending on the network structure, some of these constraints are superfluous, so we also provide conditions to determine which constraints can be removed. We show that in general the number of required constraints increases exponentially with the number of nodes in the network, however, the increase is linear for several well-known classes of network structures.

In principle, all feasible shipment decisions should be enumerated in order to obtain the optimal solution of the MDP. We exploit the fact that often multiple feasible shipment decisions lead to the same post-decision state. It suffices, therefore, to only consider minimum cost shipment decisions from a given pre-decision state to a given post-decision state. Depending on the functional form of $c_{ij}(s_{ij})$, it can be challenging to obtain minimum cost shipment decisions. In the special case where $c_{ij}(s_{ij})$ is linear in s_{ij}, minimum cost shipment decisions are readily obtained by solving a transportation problem for any pair of pre- and post-decision states.

Dropping time from the notation for ease of exposition, we determine all feasible post-decision state variables (I', S') that can result from a given pre-decision inventory I. We can assume $I \geq 0$, since when $I_i < 0$ for some i, we can calculate feasible post-decision states for on-hand inventory $I + (I)^-$ and afterwards subtract the backorders $(I)^-$ from each of these states.

Feasibility of a post-decision state depends on the shipment matrix A. Since shipments are generally not possible between any pair of nodes, we need to take into account where items come from. For a single node i, $i \in N$, it is useful to define the neighbor set $\mathcal{T}(\{i\})$ as

$$\mathcal{T}(\{i\}) = \{j : a_{ij} = 1 \mid j \in N, j \neq i\}.$$

For sets of nodes $\mathcal{S} \subset N$, the neighbor set is defined as the union of its single node neighbor sets, i.e.,

$$\mathcal{T}(\mathcal{S}) = \bigcup_{i \in \mathcal{S}} \mathcal{T}(\{i\}).$$

Feasible post-decision states (I', S') need to satisfy

$$\sum_{i \in N} (I_i - I'_i) = \sum_{i \in N} S'_i, \tag{7}$$

$$I'_i \leq a_{ii} I_i \qquad \text{for } i \in N, \tag{8}$$

$$\sum_{i \in \mathcal{S}} (I_i - I'_i) \leq \sum_{j \in \mathcal{T}(\mathcal{S})} S'_j \qquad \text{for } \mathcal{S} \subset N, \mathcal{S} \neq \emptyset. \tag{9}$$

Constraint (7) ensures that all current inventory ends up at some node. Due to constraints (8), inventory after shipments is limited to what is currently on-hand, and it is ensured that transfer nodes keep no inventory. The subset constraints (9) ensure that the total inventory in a non-empty subset of nodes ends up at least at itself or at its neighbor nodes, or, in other words, inventory cannot be shipped to non-neighbors. We let $\mathbb{F}(I)$ denote the set of all post-decision states (I', S') satisfying constraints (7-9).

For each shipment matrix A, there exist subsets $\mathcal{S} \subset N$ for which the subset constraints (9) are superfluous. Removal of these constraints significantly speeds up feasibility checking. The following theorem specifies which subsets can be removed; the proof is trivial and therefore omitted.

Theorem 1. *Let $\mathcal{S}$, $\mathcal{S}_1$, $\mathcal{S}_2 \subset N$ be nonempty subsets. Constraints* (9) *are superfluous for the following subsets:*

(a) *$\mathcal{S}$ with $\mathcal{T}(\mathcal{S}) = N$.*
(b) *$\mathcal{S} \subset \mathcal{S}_1$ with $\mathcal{T}(\mathcal{S}) = \mathcal{T}(\mathcal{S}_1)$.*
(c) *$\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2$ with $\mathcal{S}_1 \cap \mathcal{S}_2 = \emptyset$ and $\mathcal{T}(\mathcal{S}_1) \cap \mathcal{T}(\mathcal{S}_2) = \emptyset$.*

The results in Theorem 1 can be explained as follows. The subset in (a) has no non-neighbors, therefore it is not necessary to check whether its inventory is shipped to non-neighbors. The subset in (b) is contained in a larger subset with the same neighbors, for which it is immediate that constraint (9) is more restrictive. The subset in (c) is the union of disconnected subsets with disconnected neighbor sets. It is sufficient to consider these disconnected subsets in isolation.

For several important classes of network structures, we derive how many subset constraints are needed in total.

Theorem 2. *The following holds for classes of network structures with n nodes.*

(a) *Divergent multi-echelon networks require exactly n subset constraints.*
(b) *Convergent multi-echelon networks require at most n subset constraints.*
(c) *Undirected two-echelon networks require exactly 1 subset constraint.*
(d) *Complete graphs require exactly n subset constraints.*
(e) *For odd-numbered n, $n \geq 3$, friendship graphs requires exactly* $2^{n-1} - 1$ *subset constraints.*

Proof. (a) Consider a divergent multi-echelon network, defined as a connected directed network with a root node, where each node is reachable from the root node through a unique directed path. For every pair of nodes i and j, $i \neq j$, we have $\mathcal{T}(\{i\}) \cap \mathcal{T}(\{j\}) = \emptyset$. By Theorem 1 (c), it therefore suffices to consider the subsets $\{i\}$ for all $i \in N$.

(b) Consider a convergent multi-echelon network, defined as a connected directed network with an end-node, where for each node there exists a unique directed path towards the end-node. For node i, let $\mathcal{S}_i$ be its set of predecessors, i.e., $\mathcal{S}_i = \{j : \mathcal{T}(\{j\}) = \{i\}, j \in N\}$, which can be empty. By the structure of the network, all predecessors in a non-empty $\mathcal{S}_i$ have node i as only successor, hence $\mathcal{T}(\mathcal{S}_i) = \{i\}$. For all nodes i and j, $i \neq j$, we have $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ and $\mathcal{T}(\mathcal{S}_i) \cap \mathcal{T}(\mathcal{S}_j) = \emptyset$. Trivially, the same holds true when comparing subsets of $\mathcal{S}_i$ and $\mathcal{S}_j$. Let $\mathcal{S}$ be a subset containing nodes from different predecessor sets, i.e., there exist i and j such that $\mathcal{S} \cap \mathcal{S}_i \neq \emptyset$ and $\mathcal{S} \cap \mathcal{S}_j \neq \emptyset$. Then we can always divide $\mathcal{S}$ into at least two disconnected subsets with disconnected neighbor sets. Hence, by Theorem 1 (c), all subsets of nodes which include nodes from different predecessor sets can be skipped. In addition, by Theorem 1 (b), all non-empty subsets of predecessor sets can be skipped. Hence, we need only consider non-empty predecessor sets $\mathcal{S}_i$ for $i \in N$.

(c) In an undirected two-echelon network, $n - 1$ nodes are connected only to a root node. Letting node 1 be the root node, we only require the subset constraints for $\mathcal{S} = \{1\}$ and $\mathcal{S} = N \backslash \{1\}$. Due to symmetry and due to Eq. (7), the constraints can be rewritten as $(I_1 - I_1') = \sum_{i \in N \backslash \{1\}} S_i$.

(d) All nodes in complete graphs are connected to each other. All subsets $\mathcal{S}$ with $|\mathcal{S}| > 1$ have neighbor set $\mathcal{T}(\mathcal{S}) = N$, hence they can be removed due to Theorem 1 (a). The remaining subsets are the subsets $\{i\}$, $i \in N$.

(e) The friendship graph is an undirected graph graph where $k = \frac{n-1}{2}$ triangles are connected at one common node (Gallian, 2009). Let node 1 be the common node. For all nodes $i \in N \backslash \{1\}$, $\mathcal{T}(\{i\})$ includes node 1 and one other node. Hence, there exist no disconnected subsets with disconnected neighbor sets. All subsets $\mathcal{S}$ containing node 1 and at least one other node have neighbor set $\mathcal{T}(\mathcal{S}) = N$. In addition, the set $\mathcal{S} = N \backslash \{1\}$ has $\mathcal{T}(\mathcal{S}) = N$. By Theorem 1 (a), all such subsets can be removed. Therefore, we need only consider the subset $\mathcal{S} = \{1\}$ and all subsets $\mathcal{S} \subset N \backslash \{1\}$, except the empty set and $N \backslash \{1\}$ itself, giving $2^{n-1} - 1$ subsets in total. □

For convergent and divergent multi-echelon networks, the number of required subset constraints increases linearly in the number of nodes. In general, when graphs contain cycles, an exponential number of constraints is required. However, for some graphs with cycles, such as the complete graph, the number of subset constraints is linear. We empirically checked the number of required constraints for all possible undirected graphs with $n \leq 9$ nodes. Among undirected graphs with odd-numbered n, $2^{n-1} - 1$ is the maximum number of required constraints, attained by the friendship graph.

3. Application to a Shared Warehouse and Transportation Network

We apply the modeling framework to the case of a company responsible for the inventory control at all nodes in an SWTN. We first specify suitable cost functions to model this case. Except for ordering, we assume that the cost functions are linear. We also assume that unit cost parameters do not vary between nodes. For each type of cost we now motivate our choices.

Ordering incurs a fixed cost since in practice there is often a setup cost for bringing (a batch of) products into the network. Using existing connections in an SWTN, typically shipment costs are charged per unit of volume/weight and per unit of distance driven (Sarraj et al., 2014b), so linear shipment costs are used. Equal unit shipment costs on each edge are appropriate when distances between nodes in the network are similar. Shipped items are handled twice on each edge (loading and unloading). Hence, when shipment costs also include handling costs, shipment costs will be quite close to each other regardless of distance. In case shipment costs do differ significantly, dummy transfer nodes can be added to better represent travel time and travel costs on edges with longer distances.

For most practical settings it is common to have linear holding, backorder, and lost sales costs, so we explain why these unit costs do not vary between nodes. The holding cost is often estimated as missed interest over the purchase cost. Value adding processes, such as assembly and further processing, are typically not carried out at intermediate nodes in a transportation network. Therefore, we assume equal unit holding costs at each node. The lost sales and backorder cost are typically based on the difference between sales price and purchase cost in addition to loss of goodwill. As these factors should be of similar order for each node, each node has the same lost sales and the same backorder cost.

In summary, the cost functions in Eq. (6) become

$$\begin{aligned} k_i(o_i) &= K\mathcal{I}[o_i > 0], \\ c_{ij}(s_{ij}) &= cs_{ij}, \\ h_i\left(S_i'(t) + (I_i''(t))^+\right) &= hS_i'(t) + h(I_i''(t))^+, \\ b_i\left((I_i''(t))^-\right) &= b(I_i''(t))^-, \\ \ell_i\left((D_i(t) - I_i'(t) - B)^+\right) &= \ell(D_i(t) - I_i'(t) - B)^+, \end{aligned}$$

where K is the fixed order cost, and c, h, b, and ℓ are the unit costs for shipments, holding inventory, backorders, and lost sales. The indicator function $\mathcal{I}[o_i > 0]$ equals 1 if $o_i > 0$ and 0 otherwise.

In the remainder of this section, we propose simple procedures to identify and eliminate suboptimal decisions. After that, we discuss the calculation of steady-state network flows and we introduce an MDP with static shipments used for benchmarking.

3.1. Identifying and Removing Cycles

We identified the set of feasible post-decision states $\mathbb{F}(I)$ in Section 2.2. Under the assumption that shipment costs are linear and equal for each edge, we can identify and eliminate various suboptimal post-decision states from $\mathbb{F}(I)$. This is the case because some post-decision states in $\mathbb{F}(I)$ correspond to inefficient shipment decisions with cycles.

An example is shown in Figure 1. In (a), the network is shown with the starting inventory I_i at each node. In (b), post-decision states with cyclic and non-cyclic shipment decisions are shown, both leading to the same end-inventories $I' + S'$. The corresponding shipment decisions for these states are shown in (c) and (d). The shipment decisions in (d) were obtained from (c) by removing one shipment from each edge on the cycle $1 \to 2 \to 3 \to 1$. When each node has the same penalty cost for backorders and lost sales, there is no incentive to ration for nodes with higher penalty costs. This implies non-cyclic decisions are strictly better than cyclic decisions leading to the same end-inventories. Therefore, all feasible post-decision states with such cyclic decisions can be removed from $\mathbb{F}(I)$.

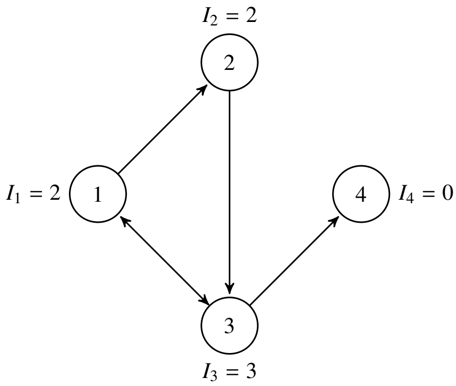
We now determine whether a post-decision state $(I', S') \in \mathbb{F}(I)$ has a cyclic shipment decision by checking whether feasible post-decision states with fewer shipments exist that yield the same end-inventories $I' + S'$. Let the set of stock-keeping nodes that simultaneously ship and receive at least one item be

$$\mathcal{S} = \{i : S_i'(I_i - I_i') > 0, a_{ii} = 1, i \in N\}.$$

If $\mathcal{S} = \emptyset$ then there is clearly no cycle. For each nonempty subset $\hat{\mathcal{S}} \subseteq \mathcal{S}$ we introduce test state $\hat{I} = I'$, $\hat{S} = S'$. For nodes $i \in \hat{\mathcal{S}}$, we update $\hat{I}_i := \hat{I}_i + 1$ and $\hat{S}_i := \hat{S}_i - 1$. If this test state satisfies constraints (7-9), then end-inventory $I' + S'$ could be achieved with fewer shipments. The original post-decision state then has a cyclic shipment decision. If the test state is infeasible for all subsets $\hat{\mathcal{S}}$, then we conclude the original post-decision state has no cyclic shipment decisions.

Figure 1 helps to illustrate the procedure. For the post-decision state with a cyclic decision, $\mathcal{S} = \{1, 2, 3\}$. The test state for $\hat{\mathcal{S}} = \{1, 2, 3\}$ is in fact the post-decision state for the non-cyclic decision, which is feasible, hence there is a cycle. Applying the same procedure to the post-decision state for the non-cyclic decision leads to infeasibility, since $\mathcal{S} = \{1\}$ and reducing the shipments to and from node 1 prohibits node 2 from reaching its end-inventory.

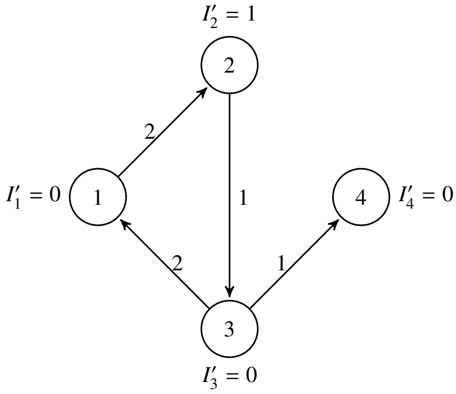
Let $\mathbb{F}_{\text{nc}}(I) \subseteq \mathbb{F}(I)$ be the set of all post-decision states with non-cyclic shipment decisions. For some combinations of A and I, the set $\mathbb{F}_{\text{nc}}(I)$ is over 90% smaller than $\mathbb{F}(I)$. In the MDP we need to evaluate all combinations of feasible order and shipment decisions, hence each fewer shipment decision reduces the total number of decisions by the size of the order set. This conserves memory and speeds up computations when storing decisions in memory for repeated use.



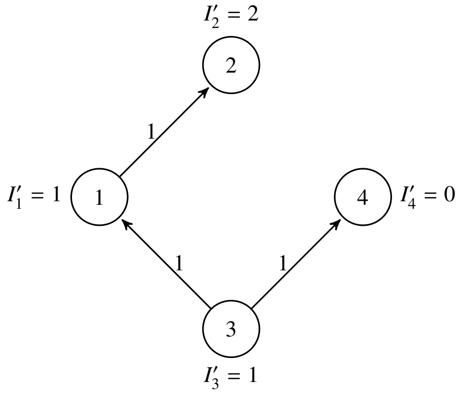
(a) Network structure and inventories

	End-inventory	Cyclic decision		Non-cyclic decision	
Node	$I'_i + S'_i$	I'_i	S'_i	I'_i	S'_i
1	2	0	2	1	1
2	3	1	2	2	1
3	1	0	1	1	0
4	1	0	1	0	1

(b) Example decisions



(c) A cyclic decision



(d) A non-cyclic decision

Figure 1: Example of a cyclic and non-cyclic shipment decision

Remark 2. Suppose that two vectors of inventory levels I and $\bar{I}$ are given with $0 \le I \le \bar{I}$. In this case, it is possible obtain $\mathbb{F}_{\text{nc}}(I)$ directly from $\mathbb{F}_{\text{nc}}(\bar{I})$. With inventory levels I, there are $\bar{I} - I$ fewer on-hand items before shipments. In order to keep non-negative inventories after shipments, we need only consider post-decision states $(I', S') \in \mathbb{F}_{\text{nc}}(\bar{I})$ for which $I' \ge \bar{I} - I$. After subtracting $\bar{I} - I$ from each post-decision state, we obtain

$$\mathbb{F}_{\text{nc}}(I) = \left\{ \left(I' - (\bar{I} - I), S' \right) : I' \ge \bar{I} - I,\ (I', S') \in \mathbb{F}_{\text{nc}}(\bar{I}) \right\}.$$

Checking for cycles is only required for inventory levels I in the set $\{I : \sum_{i \in N} I_i = I_{\max}\}$. This set is considerably smaller than the set of all possible inventory levels $\{I : \sum_{i \in N} I_i \le I_{\max}\}$.

3.2. Calculating Steady-State Network Flows

In order to compare flows through different networks in steady-state, we explain how to calculate the per-period average flows of items over an edge. After the optimal policy of the MDP has been calculated, optimal flows over all edges are obtained by solving a transportation problem for each optimal pair of pre- and post-decision states. Vogel's Approximation Method is used to obtain a starting solution of the transportation problem. Since typically multiple optimal solutions exist when shipment costs are identical, the resulting optimal flows depend on the way ties are broken and on the starting solution. The probability of being in each pre-decision state follows by calculating the steady-state distribution of the Markov chain induced by the optimal policy of the MDP (Norris, 1998). The average flows on each edge then follow from taking the expectation of optimal flows in each pre-decision state with respect to the steady-state distribution. Other performance measures, such as average orders and inventory levels, can be obtained analogously.

3.3. *Static Shipments Case*

Our model has dynamic shipments because each period it is possible to ship items between any pair of connected nodes depending on the system state. For determining the added value of such dynamic shipments, we need to introduce a benchmark with static shipments. In the static shipment case, as soon as an order is placed at node $j \in N_o$, for each item in the order it must be specified which node $i \in N$ receives the item. If Z_{ji} is the length of the shortest path from node j to i, then the items in the order at node j assigned to node i arrive at node i exactly $\bar{L}_{ji} = L_j + Z_{ji}$ periods from now. Items assigned to node i cannot be used by other nodes, so, unlike the model in Section 2, nodes may have on-hand inventory and backorders simultaneously. This situation occurs when items assigned to node i travel through another node with backorders.

We implemented an MDP to obtain optimal orders for the static shipments case. In the state variable, it is sufficient to track for each node $i \in N$ the number of items arriving $p = 1, \ldots, \max_{j \in N_o}\{\bar{L}_{ji}\}$ periods from now. Typically this MDP is more difficult to solve than its counterpart with dynamic shipments because of larger pipelines of orders, especially if the distance between order nodes and destination nodes is large.

4. Numerical Experiments

For the SWTN described in Section 3, we carry out several numerical experiments. The goal is to obtain a deeper understanding of how network structure influences optimal costs and decision-making under different circumstances. This gives insight in which connections play an important role in the network, which could be of assistance when designing new networks or picking the most suitable network from a selection of candidate networks.

The MDP has been implemented in Python 3.6 and the computations are carried out on an Intel Core i3-6100T CPU (3.2GHz) with 8GB of memory. Value iteration is used, where we set $\epsilon = 10^{-6}$ to obtain an ϵ-optimal policy (Puterman, 2009). The bottleneck in the code is computing actions and transitions, hence, for each state, indices of reachable states with corresponding costs and probabilities are stored in memory for repeated use. In every iteration, we first calculate post-decision state values, i.e., the expectation of pre-decision state values after demand. These post-decision state values can be reused for all combinations of pre-decision states and decisions leading to the same post-decision state. This approach speeds up value iteration by several orders of magnitude, at the expense of increased memory consumption.

Section 4.1 discusses the size of solvable networks. Section 4.2 introduces an experimental design that forms the basis for detailed comparisons between network structures with four warehouses, including non-arborescent structures. These comparisons are carried out in Section 4.3 and 4.4, where average costs and optimal flows of products are investigated for different network structures. Sensitivity of the results is tested in Section 4.5. Finally, in order to quantify the possible gains from using dynamic shipments, situations with dynamic and static shipments are compared in Section 4.6.

4.1. *Size of Solvable Networks*

We first investigate the maximum size of solvable networks in our implementation of the MDP. To that end, we consider serial networks, where all nodes are located on a line. Each row in Table 1 shows a network structure together with the choice for parameters, the number of states in the state space, the number of mass points in the demand distribution, and several other computational statistics. For each number of nodes n, $2 \leq n \leq 5$, the instances shown are among the largest that can be solved with 8GB of memory.

For any network in Table 1, the order node is indicated by a square and has lead time $L_i = 1$; other nodes are indicated by circles. From left to right, nodes independently face Poisson demand with the rates specified in the vector λ, which may be 0 for some nodes. The number of states in an instance depends only on n, $I_{\max}$, and B. For a given value of n, we tuned the parameters $I_{\max}$ and B to keep a tractable number of states. Accordingly, the values in λ are set as large as possible, in such a way that $I_{\max}$ does not restrict any optimal actions, see also the discussion in Remark 1. Finally, the number of demand mass points is kept finite by truncating Poisson demand at the 0.999th quantile. The cost parameters, defined in Section 3, have limited influence on computation times. Hence, those are kept constant at $K = 50$, $c = 5$, $h = 1$, $\ell = 60$, and $b = 20$.

All of the instances in Table 1 can be solved within 20 minutes and most instances require around 100 value iterations. States are vectors of length $2n$, so when $I_{\max}$ and B are kept fixed, the number of states increases quickly in

Table 1: Computational statistics of various instances. In the column 'Network', □ is an order node and ○ a regular node.

Network	n	$I_{\max}$	B	λ	States	Demand mass points	Solution time (s)	Itera- tions	Average cost
□—○	2	100	0	$(0, 20)$	4,598,126	36	126.17	216	174.03
□—○	2	100	20	$(0, 18)$	8,135,146	33	148.99	85	156.33
□—○	2	100	0	$(7, 7)$	4,598,126	289	344.62	91	90.17
□—○	2	80	20	$(6, 6)$	6,933,141	256	540.24	75	77.10
□—○—○	3	35	0	$(1, 1, 1)$	4,496,388	216	179.03	98	42.63
□—○—○	3	45	0	$(0, 0, 4)$	18,009,460	12	561.56	91	73.43
□—○—○	3	35	10	$(0, 0, 2.5)$	11,076,468	10	286.80	95	48.81
○—□—○	3	35	0	$(1, 1, 1)$	4,496,388	216	202.30	103	36.99
□—○—○—○	4	25	0	$(0, 0.5, 0.5, 0.5)$	13,884,156	125	775.60	115	35.84
□—○—○—○	4	20	2	$(0, 0.3, 0.3, 0.3)$	11,624,085	64	539.56	107	23.98
□—○—○—○—○	5	18	0	$(0, 0.2, 0.2, 0.2, 0.2)$	13,123,110	256	998.19	131	26.03

n. For $n = 2$, instances with reasonably high demand rates can be solved, regardless of whether there are backorders ($B = 20$) or lost sales ($B = 0$). For higher n, the values of $I_{\max}$ and B need to decrease quickly to keep a tractable number of states. By comparing computation times of networks with $n = 2$, we see that the choice of λ has significant impact on computation times. In particular, when both nodes face demand, the number of demand mass points is relatively high, as are the computation times. Based on the above analysis, we further investigate networks with $n = 4$ nodes, which can still be solved and feature a rich variety of network structures.

4.2. Experimental Design

In the remainder of this section, we consider an experiment with 24 instances. In each instance, several network structures are compared with each other. We consider all undirected networks with $n = 4$ nodes, where a main node (node 0) receives orders from an external supplier with lead time $L_0 = 1$ and the remaining nodes (nodes 1, 2, and 3) are demand points with independently and identically distributed demand. All 11 unique such networks are depicted in Figure 2. While seemingly simple, 7 out of the 11 networks contain cycles and, as a consequence, have never been solved analytically in the literature. Network names are based on distance and flexibility. Networks are first grouped by distance between the main node and the three demand nodes (A=[1,1,1], B=[1,2,1], C=[1,2,2], D=[1,2,3]) and then sorted in descending order by the number of edges (B2 and B3 have the same rank). As the action space of A1 includes the actions of any other network, that network structure forms the benchmark for cost comparisons.

In the experiment, we carry out the optimization at all levels of selected parameters. The following levels have been chosen for the parameters in the instances. The backorder cost takes on values $b \in \{10, 20\}$, the shipment cost $c \in \{0, 1, 5\}$, and the backorder level $B \in \{0, 2\}$. Demand distributions have mean 0.1, with either a low variance of 0.1 or high variance of 0.2. For low variance, the distribution is truncated Poisson with pdf $f(0) = 0.9048$, $f(1) = 0.0905$, $f(2) = 0.0047$. For high variance, the distribution has pdf $f(0) = 0.9433$, $f(1) = 0.025$, $f(2) = 0.02$, $f(3) = 0.0116$. As one parameter can always be fixed, the holding cost is $h = 1$ everywhere. Since backorders are typically preferred to lost sales, we set the lost sales cost at $\ell = 3b$. After testing we decided to have one value for the order cost ($K = 50$); results for other values are similar. A total inventory of $I_{\max} = 17$ was found to be sufficient for all instances.

4.3. Cost Comparison of Different Networks

Figure 3 shows the results for the 24 instances in the experimental design from Section 4.2. The bar graph in each instance shows the percentage increase in costs for the different networks compared to the benchmark network structure A1. The instance number is shown above each bar graph and parameter settings for an instance can be read from the labels above its column and to the right of its row. For example, graphs above the middle have low variance and below the middle have high variance. As one has a direct penalty cost and the other a penalty cost per time unit, heights of bars in instances with lost sales and backorders (left and right from the middle) should not be compared directly, however, it is interesting to compare how networks are ranked in terms of costs.

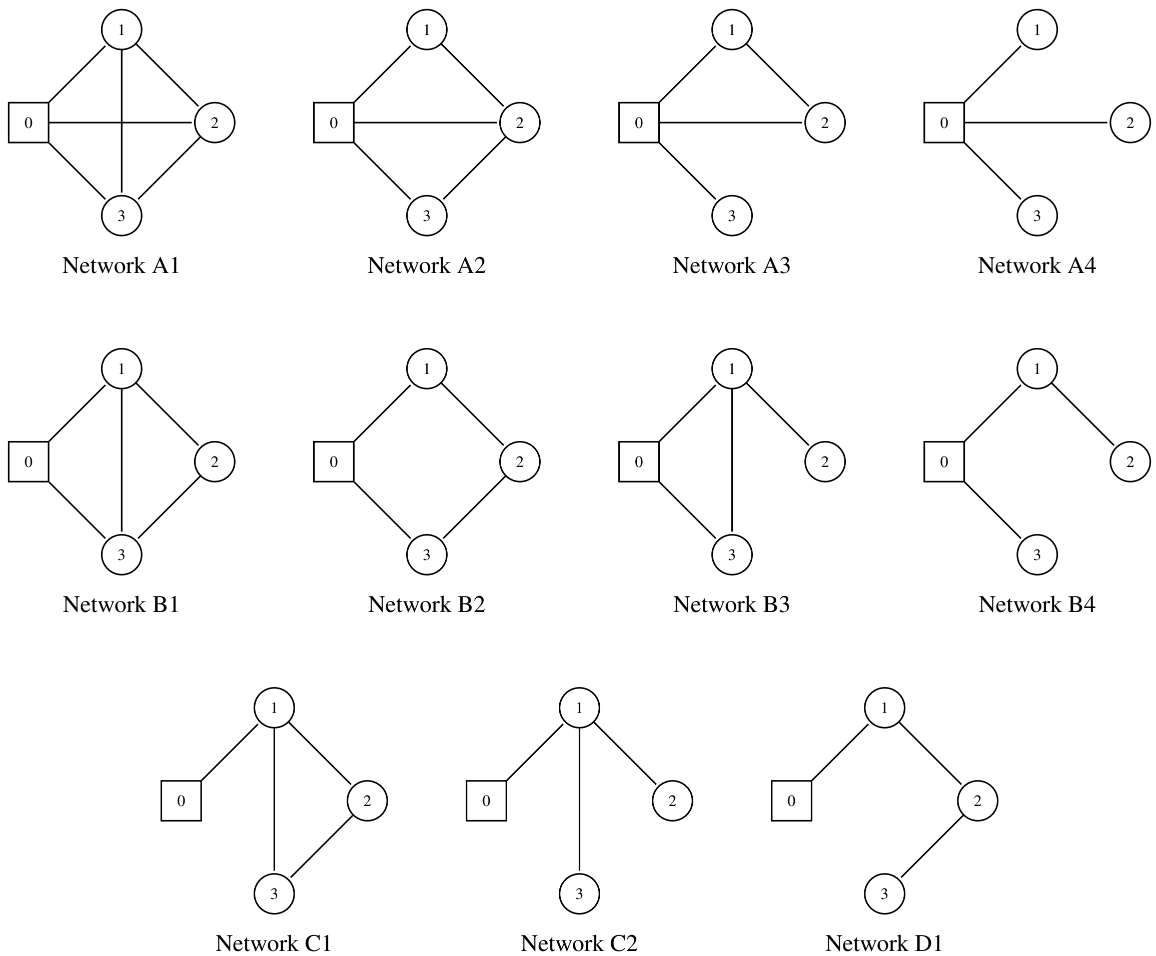


Figure 2: All possible unique undirected network structures with four nodes, assuming node 0 receives orders and nodes 1, 2, and 3 are identical demand nodes.

Figure 3 yields various interesting insights. We focus first on instances with lost sales. From the increasing bars, it is evident that the cost of networks typically follows the lexicographical ordering. Not unexpectedly, the networks with the lowest distances from the main node perform best, while among networks with the same distances those with most edges perform best. Several instances exist where network A4 is more costly than B1, showing that distance is not necessarily the leading factor for performance; this mainly occurs when there is a combination of low shipment costs and high lost sales costs ($c \leq 1$, $\ell = 60$). Within groups of networks with the same distances, the difference in percentage points between the best and worst network is mostly around 1% and at most 2.5%, showing that extra edges have some added value. Instances 9, 17, and 21 show a surprising pattern with multiple (or all) networks having the same performance. This is due to a combination of low lost sales penalties and high shipment costs ($\ell = 30, c > 0$), so that, when the distances from the main node are large enough, losing all demand is preferred to ordering and shipping inventory. For example, in instance 9, inventory is ordered in distance A and B networks, but not in distance C and D networks. A higher variance seems to have no clear effect on the relative order of networks, though the lower bars indicate that differences in costs between networks become smaller.

Under backordering, many parameter settings exist where the costs of networks deviate from the lexicographical ordering that we observed in lost sales instances. In fact, the lexicographical ordering holds only in instances with low demand variance and high shipment costs. In other instances there are many combinations of networks that do not satisfy the ordering, e.g., often network A4 is more costly than B1 and network B4 is more costly than C1. The difference between the best and worst network within groups with the same distances can be as high as 8%. As

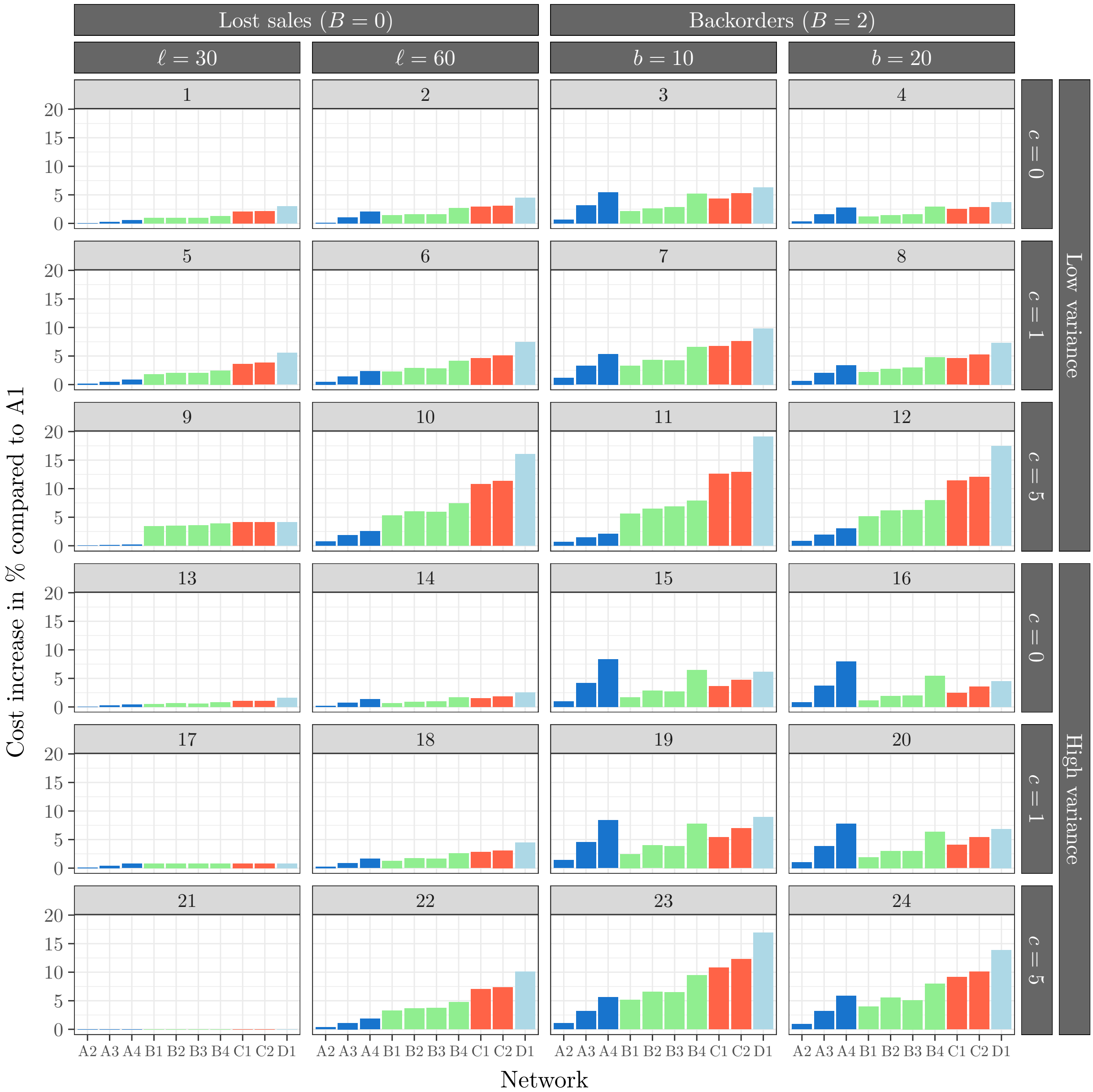


Figure 3: Cost comparison of networks.

backorders incur a cost per time unit, it important to quickly ship items between demand nodes when backorders appear, which is easier in flexible networks with more edges. Generally, we see that the added value of having more flexibility is larger when demand has high variance. Since demand is unpredictable, it is challenging to store the correct amount of inventory at each demand node, increasing the need for shipments between nodes. The height of the backorder cost does not seem to have much effect in this experiment.

Of particular interest are backordering instances with high demand variance and low shipment costs ($c \leq 1$). It is surprising that, despite having the smallest possible distances, network A4 is the most expensive network in such instances. This shows that two-echelon-like networks, commonly employed in practice and studied in the literature, are costly and ineffective in some cases. The parallel network structure prohibits shipments between demand nodes, requiring two shipments in order to meet a backorder at a node with the stock of another node. This can be mitigated by keeping reserve stock at the main node, however, then that stock is not available for immediate demand fulfillment. The cost of this parallel structure can, therefore, be higher than a series structure with the same number of edges (network D1), in which, despite the large distances, it is possible to reach all other demands nodes with one shipment without keeping stock at the main node. A myriad of other networks can be pairwise compared for similar insights.

By comparing costs for networks based on the number of edges, insight can be obtained in the most effective way of setting up network edge by edge. In our experiment, in most cases it is best to choose the following order: A4 → A3 → A2 → A1. However, in case of backordering, low shipment costs, and highly variable demand, the order C2 → B3 → A2 → A1 is better. Differences in product characteristics can thus lead to different effective strategies for setting up networks.

In conclusion, generally we see that it is vital to limit distance to the main node when transportation costs are high. When transportation costs are relatively low, flexibility starts to become important, especially in cases with backordering and highly variable demand. In such cases, traditional network structures may be far from cost-effective. When setting up a new network, product characteristics need to be taken into account.

4.4. *Comparison of Optimal Network Flows*

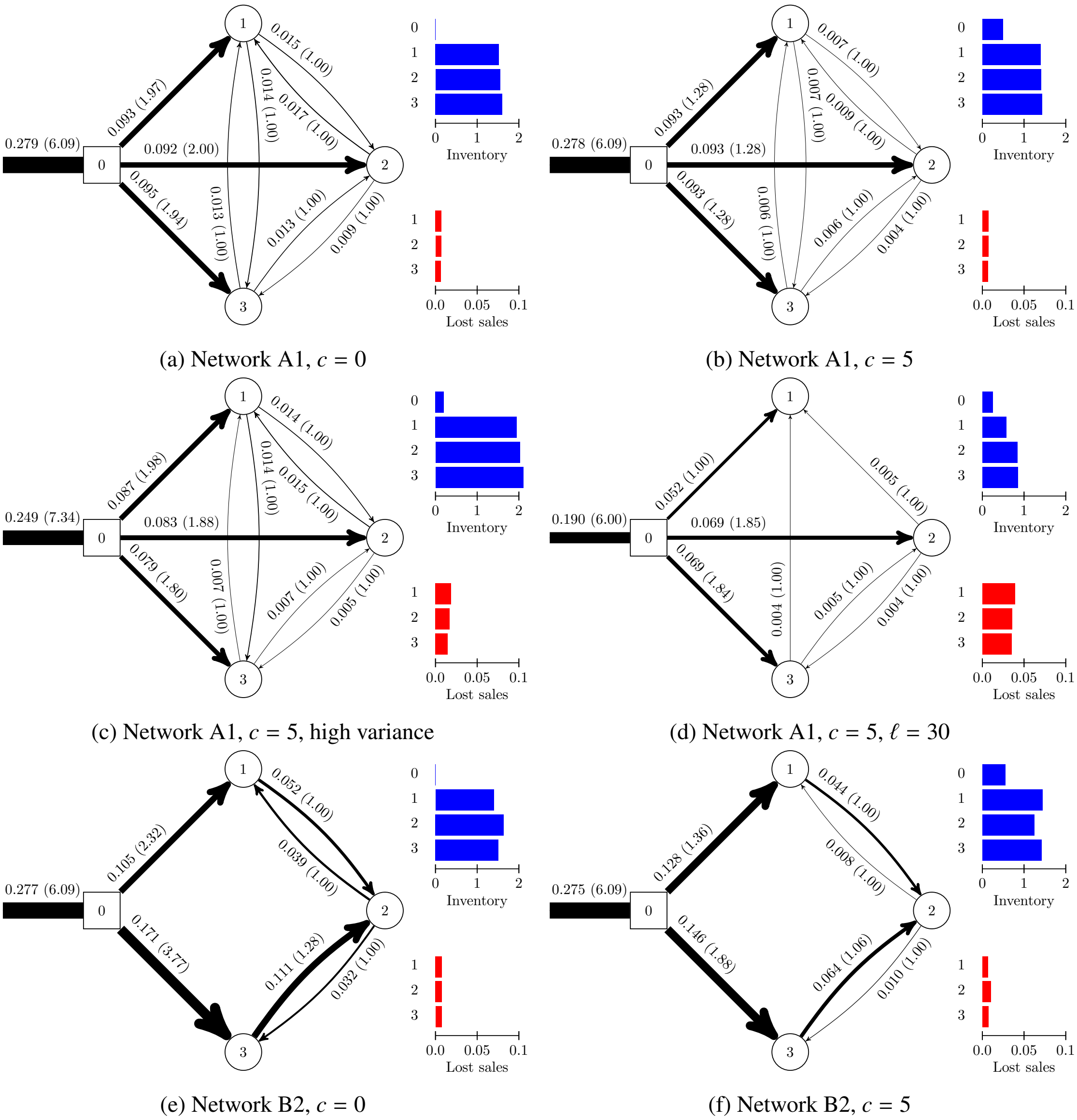
In order to obtain insights into effective decision-making in SWTNs with dynamic shipments, we investigate steady-state optimal decisions for different networks in different instances. We illustrate the optimal decisions with network flow graphs in Figures 4 and 5. Each edge in a flow graph has two values. The first value is called the average flow and shows the average number of items shipped over an edge per period. The second value, between parenthesis, is called the conditional flow and shows the average size of a shipment whenever something is shipped over an edge. As the size of a shipment is an integer, conditional flows are always at least 1. The average and conditional flows have been calculated using the approach of Section 3.2. Edges entering node 0 from outside the network represent incoming orders from the external supplier. To the right of each flow graph, the average end-of-period inventory and backorders/lost sales at each node are shown. Since it is excessive to show all 264 (= 24 × 11) network flow graphs, we made a selection illustrating the most interesting effects. The other graphs are available online as supplementary material.

As before, we focus first on instances with lost sales, shown in Figure 4. We call edges on the shortest paths from the main node to the demand nodes *direct edges* and all other edges *indirect edges*. As indicated by the thicker arrows in Figure 4, flows are always largest on the direct edges. Occasionally indirect edges are used to ship 1 item, mostly when a demand node is out of stock while another has multiple items. In all flow graphs, the average order per time unit is smaller than the total demand per time unit ($3 \times 0.1 = 0.3$), accordingly some demand is lost.

We study the effects of several parameters by making pairwise comparisons. In (a) and (b), shipment costs in network A1 are $c = 0$ and $c = 5$, respectively. The average flows on direct edges are near-identical in both situations, however, flows on indirect edges are almost 100% higher when $c = 0$ than when $c = 5$. Differences in average inventory levels between the nodes are also substantial. Shipping stock to a demand node requires a significant investment when $c = 5$, hence part of the inventory is stored at the main node until more inventory information becomes available at the demand nodes. When $c = 0$, shipping to and between demand nodes is inexpensive, hence in that case almost all stock at the main node is immediately shipped to the demand nodes. Low shipment cost decisions thus aim to directly solve any imbalances in inventories, while high shipment cost decisions are more strategic.

In (c) the variance is higher than in (b). This leads to an increase in lost demand, despite that this is mitigated in part through higher inventory levels. Flows on indirect edges are larger, as well as the conditional order and shipment sizes. Higher variance thus leads to ordering and shipping in larger quantities. On average less stock is kept at the main node, since the lumpy and unpredictable nature of demand requires more stock at the demand nodes in order to avoid lost sales. In (d) the lost sales cost is lower than in (b). Meeting demand is less crucial and as a result the average inventories are lower and several indirect edges remain unused.

In (e) and (f), the shipment costs in network B2 are varied. In both cases, node 2 is supplied both through node 1 and 3, although the flows through those nodes are not necessarily the same. Here, we again see proactive balancing of inventory when shipment costs are low. Flows on indirect edges with $c = 0$ are almost 4 times higher than with $c = 5$. In addition, with $c = 5$ extra stock is kept at the main node, while with $c = 0$ it is kept at node 2 instead. Keeping stock at node 2 provides full flexibility for supplying the other nodes, with the advantage that less demand is lost since all stock is kept at demand nodes.



(a) Network A1, $c = 0$

(b) Network A1, $c = 5$

(c) Network A1, $c = 5$, high variance

(d) Network A1, $c = 5$, $\ell = 30$

(e) Network B2, $c = 0$

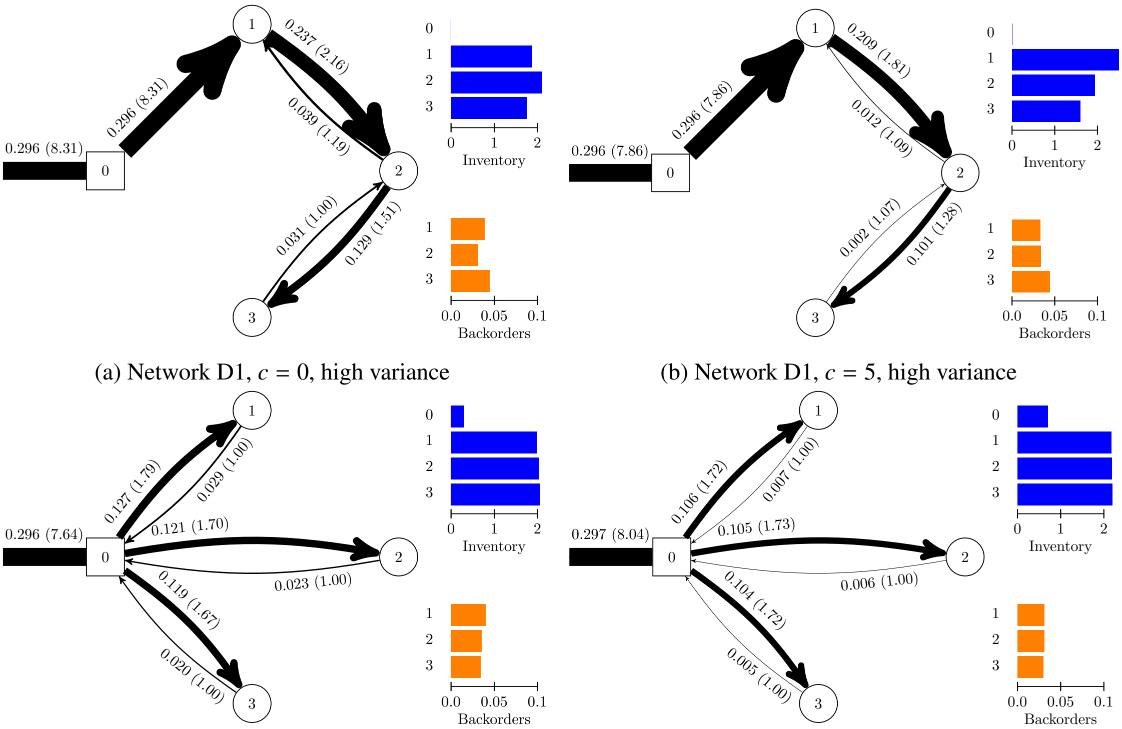
(f) Network B2, $c = 5$

Figure 4: Flow graphs for lost sales instances. If unspecified, variance is low and $\ell = 60$

Comparing (a) and (e), or (b) and (f), we see that the flow on each individual indirect edge in network B1 is higher than in network A1, though the flow on all indirect edges combined is smaller. With fewer indirect edges, fewer products in total are shipped over these edges, but each edge is used more intensively. This is generally the case when a network is a subnetwork of another network.

Flow graphs for backorder instances are shown in Figure 5. The number of orders per time unit in all graphs is nearly 0.3, hence (almost) all demand is met. We emphasize that backorder and lost sales instances have nearly the same insights regarding the effect of shipment costs and variability on where inventory is stored in the network. One effect that we did not yet discuss is visible in (a) and (b). Since in network D1 all stock needs to pass through node 1 to reach the other nodes, it is obvious that no stock should ever be kept at the main node; the same holds for networks C1 and C2. For similar reasons as in the lost sales instances, most stock is stored at node 2 when $c = 0$ and at node 1 when $c = 5$.

Comparing (a) and (c), we can understand why network D1 is less costly than network A4 in situations with



(a) Network D1, $c = 0$, high variance

(b) Network D1, $c = 5$, high variance

(c) Network A4, $c = 0$, high variance

(d) Network A4, $c = 5$, high variance

Figure 5: Flow graphs for backorder instances with $b = 20$.

$c = 0$ and high variance. In order to prevent roughly the same number of backorders, A4 requires significantly higher inventory levels than D1, resulting in higher holding costs. As shipment costs are 0 and the backorder costs are nearly equal for both networks, the total cost for A4 is higher. With $c = 5$ in (b) and (d), A4 is less costly than D1. The holding costs in A4 are still higher and the backorder costs are nearly equal, however, the shipment costs in network A4 are almost 50% lower, resulting in A4 having the lower total cost.

4.5. Sensitivity Analysis

Through sensitivity analysis, we further investigate how the values for various parameters influence the costs and the ordering of different network structures. We vary parameters not yet varied in our previous experiments: the demand rate λ of the Poisson distribution and the order cost K. The demand rate is of interest because it impacts the number of items to be ordered and shipped each time unit. Similarly, the order cost is of interest because it impacts how often orders are placed.

The results of the sensitivity analysis are shown in Figures 6 and 7. The graphs show the relative cost of the 11 network structures compared to network A1 as a function of λ and K. Note that the graphs have different scales. Based on our previous experiments, instances with backorders and lost sales are shown under different shipment costs ($c = 0$ and $c = 5$). The backorder and lost sales costs are kept constant at $b = 10$ and $\ell = 30$. Furthermore, $K = 50$ in Figure 6 and $\lambda = 0.1$ in Figure 7.

We observe in most graphs that networks form groups based on distance. The gaps between networks with different distances typically grow with λ and shrink with K, except in cases with backordering and $c = 0$. As λ increases, distance becomes a more prominent factor than flexibility due to higher flows through the network. The reason why the gaps shrink in K is because the order costs start to dominate the total costs. Near $K = 0$, the slopes are particularly negative for most networks. In most graphs, there are no crossing lines, meaning that the relative order of networks remains constant in λ and K. We conclude that the percentage costs differences are rather sensitive to changes in the values of λ and K, while the ordering of networks is rather insensitive. Finally, when λ is very low or K is very high, we observe the familiar effect with lost sales that losing all demand is preferred to placing orders.

4.6. Gains from Dynamic Shipments

In this experiment, we quantify the gains from carrying out dynamic shipments in SWTNs. In addition, we want to determine in which cases those gains are substantial. To that end, in our set of benchmark instances we compare the costs of dynamic shipments with the costs of the static shipments case from Section 3.3.

Figure 8 shows the percentage cost reduction of dynamic shipments compared to the static shipments case. Graphs are shown only for instances with $c = 0$ or $c = 5$, as the instances with $c = 1$ yield very similar results. In order to make the relation between distances from the main node and the cost reduction visible, we only include networks A1, B1, C1 and D1 in the graphs. These networks in fact have larger cost reductions than other networks with the same distances. For example, A1 has a larger cost reduction than A2, because A1 is less costly than A2 under dynamic shipments and equally costly under static shipments. Hence, the less flexible networks not shown here gain less from using dynamic shipments.

In most instances in Figure 8, there are significant benefits to using dynamic shipments over static shipments. For lost sales instances, the cost reduction varies drastically between instances. It ranges between 4% and 14% in instances with $\ell = 60$, and between 0% and 5% in instances with $\ell = 30$. In cases where the cost reduction is 0%, losing all demand is preferred over ordering inventory. This same effect also explains why the cost reduction is relatively small in lost sales instances with high variance. For backorder instances, the cost reduction is around 18% when shipment costs are low and around 12% when shipment costs are high. The influence of variance seems limited in backorder instances.

Based on the above observations, dynamic shipments seem most useful in backorder instances. As backorder costs are incurred each time unit, it is preferred to fulfill backorders in the next period by carrying out a shipment. Backorders are fulfilled much faster with dynamic shipments than with static shipments, where nodes with backorders have to wait for an arriving order. In networks with long distances, nodes have relatively long lead times between placing and receiving orders. This explains why in Figure 8 the cost reduction in backorder instances mostly increases with the distances from the main node. For lost sales instances, the cost reduction instead decreases with the distances. This seems counter-intuitive, however, it can be understood as follows. When distance increases, the number of lost sales per time unit increase considerably, both under dynamic and static shipments. The decrease in lost sales from using dynamic shipments is not large enough to make up for the increase in lost sales due to distance, hence the overall cost reduction decreases with distance. This same effect does not occur for backordering, as the number of backorders per time unit is relatively constant in the distance.

All in all, dynamic shipments seem to have a large potential for reducing costs. This occurs even though shipments arrive in the next period, so we expect even larger cost reductions when introducing shipments that arrive within the current period.

5. Discussion

Important goals of SWTNs are improving demand fulfillment and transport efficiency. With a high degree of consolidation, it seems reasonable to expect that shipping costs in an SWTN become quite low. Our results indicate that this gives companies incentives to ship more often than they normally would in order to balance their inventories. Inventory balancing evidently complies with the goal of improving demand fulfillment. With regard to environmental goals, however, this can have positive as well as negative effects. On one hand, better inventory control may reduce waste, since fewer products need to be disposed of when stored at their most suitable locations. On the other hand, the required number of vehicles/trips increases, possibly leading, among others, to higher emissions, congestion, and capacity problems. Emerging trends for CO_2 neutral transportation, such as electrical vehicles, may help to partially address these negative effects. If, despite this, the negative effects still have severe impact, then certain edges in the network could be disabled or incentive schemes could be designed to trigger fair use. We believe it is important to further study this trade-off between positive and negative effects to create truly sustainable transportation.

For network design decisions, we see that different combinations of shipment costs, stock-out behavior, and demand variability lead to considerable changes in effectiveness of different network structures. It is intriguing that

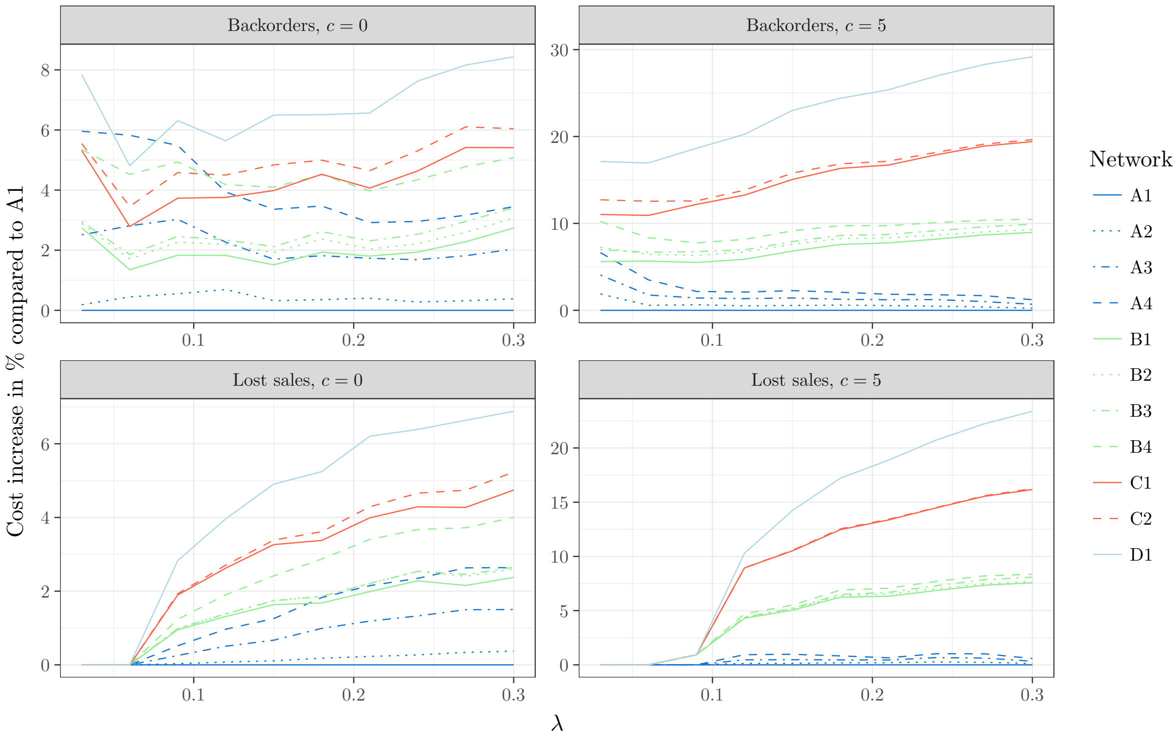


Figure 6: Effect of varying λ

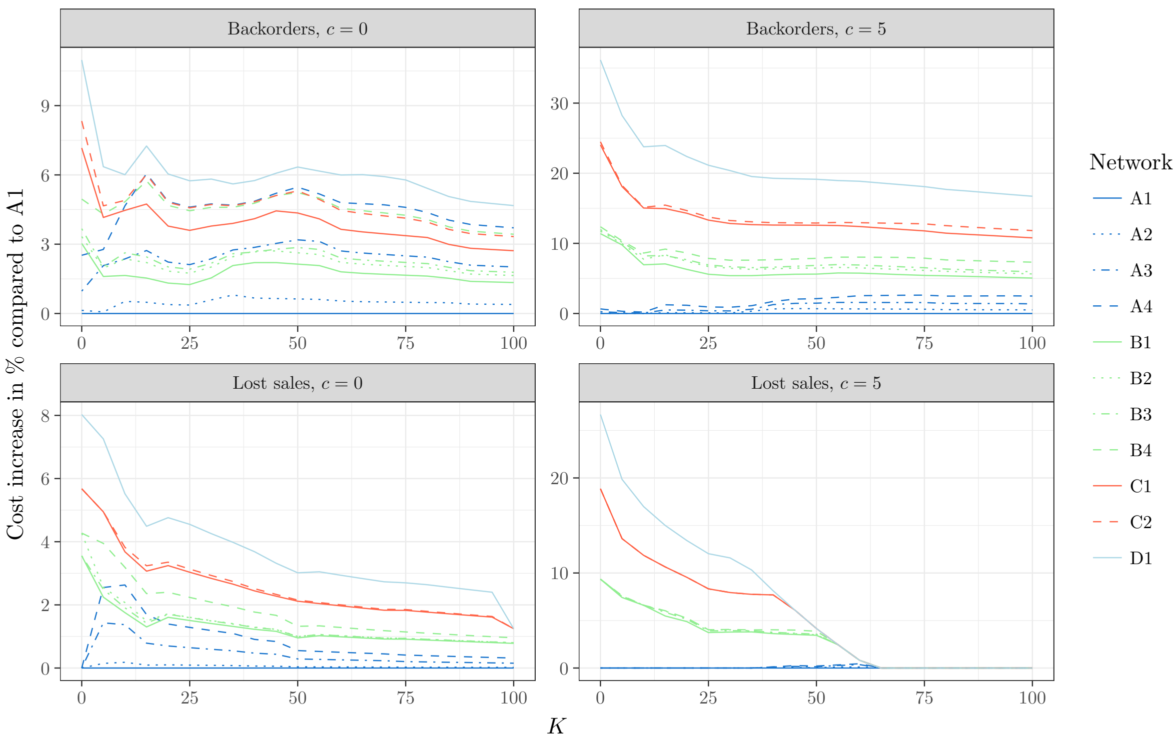


Figure 7: Effect of varying K

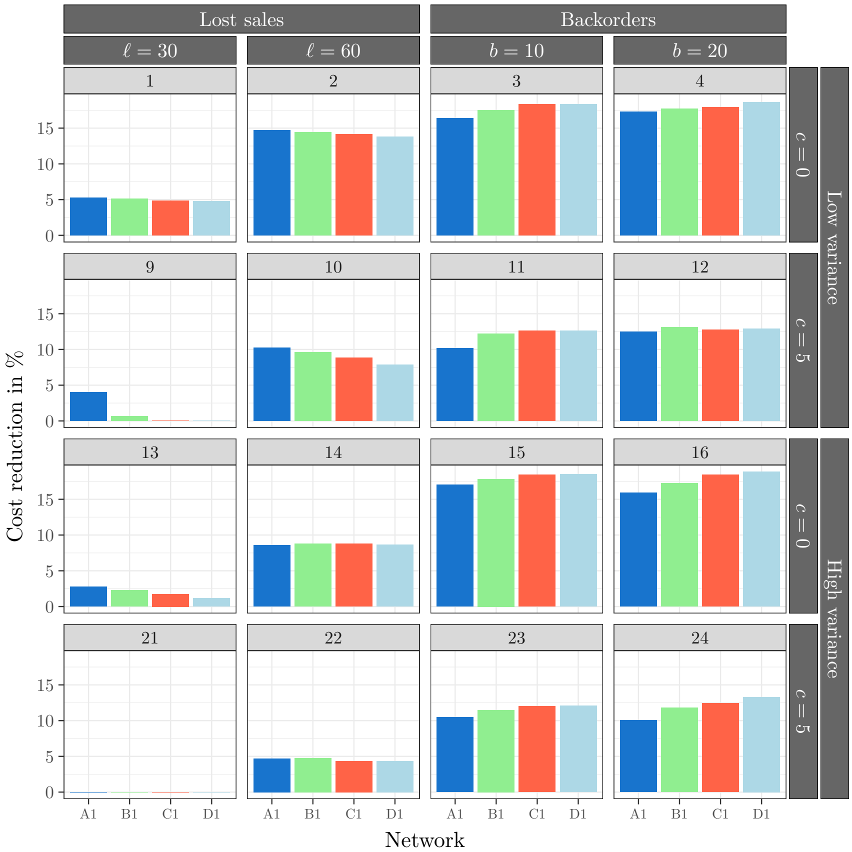


Figure 8: Cost reduction of using dynamic shipments instead of static shipments in networks with different distances from the main node.

perhaps the most intuitive choice for a network structure, i.e., two-echelon, leads to poor performance in a considerable number of cases. When there is freedom in designing a new network or establishing a new connection between warehouses, taking into account the product's characteristics in this choice can lead to cost savings. In many cases the underlying geographic structure of a region determines its network's shape. By comparing logistic costs for different regions, partnering companies can make strategic decisions regarding in which regions they want to setup SWTNs. Ultimately, connecting many such regional networks may result in a global Physical Internet.

We generated various insights regarding distance and flexibility that assist decision-making in real-life SWTNs. For small-sized SWTNS we obtained optimal solutions, but there exist SWTNs in practice that are much larger. For example, FLEXE currently offers warehouse space in over 850 locations throughout America (FLEXE, 2018). Such a real-life network can be modeled straightforwardly in our framework by determining the network structure, estimating all parameters, and selecting a period length that appropriately matches the travel times. When travel times between certain nodes are longer than 1 period, it is possible to add intermediate transfer nodes, i.e., nodes without storage and without demand. However, it is unrealistic to expect that real-life instances of this size will be solved to optimality anywhere in the near future. In order to support real-life decision-making, a fruitful area for future research is developing approximate solution procedures for large instances. In that regard, a particularly interesting approach seems to be approximate dynamic programming, where (machine) learning algorithms and simulation are applied to approximate the value function of a Markov decision process (Powell, 2011). If the value function can be approximated by a piecewise linear function, then the decision in each state can be efficiently optimized using linear programming. Another alternative would be the development of heuristics. We realize it is challenging to create a heuristic performing well for any network structure and any type of stock-out behavior, since for some special cases of the model developing effective heuristics is already challenging by itself. However, the results from the numerical experiments in this paper may provide guidance on what the objective of such heuristics could be.

6. Conclusion

In this article we propose a modeling framework and use it to determine optimal decisions for dynamic shipments in shared warehouse and transportation networks. In numerical experiments, we show how stock-out behavior, demand variability, and other cost parameters influence the costs of and flows through different networks. In most cases, distance is the main driver for costs, however, situations with low shipment costs and highly variable demand favor networks with a high number of connections over those with short distances. Especially under backordering, the ordering of different networks is sensitive to changes in shipment costs and demand variability. The possibility to dynamically ship products based on the latest inventory information leads to significant cost savings in many cases.

Acknowledgement

We thank the four anonymous reviewers for their helpful comments for improving this paper. This research has been funded by NWO TKI Dinalog, project number 438-15-525.

References

M. Ahmadi, A. Seifi, and B. Tootooni. A humanitarian logistics model for disaster relief operation considering network failure and standard relief time: A case study on San Francisco district. *Transportation Research Part E: Logistics and Transportation Review*, 75:145–163, 2015.

D. Alem, A. Clark, and A. Moreno. Stochastic network models for logistics planning in disaster relief. *European Journal of Operational Research*, 255(1):187–206, 2016.

K.D. Cattani, F.R. Jacobs, and J. Schoenfelder. Common inventory modeling assumptions that fall short: Arborescent networks, Poisson demand, and single-echelon approximations. *Journal of Operations Management*, 29(5):488–499, 2011.

J.M. Chen, I.C. Lin, and H.L. Cheng. Channel coordination under consignment and vendor-managed inventory in a distribution system. *Transportation Research Part E: Logistics and Transportation Review*, 46(6):831–843, 2010.

L.C. Coelho, J.F. Cordeau, and G. Laporte. Thirty years of inventory routing. *Transportation Science*, 48(1):1–19, 2013.

T. De Kok, C. Grob, M. Laumanns, S. Minner, J. Rambau, and K. Schade. A typology and literature review on stochastic multi-echelon inventory models. *European Journal of Operational Research*, 2018. In press.

V. Deshpande, M.A. Cohen, and K. Donohue. A threshold inventory rationing policy for service-differentiated demand classes. *Management Science*, 49(6):683–703, 2003.

S.M. Disney, A.T. Potter, and B.M. Gardner. The impact of vendor managed inventory on transport operations. *Transportation Research Part E: Logistics and Transportation Review*, 39(5):363–380, 2003.

Y. Dong and K. Xu. A supply chain model of vendor managed inventory. *Transportation Research Part E: Logistics and Transportation Review*, 38(2):75–95, 2002.

EC. Horizon 2020 working programme 2016-2017 11. smart, green and integrated transport. Available at: `http://ec.europa.eu/research/participants/data/ref/h2020/wp/2016_2017/main/h2020-wp1617-transport_en.pdf`, 2017. Accessed 2018-02-19.

P. Escalona, F. Ordóñez, and V. Marianov. Joint location-inventory problem with differentiated service levels using critical level policy. *Transportation Research Part E: Logistics and Transportation Review*, 83:141–157, 2015.

ETP-Alice. Roadmaps. Available at: `http://www.etp-logistics.eu/`, 2017. Accessed 2018-02-19.

FLEXE. About FLEXE. Retrieved from `https://www.flexe.com/company`, 2018. Accessed 2018-04-30.

J.A. Gallian. A dynamic survey of graph labeling. *The Electronic Journal of Combinatorics*, 16(6):1–219, 2009.

K. Govindan. Vendor-managed inventory: a review based on dimensions. *International Journal of Production Research*, 51(13):3808–3835, 2013.

Ü. Gürler, O. Alp, and N.Ç. Büyükkaramikli. Coordinated inventory replenishment and outsourced transportation operations. *Transportation Research Part E: Logistics and Transportation Review*, 70:400–415, 2014.

A.Y. Ha. Inventory rationing in a make-to-stock production system with several demand classes and lost sales. *Management Science*, 43(8): 1093–1103, 1997.

U.S. Karmarkar. The multiperiod multilocation inventory problem. *Operations Research*, 29(2):215–228, 1981.

A. Madadi, M.E. Kurz, and J. Ashayeri. Multi-level inventory management decisions with transportation cost consideration. *Transportation Research Part E: Logistics and Transportation Review*, 46(5):719–734, 2010.

G. Marquès, C. Thierry, J. Lamothe, and D. Gourc. A review of vendor managed inventory (VMI): from concept to processes. *Production Planning & Control*, 21(6):547–561, 2010.

B. Montreuil. Toward a physical internet: meeting the global logistics sustainability grand challenge. *Logistics Research*, 3(2-3):71–87, 2011.

J.A. Muckstadt and L.J. Thomas. Are multi-echelon inventory methods worth implementing in systems with low-demand-rate items? *Management Science*, 26(5):483–494, 1980.

J.R. Norris. *Markov Chains*. Cambridge University Press, Cambridge, 1998.

L. Özdamar, E. Ekinci, and B. Küçükyazici. Emergency logistics planning in natural disasters. *Annals of Operations Research*, 129(1):217–245, 2004.

D. Özdemir, E. Yücesan, and Y.T. Herer. Multi-location transshipment problem with capacitated transportation. *European Journal of Operational Research*, 175(1):602–621, 2006.

S. Pan, M. Nigrelli, E. Ballot, R. Sarraj, and Y. Yang. Perspectives of inventory control models in the Physical Internet: A simulation study. *Computers & Industrial Engineering*, 84:122–132, 2015.

S. Pan, E. Ballot, G.Q. Huang, and B. Montreuil. Physical internet and interconnected logistics services: research and applications. *International Journal of Production Research*, 55(9):2603–2609, 2017.

V. Pillac, M. Gendreau, C. Guéret, and A.L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–11, 2013.

W.B. Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, volume 842. John Wiley & Sons, 2011.

M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New York, 2009.

S.J. Rennemo, K.F. Rø, L.M. Hvattum, and G. Tirado. A three-stage stochastic facility routing model for disaster response planning. *Transportation Research Part E: Logistics and Transportation Review*, 62:116–135, 2014.

U. Ritzinger, J. Puchinger, and R.F. Hartl. A survey on dynamic and stochastic vehicle routing problems. *International Journal of Production Research*, 54(1):215–231, 2016.

R. Sarraj, E. Ballot, S. Pan, D. Hakimi, and B. Montreuil. Interconnected logistic networks and protocols: simulation-based efficiency assessment. *International Journal of Production Research*, 52(11):3185–3208, 2014a.

R. Sarraj, E. Ballot, S. Pan, and B. Montreuil. Analogies between internet network and logistics service networks: challenges involved in the interconnection. *Journal of Intelligent Manufacturing*, pages 1–13, 2014b.

H. Sternberg and A. Norrman. The physical internet–review, analysis and future research agenda. *International Journal of Physical Distribution & Logistics Management*, 47(8):736–762, 2017.

H. Treiblmaier, K. Mirkovski, and P.B. Lowry. Conceptualizing the physical internet: Literature review, implications and directions for future research, 2016. Working paper. Available at: `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2861409`.

Y. Yang, S. Pan, and E. Ballot. Innovative vendor-managed inventory strategy exploiting interconnected logistics services in the physical internet. *International Journal of Production Research*, 55(9):2685–2702, 2017a.

Y. Yang, S. Pan, and E. Ballot. Mitigating supply chain disruptions through interconnected logistics services in the physical internet. *International Journal of Production Research*, 55(14):3970–3983, 2017b.

Y. Yao and M. Dresner. The inventory value of information sharing, continuous replenishment, and vendor-managed inventory. *Transportation Research Part E: Logistics and Transportation Review*, 44(3):361–378, 2008.

Q. Zhao, S. Chen, S.C.H. Leung, and K.K. Lai. Integration of inventory and transportation decisions in a logistics system. *Transportation Research Part E: Logistics and Transportation Review*, 46(6):913–925, 2010.