MIT Global Supply Chain and Logistics Excellence Network

WORKING PAPER

TITLE:

A Discrete Simulation-Based Optimization Algorithm for the Design of Highly Responsive Last-Mile Distribution Networks

AUTHORS AND AFFILIATIONS:

André Snoeck
Matthias Winkenbach
Massachusetts Institute of Technology

DATE:

May 28, 2020

NUMBER:

2020-mitscale-ctl-04

MIT Global Supply Chain and Logistics Excellence Network
Working Paper Series

TITLE:

A Discrete Simulation-Based Optimization Algorithm for the Design of Highly Responsive Last-Mile Distribution Networks

AUTHORS AND AFFILIATIONS:

André Snoeck
Matthias Winkenbach
Massachusetts Institute of Technology

ABSTRACT:

Online and omnichannel retailers are proposing increasingly tight delivery deadlines, moving closer towards instant on-demand delivery. To operate last-mile distribution systems with such tight delivery deadlines efficiently, defining the right strategic distribution network design is of paramount importance. However, this problem exceeds the complexity of the strategic design of traditional last-mile distribution networks for two main reasons: (1) the reduced time available for order handling and delivery, and (2) the absence of a delivery cut-off time that clearly separates order collection and delivery periods. This renders state-of-the-art last-mile distribution network design models inappropriate, as they assume periodic order fulfillment based on a delivery cut-off.

In this study, we propose a metamodel simulation-based optimization (SO) approach to strategically design last-mile distribution networks with tight delivery deadlines. Our methodology integrates an in-depth simulator with traditional optimization techniques by extending a traditional black-box SO algorithm with an analytical model that captures the underlying structure of the decision problem. Based on a numerical study inspired by the efforts of a global fashion company to introduce on-demand distribution with tight delivery deadlines in Manhattan, we show that our approach outperforms contemporary SO approaches as well as deterministic and stochastic programming methods. In particular, our method systematically yields network designs with superior expected cost performance. Furthermore, it converges to good solutions with a lower computational budget and is more consistent in finding high-quality solutions. We show how congestion effects in the processing of orders at facilities negatively impact the network performance through late delivery of orders and reduced potential for consolidation. In addition, we show that the sensitivity of the optimal network design to congestion effects in order processing at the facilities increases as delivery deadlines become increasingly tight.

KEYWORDS:

last-mile distribution; simulation-based optimization; network design

CORRESPONDING AUTHOR:

André Snoeck – asnoeck@mit.edu

A Discrete Simulation-Based Optimization Algorithm for the Design of Highly Responsive Last-mile Distribution Networks

André Snoeck[a,*], Matthias Winkenbach[a]

[a] *Massachusetts Institute of Technology, Cambridge, Massachusetts 02139,*

Abstract

Online and omnichannel retailers are proposing increasingly tight delivery deadlines, moving closer towards instant on-demand delivery. To operate last-mile distribution systems with such tight delivery deadlines efficiently, defining the right strategic distribution network design is of paramount importance. However, this problem exceeds the complexity of the strategic design of traditional last-mile distribution networks for two main reasons: (1) the reduced time available for order handling and delivery, and (2) the absence of a delivery cut-off time that clearly separates order collection and delivery periods. This renders state-of-the-art last-mile distribution network design models inappropriate, as they assume periodic order fulfillment based on a delivery cut-off.

In this study, we propose a metamodel simulation-based optimization (SO) approach to strategically design last-mile distribution networks with tight delivery deadlines. Our methodology integrates an in-depth simulator with traditional optimization techniques by extending a traditional black-box SO algorithm with an analytical model that captures the underlying structure of the decision problem. Based on a numerical study inspired by the efforts of a global fashion company to introduce on-demand distribution with tight delivery deadlines in Manhattan, we show that our approach outperforms contemporary SO approaches as well as deterministic and stochastic programming methods. In particular, our method systematically yields network designs with superior expected cost performance. Furthermore, it converges to good solutions with a lower computational budget and is more consistent in finding high-quality solutions. We show how congestion effects in the processing of orders at facilities negatively impact the network performance through late delivery of orders and reduced potential for consolidation. In addition, we show that the sensitivity of the optimal network design to congestion effects in order processing at the facilities increases as delivery deadlines become increasingly tight.

Keywords: last-mile distribution, simulation-based optimization, network design

1. Introduction

The rise of the on-demand economy and on-demand consumerism constitutes a paradigm shift in customer service (Colby and Bell 2016). E-commerce and logistics services are no exception to the "I-want-it-now" instant-gratification mindset. 78% of logistics companies expect to provide same-day delivery by 2023, while 39% even anticipate delivery within a two-hour window by 2028 (Zebra Technologies 2018). Amazon Prime Now, JD Express, and Instacart Express are examples of e-commerce companies promising one hour delivery already today. Even traditional retail brands begin to differentiate themselves with highly responsive delivery services. For example,

*Corresponding author, asnoeck@mit.edu

Mediamarkt offers two-hour delivery in Spain, and Gucci offers 90-minute delivery in various cities globally (Farfetch 2017, MediaMarkt 2020). These near-instant delivery services pose a major challenge for online and offline retailers. Worldwide retail e-commerce sales will increase to $4.48 trillion by the end of 2021, up from $2.29 trillion in 2017 (eMarketer 2017). Rising customer expectations regarding lead time, time windows, and late customization of shipments lead to a larger variety of delivery requirements and greater uncertainty in the timing of customer demands. These trends force companies to be more responsive in their distribution operations. Furthermore, tight promised delivery deadlines limit the available time for handling and transportation. The increasing responsiveness to comply with tight delivery deadlines puts additional pressure on available capacity and cause the network architecture and performance to be increasingly sensitive to facility processing congestion and associated picking queues. Notwithstanding this increase in complexity, providing a high-quality delivery service remains desirable for retailers, as 90% of consumers state that the delivery service affects the brand perception of the seller (Zebra Technologies 2018).

In this paper, we study the strategic design of last-mile distribution networks with continuous response (CR) and tight delivery deadlines. We focus on networks that enable delivery promises within a few hours after order placement. Such services are increasingly relevant to online and omnichannel retailers, as they help closing the gap of instant gratification between online and brick-and-mortar shopping (Ulmer and Thomas 2018). The increasing prevalence of such delivery services leads to increasing attention in the recent literature (Savelsbergh and Van Woensel 2016). However, to the best of our knowledge, the existing literature is limited to solving the operational delivery problem given a fixed network design (see, e.g., Klapp et al. 2018a, Voccia et al. 2019). Meanwhile, literature that focuses on the strategic design of traditional urban distribution networks has shown the importance and value of choosing the right strategic network configuration (Crainic et al. 2004, Winkenbach et al. 2016a, Snoeck and Winkenbach 2020). However, the complexity of the strategic design problem for urban distribution networks with continuous response and tight delivery deadlines exceeds that of traditional urban distribution networks due to two main reasons: (1) the absence of a delivery cut-off time that clearly separates the order collection period and the delivery period, and (2) the reduced time available for order handling and delivery.

Traditional urban distribution networks with loose delivery deadlines are typically operated as periodic order fulfillment systems, i.e., there exists a segregation of the order collection period and the order delivery period for a specific set of orders by means of a cut-off time. After the cut-off time, the company constructs an operational plan to deliver the accrued orders. This problem

is referred to as the day-before planning problem (Crainic et al. 2009). Existing models that address the strategic design of last-mile distribution networks are based on the day-before planning assumption. This simplifies the network design problem in two ways. First, as soon as the delivery period starts, demand is assumed to be known and the operational problem is deterministic. Second, the order cut-off renders different delivery periods mutually independent, i.e., it eliminates time dependency in the network design problem.

On the contrary, the order collection and delivery periods are intertwined for distribution systems with tight delivery deadlines. Distribution systems with CR and tight deadlines are characterized by dynamically arriving delivery requests throughout the service period (Voccia et al. 2019). Each delivery request needs to be served within a promised time-frame, thus the time of occurrence of the request defines the delivery deadline. Furthermore, delivery requests arrive stochastically. The stochastic nature of arriving orders, combined with the tight delivery deadlines, gives rise to an inherent trade-off in the vehicle dispatching decisions. Delaying vehicle dispatch, i.e., waiting, enables more consolidated and cost-effective delivery routes, while it increases the risk of late delivery. The nature of distribution systems with CR and tight deadlines makes the operational route planning an inherently stochastic and dynamic problem, as opposed to the time-independent deterministic operational problem of traditional last-mile distribution networks. Consequently, we cannot identify independent deterministic time periods to simplify the network design. This increases the complexity of the strategic last-mile distribution network design problem and consequently renders it intractable with existing methods discussed in the literature.

In addition, in distribution networks with tight deadlines, the order processing and delivery time is large relative to the time until the delivery deadline. Consequently, delays in order processing have a large impact on the ability to deliver orders on time and the network performance is susceptible to congestion effects in order processing, e.g., due to capacity bottlenecks during order picking at distribution facilities. Furthermore, the emergence of order processing queues is exacerbated by the stochastic nature of order arrivals. Consequently, it is important to incorporate order processing congestion in the network design methodology. Strategic distribution network design approaches for traditional networks do not address this challenge since the absence of tight delivery deadlines reduces the impact of stochastic fluctuations in the order arrival process. In practice, there are often periods with a lower rate of incoming orders, e.g., at night, and the available processing capacity in these periods could be used to eliminate existing processing queues. Since this is not an option in distribution networks with tight delivery deadlines, we need to incorporate these queuing effects

explicitly, limiting the applicability of contemporary distribution network design approaches.

We propose a SO method to support the strategic design of CR last-mile distribution networks with tight delivery deadlines to incorporate the additional complexity that stems from the dynamic and stochastic arrival of orders and the susceptibility of the network to facility congestion. State-of-the-art simulation models are able to capture disaggregate agent behavior, interactions with the distribution network, and demand patterns (Osorio and Bierlaire 2013). Simulators provide detailed performance indicators of the network, including cost, service level, and utilization. Furthermore, we can implement a detailed operational order allocation and delivery vehicle routing logic and capture the non-linear queuing effects in order processing, allowing us to acquire good approximations of the (disaggregate) performance of a given network. Therefore, simulators are often used in the context of what-if or sensitivity analyses (see, e.g., Bektaş et al. 2017, Govindan et al. 2017, for examples in the context of urban distribution), or to evaluate a set of predetermined network designs. In theory, access to an in-depth simulator allows us to evaluate every potential network design. However, in real life, the set of feasible strategic decisions is often too large for a total enumeration approach, giving rise to the need for an alternative method to determine a near-optimal strategic network design. SO is an umbrella term that refers to the techniques used to optimize stochastic simulations, i.e., to search for the specific settings of the input parameters of the simulation that optimize the objective (Amaran et al. 2016). However, due to the complexity of instant delivery operations, these simulation models are computationally expensive to evaluate. Therefore, using simulators to derive optimal designs is an intricate task (Osorio and Bierlaire 2013). In this work, we build on the discrete SO metamodel approach introduced by Zhou et al. (2019) and on earlier work by Osorio and Bierlaire (2013) for continuous problems. In line with Zhou et al. (2019), we extend the so-called adaptive hyperbox algorithm (AHA) proposed by Xu et al. (2013) by introducing a metamodel, i.e., an analytical approximation of the objective function. This allows us to incorporate our knowledge of the underlying last-mile delivery system in the SO algorithm, leading to a significant reduction in the number of iterations required to arrive at satisfactory solutions.

The contribution of this research is threefold. First, we propose a methodology to design the strategic network for highly responsive urban distribution systems that explicitly incorporates congestion effects in order processing at distribution facilities. We deploy a SO based metamodel approach, which relies on (1) an analytical mixed-integer linear program to model the strategic design problem; (2) an in-depth simulator of an operational distribution network with CR and

tight delivery deadlines; and (3) a discrete SO algorithm that exploits the structure of the analytical model and approximates the congestion effects in order processing at facilities. Our approach leads to improved performance in terms of solution quality, consistency, and speed compared to traditional SO algorithms. Second, we numerically show how order-processing queues affect the performance of the network due to late deliveries and reduced opportunities for order consolidation. Third, we analyze the effect of the promised lead-time on the resulting network design and performance based on a real-world study with data from a global fashion retailer in Manhattan.

2. Literature Review

In this section, we review the relevant literature on last-mile distribution with tight delivery deadlines to contextualize our work. Furthermore, we discuss existing literature on urban distribution network design, stochastic location problems with facility congestion, and SO to motivate and position our methodology. We conclude by discussing relevant gaps in the available literature.

2.1. Categorization of Last-Mile Distribution Networks

The growing need for tight delivery deadlines in last-mile distribution is a recent phenomenon, both in industry and literature (Savelsbergh and Van Woensel 2016, Lim and Winkenbach 2019). Operationally, we can categorize the last-mile distribution problem along the two dimensions presented in Figure 1. Along the first dimension, responsiveness, we identify two problem variants. In problems with periodic response (PR) customers choose from a pre-defined set of available delivery deadlines (see, e.g. Klapp et al. 2018a,b). Here, all incoming orders have to be delivered at the end of a fixed-duration operating period, independent of their time of occurrence. In problems with CR, orders have to be delivered within a limited fixed time after the placement of the individual order, irrespective of the time of occurrence (see, e.g. Voccia et al. 2019, Ulmer and Thomas 2018). In both PR and CR problems, orders arrive stochastically and dynamically. However, the type of deadline impacts the nature and complexity of dispatching decisions.

The second dimension captures the tightness of the delivery deadline, i.e., the length of the required order processing and delivery time, relative to the available time between the delivery deadline and the time of order placement. In problems with tight delivery deadlines, the time required for processing and delivery of the order is large compared to the time available until the delivery deadline, adding a sense of urgency to order processing. This reduces the potential for order consolidation in order processing at the facility as well as during delivery. In addition, it

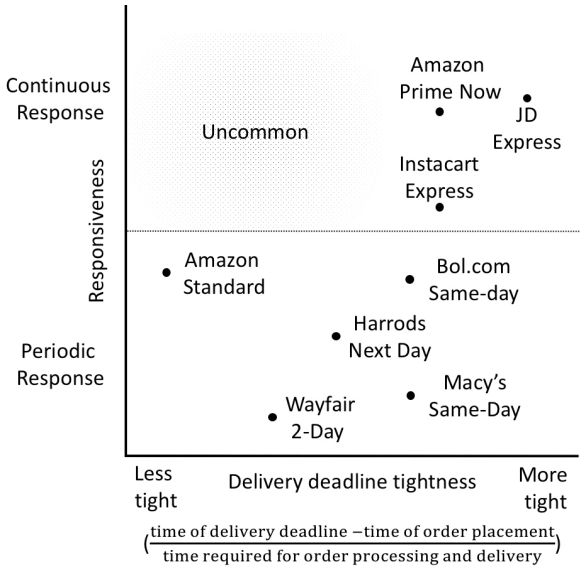


Figure 1: Categorization of last-mile distribution networks with examples.

limits the possibility of load balancing of capacity over time. In practice, tight delivery deadlines lead to overlapping order acceptance and delivery periods, rendering the problem highly dynamic. On the contrary, in problems with loose delivery deadlines, order processing is less urgent.

Note that the delivery deadline tightness is defined by a relative value and should not be confused with absolute shortness of time until the deadline. Delivery deadlines set multiple days after order placement might be considered loose if the ordered goods are readily available for shipment in close proximity to demand. However, if additional processing of the shipment is required, or if the goods need to be shipped from another city, the time required for processing and delivery of the order is large compared to the available time until the deadline rendering the delivery deadline tight.

In recent years, we observe an increasing interest in problems with tight delivery deadlines. The majority of the literature focuses on problems with PR. We refer the reader to van Heeswijk et al. (2019), Klapp et al. (2018b,a), Ulmer et al. (2019) for recent examples of research into the associated operational problem, and to Stroh et al. (2019) for an example that focuses on the associated tactical network design problem.

In this paper, we study the strategic network design of last-mile distribution networks with CR and tight delivery deadlines. Recently, this problem variant has become more prevalent in last-mile distribution, see, e.g., Amazon Prime Now (Amazon 2019). Despite recent advances in addressing the operational challenges of delivery problems with CR and tight delivery deadlines (see, e.g., Ulmer and Thomas 2018, Voccia et al. 2019), limited research exists that specifically focuses on the

strategic design of the associated distribution networks. Both papers assume a fixed set of depots and a fixed, typically homogeneous, vehicle fleet. Ulmer (2017) conducts a simulation study to explore the effect of the tightness of delivery deadlines on the cost and layout of the distribution network. However, he does not explicitly optimize the distribution network design. Therefore, the literature provides limited insights into the effect of tight delivery deadlines and the dynamic nature of incoming orders on the strategic design of last-mile distribution networks.

Distribution networks with CR and loose deadlines are uncommon in practice. To the best of our knowledge, no last-mile distribution operation exists that conducts CR for loose deadlines. However, networks with PR and loose deadlines are the de facto standard in last-mile distribution. They typically feature a delivery cut-off time and facility congestion effects are typically negligible. We review the extensive literature on the strategic design of such networks in Section 2.2.

2.2. Urban Distribution Network Design

Following Bektaş et al. (2017), urban distribution network design involves three levels of decisions. The long-term strategic network design includes decisions on flow, facilities, layout, and transportation components of the network. The medium-term tactical decisions entail the size and composition of the vehicle fleet at each facility, and the short-term operational decisions focus on vehicle routing. The strong interrelatedness of strategic location and operational routing decisions renders the independent solution of vehicle routing and facility location problems inappropriate for designing realistic urban logistics networks (Salhi and Rand 1989). Therefore, strategic network design decisions need to be informed by integrated location routing problems (LRPs) that optimize facility location and vehicle routing jointly and simultaneously. For comprehensive surveys of the existing LRP literature, we refer to Prodhon and Prins (2014), and Schneider and Drexl (2017). For applications in the context of (stochastic) urban (multi-echelon) distribution network design, we refer to Crainic et al. (2004), Boccia et al. (2011), Winkenbach et al. (2016a,b), Janjevic et al. (2019), Snoeck and Winkenbach (2020) and the references therein.

Despite recent advances, applying LRPs with explicit routing decisions to real-world urban distribution problem instances in an urban context, which often includes more than 100,000 customers, remains computationally infeasible (Merchán et al. 2020). For example, Schneider and Löffler (2019) are only able to solve capacitated location-routing problem (CLRP) instances with up to 600 customers and 30 depots with average run times below 4.5 hours. However, operational level routing decisions play a subordinate role in such large-scale LRPs, as their focus lies on ob-

taining optimal strategic design decisions. Winkenbach et al. (2016a) address this challenge by discretizing the city into a large number of adjacent rectangular pixels based on the raster data model (Singleton et al. 2018) and approximate the routing cost using an augmented route cost estimation (ARCE). Aggregating demand within each pixel simplifies the problem by reducing the number of potential demand points, while still capturing the geographic, infrastructure, and demand heterogeneity within the city. We leverage the discretization of the city in pixels, to simplify the analytical component of our SO approach.

A key difference between distribution networks with loose and tight delivery deadlines is the effect of picking queues at distribution facilities. Berman and Krass (2015) survey the literature on stochastic location models with facility congestion and immobile servers. Applications of these models can be found in the design of, for example, public service facility networks such as hospitals (see, e.g. Aboolian et al. 2016) and retail store networks (see, e.g. Schön and Saini 2018). Three key similarities exist between this stream of research and the design of last-mile distribution networks with CR and tight delivery deadlines: i) customers generate a stochastic stream of demand, ii) facilities contain a capacitated set of servers, and iii) due to stochasticity, facility congestion, i.e., processing queues at facilities, could occur, causing a deterioration in service.

However, last-mile distribution networks with CR and tight delivery deadlines differ from the class of problems studied by Berman and Krass (2015) in three ways. First, in stochastic location models with facility congestion demand is assumed to occur directly at facility locations (Berman and Krass 2015). Typically, demand is either allocated to facilities by the decision-maker (see, e.g. Vidyarthi and Jayaswal 2014), or clients choose the facility that maximizes their utility (see, e.g. Dan and Marcotte 2019). The latter assumption is typically made when studying supply chain network design with facility congestion (see, e.g. Vidyarthi et al. 2009). However, in last-mile distribution networks, demand occurs at spatially dispersed individual consumer locations, and the company is responsible for the distribution of goods to the consumer. This adds additional complexities such as the potential to delay allocation and the incorporation of vehicle fleet composition and vehicle allocation decisions. Second, the effect of demand on the wait time of clients at facilities is typically captured through standard queuing formulas that assume long-run stationary behavior of the service system (Schön and Saini 2018). These approaches are unable to capture the time-heterogeneity of demand often observed in last-mile operations. Third, most stochastic location models consider a cost associated with the expected waiting time in the system (see, e.g. Aboolian et al. 2016, Berman and Krass 2015, Schön and Saini 2018). Alternatively, some authors

propose a constraint to limit the probability that the waiting time (or queue length, or number of customers lost) exceeds a specific threshold (see, e.g. Boffey et al. 2010, Jayaswal and Vidyarthi 2017). However, in last-mile distribution networks, determining if an order is late also depends on the delivery time, i.e., the travel time from the facility to the customer, rather than solely on the waiting time for order processing at a facility. This further complicates the determination of a late delivery threshold.

2.3. Simulation-based Optimization

Linear programming is a fundamental building block of supply chain and logistics decision making (Powell 2014). However, this approach is limited to deterministic problems for which an algebraic model is available (Amaran et al. 2016). The inherent stochasticity of dynamic distribution networks with tight delivery deadlines limits the applicability of linear programming. A large body of literature exists that addresses the supply chain network design problem under stochasticity using stochastic programming (see, e.g. Santoso et al. 2005, Schütz et al. 2009, Snoeck et al. 2019) or robust optimization (see, e.g. Pishvaee et al. 2011, Maggioni et al. 2017). However, the dynamic nature and complex interdependencies in distribution networks with tight delivery deadlines make the problem sufficiently complex to render state-of-the-art stochastic programming and robust optimization methods intractable.

In the context of urban logistics and supply chain network design, simulation models have mostly been used to evaluate the performance of network designs obtained from analytical models (Bektaş et al. 2017). They have the advantage that complex, nonlinear, nonconvex objective functions can be evaluated. However, simulation models by themselves do not optimize the network design. Therefore, they need to be incorporated in SO approaches to search for the inputs that optimize the objective (Amaran et al. 2016). While such approaches are uncommon in supply chain design, a large body of literature explores SO. We refer to Andradóttir (1998) and Fu et al. (2005) for reviews on methodological advancements in SO and its applications.

The majority of SO research focuses on problems with continuous decision variables. However, the nature of urban distribution network design renders the majority of decision variables discrete, e.g., the choice of facility locations, fleet size, and inventory levels. The reviews of Nelson (2014) and Hong et al. (2015) focus on discrete SO algorithms. Examples of discrete algorithms include Convergent Optimization via Most-Promosing-Area Stochastic Search (COMPASS) and the adaptive hyperbox algorithm (AHA), both guaranteeing local convergence (Hong and Nelson 2006, Xu

et al. 2013). By focusing on finding a local optimum, locally convergent algorithms can efficiently search the solution space and deliver good finite-time performance because they only need to explore a small fraction of the feasible solution space (Hong and Nelson 2006, Xu et al. 2013). Xu et al. (2010) propose a framework integrating COMPASS into a global search algorithm. The global search phase explores the solution space to identify promising areas for intensive local search, which in turn are being explored using COMPASS. Xu et al. (2013) develop a similar algorithm based on AHA.

However, there continues to exist a significant gap between research and practice in terms of algorithmic approaches (Fu et al. 2000, Tekin and Sabuncuoglu 2004, Hong and Nelson 2009). The majority of the extant research focuses on statistical guarantees and asymptotic convergence properties, leading to a narrow focus on long-term performance on test problems of limited size, while practitioners aim for good results within reasonable computational time. Most discrete SO approaches make no assumptions about any algebraic description of the model, but solely depend on input-output data to optimize the objective function (Amaran et al. 2016). Such black-box algorithms do not attempt to exploit the structure of the underlying decision problem (Bierlaire 2015). However, in supply chain research and practice, we often have an understanding of the system and the structure of the associated decision problem we are modeling, which allows us to exploit this knowledge. Furthermore, real-world urban distribution network design problems rely on finding solutions in reasonable time.

To address these challenges, inspired by urban transportation problems, Osorio and Bierlaire (2013) introduce a so-called metamodel SO approach for continuous problems. A metamodel is an analytical approximation of the objective function. In a first step, the metamodel parameters are estimated based on a set of simulation observations. Then, the metamodel is optimized to derive a new trial point that is evaluated by the simulator, leading to an updated set of observations. By iterating these two steps, the accuracy of the metamodel improves, leading to better trial points. Osorio and Bierlaire (2013) combine a physical and a functional metamodel. Physical metamodels are problem-specific functions that attempt to capture the structure of the underlying decision problem, while functional metamodels are general-purpose functions chosen based on their mathematical proprieties (Søndergaard 2003). The most common form of metamodels used to perform SO are functional metamodels, since they can be used to approximate any objective function (Osorio and Bierlaire 2013). The use of physical metamodels is still limited and most applications focus on continuous problems (see, e.g., Osorio and Chong 2015, Osorio and Nanduri 2015). However,

only with a physical metamodel component we can fully exploit our knowledge about the problem structure. Recently, Zhou et al. (2019) explore the use of a metamodel SO approach for a discrete problem focused on large-scale car-sharing network design problems. We leverage this work by extending the AHA with a metamodel approach focused on last-mile distribution network design by explicitly incorporating order processing congestion at facilities in the SO algorithm.

2.4. Research Gap

To the best of our knowledge, the strategic decision of last-mile distribution networks with CR and tight delivery deadlines has not been studied, despite the fact that such networks are increasingly prevalent in practice (Savelsbergh and Van Woensel 2016). Further, the literature on traditional last-mile distribution networks does not capture the impact of non-linear queuing effects of order processing at facilities (see, e.g., Winkenbach et al. 2016a), and the literature on stochastic location models with congestion does not capture the delivery considerations of last-mile distribution networks (see, e.g., Berman and Krass 2015). Consequently, the proposed methods in literature for last-mile distribution network design are insufficient to capture the complexity of CR and tight delivery deadlines. We address these gaps by proposing a computationally efficient metamodel SO methodology that, by leveraging the problem structure, explicitly captures the effect of order processing congestion at facilities on last-mile distribution networks with tight delivery deadlines. This methodology enables us to optimize and study the effect of the strategic network design on the performance of last-mile distribution networks with CR and tight delivery deadlines.

3. Methodology

In this section, we outline the methodology to solve the strategic network design problem of CR last-mile distribution networks with tight delivery deadlines. We start by defining the last-mile distribution network and the associated strategic and operational decisions. Then, we define the associated network design problem, before introducing the components of our SO solution approach.

3.1. Distribution Network

Any generic last-mile distribution network can be described as a collection of capacity-constrained facilities and capacity-constrained transportation agents. In our model, we define a set of candidate facilities, $\mathcal{F}$. The capacity of facility f is determined by the maximum potential parallel order processing capacity, $e_f^{\max}$. $\mathcal{V}$ is the set of capacity-constrained transportation types. Transportation

agents of type v can serve up to ξ_v^c customers per trip. We consider two categories of transportation agent types. Scheduled transportation agent types, $\mathcal{V}^t \subset \mathcal{V}$, are paid per hour and require upfront decisions on overall committed capacity, e.g., in terms of own employees and vehicles, or through external capacity. On-demand transportation agent types, $\mathcal{V}^o \subset \mathcal{V}$, are summoned for single-stop delivery trips, i.e., $\xi_v^c = 1$. Customer requests arrive throughout the service period with length T. Each customer request c is characterized by a location ϕ_c and placement time τ_c and needs to be fulfilled within a promised time l, i.e., the delivery deadline is at time $\tau_c + l$. The arrival of requests follows arbitrary, potentially non-stationary, geographic and temporal distributions.

3.2. General Problem Definition

The strategic network design is the first set of decisions in a sequential last-mile distribution system design problem. The strategic facility and fleet decisions at time $t = 0$ are made while future demand realizations are still unknown. They influence the operational decisions, such as order allocation and transportation agent dispatching from $t = 1$ onward. The operational decisions at time t influence the future state of the system, and consequently the decisions from $t + 1$ onward.

Strategic decisions.. We consider two types of strategic decisions. First, we decide on the activation of facility location $f \in \mathcal{F}$ with a specific parallel order processing capacity $e_f \in \mathbb{Z}_0$. Here, $e_f = 0$ denotes facility location f being inactive. Let $\mathbf{e}$ be a vector containing the facility processing capacity decisions for all candidate facilities. Second, we decide to contract a certain number of scheduled transportation agents of type $v \in \mathcal{V}^t$, denoted by q_v^t. Arguably, these decisions are of a tactical nature, since they can be revised more frequently than the facility location decisions. However, since transportation contracting decisions are made under uncertainty and constrain the operational decisions, we consider them as strategic for the sake of clarity of our arguments. We set $\mathbf{Q}$ as a vector containing all contracting decisions across the scheduled transportation agent types available. We use $\mathbf{y} = \mathbf{e} \cap \mathbf{Q}$ when we refer to the combined set of strategic decisions.

Operational decisions.. The strategic decisions $\mathbf{y}$ limit the operational decisions of planners in allocating customer requests to particular facility and transportation agent combinations. These daily operational decisions depend not only on the prior strategic decisions, but also on a particular realization of exogenous uncertain parameters, e.g., the dynamic arrival of customer requests. We refer to such a particular realization as a scenario, and $\mathcal{S}_\omega$ captures the realization for the set of uncertain parameters of scenario ω. Furthermore, the operational decisions depend on the

operational decision policy π, which captures the decision logic that specifies how logistics planners run the delivery operations. For each scenario ω, we aim to deliver demand at the lowest cost by making three decisions. First, each customer request c is allocated to a particular combination of facility f and individual transportation agent of type v within the constraints imposed by the strategic design. The order can either be allocated to an existing planned trip of a scheduled transportation agent (i.e., consolidated) or trigger the creation of a new trip. Second, if the order is allocated to an existing trip, we decide on the sequence of deliveries on that trip. Third, we decide when to dispatch each transportation agent based on the trade-off between the likelihood of delivering late and the potential for future order consolidation. Order picking at facilities happens on a first-come, first-serve basis. We let $\mathbf{x}$ denote the vector of all operational decisions.

3.3. Problem Formulation

We proceed by formally defining the CR last-mile distribution network design problem with tight delivery deadlines as a sequential stochastic decision problem, based on the notation introduced by Powell (2014). Let $S_s = (R_s, I_s, K_s)$ be the state of the system at time s. The state of the system is defined by (i) the physical state R_s, which includes the physical structure of the network captured by our set of strategic decisions $\mathbf{y}$, and the location of resources such as transportation agents; (ii) the information state I_s, which captures the received orders and the planned trips; and (iii) the knowledge state K_s, which captures the belief about uncertain variables such as geographic and temporal order distributions and travel time distributions. Furthermore, let W_s be a random variable that captures the exogenous information that becomes available at time s. To capture the transition from S_s to S_{s+1}, given $\mathbf{x}_s$ and W_{s+1}, we define transition function $S^m(\cdot)$ such that

$$S_{s+1} = S^m(S_s, \mathbf{x}_s, W_{s+1}). \tag{1}$$

We define an operational decision making policy π such that $\mathbf{x}_s = X^\pi(S_s)$, and Equation (1) reduces to $S_{s+1} = S^m(S_s, W_{s+1})$. Note that the decision points s are defined as the moments in time at which we make operational decisions, e.g., upon the arrival of orders or the completion of a trip. These decision points are not necessarily uniformly distributed over the time horizon. We

can define our general objective function as

$$\min_{\mathbf{y}} g(\mathbf{y}) = \sum_{f \in \mathcal{F}} C_f(e_f) + \sum_{v \in \mathcal{V}} C_v(q_v^t) + \mathbb{E}[\sum_{s=1}^{T} C(S_s, \mathbf{x}_s)] \tag{2}$$

$$\text{subject to} \quad \mathbf{x}_s = X^\pi(S_s), \qquad 1 \leq s \leq T, \tag{3}$$

$$S_1 = S^m(S_0, \mathbf{e}, \mathbf{Q}, W_1), \tag{4}$$

$$S_{s+1} = S^m(S_s, W_{s+1}) \qquad 1 \leq s \leq T, \tag{5}$$

$$e_f \leq e_f^{\max}, \qquad f \in \mathcal{F}, \tag{6}$$

$$Q_v \leq Q_v^{t,\max}, \qquad v \in \mathcal{V}, \tag{7}$$

$$e_f, Q_v \in \mathbb{Z}, \qquad v \in \mathcal{V}, f \in \mathcal{F}, \tag{8}$$

where $C_f(e_f)$ and $C_v(Q_v)$ capture the cost associated to the strategic design decisions, and $C(S_s, \mathbf{x}_s)$ captures the operational cost at time-period s. Note that $C_f(e_f)$ and $C_v(Q_v)$ are deterministic. We know the current state of the system, and consequently, the cost incurred at time $s = 0$. However, the strategic decisions constrain the subsequent operational decisions via Equations (3) through (5). Equations (6) through (8) limit the domain of the decision variables and impose a maximum capacity constraint at facilities and a maximum number of available transportation agents.

3.4. Simulation-based Optimization Solution Approach

Although the problem defined by Equations (2) through (8) captures the dynamics of the underlying system, it is computationally impractical. Defining optimal strategic decisions $\mathbf{y}$ suffers from the 'curse of dimensionality' associated to the size of the decision space, state space, and action space, and the non-trivial distribution for W_t that captures multiple sources of uncertainty (Powell 2019). Therefore, the problem is intractable with the methods presented in the literature. Note that we are mostly interested in making near-optimal strategic decisions $\mathbf{y}$. Further, while $\mathbb{E}[\sum_{s=1}^{T} C(S_s, \mathbf{x}_s)]$ and its associated constraints are hard to capture algebraically, simulation can provide good approximations for $\sum_{s=1}^{T} C_\omega(S_s, \mathbf{x}_s)$ for individual scenarios ω. By performing multiple simulation runs, e.g., r, we can thus approximate $\mathbb{E}[\sum_{s=1}^{T} C(S_s, \mathbf{x}_s)]$ by its sample average $\frac{1}{r}\sum_{\omega=1}^{r}\sum_{s=1}^{T} C_\omega(S_s, \mathbf{x}_s)$.

For small-sized problems with a limited set of network decisions and potential demand scenarios, the existence of an in-depth simulator allows us to enumerate every possible network design and evaluate its performance on every possible scenario to eventually pick the best performing design.

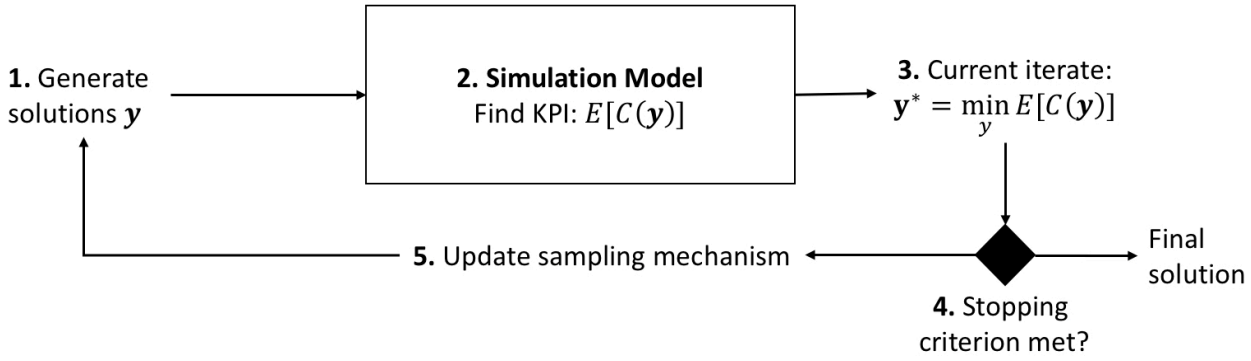


Figure 2: High level overview of SO solution approach

However, for real-world problems, such an approach is not feasible for two reasons. First, the number of decisions is too large to enumerate and simulate every potential solution. Second, the number of scenarios is typically infinite, since orders arrive in continuous time. To address this challenge, we introduce a metamodel SO solution approach, based on the continuous SO framework of Osorio and Bierlaire (2013) and the adapted discrete variant of Zhou et al. (2019). We continue by presenting a high-level overview of the solution approach, before introducing the individual algorithm components and formal definition of the algorithm in Sections 3.5 to 3.7.

Adaptive hyperbox algorithm.. The AHA is a discrete SO locally convergent random search (LCRS) algorithm, i.e., it converges with probability 1 to a local optimum in a solution space defined by discrete variables (Xu et al. 2013). The algorithm converges by iteratively following the loop defined in Figure 2 consisting of the five steps outlined in Algorithm 1.

Algorithm 1 High-level overview of AHA Algorithm

Step 1: Generation of feasible solutions according to a specified sampling mechanism.

Step 2: Performance evaluation of generated solutions leveraging a simulator.

Step 3: Determination of current iterate, i.e., the best found solution hitherto.

Step 4: Evaluation of current iterate against stopping criterion.

Step 5: Update of sampling mechanism based on previously evaluated solutions.

A key component of AHA is the hyperbox, which defines the most promising area in the solution space. The generation of solutions is concentrated on the hyperbox. To formally define the hyperbox, let $y^{(d)}$ be the d^{th} coordinate of the decision vector $\mathbf{y}$, which consists of D elements, and

$l_k^{(d)}$ and $u_k^{(d)}$ be the lower and upper bound of the hyperbox for coordinate d at algorithm iteration k,

$$\mathcal{H}(k) = \{\mathbf{y} : l_k^{(d)} \leq y^{(d)} \leq u_k^{(d)}, 1 \leq d \leq D\}. \tag{9}$$

To update the hyperbox, i.e., to update the lower and upper bounds of each element of $\mathbf{y}$, we compare the current iterate $\mathbf{y}_k^*$ to the set of other sampled solutions, $\mathcal{L}$. Specifically

$$l_k^{(d)} = \begin{cases} \max_{\mathbf{y}\in\mathcal{L}, \mathbf{y}\neq\mathbf{y}_k^*}\{y^{(d)} : y^{(d)} < y^{*,(d)}\} & \text{if it exists,} \\ -\infty & \text{otherwise,} \end{cases} \tag{10}$$

$$u_k^{(d)} = \begin{cases} \min_{\mathbf{y}\in\mathcal{L}, \mathbf{y}\neq\mathbf{y}_k^*}\{y^{(d)} : y^{(d)} > y^{*,(d)}\} & \text{if it exists,} \\ \infty & \text{otherwise.} \end{cases} \tag{11}$$

Colloquially speaking, the hyperbox is bounded from above (below) in the d^{th}-dimension by the solution with the lowest (highest) value for $y^{(d)}$ that is higher (lower) than the value of the d^{th} element of the current iterate. Throughout the algorithm, the hyperbox changes in size and position in two ways: due to the exploration of new solutions, and by increasing the number of simulations for already explored solutions. To avoid a premature convergence of AHA and to not spend significant computation resources on exploring a small area around a potentially sub-optimal local minimum, Xu et al. (2013) have combined it with the multi-start Industrial Strength COMPASS framework of Xu et al. (2010).

MetaAHA.. Building on the metamodel-based AHA (MetaAHA) introduced by Zhou et al. (2019), the SO approach introduced in this paper extends the AHA by incorporating an analytical metamodel into the sampling mechanism. At Step 1 of every iteration of Algorithm 1, in addition to sampling solutions from the hyperbox, we solve several instances of a mixed-integer linear program (MILP) that approximates the non-linear, probabilistic, and non-convex last-mile distribution network design problem. Furthermore, at Step 5 of every iteration, we update the MILP (i.e., we learn the value of the metamodel parameters) to integrate the information obtained from running the simulator. We introduce the simulation model in Section 3.5, the analytical metamodel in Section 3.6, and we define the SO algorithm that ties both together in Section 3.7.

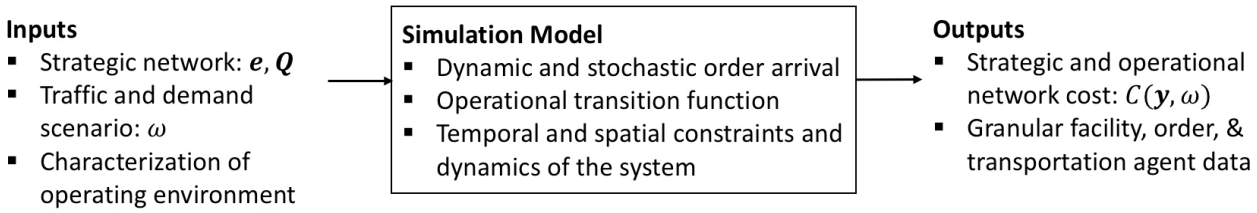


Figure 3: High-level overview of the inputs and outputs of the simulation model.

3.5. Simulator

Step 2 of Algorithm 1 relies on the availability of a disaggregate, in-depth operational simulator that allows for an accurate evaluation of a proposed network design. Specifically, the simulator should accurately capture the repeated execution of the transition function specified in Equation (5). This includes i) the dynamic and stochastic arrival of orders, ii) the operational allocation, dispatching, and routing decisions $\mathbf{x}$, introduced in Section 3.2, and iii) the underlying temporal and spatial dynamics of the system and constraints, including the network design specified by $\mathbf{y}$. We provide a high-level overview of the necessary inputs and outputs in Figure 3. The simulator requires three major inputs: i) a strategic network design, including enabled facilities and parallel processing capacity levels, and the number of scheduled transportation agents; ii) an operational scenario, which consists of a demand scenario, i.e., a realization of customer locations, order timing, and drop sizes, and a traffic scenario, i.e., a realization of the travel times at each time interval; and iii) a characterization of the environment, including a graph representation of the road network. Based on the inputs, the simulator provides detailed KPIs (e.g., (disaggregate) network cost, and facility utilization) of a particular network design for a particular scenario. For the purpose of this paper, we rely on the simulator introduced by Lavenir (2019). The author develops a discrete event simulation (DES) simulator using the SimPy library in Python. We refer to Appendix Appendix B for an in-depth overview of the simulator. Note that this simulator could be replaced by any simulator that best captures the specific network under study and meets the requirements specified above.

3.6. Analytical Metamodel

We define the metamodel optimization problem at iteration k of the solution algorithm as

$$\min_{\mathbf{y}} m_k(\mathbf{y}; \alpha_k, \beta_k) = \alpha_k g_{A,k}(\mathbf{y}) + \phi(\mathbf{y}; \beta_k). \tag{12}$$

The metamodel contains the objective function of a problem-specific MILP, $g_{A,k}(\mathbf{y})$, which attempts to capture the structure of the underlying last-mile urban distribution network, and is corrected for parametrically by a scaling term α_k and an additive error term, $\phi(\mathbf{y};\beta_k)$ (Zhou et al. 2019). The error-term is a general-purpose polynomial, for which the parameters are fitted in every iteration based on the available simulation results.

Physical metamodel component.. When defining the physical metamodel, we ensure that it is i) an accurate representation of the non-linear, non-convex, and stochastic objective function $g(\mathbf{y})$, ii) scalable, to address real-world problems in urban distribution, and iii) computationally efficient, to justify the integration of the metamodel in every iteration instead of running additional simulations. To this end, our MILP focuses on the strategic network design decisions, while approximating the operational decisions. To enable the specification of a tractable model, we make four simplifications.

First, we develop an expected value based deterministic analytical approximation of the stochastic decision problem. This implies that there is no uncertainty about the location or timing of demand, i.e., we exactly know which order is going to occur when, and where it has to be delivered.

Second, we aggregate demand temporally. We divide the day into a set of discrete periods $\mathcal{T}$ and aggregate orders that fall within each time period. Furthermore, we assume that the temporal distributions of demand within each of these time periods are stationary. This implies that we assume that demand is distributed uniformly over time within each time period, while we capture demand fluctuations throughout the day, since each time period is characterized by a different demand level and a unique spatial distribution. We define Δ_t as the length of demand period $t \in \mathcal{T}$.

Third, we aggregate demand geographically. In line with Winkenbach et al. (2016a) and Merchán and Winkenbach (2018), we discretize the city into a large set of adjacent rectangular pixels, $\mathcal{I}$. We aggregate demand within each of these pixels, which simplifies the problem by reducing the number of distinct demand points, while still capturing the geographic, infrastructural, and demand related heterogeneity within the city. Each pixel is defined by a set of parameters describing its geographical location, shape, and demand characteristics. More specifically, for each pixel i and time period t we define the total demand, i.e., the total number of orders, as γ_{it}.

Fourth, we aggregate transportation capacity based on the expected number of transportation agents required per pixel and time period, rather than modeling trips of individual agents. We approximate the travel distance to every customer by the travel distance to the centroid of its associated pixel, d_{if}. Furthermore, we introduce a pixel, facility, transportation agent type, and time

period specific consolidation factor, k_{ifvt}, to account for the reduction in transportation capacity requirements due to consolidation. We define k_{ifvt} in Appendix Appendix C.2. We capture the operational allocation decisions by x_{ifvt}, i.e., the allocation of pixel i in period t to facility f and transportation agent type v. We denote the time a scheduled transportation agent spends on an order t^{o}_{ivft}.

We introduce a set of binary indicator variables a_f, to indicate if a facility is opened at location f, i.e., $a_f = 1$ if $e_f \geq 1$, allowing us to model the non-linear fixed set-up cost incurred to enable capacity at a facility, such as rent, hiring cost, equipment, etc. We summarize the notation used throughout this paper in Tables A.5 through A.7 in Appendix Appendix A, and proceed by formally introducing the iteration-independent physical component of the metamodel $g_A(\mathbf{y})$. At the end of this section, we introduce the iteration-dependent extension of $g_A(\mathbf{y})$, $g_{A,k}(\mathbf{y})$ incorporated into the SO algorithm.

$$\min_{\mathbf{a},\mathbf{e},\mathbf{q}^t,\mathbf{x},\mathbf{q}^o} \sum_{f\in\mathcal{F}}(K^f_f a_f + c^e_f e_f) + \sum_{v\in\mathcal{V}}\sum_{t\in\mathcal{T}} \Delta_t c^t_v q^t_v + \sum_{t\in\mathcal{T}}\sum_{v\in\mathcal{V}} c^o_v q^o_{vt} + \sum_{t\in\mathcal{T}}\sum_{v\in\mathcal{V}}\sum_{I\in\mathcal{I}} c^d_v \sum_{f\in\mathcal{F}} d_{if} k_{ifvt} x_{ifvt} + \sum_{i\in\mathcal{T}}\sum_{I\in\mathcal{T}} c^{ls}\gamma_{it}(1 - \sum_{f\in\mathcal{F}}\sum_{v\in\mathcal{V}} x_{ifvt}) \tag{13}$$

s.t.

$$\sum_{f\in\mathcal{F}}\sum_{v\in\mathcal{V}} x_{ifvt} \leq 1, \qquad i\in\mathcal{I}, t\in\mathcal{T}, \tag{14}$$

$$\sum_{i\in\mathcal{I}}\sum_{v\in\mathcal{V}} \gamma_{it} x_{ifvt} \leq \xi^h_f \Delta_t e_f, \qquad f\in\mathcal{F}, t\in\mathcal{T}, \tag{15}$$

$$\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}} k_{ifvt} x_{ifvt}(t^{\mathrm{o}}_{ifvt}\gamma_{it}\Delta_t - \sum_{\tau=t+1}^{T} f_{ifv\tau}(t)) + \sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}}\sum_{\tau=0}^{t-1} k_{ifv\tau} x_{ifv\tau} f_{ifvt}(\tau) \leq q^t_v \Delta_t, \qquad v\in\mathcal{V}^t, t\in\mathcal{T}, \tag{16}$$

$$\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}} \gamma_{it} x_{ifvt} \leq q^o_{vt}, \qquad v\in\mathcal{V}^o, t\in\mathcal{T}, \tag{17}$$

$$q^o_{vt} \leq \Delta_t Q^{o,\max}_v, \qquad v\in\mathcal{V}^o, t\in\mathcal{T}, \tag{18}$$

$$x_{ifvt} = 0, \qquad i \notin \mathcal{I}_{fvt}, v \in \mathcal{V}, f \in \mathcal{F}, t \in \mathcal{T}, \tag{19}$$

$$x_{ifvt} \geq 0, \qquad i \in \mathcal{I}_{fvt}, v \in \mathcal{V}, f \in \mathcal{F}, t \in \mathcal{T}, \tag{20}$$

$$a_f \in \{0,1\}, \qquad f \in \mathcal{F}, \tag{21}$$

$$e_f \leq e_f^{\max}, \qquad f \in \mathcal{F}, \tag{22}$$

$$q_v^t \leq Q_v^{t,\max}, \qquad v \in \mathcal{V}, \tag{23}$$

$$e_f, q_v^t, q_{vt}^o \in \mathbb{Z}, \qquad v \in \mathcal{V}, f \in \mathcal{F}. \tag{24}$$

The objective (13) aims to minimize the total network cost, consisting of: i) fixed cost of opening facilities, $K_f^f a_f$, ii) cost of parallel order processing capacity at the facility, $c_f^e e_f$, iii) cost of scheduled transportation capacity, $\Delta_t c_v^t q_v^t$, iv) cost of on-demand transportation agents that are hired per delivery, $c_v^o q_{vt}^o$, v) total distance based travel cost of transportation agents, $c_v^d d_{if} k_{ifvt} x_{ifvt}$, and vi) cost of lost sales, $c^{ls}\gamma_{it}(1 - \sum_{f\in\mathcal{F}}\sum_{v\in\mathcal{V}} x_{ifvt})$. Constraints (14) ensure that no more than the total demand is allocated to facility-agent combinations. Unallocated demand is considered as lost sales. Constraints (15) ensure that the available order processing capacity at facilities, $\xi_f^h \Delta_t e_f$, is not exceeded by the allocated demand in a particular time period. Constraints (16) translate the demand allocation to a number of scheduled transportation agents required to satisfy demand. The demand in previous time periods influences the scheduled transportation capacity required in the current time period, since there are spillover effects of orders that are being delivered or still need to be delivered. The left-hand side of these constraints computes the resulting total quantity of transportation time required in a particular time period. We define the transportation capacity overflow $f_{ifv\tau}(t)$ and the consolidation factor k_{ifvt} in Appendix Appendix C.2. Constraints (17) ensure that the number of each type of on-demand transportation agents, q_{vt}^o, can handle the allocated demand in time period t, while Constraints (18) impose a cap on the number of on-demand agents that can be deployed per hour, based on the average deployed on-demand agents per time period. Constraints (19) ensure that pixels are not allocated to a facility-transportation combination that would lead to a guaranteed late delivery in time period t. To ensure this, we introduce the set $\mathcal{I}_{fvt} = \{i \in \mathcal{I} | t_{ifvt}^d \leq l\}$, where t_{ifvt}^d is the minimum time required to deliver an order, including order processing and delivery. Constraints (20) through (24) limit the domain of the decision variables. Notably, Constraints (22) limit the parallel order processing capacity per facility and Constraints (23) limit the maximum number of scheduled transportation agents that can be contracted.

Functional metamodel component.. The physical component of the analytical metamodel proposed in Equations (13) through (24) is a deterministic linear programming model, which does not capture the randomness associated to the arrival of orders. Therefore, it is less adequate in capturing non-linear and stochastic effects. As a consequence, the model does not appropriately capture the role of utilization of parallel processing capacity at facilities and the associated formation of processing queues. Since networks with CR and tight delivery deadlines rely on efficiency at every step of the fulfillment process, facility processing queues hinder the ability to make the promised delivery deadlines. Increasing queue lengths lead to late deliveries of orders, potentially affecting the network performance substantially in terms of service level and expected cost. Therefore, we aim to capture queuing effects of facilities in the functional component of the metamodel, $\phi(\mathbf{y};\beta_k)$.

It is well known in queuing theory that the relationship between queue length, or equivalently the waiting time through Little's law, and server utilization is exponential (Little 1961). However, we refrain from capturing this exponential effect explicitly in our metamodel, since it would render the metamodel non-linear at the cost of additional computational complexity, contradicting our goal of defining a metamodel that is computationally efficient. Rather, we approximate this non-linear effect in the functional component of our metamodel by defining binary dummy variables u_j, to indicate that the network utilization ρ, i.e., the aggregate utilization of order processing capacity over all facilities in the network, lies within a specific interval $[\rho_j^{\min}, \rho_j^{\max})$ for J contiguous intervals on $[0,1]$, and associated error terms. In addition, we add error terms for the strategic design decisions $\mathbf{y}$. The resulting functional component of the metamodel, is the linear polynomial

$$\phi(\mathbf{y};\beta_k) = \beta_{k,0} + \sum_{i=1}^{|\mathbf{y}|} \beta_{k,i}^{y} y_i + \sum_{i=1}^{J} \beta_{k,i}^{\rho} u_j, \tag{25}$$

which results in the metamodel formulation

$$m_k(\mathbf{y};\alpha_k,\beta_k) = \alpha_k g_A(\mathbf{y}) + \beta_{k,0} + \sum_{i=1}^{|\mathbf{y}|} \beta_{k,i}^{y} y_i + \sum_{j=1}^{J} \beta_{k,j}^{\rho} u_j, \tag{26}$$

subject to Constraints (14) through (24), to ensure that that $\mathbf{y}$ is a feasible solution. Further, to

ensure that the binary indicators u_j accurately capture the system wide facility utilization, we add

$$\frac{\sum_{f\in\mathcal{F}} e_f \xi_f^h \sum_{t\in\mathcal{T}} \Delta_t}{\sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} \gamma_{it}} \leq \rho_j^{\max} M(1-u_j), \qquad 1 \leq j \leq J, \tag{27}$$

$$\frac{\sum_{f\in\mathcal{F}} e_f \xi_f^h \sum_{t\in\mathcal{T}} \Delta_t}{\sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}} \gamma_{it}} \geq \rho_j^{\min} - u_j, \qquad 1 \leq j \leq J, \tag{28}$$

$$\sum_{j=1}^{J} u_j = 1, \tag{29}$$

$$u_j \in \{0,1\}, \qquad 1 \leq j \leq J. \tag{30}$$

Constraints (27) through (29) ensure that only one utilization indicator is equal to one, based on the total expected demand and allocated processing capacity over all facilities. Here, M is a sufficiently large number. Constraints (30) limit the domain of the indicator variables. By defining the network utilization ρ as the aggregate utilization over all facilities, we avoid defining facility-specific utilization indicators and thus control the number of parameters to be fitted in the metamodel. However, we fit parameters for the processing capacity at every facility, thus implicitly controlling the utilization at each individual facility by penalizing the facility capacity.

3.7. Discrete SO algorithm

We integrate the simulation model (Section 3.5) and the metamodel (Section 3.6) into Algorithm 2, referred to as MetaAHA+ in the following. Compared to MetaAHA, MetaAHA+ contains two conceptual extensions to the metamodel: i) it partitions the solution space based on the utilization of the parallel order processing capacity at facilities of the current iterate, and ii) it captures the non-linear queuing effects on network performance in the functional component of the metamodel.

Utilization partition.. We account for non-linear queuing effects at facilities by introducing utilization specific error terms in the metamodel in Section 3.6. In addition, to stimulate exploration of the solution space, we partition the solution space at every iteration of the algorithm based on the aggregate network utilization of parallel processing capacity ρ, and the aggregate network utilization of parallel processing capacity per time period t, ρ_t. Based on the outputs of the simulation and the current iterate y_k^*, we determine iteration specific network utilization factors ρ_k and ρ_{tk}. Subsequently, at iteration $k+1$, we solve two instances of the metamodel. First, in Step 1.3 of Algorithm 2, we limit the systemwide facility utilization in every time period to at most ρ_{tk}, and

Algorithm 2 MetaAHA+ Algorithm

Initialization:

0.1 $k = 0$, $\mathcal{H}(k) = \Omega$, generate $\mathbf{y}_0 \in \mathcal{H}(k)$, and set $\mathbf{y}_0^* = \mathbf{y}_0$, $\mathcal{L}(k) = \{\mathbf{y}_0\}$

0.2 Simulate $\mathbf{y}_0$ and determine $G(\mathbf{y}_0)$

Step 1: Determine $\mathcal{L}(k)$

1.1 $k = k + 1$

1.2 Obtain r points in $\mathcal{H}(k)$ based on the asymptotically uniform sampling mechanism of AHA

1.3 Obtain $\mathbf{y}_k^{\text{meta-}\rho-}$, the solution to Problem (26) with Constraints (31) and (32)

1.4 Obtain $\mathbf{y}_k^{\text{meta-}\rho+}$, the solution to Problem (26) with Constraints (33) and (34)

1.5 Obtain $\mathbf{y}_k^{\text{meta-hyper}}$, the solution to Problem (12) with additional hyperbox constraints $\mathbf{y} \in \mathcal{H}(k)$

Step 2: Simulate performance of solutions $\mathbf{y} \in \mathcal{L}(k)$

2.1 Determine $\mathcal{A}_k(\mathbf{y})$ based on Equation (E.13) in Appendix Appendix E.2

2.2 Simulate and determine $G(\mathbf{y})$ based on all current and historic simulations

2.3 Determine $\mathbf{y}_k^* = \text{argmin}_y G(\mathbf{y})$

2.4 Determine $\mathcal{H}(k)$ based on Equations (10) and (11)

Step 3: Check for termination criteria

3.1 If $\mathbf{y}_k^*$ is a local optimum following the procedure of AHA: Stop.

3.2 If the computational budget is depleted: Stop

Step 4: Update the metamodel

4.1 Determine $g(\mathbf{y})$ for any non-evaluated solution in $\mathcal{L}(k)$ using optimization problem (13)

4.2 Fit the metamodel parameters using Equation (38)

the utilization of every facility over the entire horizon to at most ρ_k by adding constraints

$$\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}}\sum_{v\in\mathcal{V}}\gamma_{it}x_{ifvt} > \rho_{tk}\Delta_t\sum_{f\in\mathcal{F}}\xi_f^h e_f, \qquad t\in\mathcal{T}, \tag{31}$$

$$\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{I}}\sum_{v\in\mathcal{V}}\gamma_{it}x_{ifvt} > \rho_k\xi_f^h\Delta_t e_f, \qquad f\in\mathcal{F}, \tag{32}$$

to the metamodel. We define the solution to this problem as $\mathbf{y}_k^{\text{meta-}\rho-}$. Second, in Step 1.4 we find $\mathbf{y}_k^{\text{meta-}\rho+}$ by providing ρ_{tk} as a lower bound on the systemwide facility utilization in every time period, and ρ_k as a lower bound for every facility over the entire horizon by adding the constraints

$$\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}}\sum_{v\in\mathcal{V}}\gamma_{it}x_{ifvt} < \rho_{tk}\Delta_t\sum_{f\in\mathcal{F}}\xi_f^h e_f, \qquad t\in\mathcal{T}, \tag{33}$$

$$\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{I}}\sum_{v\in\mathcal{V}}\gamma_{it}x_{ifvt} < \rho_k\xi_f^h\Delta_t e_f, \qquad f\in\mathcal{F}. \tag{34}$$

Metamodel fit.. In Step 4.2 of Algorithm 2, the metamodel parameters are updated to include new observations by solving a weighted least squares problem. We find the metamodel parameter vectors α and β that minimize the weighted distance function between the metamodel $m_k(\mathbf{y})$ and the simulation observations $\hat{g}_k(\mathbf{y})$. We propose a two-step approach to piecewise linearize the non-linear curve that governs the relationship between the network cost and the systemwide utilization at facilities. This ensures that we preserve as much information as possible on its non-linear nature when fitting the metamodel. First, we approximate the exponential relationship between the network performance and the network utilization leveraging the P-th Taylor polynomial. Therefore, we provide an alternative metamodel formulation, $\hat{m}_k(\mathbf{y};\hat{\alpha}_k,\hat{\beta}_k)$,

$$\hat{m}_k(\mathbf{y};\hat{\alpha}_k,\hat{\beta}_k) = \hat{\alpha}_k g_A(\mathbf{y}) + \hat{\beta}_{k,0} + \sum_{i=1}^{|\mathbf{y}|}\hat{\beta}_{k,i}^y y_i + \sum_{p=0}^{P}\hat{\beta}_{k,p}^\rho \rho^p. \tag{35}$$

We fit the alternative parameters $\hat{\beta}_{k,p}^\rho$ for every term of the polynomial to approximate the exponential effect of the system-wide facility utilization on network performance. We solve the least squares problem

$$\min_{\hat{\alpha}_k,\hat{\beta}_k}\sum_{\mathbf{y}\in\mathcal{L}}[w_k(\mathbf{y})(g_A(\mathbf{y}) - \hat{m}_k(\mathbf{y};\hat{\alpha}_k,\hat{\beta}_k))]^2 + (w_0(\hat{\alpha}_k-1))^2 + \sum_{i=0}^{|\mathbf{y}|}(w_0\hat{\beta}_{k,i}^y) + \sum_{p=0}^{P}(w_0\hat{\beta}_{k,p}^\rho), \tag{36}$$

to find parameters $\hat{\beta}^{\rho}_{k,p}$. Here, the weight function $w_k(\mathbf{y})$ is defined as

$$w_k(\mathbf{y}) = 1/(1 + ||\mathbf{y} - \mathbf{y}^*_k||_2). \tag{37}$$

Consequently, the least squares problem minimizes a weighted distance between the simulated profit estimates $\hat{g}$ and the alternative metamodel predictions $\hat{m_k}$, where each point is weighted based on their proximity to the current optimal solution $\mathbf{y}^*_k$ to improve the local fit of the metamodel around the current iterate. The additional terms and associated weights w_0 ensure a full rank least squares matrix when the number of observations is still smaller than the number of parameters to be fitted. This estimation approach is formulated and discussed in greater detail in Osorio and Bierlaire (2013).

Second, we find a piecewise linearization for $U(\rho) = \sum_{p=0}^{P} \hat{\beta}^{\rho}_{k,p}\rho^p$ to render our metamodel m_k linear. We are particularly interested in $\rho \in [0, 1)$, since for $\rho \geq 1$ the queue grows without bound and becomes unmanageable. We propose Algorithm 3 to piecewise linearize $U(\rho)$ in J partitions.

Algorithm 3 Piecewise linearization of relationship between utilization and performance

Step 1. Find $U^{\max} = U(1)$ and $U^{\min} = U(0)$

Step 2. Find sub-range size $U^{\text{piece}} = \frac{U^{\max}=U(1)-U^{\min}=U(0)}{J}$

Step 3. For every sub-range in $j \in \{1, \cdots, J\}$ find

- Cost interval $[U^{\min}_j, U^{\max}_j) = [U^{\min} + (j-1)U^{\text{piece}}, U^{\min} + jU^{\text{piece}})$
- Find the interval for utilization $[\rho^{\min}_j, \rho^{\max}_j)$ that solves $[U^{\min}_j = U(\rho^{\min}_j), U^{\max}_j = U(\rho^{\max}_j))$

Step 4. Define $u_j = 1$ if $\rho \in [\rho^{\min}_j, \rho^{\max}_j)$ and 0 otherwise

Step 5. Find parameters $\beta^{u}_{k,n}$ by minimizing

$$\min_{\alpha,\beta} \sum_{\mathbf{y}\in\mathcal{L}} [w_k(\mathbf{y})(g_A(\mathbf{y}) - m_k(\mathbf{y}; \alpha_k, \beta_k))]^2 + (w_0(\alpha - 1))^2 + \sum_{i=0}^{|\mathbf{y}|}(w_0\beta^{y}_{k,i}) + \sum_{j=1}^{J}(w_0\beta^{\rho}_{k,j}). \tag{38}$$

Based on this approach, we estimate parameters $\beta^{\rho}_{k,j}$ for every interval j. The proposed algorithm results in larger (smaller) intervals for lower (higher) utilization levels, where utilization has a lower (higher) impact. Furthermore, solving Equation (38) provides us with the parameters α_k and β^{y}_{k}, and thus completes our specification of the metamodel defined in Equation (12).

4. Case Study

The problem instances supporting our analysis are based on a large-scale, real-life deployment of a last-mile distribution network with CR and tight delivery deadlines by a global fashion company (GFC) in Manhattan. The GFC offers an e-commerce service in addition to its extensive global

network of physical stores and resellers. In select areas, it already provides same day delivery (SDD), but the company is exposed to competitive pressure from firms such as Amazon, that are rolling out delivery options with tighter deadlines such as two-hour delivery. The GFC considers being an early mover as a strategic opportunity. In the fashion market, brand recognition is an important asset, and the company believes providing one to two-hour delivery deadlines in their key markets contributes to being perceived as a premium brand. All parameters, including vehicle speed, cost, and capacities, facility cost and capacities are based on actual data obtained from the GFC. To protect this proprietary information we present aggregated and normalized data and results.

To validate our methodology for various demand scenarios, we develop six stylized problem instances inspired by real data from the GFC, varying the systemwide demand characteristics. Each problem instance is characterized by a systemwide demand density distribution, i.e., a distribution that governs the interarrival time of orders in the system, and a geographical distribution, i.e., a distribution that governs the location of each arriving order. The systemwide demand density can either be *stationary* (S), i.e., constant throughout the day, or *dynamic* (D), i.e., varying throughout the day. We define three types of geographic distributions: i) *uniform* (U), i.e., uniformly distributed over the demand area, ii) *concentrated* (C), i.e., the majority of demand is concentrated in one geographic area, and iii) *evolving* (E), i.e., demand is concentrated, but the centroid of the concentration moves throughout the day. We refer to these problem instances by their combination of demand density and geographic distribution, e.g., D-U refers to problem instance with a dynamic demand interarrival distribution and a uniform geographical distribution. Details on the problem instances can be found in Appendix Appendix D.1. The remaining parameters are the same for every instance and based on data of the GFC. Inspired by Manhattan, we define the demand area as a rectangular area of 100km^2 (5x20 km). We generate 10 realistic potential facility locations using Algorithm 4 in Appendix Appendix D.1 and we use a Euclidean distance metric throughout the analysis.

5. Results

Based on the six stylized problem instances defined in Section 4, we first evaluate the algorithmic performance of MetaAHA+ by comparing it to MetaAHA, AHA, and both a stochastic programming (SP) and deterministic programming (DP) approach. Next, we evaluate the effect of aggregate systemwide facility utilization and facility processing queues on the network performance. Further, we evaluate the effect of tightness of the delivery deadline on network design and

performance.

5.1. Algorithmic Performance

We evaluate the performance of MetaAHA+ on three dimensions: expected cost of the proposed design, inter-restart consistency, and its performance under tight computational budgets. We compare our algorithm to MetaAHA (Zhou et al. 2019, Algorithm 1, p.18), AHA (Xu et al. 2013, Algorithm 1, p.136), the DP model presented by Equations (13) through (24), and its SP variant solved using a sample average approximation (SAA)-based approach (Kleywegt et al. 2001). Note that we use the same set of scenarios, i.e., uncertainty realizations, across methods to ensure a fair comparison. We refer the reader to Appendix Appendix E.2 for a definition of the SP formulation and an overview of the algorithm parameters.

Cost performance.. Table 1 summarizes the cost performance of the best of 10 algorithm restarts for each solution approach. The cost performance of MetaAHA+ and MetaAHA is comparable at termination. On average, the expected network cost of the network design suggested by MetaAHA exceed the cost of MetaAHA+ by 0.6%. However, both methods outperform the other solution approaches. The design found by MetaAHA+ is on average 3.1% better than the design found by AHA, 9.3% better than the design found by SP, and 56.8% better than the design found by DP. This indicates that there is significant value in deploying a metamodel SO approach to design CR networks with tight delivery deadlines compared to traditional SO, SP, and DP.

Table 1: Cost performance of deployed solution methods averaged over six problem instances at algorithm termination and after 150 simulation runs (see disaggregate results in Table F.9, Appendix Appendix F)

	At Termination		**At Early Termination**		
Method	**Cost ($)**	**Relative Gap to MetaAHA+ Solution (%)**	**Cost ($)**	**Relative Gap to Termination Solution (%)**	**Relative Gap to MetaAHA+ Solution (%)**
MetaAHA+	96.6	0.0	100.3	4.0	0.0
MetaAhA	97.2	0.6	105.6	8.8	5.3
AHA	99.8	3.1	170.2	72.0	70.2
DP	151.2	56.8	151.2	0.0	50.6
SP	104.9	9.3	104.9	0.0	5.1

Performance with limited computational budget.. MetaAHA+ and MetaAHA differ substantially with regards to their speed of convergence towards an optimal solution. This is particularly important if a design has to be determined on a tight computational budget. Table 1 shows that after 150

simulation runs, the expected network cost for the best solution found by MetaAHA+ exceed the cost of the best-found solution at algorithm termination cost by 4.0%. The algorithm terminates after 678 simulation runs on average, i.e., the solution found within 22% of the computational budget performs only 4.0% worse. The cost of the solution found by MetaAHA at early termination exceed the cost of the best-found solution at termination by 8.8%, indicating that MetaAHA+ finds good solutions faster than MetaAHA. Figure 4 confirms this result. While MetaAHA generally finds good solutions, it exhibits a less steep gradient of the objective value for the first iterations. The reason for this faster convergence is the effectiveness of the metamodel in finding updated solutions. In the case of MetaAHA, the current iterate is updated 632 times over all problem instances and individual restarts, of which 219 updates fall within the first 150 simulation runs. 34.0% of the solutions that improved the current iterate are proposed by the metamodel, i.e., a metamodel update. In the first 150 simulation runs, 62.6% of the updates of the current iterate are metamodel updates. For MetaAHA+, the current iterate is updated 567 times (thereof 46.4% are metamodel updates), and 244 updates occur during the first 150 simulation runs (thereof 72.5% are metamodel updates). We see that MetaAHA+ incurs more updates in the first 150 simulation runs, and a larger percentage of these updates is a solution to the analytical metamodel. This indicates that learning about the system-wide facility utilization level is a key enabler for fast convergence.

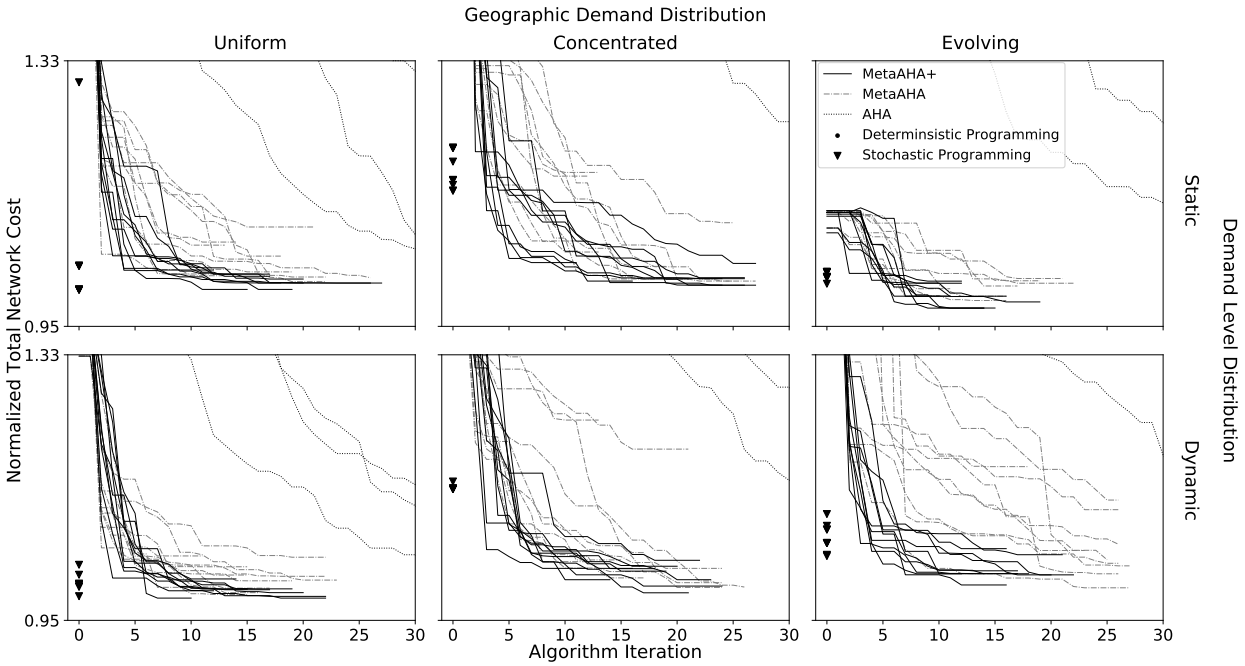


Figure 4: Total network cost evolution for each solution method and algorithm restart for each problem instance (normalized). Some results for AHA and DP fall outside the chart, see Figure F.11 in Appendix Appendix F.

Inter-restart consistency.. MetaAHA+ outperforms MetaAHA in terms of consistency of solution quality. Table 2 confirms the results of Figure 4 that for each problem instance, the performance is most consistent for MetaAHA+ and least consistent for AHA. While the best cost performance for each problem instance is about equal for MetaAHA and MetaAHA+, the average coefficient of variation of the ten different restarts for each instance is 3.4 times as large for MetaAHA: 3.8% for MetaAHA, and 1.1% for MetaAHA+. The absolute gap between the coefficients of variation is larger in the case of termination after 150 simulation runs, with an average coefficient of variation of 7.1% for MetaAHA, and 4.8% for MetaAHA+.

Table 2: Cost performance of individual restarts of deployed solution methods averaged over six problem instances (see disaggregate results in Table F.10, Appendix Appendix F)

Method	At Termination			At Early Termination		
	Mean ($)	**Standard Deviation**	**Gap to Best Found Solution (%)**	**Mean ($)**	**Standard Deviation**	**Gap to Best Found Solution (%)**
MetaAHA+	97.5	1.1	1.6	106.8	5.2	11.5
MetaAHA	100.4	4.0	4.7	117.7	9.3	22.7
AHA	100.7	1.9	4.9	184.7	12.9	93.2
DP	148.2	0.0	53.9	148.2	0.0	53.9
SP	103.6	2.4	8.4	103.6	2.4	8.4

Summarizing, both MetaAHA+ and MetaAHA outperform AHA, DP, and SP in terms of expected cost. However, MetaAHA+ outperforms MetaAHA by finding better solutions under a tight computational budget and by more consistently finding good solutions across algorithm restarts. Both characteristics are paramount in actual business applications. The computational budget is limited, driven by time pressure in decision making, and model-based business recommendations need to be robust. Both fast convergence and high inter-restart consistency reduce the need for computational resources and build trust and managerial buy-in to the solution obtained.

5.2. Network Design

Table F.11 in Appendix Appendix F summarizes the proposed network design for each problem and solution instance at termination and at early termination after 150 simulation runs, if applicable. We can draw three main conclusions from these results.

Solution evolution between 150th simulation and algorithm termination.. For every problem instance, the best network found after 150 simulation runs is different from the best network found at the termination of each of the SO algorithms. Notably, at early termination, the proposed ag-

gregated parallel processing capacity is larger or equal than suggested at algorithm termination. This indicates that the cost performance of the last-mile distribution problem with CR and tight delivery deadlines is sensitive to undercapacity.

Comparison to optimization.. The number of active facilities and the aggregated parallel processing capacity proposed by MetaAHA+ is always greater than or equal to the design proposed by SP, and greater than the design proposed by DP. Further, the suggested facility locations in MetaAHA+ differ in five (four) out of six problem instances from those suggested in DP (SP). This confirms that traditional optimization methods are inappropriate to determine the network design. Such methods do not capture order processing congestion at facilities. Therefore, there appears to be no need for excess processing capacity when solving the problem, which systematically leads to sub-optimal network designs.

Comparison between problem instances.. Across all instances and solution algorithms, the best solution found activates three facilities. However, Figure 5 shows the differences in cost structure for the network design proposed by MetaAHA+ for each of the problem instances. For problem instances where the demand density distribution is dynamic, we observe a larger investment in parallel processing capacity, a smaller investment in scheduled transportation capacity, and a larger use of on-demand transportation capacity. Notably, the parallel processing capacity is higher to ensure capacity during peak demand. Further, scheduled transportation capacity is lower, since lower demand during off-peak hours does not justify a high transportation capacity while order delivery can be outsourced during peak hours. Studying the value of flexibly adjusting capacity would be a worthwhile extension of this work. Moreover, Figure 5 shows that a concentrated geographical demand distribution leads to lower transportation cost, as it encourages the activation of facilities in high demand density areas, leading to shorter travel distances, increasing the potential to consolidate, and reducing the on-demand transportation capacity cost. While the problem instances with an evolving geographical demand distribution also benefit from high density, the changing location of the center of gravity of demand causes longer travel distances, leading to a slight decrease in consolidation potential and an increase in on-demand transportation capacity cost.

5.3. Effects of Facility Congestion

The network performance is affected by facility congestion, driven by the network utilization, i.e., the aggregate utilization of order processing capacity over all facilities in the network. In this

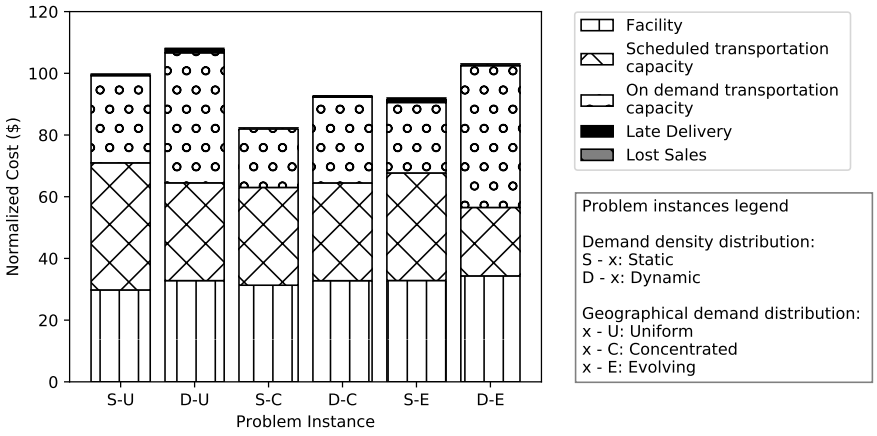
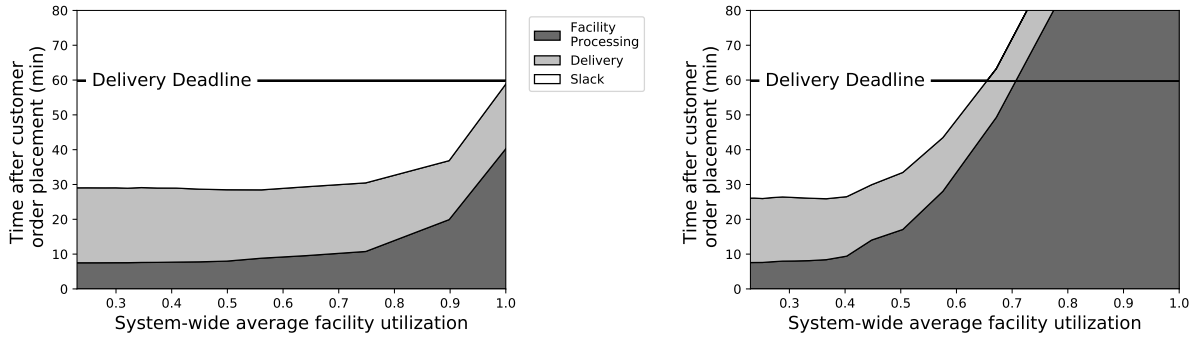


Figure 5: Network cost broken down in major components (normalized)

section, we show how an increase in network utilization drives an increase in late delivery and a reduction in potential for consolidation. Next, we analyze a breakdown of the network cost to show that the network performance is more sensitive to undercapacity than to overcapacity and to quantify the fixed vs. variable cost trade-off. We illustrate our discussion based on results for the S-C and D-C problem instances. However, the insights are derived from an analysis of all problem instances for which results can be found in Appendix Appendix F. For this analysis, the activated facility locations are fixed to the locations proposed by MetaAHA+ in Section 5.2. To analyze the effect of network utilization, we optimally allocate a gradually increasing level of parallel processing capacity in the network over the active facilities. Furthermore, we fix the scheduled transportation capacity to the optimal value based on the best design found by MetaAHA+.



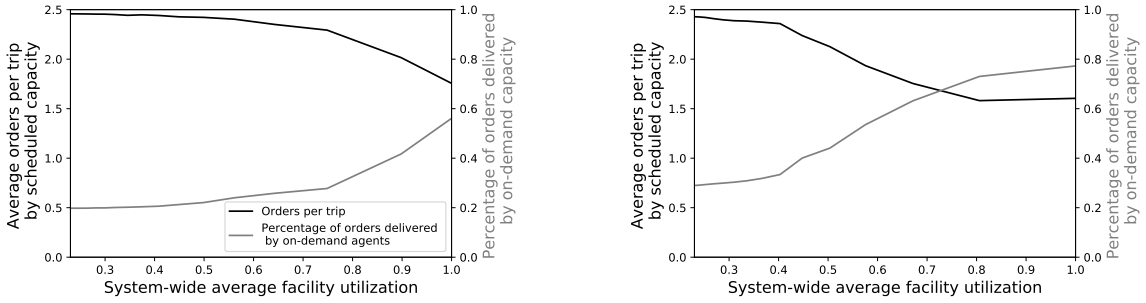
(a) Problem instance S-C

(b) Problem instance D-C

Figure 6: Average time distribution of an order with changing system-wide average facility utilization. See Figure F.14 in Appendix Appendix F for the other problem instances.

Figure 6 illustrates the relationship between the network utilization and the breakdown of the average time spent by an order from placement until arrival at the customer. As the network

utilization increases, the facility processing time increases exponentially, indicating the emergence of processing queues. Initially, an increase in network utilization does not threaten the on-time delivery, since the sum of facility processing and delivery time does not exceed the total available time until the delivery deadline. However, with an increase in utilization, the available time to wait and consolidate multiple orders, i.e., the slack time, reduces. This leads to a reduction in the average number of orders per trip by scheduled transportation agents, and a larger reliance on on-demand transportation agents (see Figures 7 and 8). Consequently, an increase in network utilization leads to an increase in transportation cost. As utilization increases further, the sum of the average facility processing and delivery time exceeds the available time until the delivery deadline, leading to additional cost due to late deliveries (see Figure 8).



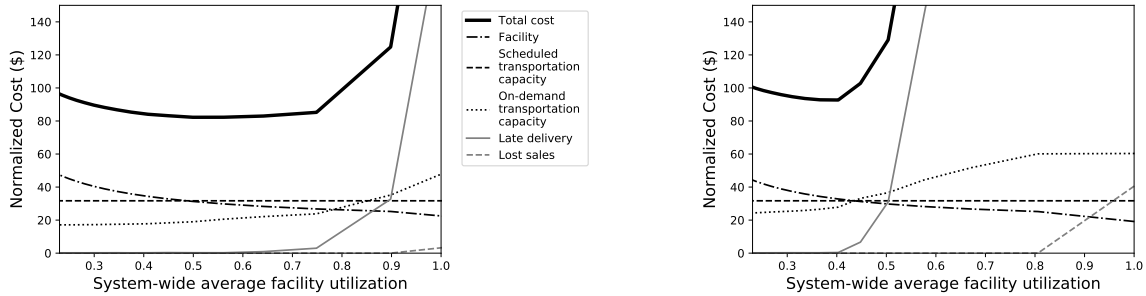
(a) Problem instance S-C

(b) Problem instance D-C

Figure 7: Average performance of transportation capacity with changing system-wide average facility utilization. See Figure F.15 in Appendix Appendix F for the other problem instances.

In addition, based on Figure 8, when deviating from the optimal network design, the expected network cost increases rapidly as network utilization increases, while it only increases slowly as utilization decreases. At the optimal network utilization level, the fixed cost dominate the total cost. Until late deliveries play a dominant role, the optimal cost curve is rather flat, in particular for utilization levels slightly lower than the optimal, as the increase in on-demand courier cost and the decrease in fixed facility cost approximately offset each other. However, as network utilization increases beyond the optimal level, late delivery cost start to dominate the total cost, causing the total network cost to grow exponentially. Thus, the network performance is more sensitive to undercapacity than to overcapacity, due to the non-linear relationship between facility processing congestion and the cost of late deliveries and additional on-demand transportation. In this study, we assume a company has limited short-term flexibility on altering the parallel processing capacity at facilities, e.g., by hiring additional employees. Therefore, the company is facing a managerial trade-off between lower fixed cost with a higher cost performance risk versus higher fixed cost with a

lower cost performance risk. In addition, Figure 8 supports the insights of Section 5.2 that problem instances with dynamic interarrival time distributions should operate at lower network utilization levels to obtain optimal performance.



(a) Problem instance S-C

(b) Problem instance D-C

Figure 8: Relationship between total network cost and individual components and system-wide average facility utilization. Note that the cost axis is bounded in these figures, for a total overview, for all problem instances, see Figure F.13 in Appendix Appendix F

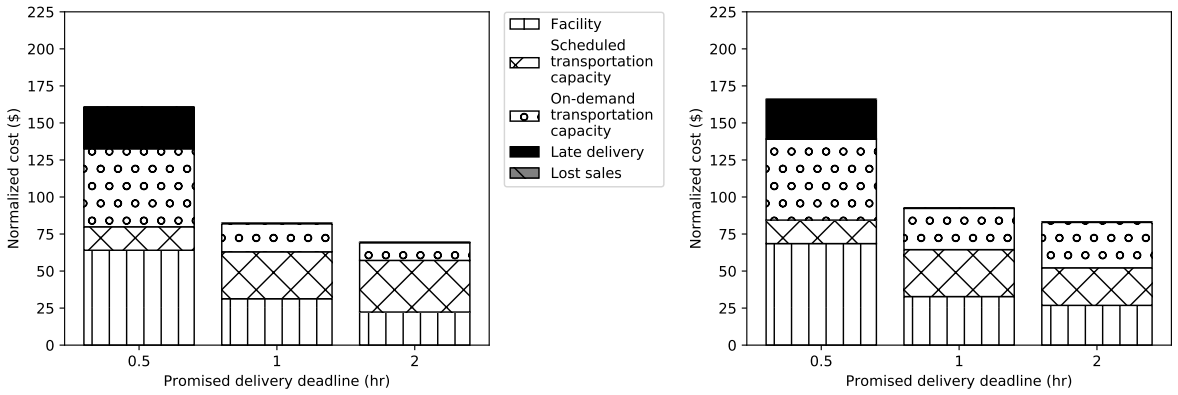
5.4. Effects of Tightness of Delivery Deadline

Figure 9 shows how the total network cost increase non-linearly with an increase in tightness of the delivery deadline. The cost increase associated with an increase in tightness from 2 to 1 hours is 15.5%, while the cost increase from 1 to 0.5 hours is 99.5%. The main drivers for the increase in cost are threefold and can be explained by the details of the average proposed network designs presented in Table 3. First, increasing tightness of the delivery deadline leads to a decrease in the potential for order consolidation, and to an increase in transportation cost. Scheduled transportation capacity is used for circa three times as many deliveries in the case of a 2-hour deadline compared to a 0.5-hour deadline, while the average scheduled transportation capacity is only circa two times as large.

Table 3: Network Design and KPIs averaged over six problem instances obtained by MetaAHA+ for various levels of delivery deadline tightness. See Table F.12 in Appendix Appendix F for disaggregate results

Deadline (hr)	**Number of Facilities**	**Parallel Processing Capacity**	**Scheduled Transportation Capacity**	**(A)**	**(B)**	**(C)**
0.5	8.3	19.8	4.8	76.5	40.8	27.2
1	3.0	9.5	11.8	36.0	74.9	61.7
2	2.0	7.8	10.2	23.2	96.5	68.6

(A) Percentage of orders served by on-demand transportation agents.
(B) Average utilization of scheduled transportation agents per trip.
(C) Utilization level at which late delivery cost constitutes 25 percent of the total cost.



(a) Problem instance S-C

(b) Problem instance D-C

Figure 9: Network cost broken down in major components for different promised delivery deadline tightness (normalized). See Figure F.16 in Appendix Appendix F for the other problem instances.

Second, increasing tightness of the delivery deadline leads to a network with additional facilities and less scheduled transportation capacity. Additional facilities ensure that the entire demand area can be served from a facility by the promised delivery deadline. In addition, having more facilities closer to demand increases the potential to consolidate orders for scheduled transportation capacity, and reduces the distance-based cost of on-demand transportation capacity.

Third, increasing tightness of the delivery deadline leads to an increase in parallel processing capacity. Tighter delivery deadlines make the network more susceptible to facility congestion and the associated cost of late delivery and reduced consolidation. The tighter the delivery deadline, the lower the system-wide average facility utilization beyond which late delivery cost become a dominant cost component. Figure 9 also reveals that it becomes optimal to accept some level of late deliveries rather than planning to fulfill all demand when delivery deadlines become extremely tight. In Section 5.3, we identified that facility congestion also impacts the network cost through a reduction in consolidation. However, for tighter delivery deadlines, this effect becomes less pronounced, since the opportunities for consolidation are lower regardless.

5.5. Scalability

So far, our analysis focuses on the earlier introduced stylized problem instances, which are based on a real-life study in Manhattan. The area covered by the instances is 100 km^2 and the expected demand is 500 orders per day. In this section, we explore the scalability of MetaAHA+ to larger real-world instances by assessing its computational requirements, speed of convergence, and consistency. Specifically, we increase the expected demand of the stylized problems to 2,500 orders. To evaluate the effect of geographical scope, we increase the area to 400 km^2, comparable

to Denver, CO, or Vienna, Austria, and increase the number of potential facility locations to 20, while keeping the demand at 2,500 orders per day. We draw three conclusions based on the results in Table 4.

Table 4: Computational and algorithmic performance for various problem sizes

Demand (orders/ day)	**Area (km^2)**	**Gap: 150 to final (%)**	**Coefficient of variation**	**Time spent optimizing (%)**	**Time per simulation run (s)**	**Time per SP run (s)**
500	200	1.1	0.9	37.5	5.7	1170.7
2500	200	1.2	1.6	7.9	68.1	15264.7
2500	400	5.9	1.0	7.2	187.5	22007.2

First, the computational performance gets increasingly dominated by the performance of the simulation rather than the optimization as problem instance size increases. Second, the cost performance results after 150 simulation runs for the larger instances, confirm the high speed of convergence of MetaAHA+. Even though the reported instances are larger and more complex than the previously considered stylized instances, the gap in performance between the best solution after 150 simulation runs and the best solution found after algorithm termination is smaller than the gaps reported for MetaAHA and AHA in Table 1. Third, the consistency of the solutions matches the results presented in Table 2. The coefficient of variation is within or close to the range of the values found for our initial stylized problem instances (0.8% to 1.4%), and so are the gaps between the best performance and the mean performance over all individual restarts (1.0% to 2.7%).

6. Conclusion

Last-mile distribution networks with tight delivery deadlines are increasingly prevalent. However, state-of-the-art last-mile distribution network design models fail to support the strategic design of networks with tight delivery deadlines due to two main reasons: (1) the absence of a delivery cut-off time that separates the order collection period and the delivery period, and (2) the reduced time available for order handling and delivery. Therefore, we present a metamodel SO approach to solve the strategic network design problem for last-mile distribution networks with CR and tight delivery deadlines. We show that our method outperforms contemporary SO and traditional DP and SP methods based on a numerical study based on real data from a global fashion company aiming to deploy one to two-hour delivery lead-times in Manhattan. In particular, explicitly incorporating the non-linear effects of congestion of order processing at facilities on net-

work performance in our metamodel formulation enables us to achieve better final solutions, better performance under a tight computational budget, and more consistent results between algorithm restarts.

In addition, we show that last-mile distribution networks with tight delivery deadlines are susceptible to facility processing congestion in two ways. First, if the processing and delivery time of orders exceed the time available until the deadline, we observe a direct impact through late deliveries. Second, even if the processing queues do not cause the order to be delivered late, facility congestion reduces the potential to consolidate orders on delivery routes, leading to an increase in transportation cost. Furthermore, the negative impact of processing queues increases with an increase in the tightness of the delivery deadline. However, the relative impact of the reduced potential for consolidation is larger for looser delivery deadlines.

Our results indicate several additional potential avenues for future research. First, our proposed model only considers demand uncertainty. In practice, the performance of a last-mile distribution network is also influenced by other sources of uncertainty, such as travel time and order processing uncertainty. Including other sources of uncertainty into the strategic design process allows for a better understanding of their impact on the resulting design and associated network performance. Second, companies often respond to uncertainty by deploying various measures of distribution flexibility. In this study, we consider the option to outsource delivery to on-demand transportation agents. However, earlier work (see, e.g., Snoeck and Winkenbach 2020) shows the potential value of flexibly adjusting the facility capacity in response to demand uncertainty. There is a need to understand how physical distribution flexibility can be effectively deployed in distribution networks with tight delivery deadlines to actively control the system-wide average facility utilization in response to changing demand. Third, we focus on the supply side of the last-mile distribution problem by aiming to design the optimal distribution network. However, companies increasingly manage the demand side of last-mile distribution as well by deploying revenue management techniques (see, e.g., Klein et al. 2019). Further research is required to understand how managing the demand can control the order processing utilization at facilities and impact the resulting network design.

References

Aboolian, R., Berman, O., Verter, V., 2016. Maximal accessibility network design in the public sector. Transportation Science 50, 336–347.

Amaran, S., Sahinidis, N.V., Sharda, B., Bury, S.J., 2016. Simulation optimization: a review of algorithms and applications. Annals of Operations Research 240, 351–380.

Amazon, 2019. Amazon prime now. URL: https://primenow.amazon.com/learn-more. accessed September 30, 2019.

Andradóttir, S., 1998. Simulation optimization, in: Banks, J. (Ed.), Handbook of simulation: Principles, methodology, advances, applications, and practice. John Wiley & Sons, New York. chapter 9, pp. 307–333.

Bektaş, T., Crainic, T.G., Van Woensel, T., 2017. From managing urban freight to smart city logistics networks, in: Gakis, K., Pardalos, P. (Eds.), Network Design and Optimization for Smart Cities. World Scientific, Singapore, pp. 143–188.

Berman, O., Krass, D., 2015. Stochastic location models with congestion, in: Laporte, G., Nickel, S., Saldanha da Gama, F. (Eds.), Location Science. Springer International Publishing, Cham, pp. 443–486.

Bierlaire, M., 2015. Simulation and optimization: A short review. Transportation Research Part C: Emerging Technologies 55, 4 – 13.

Boccia, M., Crainic, T.G., Sforza, A., Sterle, C., 2011. Location-routing models for designing a two-echelon freight distribution system. Technical Report, CIRRELT-2011-06, Université de Montréal .

Boffey, B., Galvão, R.D., Marianov, V., 2010. Location of single-server immobile facilities subject to a loss constraint. Journal of the Operational Research Society 61, 987–999.

Colby, C., Bell, K., 2016. The on-demand economy is growing, and not just for the young and wealthy. Harvard Business Review URL: https://hbr.org/2016/04/the-on-demand-economy-is-growing-and-not-just-for-the-young-and-wealthy.

Crainic, T.G., Ricciardi, N., Storchi, G., 2004. Advanced freight transportation systems for congested urban areas. Transportation Research Part C: Emerging Technologies 12, 119–137.

Crainic, T.G., Ricciardi, N., Storchi, G., 2009. Models for evaluating and planning city logistics systems. Transportation Science 43, 432–454.

Dan, T., Marcotte, P., 2019. Competitive facility location with selfish users and queues. Operations Research 67, 479–497.

eMarketer, 2017. A brief overview of the global ecommerce market. URL: https://retail.emarketer.com/article/brief-overview-of-global-ecommerce-market/59690010ebd40005284d accessed April 21, 2018.

Farfetch, 2017. Gucci in 90 minutes. URL: https://www.farfetch.com/editorial/gucci-in-90-minutes.aspx. accessed September 30, 2019.

Fu, M.C., Andradóttir, S., Carson, J.S., Glover, F., Harrell, C.R., Ho, Y.C., Kelly, J.P., Robinson, S.M.,

2000. Integrating optimization and simulation: research and practice, in: Joines, J.A., Barton, R.R., Kang, K., Fishwick, P.A. (Eds.), Proceedings of the 2000 Winter Simulation Conference, IEEE. pp. 610–616.

Fu, M.C., Glover, F.W., April, J., 2005. Simulation optimization: a review, new developments, and applications, in: Kuhl, M.E., Steiger, N.M., Armstrong, F.B., Joines, J.A. (Eds.), Proceedings of the 2005 Winter Simulation Conference, IEEE. pp. 83–95.

Govindan, K., Fattahi, M., Keyvanshokooh, E., 2017. Supply chain network design under uncertainty: A comprehensive review and future research directions. European Journal of Operational Research 263, 108–141.

van Heeswijk, W.J., Mes, M.R., Schutten, J.M., 2019. The delivery dispatching problem with time windows for urban consolidation centers. Transportation science 53, 203–221.

Hong, L.J., Nelson, B.L., 2006. Discrete optimization via simulation using compass. Operations Research 54, 115–129.

Hong, L.J., Nelson, B.L., 2009. A brief introduction to optimization via simulation, in: Rossetti, M.D., Hill, R.R., Johansson, B., Dunkin, A., Ingalls, R.G. (Eds.), Proceedings of the 2009 Winter Simulation Conference, IEEE. pp. 75–85.

Hong, L.J., Nelson, B.L., Xu, J., 2015. Discrete optimization via simulation, in: Fu, M. (Ed.), Handbook of simulation optimization. International series in Operations Research & Management Science. Springer, New York, NY. volume 216, pp. 9–44.

Janjevic, M., Winkenbach, M., Merchán, D., 2019. Integrating collection-and-delivery points in the strategic design of urban last-mile e-commerce distribution networks. Transportation Research Part E: Logistics and Transportation Review 131, 37 – 67.

Jayaswal, S., Vidyarthi, N., 2017. Facility location under service level constraints for heterogeneous customers. Annals of Operations Research 253, 275–305.

Klapp, M.A., Erera, A.L., Toriello, A., 2018a. The dynamic dispatch waves problem for same-day delivery. European Journal of Operational Research 271, 519–534.

Klapp, M.A., Erera, A.L., Toriello, A., 2018b. The one-dimensional dynamic dispatch waves problem. Transportation Science 52, 402–415.

Klein, R., Koch, S., Steinhardt, C., Strauss, A.K., 2019. A review of revenue management: Recent generalizations and advances in industry applications. European Journal of Operational Research In Press.

Kleywegt, A.J., Shapiro, A., Homem-de-Mello, T., 2001. The sample average approximation method for stochastic discrete optimization. SIAM Journal on Optimization 12, 479–502.

Lavenir, X., 2019. The Strategic Design and Environmental Footprint of Highly Responsive Urban Distribution Networks. Master's thesis. Massachusetts Institute of Technology, Cambridge, MA.

Lim, S.F.W., Winkenbach, M., 2019. Configuring the last-mile in business-to-consumer e-retailing. California Management Review 61, 132–154.

Little, J.D., 1961. A proof for the queuing formula: L = λw. Operations Research 9, 383 – 387.

Maggioni, F., Potra, F.A., Bertocchi, M., 2017. A scenario-based framework for supply planning under uncertainty: stochastic programming versus robust optimization approaches. Computational Management Science 14, 5–44.

MediaMarkt, 2020. Entrega inmediata en 2 horas. URL: `https://specials.mediamarkt.es/entrega-inmediata-2-horas`. accessed January 15, 2020.

Merchán, D., Winkenbach, M., 2018. High-resolution last-mile network design, in: Taniguchi, E., Thompson, R.G. (Eds.), City Logistics 3: Towards Sustainable and Livable Cities. ISTE, London. chapter 11, pp. 201–214.

Merchán, D., Winkenbach, M., Snoeck, A., 2020. Quantifying the impact of urban road networks on the efficiency of local trips. Transportation Research Part A: Policy and Practice 135, 38–62.

Nelson, B.L., 2014. Optimization via simulation over discrete decision variables, in: INFORMS TutORials in Operations Research, pp. 193–207.

Osorio, C., Bierlaire, M., 2013. A simulation-based optimization framework for urban transportation problems. Operations Research 61, 1333–1345.

Osorio, C., Chong, L., 2015. A computationally efficient simulation-based optimization algorithm for large-scale urban transportation problems. Transportation Science 49, 623–636.

Osorio, C., Nanduri, K., 2015. Energy-efficient urban traffic management: a microscopic simulation-based approach. Transportation Science 49, 637–651.

Pishvaee, M.S., Rabbani, M., Torabi, S.A., 2011. A robust optimization approach to closed-loop supply chain network design under uncertainty. Applied Mathematical Modelling 35, 637–649.

Powell, W.B., 2014. Clearing the jungle of stochastic optimization, in: INFORMS TutORials in Operations Research. Informs, pp. 109–137.

Powell, W.B., 2019. A unified framework for stochastic optimization. European Journal of Operational Research 275, 795–821.

Prodhon, C., Prins, C., 2014. A survey of recent research on location-routing problems. European Journal of Operational Research 238, 1–17.

Salhi, S., Rand, G.K., 1989. The effect of ignoring routes when locating depots. European Journal of Operational Research 39, 150–156.

Santoso, T., Ahmed, S., Goetschalckx, M., Shapiro, A., 2005. A stochastic programming approach for supply chain network design under uncertainty. European Journal of Operational Research 167, 96–115.

Savelsbergh, M., Van Woensel, T., 2016. 50th anniversary invited article-—city logistics: Challenges and opportunities. Transportation Science 50, 579–590.

Schneider, M., Drexl, M., 2017. A survey of the standard location-routing problem. Annals of Operations Research 259, 389–414.

Schneider, M., Löffler, M., 2019. Large composite neighborhoods for the capacitated location-routing problem. Transportation Science 53, 301–318.

Schön, C., Saini, P., 2018. Market-oriented service network design when demand is sensitive to congestion. Transportation Science 52, 1253–1275.

Schütz, P., Tomasgard, A., Ahmed, S., 2009. Supply chain design under uncertainty using sample average approximation and dual decomposition. European Journal of Operational Research 199, 409–419.

Shapiro, A., 2003. Monte carlo sampling methods, Elsevier, Amsterdam. volume 10 of *Handbooks in Operations Research and Management Science*, pp. 353 – 425.

Singleton, A.D., Spielman, S., Folch, D., 2018. Urban analytics. SAGE, London.

Snoeck, A., Udenio, M., Fransoo, J.C., 2019. A stochastic program to evaluate disruption mitigation investments in the supply chain. European Journal of Operational Research 274, 516 – 530.

Snoeck, A., Winkenbach, M., 2020. The value of physical distribution flexibility in serving dense and uncertain urban markets. Transportation Research Part A: Policy and Practice 136, 151–177.

Søndergaard, J., 2003. Optimization using surrogate models - by the space mapping technique. Ph.D. thesis. Technial University of Denmark, Kgs. Lyngby, Denmark.

Stroh, A.M., Erera, A.L., Toriello, A., 2019. Tactical design of same-day delivery systems. Georgia Institute of Technology Working Paper.

Tekin, E., Sabuncuoglu, I., 2004. Simulation optimization: A comprehensive review on theory and applications. IIE transactions 36, 1067–1081.

Ulmer, M., 2017. Delivery deadlines in same-day delivery. Logistics Research 10, 1–15.

Ulmer, M.W., Thomas, B.W., 2018. Same-day delivery with heterogeneous fleets of drones and vehicles. Networks 72, 475–505.

Ulmer, M.W., Thomas, B.W., Mattfeld, D.C., 2019. Preemptive depot returns for dynamic same-day delivery. EURO Journal on Transportation and Logistics 8, 1–35.

Vidyarthi, N., Elhedhli, S., Jewkes, E., 2009. Response time reduction in make-to-order and assemble-to-order supply chain design. IIE transactions 41, 448–466.

Vidyarthi, N., Jayaswal, S., 2014. Efficient solution of a class of location–allocation problems with stochastic demand and congestion. Computers & Operations Research 48, 20–30.

Voccia, S.A., Campbell, A.M., Thomas, B.W., 2019. The same-day delivery problem for online purchases. Transportation Science 53, 167–184.

Winkenbach, M., Kleindorfer, P.R., Spinler, S., 2016a. Enabling urban logistics services at La Poste through multi-echelon location-routing. Transportation Science 50, 520–540.

Winkenbach, M., Roset, A., Spinler, S., 2016b. Strategic redesign of urban mail and parcel networks at La Poste. Interfaces 46, 445–458.

Xu, J., Nelson, B.L., Hong, L.J., 2010. Industrial strength compass: A comprehensive algorithm and software for optimization via simulation. ACM Transactions on Modeling and Computer Simulation (TOMACS) 20, 1–29.

Xu, J., Nelson, B.L., Hong, L.J., 2013. An adaptive hyperbox algorithm for high-dimensional discrete optimization via simulation problems. INFORMS Journal on Computing 25, 133–146.

Zebra Technologies, 2018. Reinventing the supply chain: the future of fulfillment vision study. Technical Report.

Zhou, T., Osorio, C., Fields, E., 2019. Large-scale data-driven simulation-based car-sharing network design. MIT working paper.

Appendix A. Notation

In this section, we summarize the notation used throughout this paper. Note that most parameters and decision variables are defined for indices (e.g., for pixel i, or facility f). If we drop (a subset of) the indices and bold the symbol, we refer to the vector of parameters or decision variables, respectively.

Table A.5: Notation: Decision variables

a_f	binary variable indicating whether a facility at location f is activated
e_f	parallel order processing capacity at facility f
q_v^t	quantity of scheduled transportation agents of type v
q_{vt}^o	quantity of on-demand transportation agents type v in time period t
u_j	binary variable to indicate if network utilization is in interval j
$\mathbf{x}$	vector containing operational decision variables
x_{ifvt}	fraction of pixel i served from facility f by transportation agent type v in time period t
$\mathbf{y}$	vector containing all strategic decision variables

Table A.6: Notation: Sets

$\mathcal{F}$	set of potential facility locations
$\mathcal{I}$	set of pixels
$\mathcal{I}_{vft}$	set of pixels within reach of facility f within l using vehicle type v in time period t
$\mathcal{S}_\omega$	the set of realizations for the uncertain exogenous parameters in scenario ω
$\mathcal{T}$	set of time periods
$\mathcal{V}$	set of transportation agent types, where $\mathcal{V}^o$ and $\mathcal{V}^t$ are the sets of on-demand and scheduled transportation agents types

Appendix B. Simulator

In this section, we supplement the high-level overview of the simulator introduced in Section 3.5 by explaining the main components and decisions of the simulation model. Besides, we refer to the relevant Algorithms and Figures in Lavenir (2019) that provide a detailed description of the simulator.

Lavenir (2019) identifies the life cycle of an e-commerce order as the basis to define design requirements for the simulator (Lavenir 2019, Figure 3.4). The key phases relevant for our application include order placement, order allocation, facility processing, courier processing, and delivery.

Table A.7: Notation: Metamodel and network parameters

c_f^e	cost for one employee at facility f
c_v^t	operational cost of a scheduled transportation agents of type v
c_v^o	cost of summoning an on-demand transportation agent of type v
c_v^d	distance based cost for a transportation agent of type v
c^{ls}	lost sales cost per order
d_{if}	travel distance between pixel i and facility f
$e_f^{\max}$	maximum number of employees at facility f
$f_{ifvt}(t')$	time transportation agents of type v spend in period t on orders placed in time period t' and delivered from facility f to pixel i, see Equation (C.2).
k_{ifvt}	consolidation factor that approximates the effect of consolidating multiple orders into one trip, see Equation (C.5).
K_f^f	daily facility fixed cost for facility f
l	promised lead-time, i.e., available time to deliver order after customer request
$Q_v^{o,\max}$	maximum nr. of on-demand transportation agents of type v that can be summoned per unit of time
$Q_v^{t,\max}$	maximum number of hireable scheduled transportation agents of type v
S_s	state of the last-mile distribution system at time s
R_s, I_s, K_s	physical, information, and knowledge states of the system at time s
T	length of service period
t_{ifvt}^{o}	time required for transportation agent of type v to serve pixel i from facility f in period t
t_{ifvt}^{d}	minimum time required to delivery an order in pixel i from facility f using a transportation agent of type v in time period t
W_s	random variable that captures the exogenous information that becomes available at time s
α_k	correction parameter on the physical component of the metamodel
$\beta_{k,0}$	constant correction parameter in functional component of metamodel
β_k^y, β_k^ρ	correction parameters on decision variables and utilization indicators in functional component of metamodel
$\hat{\alpha}_k, \hat{\beta}_k$	correction parameters of alternative metamodel formulation to support fitting the non-linear relationship between network utilization and network cost
Δ_t	length of period t
γ_{it}	quantity of orders in pixel i at time period t
ξ_v^c	carrying capacity of a transportation agent of type v
ξ_f^h	handling capacity per unit of time of parallel processing capacity at facility f
τ_c	order arrival time of customer request c
ϕ_c	delivery location associated to customer request c
π	operational decision making policy
ρ	systemwide facility processing utilization of the distribution network
ρ_{tk}	systemwide facility processing utilization in period t of current iterate at iteration k

Order Placement - Demand Generation.. Consumer demand is specified and a deterministic input to the simulator for every scenario. This means that location, time, promised delivery, and deadline

Table A.8: Notation: Definition of MetaAHA+ (Algorithm 2)

$\mathcal{A}_k(\mathbf{y})$	number of additional simulations for solution $\mathbf{y}$ in iteration k
$G(\mathbf{y})$	the average simulation performance of solution $\mathbf{y}$
$\mathcal{H}(k)$	hyperbox at iteration k
k	current iteration
$\mathcal{L}$	set of sampled solutions
$\mathcal{L}(k)$	set of sampled solutions in iteration k
$l_k^{(d)}$	lower bound of the hyperbox for coordinate d at iteration k of the algorithm
$N_k(\mathbf{y})$	total number of simulations for solution $\mathbf{y}$ until iteration k
$u_k^{(d)}$	lower bound of the hyperbox for coordinate d at iteration k of the algorithm
w_0	base weight to ensure full rank matrix in fitting of metamodel
$w_k(\mathbf{y})$	weight of solution y in iteration k in fitting of metamodel
$\mathbf{y}_k^*$	current iterate at iteration k, i.e., best solution until iteration k
$\mathbf{y}_k^{\text{meta-}\rho-}$	solution to the metamodel problem in iteration k with upper bound on network and facility utilization
$\mathbf{y}_k^{\text{meta-}\rho+}$	solution to the metamodel problem in iteration k with lower bound on network and facility utilization
$\mathbf{y}_k^{\text{meta-hyper}}$	solution to the hyperbox constrained metamodel problem in iteration k
Ω	feasible solution space

for every customer request are fully known. However, the demand generation module dynamically reveals demand to the system to mimic the dynamic nature of arriving orders (Lavenir 2019, Algorithm 1).

Order Allocation.. The order allocation module takes the role of dispatcher and decides how to serve incoming customer requests. The module is triggered when a new request arrives, or when a new circumstance has arisen that might make it possible to assign previously unassigned jobs, e.g., when a scheduled transportation agent is activated (Lavenir 2019, Figure B-3). First, it checks the feasibility of a customer request based on the available inventory and the existing facility processing times, including picking queues. If inventory is available, and the request can be delivered before the end of the service time, a job is created. Second, the order allocator attempts to assign each newly created job to a facility and a courier in sequential order:

1. The allocator attempts to consolidate the job with an existing trip, i.e., an existing planned route to be executed by a transportation agent (Lavenir 2019, Algorithm 3). A job can be consolidated if, post-consolidation, i) inventory is available at the facility, ii) the transportation agent has remaining capacity, iii) all jobs belonging to a trip can be served before their internal delivery deadlines, and iv) jobs on all future trips assigned to a transportation agent

can be served before their internal delivery deadline. Trips with earlier latest departure times, i.e., the time a courier needs to depart to ensure all jobs are delivered by their internal delivery deadline, have precedence, since future incoming jobs have a higher probability of being consolidated with trips that depart later.

2. If a job cannot be consolidated, the module finds the facility-courier combination that can serve the job at minimum cost in the shortest time. If a facility-courier combination exists that can serve the job before the end of the service time, a new trip is created, and other jobs can be consolidated on the trip until its latest departure time (Lavenir 2019, Figure 3.5, Algorithm 4).

3. If no facility-courier combination exists that can serve the job before the end of the service time, it is not assigned.

Facility Processing.. Once a job is allocated to a facility, it is picked first-come, first-served. The facility processing module assigns the first job in the queue to the first employee that becomes available (Lavenir 2019, Figure 3.6).

Courier processing.. Transportation agents are guided by trips. The execution of a trip consists of four steps (Lavenir 2019, Figure B.2). First, the transportation agent travels to the facility. If the agent is of the on-demand type, it is newly generated. Second, it waits for all jobs to be picked and loaded. Third, if the transportation agent has no remaining capacity or the latest departure time is reached, the agent starts to deliver the jobs on the trip. Fourth, if the trip is executed by an on-demand agent, once it is finished, the agent disappears from the system. If the trip is executed by a scheduled agent, the agent starts its next trip, or it travels to the closest facility if no trips are assigned yet.

Appendix C. Supporting Parameters

In this section, we define two auxiliary variables used in the model defined by Equations (13) through (24). In particular, we define the scheduled transportation capacity overflow function $f_{ifvt}(t')$ and the consolidation parameter k_{ifvt}.

Appendix C.1. Scheduled transportation capacity overflow

The scheduled transportation capacity overflow variable ensures that agents that start a delivery in one period do not suddenly finish as soon as the period finishes, their work carries over into the

next period(s). More precisely, we consider the time an agent spends in the subsequent periods after starting a delivery in a certain period. We define $\tau_{t't}$ as the time that has passed since the start of period t' and the start of period t,

$$\tau_{t't} = \sum_{j=t'}^{t-1} \Delta_t. \tag{C.1}$$

We can define nine different cases (A to I, see Figure C.10) with potential overflow of scheduled courier capacity from period t' into period t, that we categorize into two broader categories.

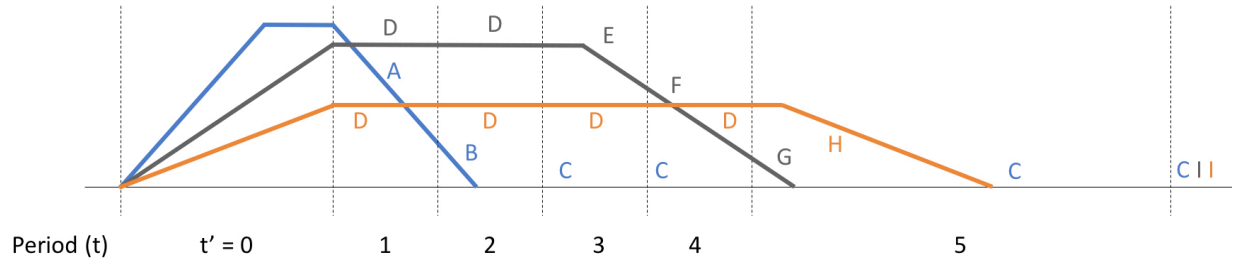


Figure C.10: Courier Overflow

First, the time a transportation agent spends on one order is smaller than the length of time period t', therefore, agents start to free up as soon as t' is finished with rate $\gamma_{it'}\Delta_t^{-1}$ (order density in orders/hr). The maximum number of transportation agents deployed simultaneously to deliver orders from time period t' is $\gamma_{it'}t^s_{ifv}$.

(A) At the beginning of time period t, transportation agents are finishing their job, but not all agents are finished by the end of t ($t_{ifvt'} \leq \Delta_{t'}$ and $\tau_{(t'+1)(t+1)} \leq t_{ifvt'}$). Then,

$$H_b = \gamma_{it'}\Delta_t^{-1}(t_{ifvt'} - \tau_{(t'+1)(t)}), \quad H_e = \gamma_{it'}\Delta_t^{-1}(t_{ifvt'} - \tau_{(t'+1)(t+1)}), \quad f^A_{ifvt} = 0.5\Delta_t(H_b - H_e).$$

(B) At the beginning of time period t, agents are finishing their job, and at some point in period t, all agents are done with their delivery ($t_{ifvt'} \leq \Delta_{t'}$ and $\tau_{(t'+1)(t)} \leq t_{ifvt'} \leq \tau_{(t'+1)(t+1)}$). Then,

$$H_b = \gamma_{it'}\Delta_t^{-1}(t_{ifvt'} - \tau_{(t'+1)(t)}), \quad H_h = t_{ifvt'} - \tau_{(t'+1)(t)}, \quad f^B_{ifvt} = 0.5H_bH_h.$$

(C) All agents are done delivering items from period t' in period t ($t_{ifvt'} \leq \Delta_{t'}$ and $t_{ifvt'} \leq$

$\tau_{(t'+1)(t)}$). Then,

$$f^C_{ifvt} = 0.$$

Second, the time agents spend on one order ($t_{ifvt'}$ is larger than the length of time period t', therefore, agents start to free up after $t_{ifvt'}$ with rate $\gamma_{it'}$. The maximum number of agents deployed simultaneously to deliver orders from time period t' is $\gamma_{it'}\Delta_{t'}$.

(D) All throughout period t, agents are busy delivering orders from period t' ($t_{ifvt'} \geq \Delta_{t'}$ and $\tau_{t'(t+1)} \leq t_{ifvt'}$). Then,

$$f^D_{ifvt} = \gamma_{it'}\Delta_t.$$

(E) At the beginning of time period t, all agents are still busy delivering orders from period t', but somewhere in period t, the first agents start finishing up. However, by the end of period t, not all agents are finished yet ($t_{ifvt'} \leq \Delta_{t'}$ and $\tau_{t'(t)} \leq t_{ifvt'} \leq \tau_{t'(t+1)} \leq t_{ifvt'} + \Delta_{t'}$). Then,

$$S = \gamma_{it'}(t_{ifvt'} - \tau_{t'(t)}), \quad H_b = \gamma_{it'}, \quad H_e = \gamma_{it'} - \gamma_{it'}\Delta_{t'}^{-1}(\tau_{t'(t+1)} - t_{ifvt'}),$$
$$f^E_{ifvt} = 0.5(H_b + H_e)(\tau_{t'(t+1)} - t_{ifvt'}) + S.$$

(F) At the beginning of time period t, agents are finishing their job, but not all agents are finished by the end of t ($t_{ifvt'} \leq \Delta_{t'}$ and $t_{ifvt'} \leq \tau_{t'(t)} \leq \tau_{t'(t+1)} \leq t_{ifvt'} + \Delta_{t'}$). Then,

$$H_b = \gamma_{it'} - \gamma_{it'}\Delta_{t'}^{-1}(\tau_{t'(t)} - t_{ifvt'}), \quad H_e = \gamma_{it'} - \gamma_{it'}\Delta_{t'}^{-1}(\tau_{t'(t+1)} - t_{ifvt'}), \quad f^F_{ifvt} = 0.5(H_b + H_e)\Delta_t.$$

(G) At the beginning of time period t, agents are finishing their job, and at some point in period t, all agents are done with their delivery ($t_{ifvt'} \leq \Delta_{t'}$ and $t_{ifvt'} \leq \tau_{t'(t)} \leq t_{ifvt'} + \Delta_{t'} \leq \tau_{t'(t+1)}$). Then,

$$H_b = \gamma_{it'} - \gamma_{it'}\Delta_{t'}^{-1}(\tau_{t'(t)} - t_{ifvt'}), \quad f^G_{ifvt} = 0.5H_b(t_{ifvt'} + \Delta_{t'} - (\tau_{t'(t)} - t_{ifvt'})).$$

(H) At the beginning of time period t, all agents are still busy delivering orders from period t', but somewhere in period t, the first agents start finishing up. By the end of period t, all

agents are finished ($t_{ifvt'} \leq \Delta_{t'}$ and $\tau_{t'(t)} \leq t_{ifvt'} \leq t_{ifvt'} + \Delta_{t'} \leq \tau_{t'(t+1)}$). Then,

$$S = \gamma_{it'}(t_{ifvt'} - \tau_{t'(t)}), \quad H_b = \gamma_{it'}, \quad f^H_{ifvt} = 0.5 H_b \Delta_{t'} + S.$$

(I) All agents are done delivering items from period t' in period t ($t_{ifvt'} \leq \Delta_{t'}$ and $t_{ifvt'} + \Delta_{t'} \leq \tau_{t't}$). Then,

$$f^I_{ifvt} = 0.$$

Integrating the cases presented above, the formulation for $f_{ifvt}(t')$ leads to

$$f_{ifvt}(t') = \begin{cases} f^A_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } \tau_{(t'+1)(t+1)} \leq t_{ifvt'}, \\ f^B_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } \tau_{(t'+1)(t)} \leq t_{ifvt'} \leq \tau_{(t'+1)(t+1)}, \\ f^C_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } t_{ifvt'} \leq \tau_{(t'+1)(t)}, \\ f^D_{ifvt} & \text{for } t_{ifvt'} \geq \Delta_{t'} \text{ and } \tau_{t'(t+1)} \leq t_{ifvt'}, \\ f^E_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } \tau_{t'(t)} \leq t_{ifvt'} \leq \tau_{t'(t+1)} \leq t_{ifvt'} + \Delta_{t'}, \\ f^F_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } t_{ifvt'} \leq \tau_{t'(t)} \leq \tau_{t'(t+1)} \leq t_{ifvt'} + \Delta_{t'}, \\ f^G_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } t_{ifvt'} \leq \tau_{t'(t)} \leq t_{ifvt'} + \Delta_{t'} \leq \tau_{t'(t+1)}, \\ f^H_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } \tau_{t'(t)} \leq t_{ifvt'} \leq t_{ifvt'} + \Delta_{t'} \leq \tau_{t'(t+1)}, \\ f^I_{ifvt} & \text{for } t_{ifvt'} \leq \Delta_{t'} \text{ and } t_{ifvt'} + \Delta_{t'} \leq \tau_{t't}. \end{cases} \tag{C.2}$$

Appendix C.2. Consolidation factor

To account for the reduction in transportation capacity requirements trough consolidation (i.e., assigning multiple orders to one transportation agent), we introduce a consolidation factor k_{ifvt} in the model defined by Equations (13) through (24). To approximate the effect of consolidation in pixel i, we consider the available 'slack' a courier has within the available time until the delivery deadline when delivering from facility f to pixel i, t^s_{ifvt}, and the potential consolidation density in orders per hour in pixel i and time period t, γ^c_{it}. We can compute t^s_{ifvt} by subtracting the time required to deliver the order (t^d_{ifvt}), including picking, courier response, traveling and loading time,

from the promised delivery lead-time (l) as

$$t^s_{ifvt} = l - t^d_{ifvt}. \tag{C.3}$$

Furthermore, we can compute γ^c_{it} by defining the Neighborhood of a pixel i, $\mathcal{N}(i)$ based on the maximum pixel-to-pixel consolidation distance d^c, and computing the order density in the neighborhood of pixel i as

$$\mathcal{N}(i) = \{i' | d_{ii'} \leq d^c\}, \quad \gamma^c_{it} = \sum_{i' \in \mathcal{N}(i)} \sum_{s \in \mathcal{S}} \frac{\gamma_{it}}{\Delta_t}. \tag{C.4}$$

The maximum consolidation factor, i.e., the proportion of original trips required with consolidation, is the maximum of the inverse of the carrying capacity of a vehicle (ξ_v) and a function of the density of orders arriving during the 'slack' time. We formally define the consolidation factor as

$$k_{ifvt} = \max(\frac{1}{\xi_v}, \min(1, \frac{1}{t^s_{ifvt} * \gamma^c_{it}})). \tag{C.5}$$

Appendix D. Problem Instance Definition

This appendix supports the introduction of the problem instances that leveraged for our analysis Section 4.

Appendix D.1. Stylized Problem Instances

Most parameter values in the stylized problem instances are equal to those of the actual case study, e.g., cost and capacity. However, we explicitly control demand distributions via the systemwide demand density and the geographic demand distributions. Furthermore, we generate artificial potential facility locations.

Systemwide demand density distributions.. We define to types of systemwide demand density distribution, *stationary* (S) and *dynamic* (D). In both cases, we generate order interarrival times based on the exponential distribution. In the *stationary* (S) case, the demand level distribution is governed by the exponential parameter λ. In the dynamic case, the parameter that governs the interarrival distribution is time-dependent, λ_t. Furthermore, we define benchmark parameters,

λ^{high} and λ^{low}. Given those bounds, we define λ_t as

$$\lambda_t = \begin{cases} \lambda^{\text{low}}, & t \in [0, t_1], \\ \frac{\lambda^{\text{high}}, -\lambda^{\text{low}}}{t_2 - t_1}(t - t_1) & t \in [t_1, t_2], \\ \lambda^{\text{high}}, & t \in [t_2, t_3], \\ \frac{\lambda^{\text{low}}, -\lambda^{\text{high}}}{t_4 - t_3}(t - t_3) & t \in [t_3, t_4], \\ \lambda^{\text{low}}, & t \in [t_4, t_{\text{end}}]. \end{cases} \tag{D.1}$$

When choosing the demand parameters, we ensure that the expected demand is equal for both the *stationary* (S) and *dynamic* (D) cases.

Geographic demand distribution.. We define three types of geographic distributions, *uniform* (U), *concentrated* (C), and *dynamic* (D). In the *uniform* (U) case, demand is uniformly distributed over the demand area. For both the *concentrated* and *dynamic* cases, we define a parameter ζ, to indicate the probability that an order belongs to a demand cluster. Consequently, with probability $1-\zeta$, an order does not belong to the demand cluster and is uniformly distributed over the demand area. In the *concentrated* (C) case, an order assigned to the cluster is randomly located in a circle with centroid (x^c, y^c) and radius r. Similarly, in the *dynamic* (D) case, an order is assigned to a circle with radius r, but the center of the circle depends on the time (x_t^c, y_t^c). The center of the circle moves linearly over time from (x_0^c, y_0^c) to $(x_{\text{end}}^c, y_{\text{end}}^c)$.

Facility generation algorithm.. Since the stylized problem instances do not have actual proposed facility locations, we generate those using Algorithm 4. Note that this is just one potential mechanism to generate the facility locations, any other could be used as well. First, we generate potential facility locations by solving a p-median problem. However, in real-life cases, potential facility locations are rarely found at the optimally suggested locations, particularly in dense urban areas. To mimic this additional real-life constraint, we add a random geographical shift to the locations proposed.

Appendix E. Model Parameters

In this section, we introduce the SP formulation and specify the parameters used to run the various algorithms discussed in the results.

Algorithm 4 Algorithm to generate facility locations for stylized problem instances

Step 1: Generate potential locations

1. Raster the demand area with dimensions X and Y into square pixels
2. Take the centroid of every pixel as demand location

Step 2: Solve p-median problem

1. Determine number (p) of potential facility locations to be included in the model
2. Solve p-median with demand locations

Step 3: Randomize locations

1. Define relative randomization as percentage z
2. For each location i with coordinates (x_i, y_i) suggested by the p-median solution
   - Generate two uniform random numbers from $U(-1, 1)$: u_x, u_y
   - Find randomized location $(x_i + zXu_x, y_i + zYu_y)$

Appendix E.1. SP formulation

We define a two-stage SP equivalent of the DP model defined in Equations (13) through (24). In the SP model, we distinguish the first stage strategic decisions, $\mathbf{a}$, $\mathbf{e}$, and $\mathbf{q}^t$, and the second stage operational decisions $\mathbf{x}$ and $\mathbf{q}^o$. The second stage decisions become scenario dependent, indicated by subscript ω. In addition, any scenario-specific realizations of uncertain parameters also become scenario dependent, again indicated by subscript ω. For example, $\gamma_{it\omega}$ refers to the demand in pixel i and time period t in scenario ω.

$$\begin{aligned}\min_{\mathbf{a},\mathbf{e},\mathbf{q}^t,\mathbf{x},\mathbf{q}^o} & \sum_{f\in\mathcal{F}}(K_f^f a_f + c_f^e e_f) + \sum_{v\in\mathcal{V}}\sum_{t\in\mathcal{T}}\Delta_t c_v^t q_v^t + \frac{1}{\omega}\sum_{\omega\in\Omega}\sum_{t\in\mathcal{T}}\sum_{v\in\mathcal{V}} c_v^o q_{vt\omega}^o \\ & + \frac{1}{\omega}\sum_{\omega\in\Omega}\sum_{t\in\mathcal{T}}\sum_{v\in\mathcal{V}}\sum_{I\in\mathcal{I}} c_v^d \sum_{f\in\mathcal{F}} d_{if} k_{ifvt\omega}(\gamma_{it\omega}) x_{ifvt\omega} \\ & + \frac{1}{\omega}\sum_{\omega\in\Omega}\sum_{i\in\mathcal{T}}\sum_{I\in\mathcal{T}} c^{ls}\gamma_{it\omega}(1 - \sum_{f\in\mathcal{F}}\sum_{v\in\mathcal{V}} x_{ifvt\omega})\end{aligned} \tag{E.1}$$

$$\text{s.t} \quad \sum_{f\in\mathcal{F}}\sum_{v\in\mathcal{V}} x_{ifvt\omega} \leq 1, \qquad i\in\mathcal{I}, t\in\mathcal{T}, \omega\in\Omega, \tag{E.2}$$

$$\sum_{i\in\mathcal{I}}\sum_{v\in\mathcal{V}} \gamma_{it\omega} x_{ifvt\omega} \leq \xi_f^h \Delta_t e_f, \qquad f\in\mathcal{F}, t\in\mathcal{T}, \omega\in\Omega, \tag{E.3}$$

$$\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}} k_{ifvt\omega}(\gamma_{it\omega}) x_{ifvt\omega}(t^o_{ifvt\omega}\gamma_{it\omega}\Delta_t - \sum_{\tau=t+1}^{T} f_{ifv\tau\omega}(t)) +$$

$$\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}}\sum_{\tau=0}^{t-1} k_{ifv\tau\omega}(\gamma_{it\omega}) x_{ifv\tau\omega} f_{ifvt\omega}(\tau) \leq q_v^t \Delta_t, \qquad v\in\mathcal{V}^t, t\in\mathcal{T}, \omega\in\Omega, \tag{E.4}$$

$$\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}} \gamma_{it\omega} x_{ifvt\omega} \leq q^o_{vt\omega}, \qquad v\in\mathcal{V}^o, t\in\mathcal{T}, \omega\in\Omega, \tag{E.5}$$

$$q^o_{vt\omega} \leq \Delta_t Q_v^{o\max}, \qquad v\in\mathcal{V}^o, t\in\mathcal{T}, \omega\in\Omega, \tag{E.6}$$

$$x_{ifvt\omega} = 0, \qquad i\notin\mathcal{I}(fvt\omega), v\in\mathcal{V}, f\in\mathcal{F}, t\in\mathcal{T}, \omega\in\Omega, \tag{E.7}$$

$$x_{ifvt\omega} \geq 0, \qquad i\in\mathcal{I}(fvt\omega), v\in\mathcal{V}, f\in\mathcal{F}, t\in\mathcal{T}, \omega\in\Omega, \tag{E.8}$$

$$a_f \in \{0,1\}, \qquad f\in\mathcal{F} \tag{E.9}$$

$$e_f \leq e_f^{\max}, \qquad f\in\mathcal{F} \tag{E.10}$$

$$q_v^t \leq q_v^{t\max}, \qquad v\in\mathcal{V} \tag{E.11}$$

$$e_f, q_v^t, q^o_{vt\omega} \in \mathbb{Z}, \qquad v\in\mathcal{V}, f\in\mathcal{F}, \omega\in\Omega. \tag{E.12}$$

The constraints in the model defined by Equations (E.1) through (E.1) are equivalent to the constraints in the DP model defined by in Equations (E.1) through (E.12). To solve the model we rely on the SAA introduced by Shapiro (2003). We solve a deterministic equivalent of the model based on 10 randomly generated scenarios.

Appendix E.2. Algorithmic parameters

In this section, we define the algorithmic parameter settings used in this study. We opted for similar parameter settings to Xu et al. (2013), Osorio and Bierlaire (2013), and Zhou et al. (2019) whenever possible. Consequently, we define the algorithmic parameters as follows.

- The total number of simulations per solution at iteration k is defined by

$$N_k(\mathbf{y}) = \min\{5, \lceil 5(\log k)^{1.01}\rceil\}, \quad \mathcal{A}_k(\mathbf{y}) = N_k(\mathbf{y}) - \mathcal{A}_{k-1}(\mathbf{y}). \tag{E.13}$$

- w_0 is set to 0.01
- We perform 10 algorithm iterations for MetaAHA+, MetaAHA, SP. We perform 3 iterations for AHA. Since DP is deterministic, we only solve the model once per problem instance.
- At every iteration, we generate 10 solutions.

- We piecewise linearize the utilization into 10 dummies, the Taylor series expansion is to the sixth polynomial.

Appendix F. Extended Results

In this section, we share the extended results eluded to in Section 5.5.

Table F.9: Cost performance of deployed solution methods for six problem instances at algorithm termination and after 150 simulation runs

		At Termination		At Early Termination		
Problem Instance	**Method**	**Cost ($)**	**Relative Gap to MetaAHA+ Solution (%)**	**Cost ($)**	**Relative Gap to Termination Solution (%)**	**Relative Gap to MetaAHA+ Solution (%)**
S-U	AHA	101.2	1.2	181.4	79.3	79.5
S-U	MetaAHA	100	0	103.3	3.3	2.2
S-U	MetaAHA+	100	0	101.1	1.1	0
S-U	DP	137.2	37.2	137.2	0	35.7
S-U	SP	101.3	1.3	101.3	0	0.2
D-U	AHA	115.4	6.8	172.6	49.6	55.1
D-U	MetaAHA	109.3	1.2	115.7	5.9	3.9
D-U	MetaAHA+	108	0	111.3	3.1	0
D-U	DP	156.8	45.1	156.8	0	40.8
D-U	SP	108.7	0.6	108.7	0	-2.4
S-C	AHA	83.3	1.1	159.9	91.8	81.3
S-C	MetaAHA	83.1	0.8	92.1	10.8	4.5
S-C	MetaAHA+	82.5	0	88.2	6.9	0
S-C	DP	144.4	75	144.4	0	63.7
S-C	SP	101.7	23.3	101.7	0	15.3
D-C	AHA	93.3	0.6	175.4	87.9	80.8
D-C	MetaAHA	93.3	0.5	110.4	18.3	13.8
D-C	MetaAHA+	92.8	0	97	4.5	0
D-C	DP	157	69.1	157	0	61.8
D-C	SP	106.5	14.7	106.5	0	9.8
S-E	AHA	94.8	1.8	152.1	60.4	57.7
S-E	MetaAHA	93.7	0.6	96.9	3.3	0.4
S-E	MetaAHA+	93.2	0	96.5	3.5	0
S-E	DP	108.3	16.3	108.3	0	12.3
S-E	SP	105.5	13.3	105.5	0	9.4
D-E	AHA	110.5	7.3	179.8	62.7	66.9
D-E	MetaAHA	103.7	0.7	115.3	11.2	7.1
D-E	MetaAHA+	103	0	107.7	4.6	0
D-E	DP	203.8	97.9	203.8	0	89.2
D-E	SP	105.8	2.7	105.8	0	-1.8

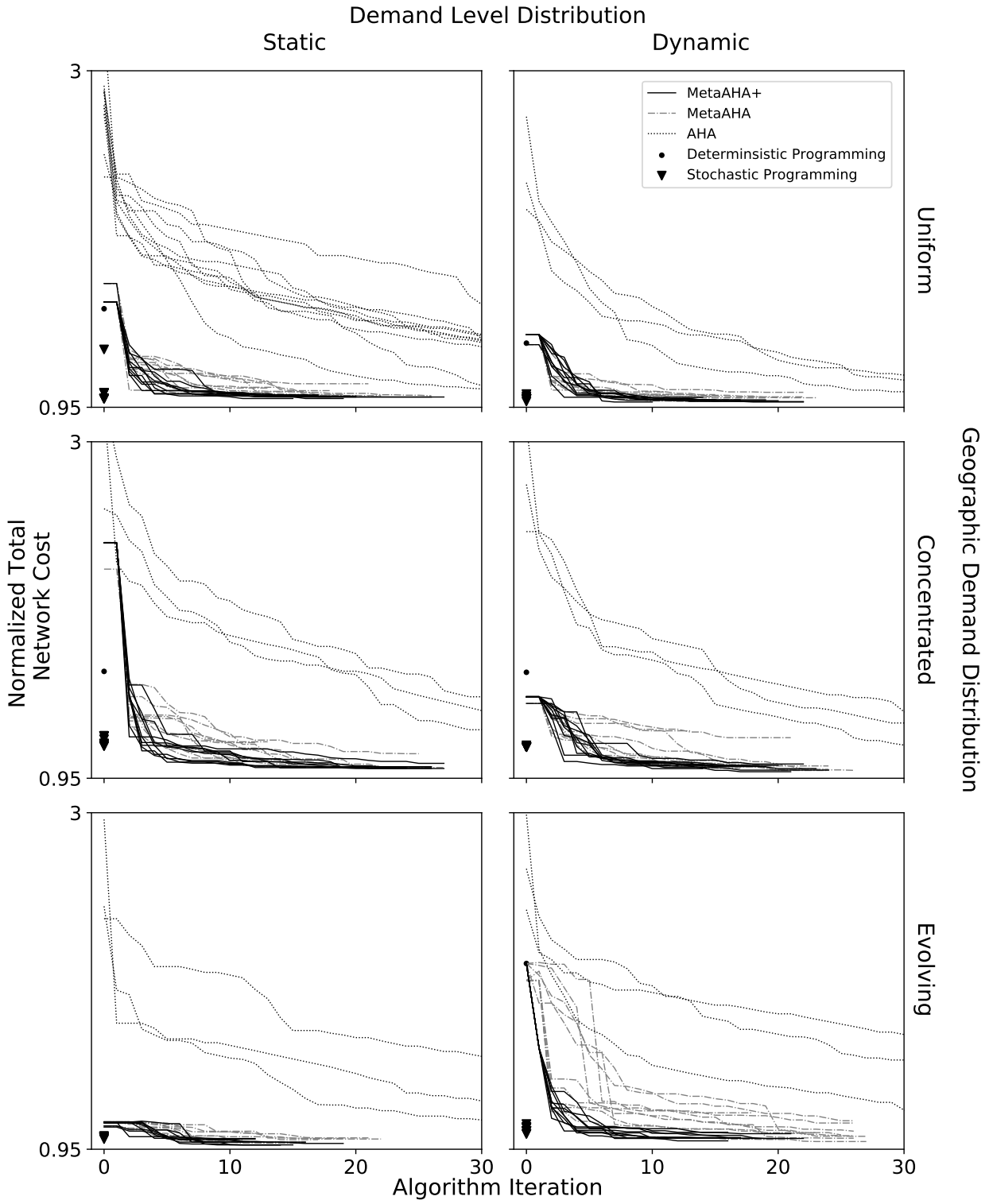


Figure F.11: Total network cost evolution for each solution method and individual run for each of the problem instances (normalized).

Table F.10: Consistency of cost performance of individual restarts of deployed solution methods for six problem instances

Problem Instance	Method	At Termination			At Early Termination		
		Mean ($)	Standard Deviation	Gap to Best Found Solution (%)	Mean ($)	Standard Deviation	Gap to Best Found Solution (%)
S-U	AHA	105.5	1.4	5.2	198.4	13.4	97.7
S-U	MetaAHA	104.2	2.6	3.8	117.8	6.3	17.4
S-U	MetaAHA+	101.5	0.9	1.2	108.8	5.1	8.4
S-U	DP	155	0	54.5	155	0	54.5
S-U	SP	104.7	8.7	4.3	104.7	8.7	4.3
D-U	AHA	111.7	1.5	5.3	183.1	8.6	72.6
D-U	MetaAHA	109.4	2	3.1	116.6	3.8	9.9
D-U	MetaAHA+	107.2	0.8	1	115.4	4.2	8.8
D-U	DP	144.8	0	36.5	144.8	0	36.5
D-U	SP	108.5	1.2	2.2	108.5	1.2	2.2
S-C	AHA	85.7	1.8	2.9	175.4	12.9	110.7
S-C	MetaAHA	87.6	4.9	5.2	106	8.9	27.3
S-C	MetaAHA+	84.1	0.9	1.1	97.3	7	16.9
S-C	DP	132	0	58.6	132	0	58.6
S-C	SP	97.2	2.1	16.8	97.2	2.1	16.8
D-C	AHA	95.1	2	3.5	183.7	8.1	100
D-C	MetaAHA	98.1	7.6	6.7	113.9	5.4	24
D-C	MetaAHA+	94.4	1.3	2.7	106.8	6.2	16.3
D-C	DP	148.1	0	61.2	148.1	0	61.2
D-C	SP	105.9	0.3	15.3	105.9	0.3	15.3
S-E	AHA	95.6	3.3	5.1	166.5	18.3	83.1
S-E	MetaAHA	93.7	1.3	3	100	3.1	9.9
S-E	MetaAHA+	92.4	1.2	1.6	98	2.6	7.7
S-E	DP	95.1	0	4.6	95.1	0	4.6
S-E	SP	95.4	0.5	4.9	95.4	0.5	4.9
D-E	AHA	110.7	1.4	7.4	200.9	16	94.9
D-E	MetaAHA	109.5	5.4	6.2	152.1	28.2	47.5
D-E	MetaAHA+	105.3	1.5	2.1	114.2	6	10.8
D-E	DP	214.4	0	107.9	214.4	0	107.9
D-E	SP	110.1	1.9	6.8	110.1	1.9	6.8

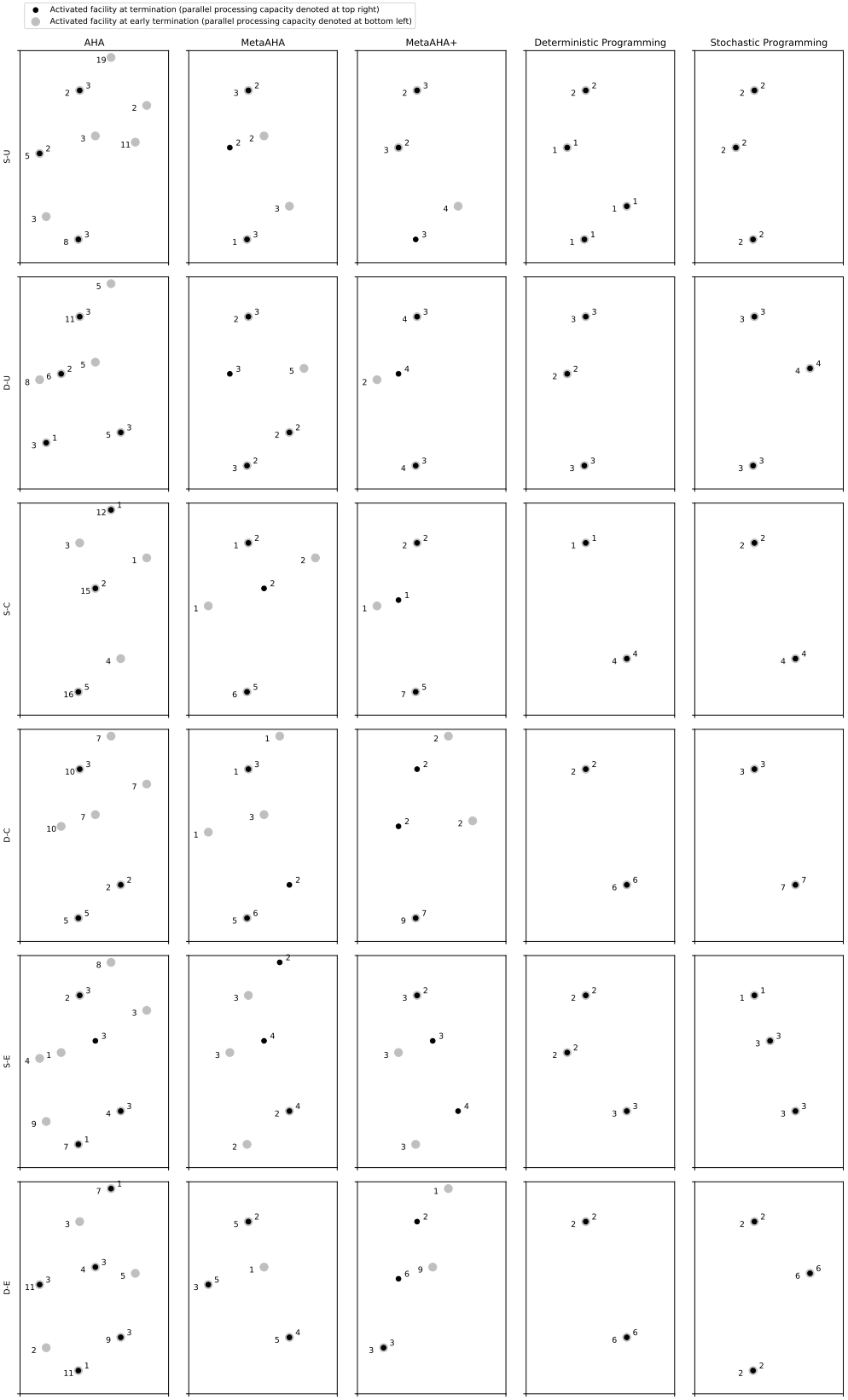
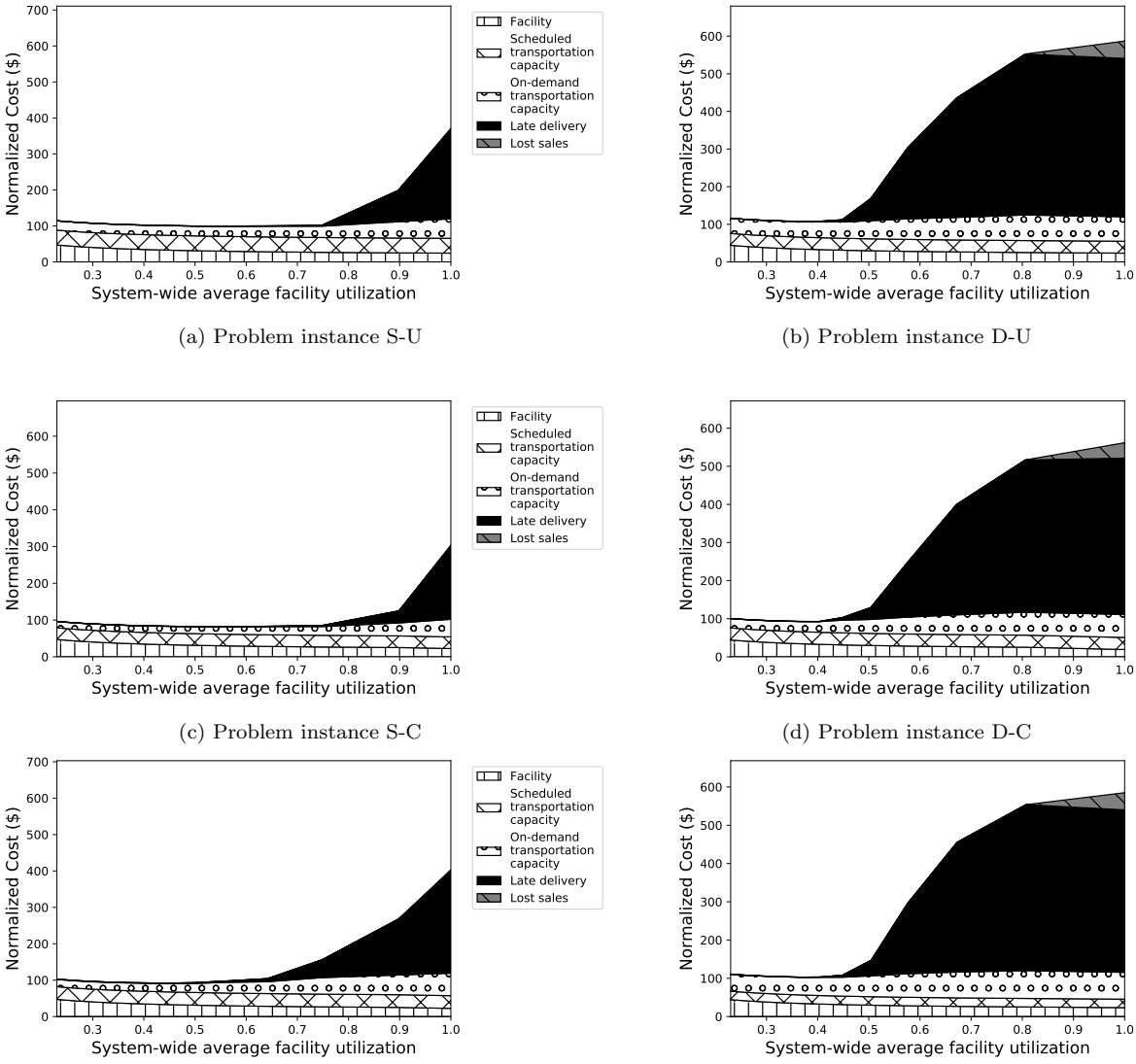


Figure F.12: Proposed network design for problem instances and solution methods at algorithm termination and after early termination at 150 simulation runs



(a) Problem instance S-U

(b) Problem instance D-U

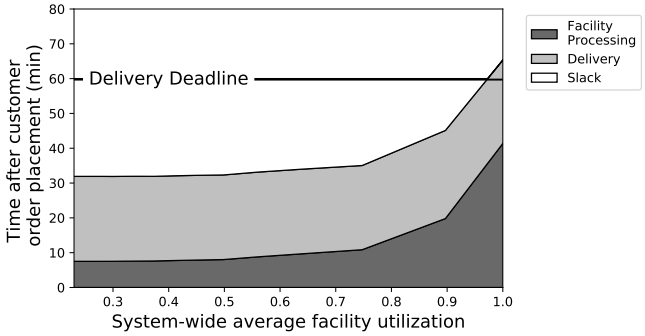
(c) Problem instance S-C

(d) Problem instance D-C

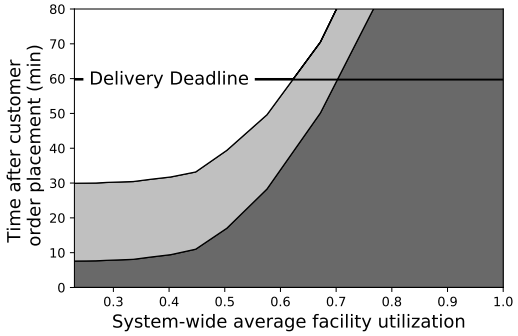
(e) Problem instance S-E

(f) Problem instance D-E

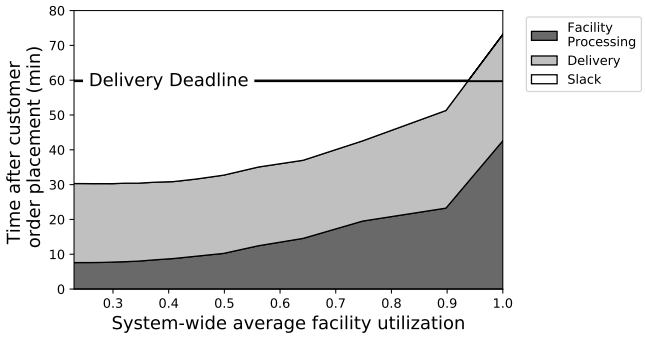
Figure F.13: Relationship between total network cost and individual components and system-wide average facility utilization.



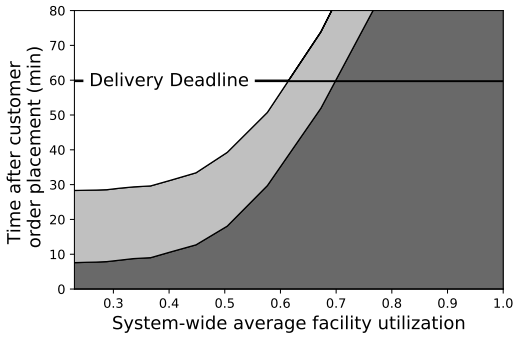
(a) Problem instance S-U



(b) Problem instance D-U



(c) Problem instance S-E



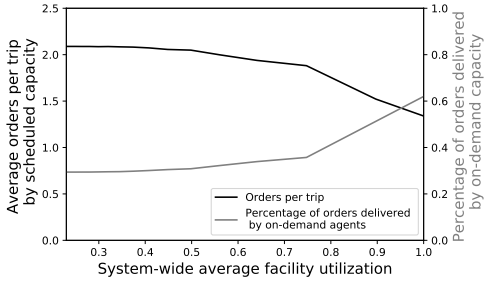
(d) Problem instance D-E

Figure F.14: Average time distribution of an order with changing system-wide average facility utilization.

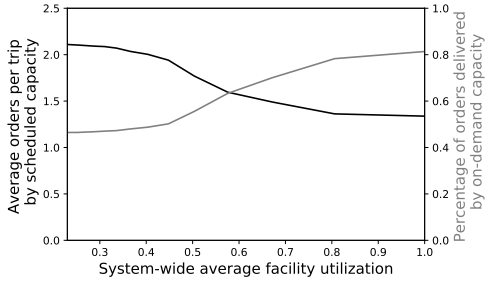
Table F.11: Proposed network design for various problem instances and solution methods at algorithm termination and after 150 simulation runs (see Figure F.12 for a geographical overview of network designs).

Prob. Inst.	Method	At Termination			At Early Termination*		
		Number of Facilities	Facility Processing Capacity	Scheduled Transp. Capacity	Number of Facilities	Facility Processing Capacity	Scheduled Transp. Capacity
S-U	MetaAHA+	3	8	13	3	9	13
S-U	MetaAHA	3	7	11	4	9	10
S-U	AHA	3	8	15	8	53	14
S-U	DP	4	5	11			
S-U	SP	3	6	11			
D-U	MetaAHA+	3	10	10	3	10	9
D-U	MetaAHA	4	10	11	4	12	9
D-U	AHA	4	9	7	7	43	5
D-U	DP	3	8	10			
D-U	SP	3	10	12			
S-C	MetaAHA+	3	8	10	3	10	14
S-C	MetaAHA	3	9	10	4	10	7
S-C	AHA	3	8	11	6	51	15
S-C	DP	2	5	10			
S-C	SP	2	6	10			
D-C	MetaAHA+	3	11	10	3	13	7
D-C	MetaAHA	3	11	10	5	11	9
D-C	AHA	3	10	7	7	48	18
D-C	DP	2	8	10			
D-C	SP	2	10	11			
S-E	MetaAHA+	3	9	11	3	9	11
S-E	MetaAHA	3	10	12	4	10	10
S-E	AHA	4	10	11	8	38	11
S-E	DP	3	7	10			
S-E	SP	3	7	10			
D-E	MetaAHA+	3	11	7	3	13	12
D-E	MetaAHA	3	11	9	4	14	12
D-E	AHA	5	11	10	8	52	7
D-E	DP	2	8	9			
D-E	SP	3	10	11			

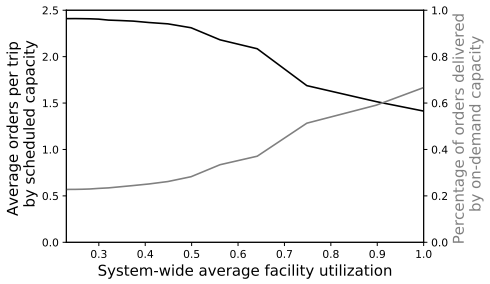
*For DP and SP the network design is determined after one iteration of solving the model, thus there are no early termination results



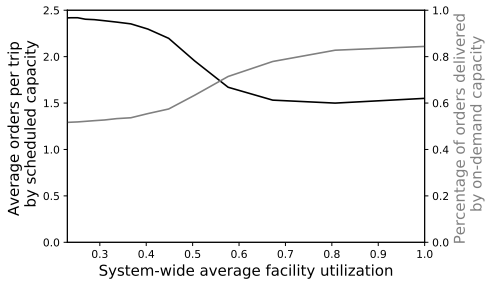
(a) Problem instance S-U



(b) Problem instance D-U

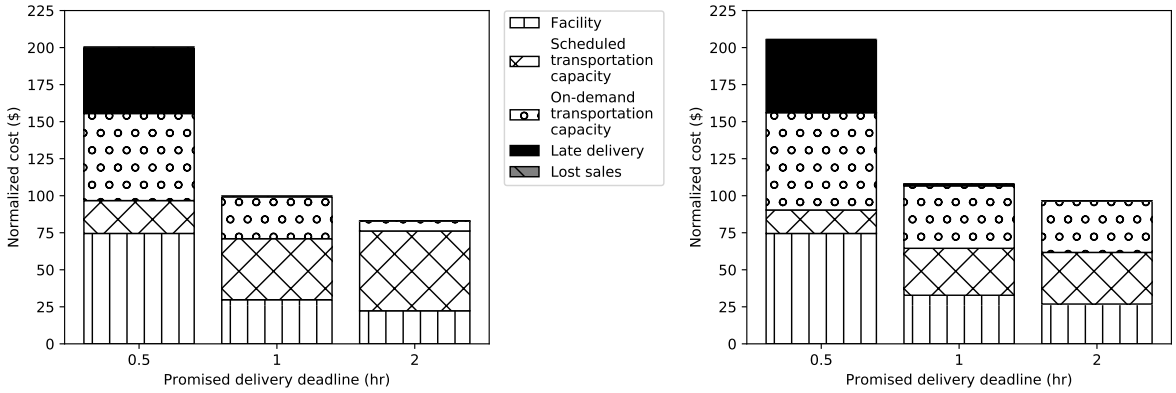


(c) Problem instance S-E



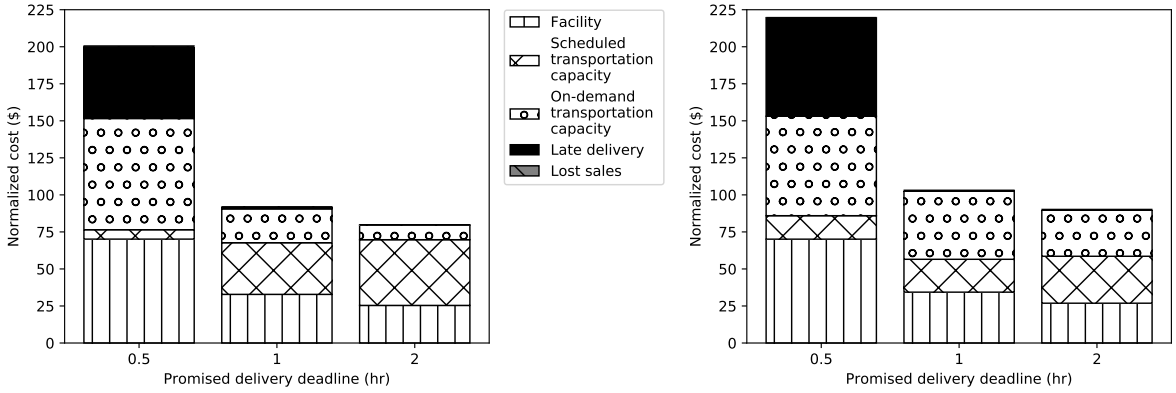
(d) Problem instance D-E

Figure F.15: Average performance of transportation capacity with changing system-wide average facility utilization.



(a) Problem instance S-U

(b) Problem instance D-U



(c) Problem instance S-E

(d) Problem instance D-E

Figure F.16: Network cost broken down in major components for different promised delivery deadline tightness (normalized)

Table F.12: Network design and KPIs averaged over six problem instances obtained by MetaAHA+ for various levels of delivery deadline tightness.

Problem Instance	**Deadline (hr)**	**# Fac.**	**Parallel Processing Capacity**	**Scheduled Transp. Capacity**	**% of Orders On-demand**	**Capacity Utilization Scheduled Couriers per Trip (%)**	**Late Delivery is 25% of Cost (Utilization)**
S-U	0.5	9	20	7	71.6	35.2	26.9
	1	3	8	13	32.2	66.6	74.7
	2	2	6	17	7.2	97.0	74.7
D-U	0.5	9	22	5	80.6	35.4	21.3
	1	3	10	10	48.7	66.8	62.1
	2	2	9	11	38.5	95.2	48.5
S-C	0.5	8	15	5	64.9	50.6	41.1
	1	3	8	10	22.1	80.7	74.9
	2	2	6	11	13.1	97.5	74.9
D-C	0.5	8	19	5	68.4	49.5	30.5
	1	3	11	10	33.4	78.6	62.8
	2	2	9	8	35.2	95.6	50.6
S-E	0.5	8	19	2	91.5	37.6	23.2
	1	3	9	11	26.2	78.4	74.8
	2	2	7	14	10.6	97.9	71.9
D-E	0.5	8	24	5	81.9	36.7	20.2
	1	3	11	7	53.7	78.4	62.1
	2	2	10	10	34.8	95.6	49.8