Coordinating International Humanitarian Inventory by Stochastic Dual Dynamic Programming

Penghui Guo, Jianjun Zhu*
College of Economics and Management, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China,
{guopenghui@outlook.com, zhujianjun@nuaa.edu.cn}

International humanitarian organizations face challenges in inventory management due to unpredictable disasters, evolving emergencies, participant coordination, long planning horizons, and broad geographical coverage. This paper develops an international humanitarian inventory coordination framework, using multi-stage stochastic programming to make long-term (monthly or quarterly) procurement, inventory, and transportation decisions. As a counterpart of monetary expense in the objective, a *target-based disutility*, that is monotonically non-increasing and convex, is proposed to measure the suffering caused by insufficient consumption. The model is solved by the generalized Stochastic Dual Dynamic Programming (SDDP), which allows convex recourse functions. The SDDP takes historical demands as input and generates an optimal *policy* for making future decisions without knowing exact demand information. Unlike deterministic equivalent formulations based on scenario trees, this policy is implementable for out-of-sample data. Extensive numerical experiments are conducted with publicly available data from the United Nations Humanitarian Response Depot, the United Nations Office for the Coordination of Humanitarian Affairs (OCHA), and the EM-DAT international disaster database. The method can generate a policy in under two hours using 216 months (18 years) of data from the 34 most disaster-vulnerable countries or territories where the OCHA works. The SDDP policy offers up to 21% cost savings over myopic or deterministic policies. Results demonstrate that good out-of-sample coordination results can be achieved with a moderate sample size, a reasonable number of iterations, and within the current OCHA organization structure.

Key words: humanitarian logistics; inventory management; multi-stage stochastic programming; stochastic dual dynamic programming

1. Introduction

From 2000 to 2023, over 15,000 disasters impacted 4.5 billion people worldwide (CRED 2024). The least developed and most disaster-vulnerable countries or territories rely on aid from international humanitarian organizations, including United Nations (UN) agencies and international non-governmental organizations (INGOs), e.g., the World Food Programme (WFP), the UN Humanitarian Response Depot (UNHRD) managed by the WFP, the UN Office for the Coordination of Humanitarian Affairs (OCHA), and over 800 INGOs (GDHO 2024). These organizations work in sectors like food security, health, nutrition, protection, and shelter, to meet the needs of affected populations. In these sectors, inventory management plays a fundamental role. However, inventory management in international humanitarian operations is challenging due to unpredictable disasters, evolving emergencies, participant coordination, long planning horizons, and broad geographical coverage. In alignment with the UN Sustainable Development Goals (SDGs) (Besiou et al. 2021): SDG2 zero hunger, SDG10 reduced inequality, and SDG17 partnerships for

* Corresponding author. *October 25, 2025*

Preprint submitted to Naval Research Logistics (DOI: 10.1002/nav.70030)

the goals, this study develops an inventory management framework for international humanitarian organizations to make long-term procurement, inventory, and transportation decisions.

For both regional INGOs and global organizations, making monthly or quarterly inventory decisions can be challenging. For regional INGOs that cover a specific region, inventory decisions are required to be made without knowing future, usually highly uncertain, demands. The evolving of emergencies, which may lead to changes in the demand pattern, also needs to be considered. For organizations that operate globally, e.g., the UN agencies, the covered regions are worldwide, making anticipating future evolving emergencies even more complex. Some regions may experience a demand peak in the third quarter (e.g., Atlantic Hurricane season and monsoon season in South Asia), while others may experience a demand peak in the first quarter (e.g., adverse impacts of the dry season on agricultural productivity in Sub-Saharan Africa and Southeast Asia). Inventory needs to be balanced spatially. Interactions among regional and global organizations also complicate decision-making. Though these organizations can have the same goal (alleviating suffering) and work together in the same region, lacking systematic coordination can lead to sub-optimal decisions. Some regions can overstock while others understock, leading to simultaneous waste and shortages. In summary, temporal and spatial uncertainty are the main challenges, and coordination among participants is essential to address these challenges.

The UNHRD and the OCHA make it possible to temporally and spatially coordinate international humanitarian inventory. The UNHRD is a network of five hubs located near disaster-prone areas to ensure swift dispatch of items within 48 hours. The UNHRD offers its partners services like procurement, storage, transportation, and capacity building, facilitating coordination and shared transport during crises (UNHRD 2024). The OCHA, as a coordinator, provides a platform for information exchange, strategic planning, and resource mobilization (OCHA 2023c). The UNHRD and the OCHA may employ various inventory management methods, such as economic order quantity and safety stock. However, the pattern of emergency-related demands is distinct from the market-driven ones in the commercial contexts, since the humanitarian demands are driven by the occurrence of disasters. In most cases, the demand is zero, but can spike significantly during a disaster (Figure 2). Furthermore, the goal of humanitarian operations is to alleviate suffering, rather than maximize profit, and the results of understocking can be severe. These make inventory management methods originating from commercial contexts not applicable, resulting in difficulties in preparing relief items at the right time and place.

In this study, the international humanitarian inventory coordination problem is modeled using multi-stage stochastic programming. The model aims at making coordinated long-term (monthly or quarterly) procurement, inventory, and transportation decisions for international humanitarian organizations. The objective is to minimize the systematic cost: financial costs and humanitarian costs. The humanitarian costs are modeled as a monotonically non-increasing and convex *target-based disutility*, which is a function of the gap between targeted and reached people. At the start of a planning horizon (year), the procurement decisions for stockpile are made, followed

by the procurement decisions at the beginning of each stage (month or quarter) without exact demand information in this stage. Then the transportation, consumption, repaid-response, and disposal decisions are made in each stage with known demands. To tackle the computational intractability of the multi-stage stochastic programming, Stochastic Dual Dynamic Programming (SDDP) (Pereira and Pinto 1991) is used. The SDDP method takes historical demand (in-sample) data as input and generates a policy for future (out-of-sample) data. It integrates nested Benders decomposition (Birge 1985) with sampling techniques to alleviate the curse of dimensionality. The SDDP method approximates the *cost-to-go* (*recourse*) function using linear inequalities — Benders cuts. After a sufficient number of iterations, the optimal (in the sense of in-sample data) policy is obtained. With the trained policy, international humanitarian organizations can make inventory decisions without needing to predict future demands. Numerical experiments are conducted with real data from the EM-DAT (CRED 2024) — a well-known disaster database, UNHRD (2024), and OCHA (2024). Our method can yield a policy for an annual plan (in months) using historical data of 18 years from 34 countries or territories, in less than two hours.

This study makes three key contributions. First, we develop a multi-stage stochastic international humanitarian inventory coordination model. In the objective, we define a target-based disutility function to capture the costs of unmet demand. The model allows inventory balance across participants, both vertically (from hubs to warehouses) and horizontally (among warehouses). Second, the SDDP method developed is significant because the policy obtained is *implementable* for any demand observation, not necessarily one of the in-sample sample paths. On the contrary, the widely used scenario-tree-based deterministic equivalent (Section 2.2) requires the actual demand to be within the scenario tree. In addition, the SDDP method developed allows convex cost-to-go functions (introduced by the targeted-based disutility) by incorporating generalized Benders decomposition (Geoffrion 1972), making it a generalization of the original linear SDDP. Third, we collect real data and generate instances for numerical studies. In addition to the numerical results and the management insights obtained, our instances can be used for future studies in humanitarian operations management, particularly problems dealing with uncertainty.

The remainder of this paper is organized as follows. Section 2 reviews related literature, before introducing the backgrounds and data in Section 3. The international humanitarian inventory coordination problem is modeled in Section 4 with the disutility function. Section 5 develops the SDDP method. Section 6 presents numerical results, followed by the conclusions in Section 7.

2. Literature Review

2.1. Humanitarian Coordination Problems

The term *coordination*, in the humanitarian context, is defined as *"the relationships and interactions among different actors operating within the relief environment"* (Balcik et al. 2010); *"a process whereby the activities of interdependent organizations are brought together to achieve certain objectives"* (Akhtar et al. 2012); *"process of organizing, aligning and differentiating of participating NGOs' actions based on regional knowledge, know-how, specialization and resource availability*

to reach a shared goal in the context of disasters" (Wankmüller and Reiner 2019); *"the alignment of activities to operate efficiently and effectively"* (Adsanver et al. 2024). Readers may refer to literature reviews by Balcik et al. (2010), Wankmüller and Reiner (2019), Adsanver et al. (2024), Tippong et al. (2022) for humanitarian coordination. Our study focuses on the coordination mechanism of international humanitarian participants in procurement, inventory, and transportation (Balcik et al. 2010, Adsanver et al. 2024) for systematic efficiency and effectiveness.

Humanitarian coordination can be *vertical* (between upstream and downstream) or *horizontal* (among peers) (Balcik et al. 2010). Our study involves both. In the following, we focus on humanitarian coordination papers that use operations research methods. Toyasaki et al. (2017) studied a horizontally coordinated inventory management problem using a newsvendor model. They analyzed the incentive of organizations to join the UNHRD network. Real-world data were used for numerical analysis. Aflaki and Pedraza-Martinez (2023) studied coordination of fundraising among humanitarian organizations using a game theory model. They considered the competition and collaboration among organizations, which can be categorized as horizontal coordination. Balcik et al. (2019) studied collaborative prepositioning of emergency items in a relief network — the Caribbean Disaster and Emergency Management Agency. The problem is modeled using two-stage stochastic programming. Their study belongs to horizontal and regional coordination. Rodríguez-Pereira et al. (2021) studied a cost-sharing mechanism under a similar context based on cooperative game theory. There are also studies on vertical coordination (Nikkhoo et al. 2018, John et al. 2022, Wang et al. 2015), which focus on contracting. Except for Balcik et al. (2010), which examines international humanitarian coordination (using game theory), most studies in the literature focus on regional problems. Our study is the first to use multi-stage stochastic programming to address vertical and horizontal global humanitarian inventory coordination. Our method is an easy-to-implement tool for humanitarian organizations to make inventory decisions.

2.2. Multi-stage Stochastic Programming

Given the uncertainty of humanitarian crises, stochastic programming is well-suited for related operations management problems. While two-stage stochastic programming has been extensively studied (Grass and Fischer 2016), research on multi-stage stochastic programming remains limited. In two-stage models, the planning horizon is divided into the *pre-disaster* and the *post-disaster* stages. These models usually focus on making pre-disaster stage decisions, such that they are optimal in the sense of expectation under all possible post-disaster scenarios. However, in practice, the planning horizon is typically divided into multiple stages, with decisions made sequentially to fit the multi-stage stochastic programming framework.

The first paper on multi-stage stochastic programming for humanitarian (disaster) operations management was published by Hu et al. (2019). Between 2019 and 2025, a total of ten papers have been published in this field (summarized in the Supporting Information), which is still in its infancy compared to multi-stage stochastic programming and humanitarian operations management. In the ten papers, the majority (Hu et al. 2019, Olanrewaju et al. 2020, Zahiri et al.

2020, Rambha et al. 2021, Fattahi et al. 2023, Hu et al. 2023) built their models based on scenario-tree discretization and Sample Average Approximation (SAA), then the scenario-tree model is solved exactly or heuristically by commercial solvers or customized algorithms (e.g., Progressive Hedging). However, it is unclear how to implement the solutions if the actual disaster outcome falls out of the scenario-tree predictions. We employ SDDP — a well-established (Section 2.3) multi-stage stochastic programming solution method — to solve the proposed multi-stage stochastic programming problem. The ability to generalize the policy to out-of-sample data is a significant advantage over existing studies. In the ten papers, only four (Kizito et al. 2022, Castro et al. 2022, Seranilla and Löhndorf 2024, Siddig and Song 2025) can handle out-of-sample data. They achieve this by using the nested L-shaped method, SDDP, shadow price approximation, and nested Benders decomposition, respectively. Their works differ from ours as they focus on regional or local problems, while our study addresses an international inventory coordination problem.

2.3. Stochastic Dual Dynamic Programming

Multi-stage stochastic programming models can be solved via *deterministic equivalent* formulated on scenario trees (Shapiro et al. 2021). However, the size of the deterministic equivalent grows exponentially, leading to the *curse of dimensionality*. To address this, a Stochastic Dynamic Programming method (Füllner and Rebennack 2025), namely SDDP (Pereira and Pinto 1991), was proposed. This approach ensures that the number of problems solved in each iteration scales proportionally with the number of scenarios and stages, overcoming the curse of dimensionality.

There are various variants of SDDP. Zou et al. (2019) extended the SDDP method to include integer variables. Ahmed et al. (2022) studied stochastic Lipschitz dynamic programming by allowing mixed-integer variables with non-convex cost-to-go functions, and constructing nonlinear Lipschitz cuts. The original SDDP operates with deterministic equivalents of stochastic problems, Gangammanavar and Sen (2021) introduced a method namely stochastic dynamic linear programming to involve new samples in the algorithm iteration. Guigues (2020), Guigues et al. (2021) studied SDDP methods that solve primal and dual problems inexactly with bounded errors for linear, differentiable nonlinear, and convex non-differentiable problems. The SDDP method has also been studied for multi-objective problems (Dowson et al. 2022) and robust optimization problems (Duque and Morton 2020). Theoretically, the convergence of SDDP has been proved by Philpott and Guan (2008), Shapiro (2011), Leclère et al. (2020) for the linear case, and by Girardeau et al. (2015) for the convex case. The complexity of SDDP was analyzed by Lan (2022). Readers may refer to Füllner and Rebennack (2025) for a comprehensive review of SDDP and Lan and Shapiro (2024) for solution methods for convex multi-stage stochastic optimization.

In application, the SDDP was initially proposed for energy planning problems and remains predominant in this field (Pereira and Pinto 1991, Ding et al. 2018, Helseth and Mo 2023, Zhu et al. 2023). The SDDP method was also applied to lot-sizing problems (Thevenin et al. 2021, 2022, Quezada et al. 2022). Though their works are related to ours, no humanitarian-specific characteristics were involved. In the humanitarian context, to our knowledge, Castro et al. (2022)

is the only one that used SDDP. They focused on hurricane relief logistics planning at the micro level, while we use the SDDP for international humanitarian inventory management. The SDDP method in this study can be seen as a generalization of SDDP with convex cost-to-go functions, which combines the SDDP method (Pereira and Pinto 1991) with the general Benders decomposition (Geoffrion 1972). The benefit is it contains a training phase and an inferring phase. In the training phase, all the available data (historical records) can be used to train a set of policies, each one for a stage. Though training can be computationally expensive, it is conducted offline, typically in the planning phase of the humanitarian operation. On the other hand, inferring requires far less computational effort, which is done with new data in the response phase.

3. Backgrounds and Data

3.1. Humanitarian Coordination Practice

The direct motivation of the humanitarian inventory management problem is the coordination practice of the OCHA, which leads and coordinates efforts of humanitarian organizations to ensure principled, timely, effective, and efficient action (OCHA 2024). The OCHA has two headquarters in New York (the United States) and Geneva (Switzerland) with six regional offices (Asia and Pacific, Europe, Latin America and the Caribbean, Middle East and North Africa, Southern and Eastern Africa, and West and Central Africa), as in Figure 1, each covering several of the most disaster-vulnerable countries or territories via country offices. In these regions, humanitarian financing, information management, response planning and implementation, and humanitarian coordination are operated by the OCHA with its partners. These activities form the mechanisms in relief chains that involve coordination, cooperation, and collaboration (Adsanver et al. 2024).

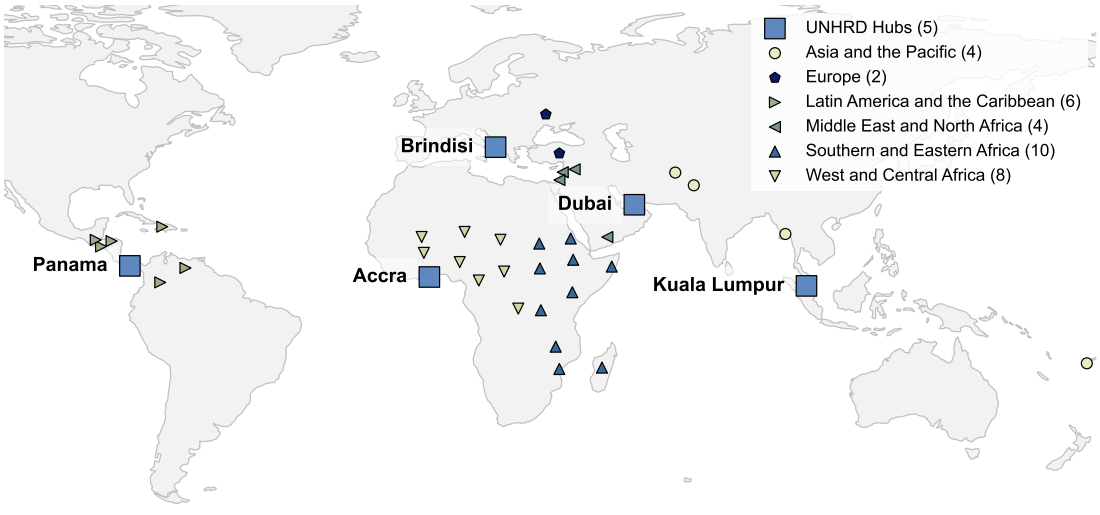


Figure 1 **The countries or territories where the OCHA works and the UNHRD Hubs.**

The countries or territories where the OCHA works are both disaster-vulnerable and least developed, heavily relying on external assistance from UN agencies and INGOs, rather than solely local governmental agencies, demonstrating multiple participants can be involved in humanitarian operations. As the "C" (coordination) in its name suggests, the OCHA is not a humanitarian organization that directly implements relief activities but plays the role of a coordinator. In 2022,

90% of the OCHA's funds were expended on humanitarian coordination (85%) and information management (5%) (OCHA 2023b). With efficient information-sharing platforms (Humanitarian Data Exchange, ReliefWeb, and HumanitarianResponse.info), the OCHA coordinates the humanitarian organizations to implement the Humanitarian Response Plans (HRPs) (OCHA 2023a) via the Cluster Approach (OCHA 2023). The clusters are groups of humanitarian organizations that work in specific sectors (e.g., logistics, food security, etc) during the life cycle of an emergency. In each cluster, there is a lead agency (usually a UN agency, e.g., the WFP leads the logistics cluster and food security cluster) and other joined agencies (usually INGOs). These agencies work collaboratively to alleviate the suffering of people affected.

Humanitarian operations are typically dynamic and stochastic. At the strategic level, the OCHA runs based on the Humanitarian Programme Cycle (HPC): (i) needs assessment and analysis, (ii) strategic planning, (iii) resource mobilization, (iv) implementation and monitoring, and (v) operational peer review and evaluation (OCHA 2023c). In (i) and (ii), Humanitarian Needs Overview (HNO) and HRPs are produced, respectively. In (iii) and (iv), to address the uncertainty and evolving (deteriorate) nature of the crisis, the Country-based Pooled Funds (CBPFs) and the Central Emergency Response Fund (CERF) are established to provide rapid and flexible resources. At the operational level, we take the WFP, a vital partner of the OCHA that plays significant roles in the logistics and food security clusters, as an example. The WFP runs the UNHRD, a network of hubs (Figure 1) located in five locations (Brindisi in Italy, Accra in Ghana, Kuala Lumpur in Malaysia, Panama City in Panama, and Dubai in the United Arab Emirates) worldwide (Eligüzel et al. 2022), to provide free logistics services to the authorized partner organizations (Dufour et al. 2018). The UNHRD and its partners dynamically manage their inventory within the UNHRD Hubs or private warehouses using historical data, current inventory levels, and demand forecasting, to ensure timely and cost-efficient responses despite the uncertainty and evolving nature of crises (WFP 2023, Acimovic and Goentzel 2016).

The model (Section 4) and the solution method (Section 5) are connected to the practices as follows. First, the relief network includes the countries or territories where the OCHA works and the UNHRD Hubs, with relief items storing in them and flowing among them. Second, a centralized coordinator (the OCHA) is assumed to share information and coordinate relief operations. Third, the dynamic and stochastic decision-making process is in alignment with the HPC — there is a need assessment and strategic planning phase ((i) and (ii) in HPC) and another resource mobilization and implementation phase ((iii) and (iv) in HPC). The SDDP training phase and inferring phase can be seen as the first and the second, respectively. Fourth, as mentioned, the OCHA allocates two types of funds: CBPFs and CERF. CBPFs focus on providing strategic, predictable, and country-specific funding to the participants as in HRPs, namely standard allocations; while CERF focuses on providing agile funding at any time to new shocks, namely *rapid response* allocations (OCHA 2021). We use budget constraints and rapid response variables to model CBPFs and CERF, respectively.

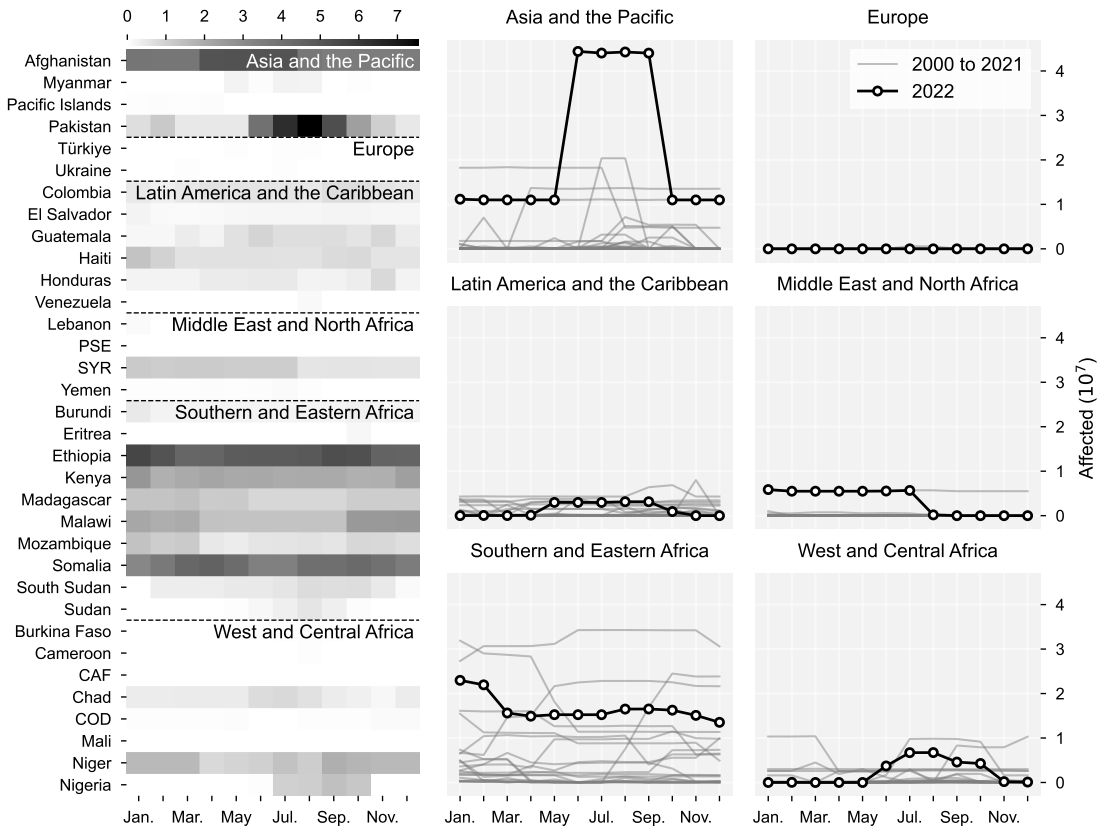
3.2. Data

This subsection presents the datasets (illustrated in Figure 1 and Figure 2) used in this study. In specific, the location of nodes in the international humanitarian logistics network collected from the official website of UNHRD (2024) and OCHA (2024); the demand (population affected) collected from EM-DAT (CRED 2024) — a well-known database that records disasters (including location, affected population, start and end date, etc) from 1900 to present. Due to the unreliability of data before 2000 (CRED 2024), we use only post-2000 data.

We first introduce the nodes in the relief network (Figure 1). There are three sets of nodes: shared hubs, organizations, and warehouses. The shared hubs are the five UNHRD Hubs. The UNHRD Hubs are used for numerical study since they are vital in international humanitarian relief (Stienen et al. 2021, Eligüzel et al. 2022). The organizations considered are the six regional offices (represented as shapes in Figure 1) of OCHA (OCHA 2024). These offices cover the most disaster-vulnerable (EC-DRMKC 2023) regions in the world. These regions frequently face disasters and lack the ability to cope, thus relying on assistance from UN agencies and INGOs. The countries or territories that are covered by each OCHA regional office are considered warehouses. Each regional office is responsible for its covered countries or territories, and coordinates with the UNHRD Hubs and other regional offices. Using the longitude and latitude of all nodes, we can estimate transportation prices using the *great-circle distance* (the shortest distance between two points on a sphere surface, e.g., the earth) and the unit cost of sea, air, and land freight.

The population affected (Figure 2) is gathered from the EM-DAT (CRED 2024) for the 34 countries or territories (Figure 1) from 2000 to 2022. The data for 2023 and 2024 are not used as they are under annual validation by the EM-DAT at the time of writing. We calculate the population affected in a certain month based on `Start Year`, `Start Month`, `End Year`, `End Month`, and `Total Affected` of disasters in the EM-DAT. For example, if a disaster starts in January 2000 and ends in February 2000, affecting a total of 100,000 people, then both months would record an affected population of this number since relief items would be needed throughout these two months. From Figure 2(a), temporal and spatial imbalances can be observed. For example, the number of people affected in Southern and Eastern Africa is higher than in other subregions; and for Pakistan, Sudan, Chad, and Nigeria, the number of people affected from March to June is higher throughout the year. This suggests the potential of inter-region coordination and the necessity of a proper inventory management strategy, which is the focus of our study. To validate the proposed method, the demand distribution for generating instances can be obtained from a subset of the historical data, e.g., 2000 to 2017; while the demand in other years, e.g., 2018 to 2022, which is not used for policy optimization, can be used for validation.

Aligned with the humanitarian coordination practices, we consider a problem that optimizes the logistics (e.g., procurement, inventory, transportation, and distribution) of critical relief items (e.g., food, water, and shelter) in humanitarian operations management. The problem takes uncertain demand information as input and outputs an optimal inventory management *policy* for the



(a) Countries or territories (b) Subregions

Figure 2 **The number of affected populations from 2000 to 2022. Figure 2(a) displays the monthly total number of people affected in each country or territory from 2000 to 2022. Figure 2(b) shows the monthly number of affected people in each region for every year.**

planning horizon. The data and code can be accessed from https://github.com/phguo/nrl25. Readers can generate their own instances for research in humanitarian operations management, especially problems under uncertainty, e.g., two-stage and multi-stage stochastic and robust programming.

4. Inventory Coordination Problem

In this section, Section 4.1 introduces the sets, parameters, and decision variables used and describes the humanitarian inventory coordination problem, followed by the target-based disutility function defined in Section 4.2, and the mathematical programming model in Section 4.3.

4.1. Problem Description

The humanitarian logistics network of interest includes hubs and warehouses run by humanitarian organizations (HOs). As mentioned earlier, the UNHRD Hubs can be used by both the WFP and other HOs. Other HOs can have their own warehouse network operated by themselves. Unlike the UNHRD Hubs, which are strategically located but may not be close to affected areas, other HOs' warehouses can be tactically placed, allowing them to be nearer to potential disaster zones. The inventory coordination problem built on this network is a multi-stage stochastic problem that determines the optimal procurement, inventory, and transportation for hubs and warehouses during the planning horizon (e.g., 12 months in a year). At each stage, the information available

is the demand and decision sequences till the previous stage, the inventory of the current stage, and estimates (probabilistic distributions) for future demands. With certain objectives (e.g., the sum of expected costs through the planning horizon), multi-stage stochastic programming makes a sequence of decisions, each corresponding to a stage. These decisions are made depending only on the observed information up to the corresponding stage, but not on the future information. This is known as *nonanticipativity*. The information (parameters of the model) in stochastic programming is a *stochastic process* (Shapiro et al. 2021). Below, we introduce the notations used for sets, parameters, and decision variables.

Sets:

$\mathcal{H}$	Set of shared hubs.
$\mathcal{R}$	Set of regions.
$\mathcal{W}_r$	Set of warehouses in region $r \in \mathcal{R}$.
$\mathcal{W}$	Set of warehouses $\mathcal{W} = \bigcup_{r\in\mathcal{R}} \mathcal{W}_r$.
$\mathcal{A}_{\mathcal{H}}$	Set of available arcs from shared hub $i \in \mathcal{H}$ to shared hub $i' \in \mathcal{H}$.
$\mathcal{A}$	Set of available arcs from shared hub $i \in \mathcal{H}$ to warehouse $j \in \mathcal{W}$.
$\mathcal{A}_{\mathcal{W}}$	Set of available arcs from warehouse $j \in \mathcal{W}$ to warehouse $j' \in \mathcal{W}$.
$\mathcal{T}$	Set of time periods. $\mathcal{T} = \{0, 1, 2, \cdots, T\}, \mathcal{T}_0 = \mathcal{T} \setminus 0, \mathcal{T}_T = \mathcal{T} \setminus T$.
Ω_r^t	Sample space of stochastic parameters of $r \in \mathcal{R}$ at $t \in \mathcal{T}_0$.

Parameters:

ω_r^t	Demand of region $r \in \mathcal{R}$ at $t \in \mathcal{T}_0$, $\left\{d_j^t \ \forall\ j \in \mathcal{W}_r\right\} = \omega_r^t \sim \Omega_r^t$.
c_k^t	Unit procurement cost of hub or warehouse $k \in \mathcal{H} \cup \mathcal{W}$ at $t \in \mathcal{T}$.
p_j^t	Unit rapid-response procurement cost of warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
b_k^t	Maximum procurement budget of hub or warehouse $k \in \mathcal{H} \cup \mathcal{W}$ at $t \in \mathcal{T}$.
q_k^t	Capacity of hub or warehouse $k \in \mathcal{H} \cup \mathcal{W}$ at $t \in \mathcal{T}$.
g_a	Unit transportation cost on arc $a \in \mathcal{A}_{\mathcal{H}} \cup \mathcal{A} \cup \mathcal{A}_{\mathcal{W}}$.
$\mathcal{U}_j^t(\cdot)$	(Disutility) Disutility function of warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
γ	(Disutility) Welfare of satisfying each beneficiary.
ρ_j^t	(Disutility) Targeted population of warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
δ_j^t	(Disutility) Consumption rate of warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
$h_j^t(\cdot)$	(Disutility) The proportion of the targeted population not reached, $h_j^t(\cdot) \in [0, 1]$.

Decision variables:

z_i^t	Procurement of hub $i \in \mathcal{H}$ at $t \in \mathcal{T}$.
$f_{ii'}^t$	Inventory balance from hub $i \in \mathcal{H}$ to hub $i' \in \mathcal{H}$ at $t \in \mathcal{T}_0$.
f_{ij}^t	Allocation from hub $i \in \mathcal{H}$ to warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
z_j^t	Procurement of warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}$.
$f_{jj'}^t$	Inventory balance from warehouse $j \in \mathcal{W}$ to warehouse $j' \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
u_j^t	Consumption of warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
x_k^t	Inventory of hub or warehouse $k \in \mathcal{H} \cup \mathcal{W}$ at $t \in \mathcal{T}$.
y_k^t	Disposal of hub or warehouse $k \in \mathcal{H} \cup \mathcal{W}$ at $t \in \mathcal{T}_0$.
v_j^t	Rapid response quantity of warehouse $j \in \mathcal{W}$ at $t \in \mathcal{T}_0$.
θ^t	(SDDP) Auxiliary variable of $t \in \mathcal{T}$ for approximating future costs.
$z_k^{'t}$	(SDDP) Auxiliary procurement variable of $k \in \mathcal{H} \cup \mathcal{W}$ at $t \in \mathcal{T}$ for nonanticipativity.

Sets. Let $\mathcal{H}$ be the set of shared hubs (e.g., the UNHRD), $\mathcal{R}$ be a set of regions that each has a set of private warehouses $\mathcal{W}_r$, and $\mathcal{W} = \bigcup_{r\in\mathcal{R}} \mathcal{W}_r$ be all the private warehouses. Transportation is allowed among all the shared hubs and warehouses, or determined by the specific organization structure of the coordination practice. Transportation is on the links $\mathcal{A}_\mathcal{H}$, $\mathcal{A}$, and $\mathcal{A}_\mathcal{W}$. Let us take $\mathcal{A}$ as an example, it contains a subset of pairs of hubs $i \in \mathcal{H}, j \in \mathcal{W}$ ($\mathcal{A} \in \mathcal{H} \times \mathcal{W}$). $\mathcal{A}_\mathcal{H} \in \mathcal{H} \times \mathcal{H}$ and $\mathcal{A}_\mathcal{W} \in \mathcal{W} \times \mathcal{W}$ are defined similarly. In contrast with $\mathcal{H} \times \mathcal{W}$, these link sets are restricted by the distance between hubs and warehouses, or organization structure. Decisions are made during the planning horizon $\mathcal{T} = \{0, 1, 2, \cdots, T\} (\mathcal{T}_0 = \mathcal{T} \setminus 0, \mathcal{T}_T = \mathcal{T} \setminus T)$, with Ω_r^t being the sample space of stochastic parameters of $r \in \mathcal{R}$ at $t \in \mathcal{T}_0$. Ω_r^t can be either discrete *scenarios* via empirical estimation or continuous random variables via data fitting.

Parameters. At each stage $t \in \mathcal{T}$, hubs and warehouses $k \in \mathcal{H} \cup \mathcal{W}$ procure relief items at the price of c_k^t to satisfy demand $\left\{d_j^t \ \forall\ j \in \mathcal{W}_r\right\} = \omega_r^t \sim \Omega_r^t$ of $r \in \mathcal{R}$. The rapid response price for $j \in \mathcal{W}$ is p_j^t. The maximum procurement budget and capacity of $k \in \mathcal{H} \cup \mathcal{R}$ at stage $t \in \mathcal{T}$ are b_k^t and q_k^t, respectively. The available inventory from the previous stage can also be used. To avoid shortage and surplus, inventory is rebalanced among hubs and warehouses at the unit cost of g_a on a specific link $a \in \mathcal{A}_\mathcal{H} \cup \mathcal{A} \cup \mathcal{A}_\mathcal{W}$. Here we let regions' warehouses be the "end-users", rather than the affected areas. The demands can be estimated either using historical data of warehouses or aggregation of affected areas' (the real end-users) demands. The disutility functions $\mathcal{U}_j^t(\cdot)$ are defined for each $j \in \mathcal{W}$ and $t \in \mathcal{T}_0$ in Section 4.2. These functions contribute to the objective function, which balances monetary costs against disutility.

Decision variables. For the operator of shared hubs, at each stage $t \in \mathcal{T}_0$, it makes the procurement decisions $z_i^t, i \in \mathcal{H}$, the disposal decision y_i^t, and the transportation decisions $f_{ii'}^t, (i, i') \in \mathcal{A}_\mathcal{H}$ and $f_{ij}^t, (i, j) \in \mathcal{A}$. For regions $r \in \mathcal{R}$, at each stage $t \in \mathcal{T}$, it makes the procurement decisions for all $j \in \mathcal{W}_r$ as z_j^t, the transportation decisions $f_{jj'}^t, (j, j') \in \mathcal{A}_\mathcal{W}$, the rapid response decisions v_j^t, and the disposal decision y_j^t. The consumption (delivery to affected people) of relief items by warehouses $j \in \mathcal{W}_r$ is denoted as u_j^t. The inventory of hubs and warehouses $k \in \mathcal{H} \cup W$ during the planning horizon is tracked with the variables x_k^t implied by the procurement (z_i^t, z_j^t), transportation $(f_{ii'}^t, f_{ij}^t, f_{jj'}^t)$, consumption (u_j^t), disposal (y_k^t), and rapid response (v_j^t) decisions. Disposal occurs when the relief items exceed the capacity of the hubs or warehouses; rapid response procurement occurs after observing the inventory is insufficient for the real demand.

4.2. Target-based Disutility

Humanitarian operations management often requires balancing costs with the impact of supply shortages. While the former can be measured by the cost of procurement, transportation, etc, the latter is more difficult to quantify. We introduce a target-based disutility function to capture the humanitarian impact. Two properties of the disutility functions $\mathcal{U}_j^t(\cdot) : u_j^t \mapsto \mathbb{R}_0^+$ are assumed: *monotonically non-increasing* and *convex*, consistent with prior studies (Keshvari Fard et al. 2022, Holguín-Veras et al. 2013, Holguin-Veras et al. 2016). The monotonicity means a higher consumption u_j^t leads to lower disutility. There are two reasons for convexity. First, convex disutility

adheres the *Diminishing Marginal Utility* (Gossen 1983, Mankiw 2024), meaning the marginal benefit of consuming an additional resource unit decreases as more resources are consumed. Second, convex disutility reflects a *risk-averse* preference (Kahneman and Tversky 1979, Mankiw 2024), favoring a sure moderate disutility over uncertain prospects that might yield either low or high disutility. One of the major goals of the HRP by the OCHA is to close the gap between people reached and *people targeted — the number of people in need that the humanitarian actors aim or plan to assist* (OCHA 2020). The disutility decreases (monotonicity) as more people are reached, but this does not suggest $\mathcal{U}_j^t(\cdot)$ being linear with u_j^t, since reaching the last ten percent of the targeted people is usually harder (convexity). For each $j \in \mathcal{W}_r, r \in \mathcal{R}$, given the people targeted $\rho_j^t \geq 0$, the welfare $\gamma > 0$ of satisfying per beneficiary, consumption rate $\delta_j^t > 0$ measured as unit per beneficiary per stage, the target-based disutility function is defined as

$$\mathcal{U}_j^t(u_j^t) = \gamma\rho_j^t \cdot h_j^t\left(\max\left\{\rho_j^t\delta_j^t - u_j^t, 0\right\}\right) \tag{1}$$

where $\gamma\rho_j^t$ is a constant that represents the welfare of satisfying all the targeted people. $h_j^t(\cdot) \in [0,1]$ is a function that represents the proportion of targeted people not reached due to a shortage $\rho_j^t\delta_j^t - u_j^t$. For succinctness, we let $(\rho_j^t\delta_j^t - u_j^t)^+ = \max\left\{\rho_j^t\delta_j^t - u_j^t, 0\right\}$. We use "target-based" in the name of (1) to emphasize that the minimum disutility is achieved when all the targeted people are reached. An example of $h_j^t(\cdot)$ is a linear function $\mathcal{LU}(u_j^t) = \varphi_j^t(\rho_j^t\delta_j^t - u_j^t)^+$, where $\varphi_j^t = 1/\rho_j^t\delta_j^t$ if $\rho_j^t \neq 0$ otherwise $\varphi_j^t = 0$ is the slope of the linear function. Since $\mathcal{LU}(u_j^t)$ is non-increasing and convex, $\gamma\rho_j^t$ is a constant, the $\mathcal{U}_j^t(u_j^t)$ is also non-increasing and convex with $\mathcal{LU}(u_j^t)$. The warehouses can have different disutility functions, which may vary from each other and from the one mentioned above. An alternative to $\mathcal{LU}(u_j^t)$ is a polynomial $\mathcal{PU}(u_j^t) = \left[\varphi_j^t(\rho_j^t\delta_j^t - u_j^t)^+\right]^{\beta_j^t}$, where $\beta_j^t \geq 1$ is a coefficient that controls the convexity. When $\beta_j^t = 1$, $\mathcal{PU}(u_j^t)$ is equivalent to $\mathcal{LU}(u_j^t)$. Since $\mathcal{PU}(u_j^t)$ is non-increasing and convex when $u_j^t \in [0, \rho_j^t\delta_j^t]$, $\gamma\rho_j^t$ is a constant, the $\mathcal{U}_j^t(u_j^t)$ is also non-increasing and convex with $\mathcal{PU}(u_j^t)$. Our solution method (Section 5) is not limited to these disutility functions, it applies whenever the two properties hold.

4.3. Model

The humanitarian inventory coordination problem is formulated as (3). The objective function (3.1) consists of two parts: (i) the objective function of the shared hub network $\mathcal{H}$, which is the sum of the expected procurement and transportation costs; (ii) the objective function of each region $r \in \mathcal{R}$, which is the sum of the expected procurement, rapid response, and the disutility weighted by λ_j. In both parts, at stage $t = 0$, there is only procurement (considered as *prepositioning* inventory) cost but no transportation cost. For the following stages $t \in \mathcal{T}_0$, $\mathcal{Q}_{\mathcal{H}}^t$ and $\mathcal{Q}_r^t$ are defined as follows.

$$\mathcal{Q}_{\mathcal{H}}^t = \sum_{i\in\mathcal{H}} c_i^t z_i^t(\omega^{t-1}) + \sum_{(i,i')\in\mathcal{A}_{\mathcal{H}}} g_{ii'} f_{ii'}^t(\omega^t) + \sum_{(i,j)\in\mathcal{A}} g_{ij} f_{ij}^t(\omega^t) + \sum_{(j,j')\in\mathcal{A}_{\mathcal{W}}} g_{jj'} f_{jj'}^t(\omega_r^t) \tag{2.1}$$

$$\mathcal{Q}_r^t = \sum_{j\in\mathcal{W}_r} \left[c_j^t z_j^t(\omega_r^{t-1}) + p_j^t v_j^t(\omega_r^t) + \lambda_j\, \mathcal{U}_j^t\left(u_j^t(\omega_r^t)\right)\right] \tag{2.2}$$

The cost of the shared hubs $\mathcal{H}$ at stage $t \in \mathcal{T}_0$ denoted as $\mathcal{Q}^t_{\mathcal{H}}$ in (2.1) is determined by decisions $z^t_i, f^t_{ii'}, f^t_{ij}$, and $f^t_{jj'}$. These decisions depend on the feasible region of decision variables and observation of the stochastic parameters $\omega^{t-1} = \bigcup_{r\in\mathcal{R}} \omega^{t-1}_r$ or $\omega^t = \bigcup_{r\in\mathcal{R}} \omega^t_r$. The superscript $t-1$ indicates that shared hubs make decisions based on information from the previous stage. Regardless of the observed ω^t, these decisions are made deterministically. This is known as the *nonanticipativity* requirement in *decision-hazard* stochastic programming. The decision-hazard decisions have the process $\cdots \rightsquigarrow X^t \rightsquigarrow \omega^t \rightsquigarrow X^{t+1} \rightsquigarrow \omega^{t+1} \cdots$, where $X^t = \{z^t_i, z^t_j\}$. In contrast, the *hazard-decision* process is $\cdots \rightsquigarrow \omega^t \rightsquigarrow X^t \rightsquigarrow \omega^{t+1} \rightsquigarrow X^{t+1} \cdots$, where $X^t = \{f^t_{ii'}, f^t_{ij}, v^t_j, f^t_{jj'}, u^t_j\}$. (2.2) contains the disutility terms (1) caused by insufficient supply. Since the disutility may not be directly comparable with the monetary cost, we introduce the coefficient λ_j to balance the cost terms and the disutility term. The complete multi-stage stochastic programming model is as follows. For conciseness, ω in the brackets in (2.1) and (2.2) is omitted.

$$\begin{aligned}
\min\ & \sum_{i\in\mathcal{H}} c^t_i z^0_i + \mathbb{E}\left[\mathcal{Q}^1_{\mathcal{H}} + \cdots + \mathbb{E}\left[\mathcal{Q}^T_{\mathcal{H}}\right]\right] \\
& + \sum_{r\in\mathcal{R}} \left[\sum_{j\in\mathcal{W}_r} c^t_j z^0_j + \mathbb{E}_{\omega^1_r\in\Omega^1_r}\left[\mathcal{Q}^1_r + \cdots + \mathbb{E}_{\omega^T_r\in\Omega^T_r}\left[\mathcal{Q}^T_r\right]\right]\right] && (3.1)\\
\text{s.t. } & c^t_i z^t_i \le b^t_i, && \forall\ i\in\mathcal{H}, t\in\mathcal{T} \quad (3.2)\\
& c^t_j z^t_j \le b^t_j, && \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in\mathcal{T} \quad (3.3)\\
& x^t_i + y^t_i \le x^{t-1}_i + z^t_i + \sum_{i'\in\mathcal{H}} f^t_{i'i} - \sum_{i'\in\mathcal{H}} f^t_{ii'} - \sum_{j\in\mathcal{W}} f^t_{ij}, && \forall\ i\in\mathcal{H}, t\in\mathcal{T}_0 \quad (3.4)\\
& x^t_j + y^t_j \le x^{t-1}_j + z^t_j + v^t_j + \sum_{i\in\mathcal{H}} f^t_{ij} + \sum_{j'\in\mathcal{W}_r} f^t_{j'j} - \sum_{j'\in\mathcal{W}_r} f^t_{jj'} - u^t_j, && \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in\mathcal{T}_0 \quad (3.5)\\
& x^t_i \le q^t_i, && \forall\ i\in\mathcal{H}, t\in\mathcal{T} \quad (3.6)\\
& x^t_j \le q^t_j, && \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in\mathcal{T} \quad (3.7)\\
& x^0_i = z^0_i, && \forall\ i\in\mathcal{H} \quad (3.8)\\
& x^0_j = z^0_j, && \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \quad (3.9)\\
& z^t_i, f^t_{ii'}, f^t_{ij}, x^t_i, y^t_i \in \mathbb{R}^+_0, && \forall\ i,i'\in\mathcal{H}, j\in\mathcal{W}, t\in\mathcal{T} \quad (3.10)\\
& z^t_j, f^t_{jj'}, x^t_j, u^t_j, y^t_j, v^t_j \in \mathbb{R}^+_0, && \forall\ j,j'\in\mathcal{W}, t\in\mathcal{T} \quad (3.11)
\end{aligned}$$

The objective function (3.1) minimizes the long-term expected cost of the shared hub network and regions. (3.2) and (3.3) are budget constraints of the shared hub network and regions, respectively. These two constraints can be seen as the CBFPs allocated in HRPs for providing strategic country-specific support, as discussed in Section 3.1. CBFPs are complemented by the CERF, a rapid response mechanism. CERF is modeled as the rapid response decisions (v^t_j) in (2.2), (3.4), and (3.5) made after the realization of stochastic parameters. The rapid response procurement decisions align with humanitarian practice, where emergency sourcing from local or regional markets to meet immediate, unforeseen demand. This action involves short lead times but incurs higher costs, differing operationally and financially from the strategic, large-scale,

and cost-efficient procurement at shared hubs. (3.4) and (3.5) are flow conservation constraints of the shared hub network and regions, respectively. These constraints calculate inventory at stage t based on the inventory (x_i^{t-1}, x_j^{t-1}) from stage $t-1$, procurement (z_i^t, z_j^t), transportation $(f_{ii'}^t, f_{ij}^t, f_{jj'}^t)$, rapid response (v_j^t), disposal (y_i^t, y_j^t), and consumption (u_j^t) at stage t. Among these decisions, disposal happens when inventory surpasses the capacity of hubs or warehouses in (3.6) and (3.7). (3.8) and (3.9) represent the prepositioning inventories, which are equal to the procurement at $t=0$. (3.10) and (3.11) define the domains for the decision variables of the shared hub network and regions, respectively. Here, $\mathbb{R}_0^+$ represents the set of nonnegative real numbers.

The model faces two challenges. First, though multi-stage stochastic problems with continuous distribution can be solved via discretized SAA models (see the Supporting Information), which can be directly fed to off-the-shelf solvers (e.g., Gurobi), the number of variables in the SAA models is related to the number of stages and scenarios sampled in each stage. For example, for a model with T stages and η scenarios sampled in each stage, the scale of variables is $\mathcal{O}(\eta^T)$. The size of decision variables grows exponentially with the number of stages, making the SAA formulation intractable. Therefore, the SAA formulation is typically unsuitable for real-world problems. Besides, even when the SAA formulation can be solved, the SAA solution is only *implementable* for the discretized problem but not for the original continuous problem. When the real observations of uncertain parameters fall out of the discretized scenario tree used for the SAA formulation, it is questionable how to apply the SAA solution (Ding 2020). Therefore, a sophisticated algorithm is needed. The SDDP method (Pereira and Pinto 1991) addresses the curse of dimensionality and generates an *implementable policy*. It is considered one of the state-of-the-art algorithms for multi-stage stochastic programming (Füllner and Rebennack 2025).

5. SDDP Method

This section proposes the SDDP (Pereira and Pinto 1991) solution method. The SDDP is a decomposition method that originated from stochastic dynamic programming and nested Benders decomposition (Birge 1985), it combines sampling techniques with cutting-plane approximations (Füllner and Rebennack 2025). The SDDP relies on *forward-simulation* steps to feed the trial solution from the current stage to the next stage, and *backward-recursion* steps to generate cutting planes from the current stage for the previous stage. In the forward-simulation steps, trial solutions are generated by sequentially solving a set of subproblems, each for a stage, with a sample path (scenario). In the backward-recursion steps, a set of scenarios is drawn from the underlying stochastic distribution. With these scenarios, we can approximate the expected value function of future costs by solving the dual of the subproblems and obtaining Benders cuts. In the forward-simulation steps, the SDDP solves stage-wise independent problems; in the backward-recursion steps, the SDDP solves stage-wise and scenario-wise independent problems. Therefore, the curse of dimensionality is broken. Shapiro (2011) proved that the SDDP forward step procedure defines an optimal *policy* defined as a sequence of mappings (each for a stage $t \in \mathcal{T}$) that each depends only on the data process up to time t (Shapiro et al. 2021). If the scenarios used in the SAA

and the SDDP are the same, the SDDP converges to the SAA solution after a sufficiently large number of iterations. The method differs from Pereira and Pinto (1991) by allowing convex (not necessarily linear) objectives. It combines the SDDP method (Pereira and Pinto 1991) with the generalized Benders decomposition method (Geoffrion 1972). The convergence was proved by Girardeau et al. (2015) for the case with both convex objective and convex constraints.

The SDDP method takes a set of discrete scenarios at each stage as input, and it outputs an optimal policy for the planning horizon, such that when a new observation of uncertain parameters is available, the policy can be directly applied to make decisions. While applying the policy is straightforward, i.e., sequentially solving a set of exact subproblems with the demand realized, we detail the policy training method below. The main components of the SDDP are the subproblem (Section 5.1) for each stage $t \in \mathcal{T}$, the forward-simulation and backward-recursion steps, and the Benders cuts generated in the backward-recursion steps (Section 5.2).

5.1. SDDP Reformulation

The model (3) is decomposed to $|\mathcal{T}|$ models SDDP^t ($t \in \mathcal{T}$). In stage $t = 0$, SDDP^0 determines the procurement decisions without post-disaster demand information. In stage $t \geq 1$, SDDP^t determines procurement, rapid response, disposal, transportation, and consumption decisions based on the decisions made in stage $t-1$. It uses demand information up to stage $t-1$ for procurement and up to stage t for the other decisions. When $\mathcal{T} = \{0, 1\}$ (a two-stage problem), SDDP resembles Benders decomposition applied to two-stage stochastic programming (Van Slyke and Wets 1969). The difference is that the SDDP samples a subset of scenarios to generate Benders cuts, while the Benders decomposition uses all the scenarios. The SDDP uses a solution trailing (forward) and cut generation (backward) process similar to the Benders decomposition. That is the SDDP^{t+1} takes the solution of SDDP^t as input and generates Benders cuts for SDDP^t, the role of SDDP^t and SDDP^{t+1} are master problem and subproblem, respectively, in the Benders decomposition. The stage $t = 0$ model, denoted as SDDP^0, is formulated as follows.

$$\min \sum_{i \in \mathcal{H}} c_i^t z_i^0 + \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{W}_r} c_j^t z_j^0 + \theta^0 \qquad [\text{SDDP}^0] \tag{4.1}$$

$$\text{s.t. } x_i^0 = z_i^0, \qquad \forall\, i \in \mathcal{H} \tag{4.2}$$

$$x_j^0 = z_j^0, \qquad \forall\, r \in \mathcal{R}, j \in \mathcal{W}_r \tag{4.3}$$

$$(6), \qquad [\text{Benders cuts}] \tag{4.4}$$

$$x_i^0 \leq q_i^0, \qquad \forall\, i \in \mathcal{H} \tag{4.5}$$

$$x_j^0 \leq q_j^0, \qquad \forall\, r \in \mathcal{R}, j \in \mathcal{W}_r \tag{4.6}$$

$$c_i^1 z_i'^0 \leq b_i^1, \qquad \forall\, i \in \mathcal{H} \tag{4.7}$$

$$c_j^1 z_j'^0 \leq b_j^1, \qquad \forall\, r \in \mathcal{R}, j \in \mathcal{W}_r \tag{4.8}$$

$$z_i'^0, z_j'^0 \in \mathbb{R}_0^+, \qquad \forall\, i \in \mathcal{H}, r \in \mathcal{R}, j \in \mathcal{W}_r \tag{4.9}$$

$$x_i^0, x_j^0, z_i^0, z_j^0 \in \mathbb{R}_0^+, \qquad \forall\, i \in \mathcal{H}, r \in \mathcal{R}, j \in \mathcal{W}_r \tag{4.10}$$

$$\theta^0 \in \mathbb{R}_0^+ \tag{4.11}$$

In the objective function (4.1), the expected future cost in (3.1) is replaced with θ^0, which denotes the expected future cost estimated at $t=0$. This term is approximated by Benders cuts during the iterations. Without loss of generality, (4.2) and (4.3) assume that the initial inventory is equal to the procurement quantity. (4.4) is the Benders cuts (detailed in Section 5.2) generated in the backward recursion steps for approximating θ^0. (4.5) and (4.6) are capacity constraints that ensure inventory does not exceed the capacity available. We introduce sets of auxiliary procurement variables $z_i^{'0}$ and $z_j^{'0}$ in (4.7)–(4.9) to ensure nonanticipativity. Since it is assumed that the procurement decisions are made before the demand is realized (decision-hazard variables), the procurement decisions at t should be independent from the demand realization at t, but only depend on the demand realization at $t-1$. This is ensured by letting the procurement decisions at t be equal to the auxiliary procurement decisions at $t-1$. Let us take $t=1$ as an example, there should be constraints $z_i^1 = z_i^{'0}$ and $z_j^1 = z_j^{'0}$ in the SDDP^1 model. These auxiliary variables are also introduced in SDDP^t $(t \geq 1)$. (4.7) and (4.8) ensure that the auxiliary procurement decisions are bounded by the budget constraints. (4.9)–(4.11) are domains of decision variables.

The inputs of SDDP^t $(t \geq 1)$ are the inventory $(\bar{x}_i^{t-1}, \bar{x}_j^{t-1})$ and auxiliary procurement decisions $(\bar{z}_i^{'t-1}, \bar{z}_j^{'t-1})$ from stage $t-1$ and the demand information from stage t. The SDDP^t $(t \geq 1)$ outputs procurement decisions $(z_i^t,\ z_j^t)$, transportation decisions $(f_{ii'}^t,\ f_{ij}^t,\ f_{jj'}^t)$, rapid response decisions (y_i^t, y_j^t), disposal decisions (v_j^t), and consumption decisions (u_j^t). The SDDP^t $(t \geq 1)$ is as follows.

$$\min\ \mathcal{Q}_{\mathcal{H}}^t + \sum_{r\in\mathcal{R}} \mathcal{Q}_r^t + \theta^t \qquad [\text{SDDP}^t] \qquad (5.1)$$

$$\text{s.t. } [\pi_i^t]\ x_i^t + y_i^t \leq \bar{x}_i^{t-1} + z_i^t + \sum_{i'\in\mathcal{H}} f_{i'i}^t - \sum_{i'\in\mathcal{H}} f_{ii'}^t - \sum_{j\in\mathcal{W}} f_{ij}^t, \qquad \forall\ i\in\mathcal{H} \qquad (5.2)$$

$$[\pi_j^t]\ x_j^t + y_j^t \leq \bar{x}_j^{t-1} + z_j^t + v_j^t + \sum_{i\in\mathcal{H}} f_{ij}^t + \sum_{j'\in\mathcal{W}_r} f_{j'j}^t - \sum_{j'\in\mathcal{W}_r} f_{jj'}^t - u_j^t, \qquad \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \qquad (5.3)$$

$$[\kappa_i^t]\ z_i^t = \bar{z}_i^{'t-1}, \qquad \forall\ i\in\mathcal{H} \qquad (5.4)$$

$$[\kappa_j^t]\ z_j^t = \bar{z}_j^{'t-1}, \qquad \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \qquad (5.5)$$

$$(6), \qquad [\text{Benders cuts}] \qquad (5.6)$$

$$x_i^t \leq q_i^t, \qquad \forall\ i\in\mathcal{H} \qquad (5.7)$$

$$x_j^t \leq q_j^t, \qquad \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \qquad (5.8)$$

$$c_i^{t+1} z_i^{'t} \leq b_i^{t+1}, \qquad \forall\ i\in\mathcal{H} \qquad (5.9)$$

$$c_j^{t+1} z_j^{'t} \leq b_j^{t+1}, \qquad \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \qquad (5.10)$$

$$z_i^{'t}, z_j^{'t} \in \mathbb{R}_0^+, \qquad \forall\ i\in\mathcal{H}, j\in\mathcal{W} \qquad (5.11)$$

$$x_i^t, z_i^t, f_{ii'}^t, f_{ij}^t, y_i^t \in \mathbb{R}_0^+, \qquad \forall\ i, i'\in\mathcal{H}, j\in\mathcal{W} \qquad (5.12)$$

$$x_j^t, z_j^t, f_{jj'}^t, u_j^t, y_j^t, v_j^t \in \mathbb{R}_0^+, \qquad \forall\ j, j'\in\mathcal{W} \qquad (5.13)$$

$$\theta^t \in \mathbb{R}_0^+ \qquad (5.14)$$

In the objective function (5.1), the definition of $\mathcal{Q}_{\mathcal{H}}^t$ and $\mathcal{Q}_r^t$ is the same as (2.1) and (2.2), θ^t is an estimator for the future costs, which is similar to the θ^0 in (4.1). (5.2) and (5.3) are flow

conservation constraints that replace inventory decision variables x_i^{t-1} and x_j^{t-1} in (3.4) and (3.5) with values $\bar{x}_i^{t-1}$ and $\bar{x}_j^{t-1}$ yielded by the previous stage, respectively. (5.4) and (5.5) are nonanticipativity constraints that ensure that the procurement decisions at stage t are independent of the demand realization at stage t, but only depend on the realization of stochastic demand process till stage $t-1$. (5.6) is the Benders cuts detailed in Section 5.2. (5.7) and (5.8) are capacity constraints that ensure inventory does not exceed the capacity available. (5.9)–(5.11) in stage $t \geq 1$ is similar to that of (4.7)–(4.9) in stage $t = 0$ for generating auxiliary procurement variables to ensure nonanticipativity. (5.12)–(5.14) are domains of decision variables. Tthe notations in square brackets in (5.2)–(5.5) represent the dual variables of these constraints, under specific $\bar{x}_i^{t-1}, \bar{x}_j^{t-1}$, and scenarios. These dual variable values are essential for generating Benders cuts.

5.2. Policy Training

The SDDP method iteratively updates Benders cuts to approximate θ^t, ultimately defining the optimal policy. The iteration mainly consists of forward-simulation and backward-recursion steps. The forward simulation solves the SDDP^t for each stage $t \in \mathcal{T}$ with a sample path drawn from the distribution of demand. The backward recursion solves the dual of SDDP^t for each stage $t \in \mathcal{T}_0$ with the trial solution from the forward simulation step and sample paths in the distribution of demand. The Benders cuts are built from the trial solution of SDDP^t and the sample average objective values and dual variable values of SDDP^{t+1}. The forward simulation, backward recursion, Benders cut generation, and stop criterion are presented in Algorithm 1.

The forward simulation step solves SDDP^t for $t \in \mathcal{T}$ using demand in a sample path drawn from $\mathcal{S}$, resulting in the trial solutions $\bar{x}_i^t$, $\bar{x}_j^t$, $\bar{z}_i^{'t}$, and $\bar{z}_j^{'t}$ for $t \in \mathcal{T}$. Then with these trial solutions, the backward recursion step solves SDDP^t for $t \in \mathcal{T}_0$ in a reverse order using all the sample paths from $\mathcal{S}$, resulting in the sample average (averaged over all $s \in \mathcal{S}$) dual values $\hat{\pi}_i^{t+1}$, $\hat{\pi}_j^{t+1}$, $\hat{\kappa}_i^{t+1}$, and $\hat{\kappa}_j^{t+1}$ associated with constraints, and sample average objective values $\hat{\mathcal{V}}^{t+1}$ for $t \in \mathcal{T}_0$. The Benders cut for SDDP^t is defined with ($\bar{x}_i^t$, $\bar{x}_j^t$, $\bar{z}_i^{'t}$, $\bar{z}_j^{'t}$, $\hat{\pi}_i^{t+1}$, $\hat{\pi}_j^{t+1}$, $\hat{\kappa}_i^{t+1}$, $\hat{\kappa}_j^{t+1}$, $\hat{\mathcal{V}}^{t+1}$) for $t \in \mathcal{T}_0$ as follows.

$$\theta^t \geq \hat{\mathcal{V}}^{t+1} + \sum_{i \in \mathcal{H}} \hat{\pi}_i^{t+1}(x_i^t - \bar{x}_i^t) + \sum_{j \in \mathcal{W}} \hat{\pi}_j^{t+1}(x_j^t - \bar{x}_j^t) + \sum_{i \in \mathcal{H}} \hat{\kappa}_i^{t+1}(z_i^{'t} - \bar{z}_i^{'t}) + \sum_{j \in \mathcal{W}} \hat{\kappa}_j^{t+1}(z_j^{'t} - \bar{z}_j^{'t}) \tag{6}$$

The Benders cut (6) approximates θ^t from below using linear inequalities. The information conveyed by the Benders cut is how the future costs (θ^t) will change if the decisions in the current stage ($\bar{x}_i^t$, $\bar{x}_j^t$, $\bar{z}_i^{'t}$, $\bar{z}_j^{'t}$) are changed in a future iteration. The scale of SDDP^t for $t \in \mathcal{T}_T$ increases with the number of cuts added. Larger SDDP^t requires more computation, though the accuracy of the approximation may be improved. Therefore, an appropriate stopping criterion is needed to balance this trade-off. An exact lower bound (`lb`) can be obtained directly from the objective value of SDDP^0. The upper bound can be estimated by conducting Mento Carlo simulation using the policy obtained and a set of sample paths $\mathcal{S}'$ independent from $\mathcal{S}$. The objective values from the $|\mathcal{S}'|$ Mento Carlo simulation replications are used to construct a confidence interval for the upper bound. We define the optimality gap as `(ci_l-lb)/ci_l`, where `ci_l` is the lower confidence bound and `lb` is the exact lower bound.

Algorithm 1: SDDP policy training method

Data: Distribution of demand Ω_r^t; time limit *Lmt.*.

Result: Approximation for θ^t for $t \in \mathcal{T}_T$ as $\mathfrak{Q}^t$.

def SDDP(*Lmt.*)**:**
  $\mu \leftarrow 0$; $\mathfrak{Q}^{t(\mu)} = \emptyset$ for $t \in \mathcal{T}_T$;
  Generate a set of discrete samples $\mathcal{S}$ from distribution of demand Ω_r^t ;
  while *Runtime* $\leq$ *Lmt.* **:**
    /* **Forward simulation** */
    Solve SDDP^0 (4) with cuts and store the values of $\bar{x}_i^0, \bar{x}_j^0$, $\bar{z}_i^{'0}$, and $\bar{z}_j^{'0}$;
    Draw a sample path s from $\mathcal{S}$;
    for t = *1, . . . , T* **:**
      Solve SDDP^t (5) with the values of $\bar{x}_i^{t-1}, \bar{x}_j^{t-1}$, $\bar{z}_i^{'t-1}$, and $\bar{z}_j^{'t-1}$, and demand s ;
      Store the values of $\bar{x}_i^t, \bar{x}_j^t$, $\bar{z}_i^{'t}$, and $\bar{z}_j^{'t}$ from SDDP^t ;
    /* **Backward recursion** */
    for t = *T, . . . , 1* **:**
      for $s' \in \mathcal{S}$ **:**
        Solve (the dual of) SDDP^t (5) with $\bar{x}_i^{t-1}, \bar{x}_j^{t-1}$, $\bar{z}_i^{'t-1}$, and $\bar{z}_j^{'t-1}$, and s' ;
        Store the objective value $\mathcal{V}^t(s')$ and dual values $\pi_i^t(s')$, $\pi_j^t(s')$, $\kappa_i^t(s')$, and $\kappa_j^t(s')$;
      /* **Cut generation** */
      Let $(\hat{\mathcal{V}}^t, \hat{\pi}_i^t, \hat{\pi}_j^t, \hat{\kappa}_i^t, \hat{\kappa}_j^t)^\top := \frac{1}{|\mathcal{S}|} \sum_{s' \in \mathcal{S}} (\mathcal{V}^t(s'), \pi_i^t(s'), \pi_j^t(s'), \kappa_i^t(s'), \kappa_j^t(s'))^\top$;
      Add cut (6) defined with $(\bar{x}_i^t, \bar{x}_j^t, \bar{z}_i^{'t}, \bar{z}_j^{'t}, \hat{\pi}_i^{t+1}, \hat{\pi}_j^{t+1}, \hat{\kappa}_i^{t+1}, \hat{\kappa}_j^{t+1}, \hat{\mathcal{V}}^{t+1})$;
      $\mathfrak{Q}^{t(\mu)} \leftarrow \{\mathfrak{Q}^{t(\mu-1)}, (6)\}$;
    /* **Stop criterion** */
    Calculate the optimality gap with a set of samples $\mathcal{S}'$ independent from $\mathcal{S}$;
    Terminate if the stop criterion is met ;
    $\mu := \mu + 1$
  return $\mathfrak{Q}^{t(\mu)}$ *for* $t \in \mathcal{T}_T$

The Algorithm 1 requires solving $|\mathcal{T}| \cdot |\mathcal{S}'|$ independent linear programming problems to estimate the upper bound and determine whether to stop. As references, the number of problems solved in the forward simulation is $|\mathcal{T}|$, and the number of problems solved in the backward recourse is $|\mathcal{T}| \cdot |\mathcal{S}|$. This shows that estimating the upper bound has a computational cost similar to the backward recursion, which is the most expensive part of the algorithm. Fortunately, upper bound estimation does not impact policy quality directly, but through determining when to stop the algorithm. To accelerate the algorithm, we can reduce the number of problems solved (determined by $|\mathcal{T}| \cdot |\mathcal{S}'|$ and estimation frequency) in the upper bound estimation step. In $|\mathcal{T}| \cdot |\mathcal{S}'|$, the former is determined by the problem inherently and the latter impacts the estimation accuracy. Therefore, we can not reduce the number of problems solved by reducing $|\mathcal{T}| \cdot |\mathcal{S}'|$. However, we can reduce the estimation frequency. Instead of checking the stop criterion in every iteration, we check it every N iterations. Specifically, the upper bound is estimated when μ (iteration index) mod $N = 0$.

6. Numerical Study

The SDDP method is implemented using Gurobi 11.0.1's Python API with Python 3.11.9. The machine that our code running on is a Virtual Private Server (VPS) with 8 cores of Intel Xeon Platinum 8369 (3.3 GHz, Turbo Boost speed of 3.8 GHz) vCPU, 16 GB RAM, and CentOS 7.9 operating system. We set the sample size $|\mathcal{S}'|$ to 500 (tuned in the Supporting Information)

for building a 95% confidence interval of upper bound, which is estimated every ten iterations, and the algorithm stops when the gap is less than 0.01% or the computation time exceeds 7,200 seconds (two hours). In this section, after generating instances in Section 6.1, Section 6.2 and Section 6.3 evaluate the in-sample and out-of-sample performance, respectively. Section 6.4 illustrates and analyzes the out-of-sample results. Section 6.5 compares the SDDP policy with two alternative policies yielded by myopic SAA and Rolling Horizon, which isolate dynamic and stochastic aspects, respectively. Section 6.6 shows the benefits of coordination. Section 6.7 conducts sensitivity analysis on several key parameters.

6.1. Instance Generation

In addition to node locations and demand distribution in Section 3.2, disutility function and cost parameters are also needed for instance generation. Recall that the disutility (1) with $\mathcal{PU}(u_j^t)$ be

$$\mathcal{U}_j^t(u_j^t) = \begin{cases} \gamma\rho_j^t \cdot \left[\frac{1}{\rho_j^t\delta_j^t} \cdot \left(\rho_j^t\delta_j^t - u_j^t\right)^+\right]^{\beta_j^t} & \text{if } \rho_j^t \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where u_j^t is the consumption quantity, γ is the welfare of satisfying per beneficiary, ρ_j^t is the people targeted, δ_j^t is the consumption rate measured as unit per beneficiary per stage, and β_j^t is a control parameter. Among them, u_j^t is a decision variable, and the others are parameters. We set $\gamma = \text{US\$}10^6$ (Guo and Zhu 2023, Holguin-Veras et al. 2016, Dore and Singh 2013), respecting the economic *value of life* — a monetary value on an individual's life for assessing damages due to loss of life (Dore and Singh 2013); ρ_j^t is set to the number of people affected, such that all the people affected are targeted; $\delta_j^t = 1$, which represents a portion of the WFP Food Basket that feeds a beneficiary for one month; and $\beta_j^t \in \{1, 2\}$. Then the right-hand side of (7) (if $\rho_j^t \neq 0$) simplifies to $10^6(\rho_j^t - u_j^t)^+$ for $\beta_j^t = 1$ and $10^6/\rho_j^t[(\rho_j^t - u_j^t)^+]^2$ for $\beta_j^t = 2$. By introducing an auxiliary nonnegative $(u_j^{'t} \geq 0)$ variable $u_j^{'t} \geq \rho_j^t - u_j^t$ representing shortage, we can express the disutility with $(\cdot)^+$ linearly as $10^6 u_j^{'t}$ or quadratically as $10^6/\rho_j^t[u_j^{'t}]^2$. The objective of our model trade-off disutility with cost under certain λ_j. Since our disutility function converts consumption into monetary value, we set $\lambda_j = 1$.

The procurement price, transportation price, budget, and capacity are also required for instance generation. The unit procurement cost was estimated at US\$13.13 (Peters et al. 2021) per beneficiary per month, with a weight of 16.89 kilograms (FSC 2022), based on the WFP Food Basket that feeds a seven-member household for a month. The local procurement (by countries or territories) price is assumed to be 50% higher (US\$13.13 $\times$ 150%) than the international procurement (by hubs) price due to economies of scale. The rapid response price is set at twice the procurement price (US\$13.13 $\times$ 200%). The estimated unit transportation prices are US\$0.18 (sea freight), US\$0.32 (land freight), and US\$6.24 (air freight) per kilometer per metric tonne (Stienen et al. 2021). Consequently, the cost per kilometer to transport enough relief items to sustain one beneficiary for a month is calculated as US\$0.18 $\times$ 16.89/1000 to US\$6.24 $\times$ 16.89/1000. We use the land freight (US\$0.32 $\times$ 16.89/1000) as it is cost effective for long-term planning (spanning months).

Though there are regions that are not connected by land, sea freight (priced similarly to land freight) can be used. Practically, for organizations not using the WFP Food Basket as a standard, procurement, rapid response, and transportation prices may vary from ours and are likely to change over time. It is worth noting that the SDDP policy is effective only when the deterministic parameters, including prices, closely match the true values. Therefore, we recommend accurately predicting these prices for each region and stage in the target year (e.g., the following year) when the trained policy is implemented. Each country or territory's capacity and budget are set to cover 100% and 70% of its average monthly demand, respectively, rounded up to the nearest integer. Hubs have unlimited capacity and budget. Potential inaccuracies in parameter estimates are addressed through sensitivity analysis in Section 6.7.

Let `M/S`-$|\mathcal{R}|$-`H/B`-$|\mathcal{S}|$ be the name of an instance, where `M/S` is the length of stages (month and season), $|\mathcal{R}| \in \{6, 3\}$ is the number of regional offices, `H/B` is the sampling method (historical records and bootstrap), and $|\mathcal{S}|$ is the number of samples. In the `M`-type instances, the planning horizon is 12 months, while in the `S`-type instances, the planning horizon is four seasons, meaning the `M`-type instances have 12 stages, and the `S`-type instances have four stages. The `S`-type instances accumulate the affected population in the three months of each season. The instances with $|\mathcal{R}| = 6$ include all the six OCHA regional offices, while the instances with $|\mathcal{R}| = 3$ include the three most disaster-frequent (from 2000 to 2022) regional offices: Southern and Eastern Africa, Asia and the Pacific, and Latin America and the Caribbean. The notation `H/B` indicates the sampling method for generating the demand samples. The `H`-type instances directly use historical records from 2000 to 2017 (other years are reserved for out-of-sample analysis in Section 6.3). For each month/season (`M`-type/`S`-type) there are 18 equal probability values ($|\mathcal{S}| = 18$), with each being exactly that month's/season's record. The `B`-type instances use the *bootstrap* (Efron 1979) resampling method to generate demand samples. For the `M`-type instances, the bootstrap method generates 12-stage sample paths by randomly selecting (with replacement) a demand record from each month's historical records, starting with January and ending with December. This procedure repeats till the required sample size $|S|$ is reached. Then in the `S`-type instances, demand samples are accumulated each season. The number of samples for the `B`-type instances is set to $|\mathcal{S}| \in \{64, 128, 256\}$ to analyze the impact of sample size on computational performance. We generate $|\{\texttt{M}, \texttt{S}\} \times \{6, 3\} \times \{\texttt{H}\} \times \{22\}| + |\{\texttt{M}, \texttt{S}\} \times \{6, 3\} \times \{\texttt{B}\} \times \{64, 128, 256\}| = 16$ instances.

6.2. In-sample Performance

This subsection evaluates the *in-sample performance* of the proposed method, as opposed to the *out-of-sample performance* in Section 6.3. In stochastic programming, there are two types of instances: in-sample instances and out-of-sample instances. The in-sample instance generates the policy, while the out-of-sample instance tests the performance of the policy obtained. With the instances generated in Section 6.1 and the algorithm proposed in Section 5, we can evaluate the performance of the proposed method on the in-sample instances. The in-sample performance focuses on the accuracy and efficiency of the proposed method.

Table 1 In-sample computational performance (linear)

Instance	LB ($\times 10^6$)	UB ($\times 10^6$)	Gap (%)	Time (s)	Time' (s)	Iter.	$\frac{\text{Time}}{\text{Iter.}}$	$\frac{\text{Time'}}{\text{Iter.}}$
S-3-H-018	2290.30	2362.75 ± 133.89	$\leq$0.00	13	3	20	0.67	0.15
M-3-H-018	2222.97	2242.25 ± 71.10	$\leq$0.00	176	38	80	2.20	0.47
S-6-H-018	2678.60	2805.59 ± 139.83	$\leq$0.00	68	15	50	1.35	0.29
M-6-H-018	2541.44	2550.25 ± 81.02	$\leq$0.00	656	133	140	4.68	0.95
S-3-B-064	2187.57	2210.69 ± 43.01	$\leq$0.00	53	27	50	1.06	0.53
M-3-B-064	2203.39	2233.03 ± 42.82	$\leq$0.00	464	229	130	3.57	1.76
S-6-B-064	2499.88	2529.80 ± 45.84	$\leq$0.00	151	74	70	2.15	1.05
M-6-B-064	2518.33	2539.42 ± 44.33	$\leq$0.00	1384	651	190	7.28	3.43
S-3-B-128	2237.67	2240.50 ± 42.32	$\leq$0.00	97	65	60	1.62	1.08
M-3-B-128	2235.46	2277.66 ± 45.95	$\leq$0.00	400	260	80	5.00	3.26
S-6-B-128	2518.68	2542.66 ± 47.24	$\leq$0.00	223	147	70	3.19	2.10
M-6-B-128	2544.91	2589.91 ± 50.01	$\leq$0.00	2763	1770	250	11.05	7.08
S-3-B-256	2185.76	2216.51 ± 41.33	$\leq$0.00	104	83	40	2.61	2.09
M-3-B-256	2208.04	2231.18 ± 42.61	$\leq$0.00	1347	1068	150	8.98	7.12
S-6-B-256	2510.09	2540.59 ± 45.44	$\leq$0.00	488	386	90	5.42	4.29
M-6-B-256	2520.48	2540.51 ± 46.37	$\leq$0.00	3915	3046	220	17.80	13.84
Mean	2381.47	-	-	769	500	106	4.91	3.09

[1] `LB`: lower bound; `UB`: 95% confidence interval of upper bound; `Gap`: $(UB_l - LB)/UB_l$, where UB_l is lower bound of UB; `Time`: time that the algorithm is terminated; `Time'`: running time excluding upper bound estimation.

In Table 1, the columns "LB" reflect the lowest cost that possibly be reached, while the columns "UB" are 95% confidence intervals when applying the policy to new samples. "Gap" calculates the relative difference between the upper bound and the lower bound. Two types of computation time are reported: "Time" is the total time that the algorithm is terminated, while "Time' " is the running time that excludes bound estimation. We also present the number of iterations and the average time per iteration in the columns "Iter.", "$\frac{\text{Time}}{\text{Iter.}}$", and "$\frac{\text{Time'}}{\text{Iter.}}$". The results here are obtained from the SDDP method. We tried to solve these instances with the SAA formulation. However, even for the smallest instance S-3-H-18, the machine that our code running on (with 18 GB of RAM) raised an out-of-memory error. The computational efficiency of the SDDP method compared with the SAA formulation is thus evident. All the instances with linear disutility reach our stop criterion (0.01%) in hundreds of iterations within the time limit (7,200 seconds), and the gaps are less than zero, indicating the lower bounds have fallen into the confidence intervals of the upper bounds. However, the computation time for instances varies.

The instance names classify the instances into different categories: {S/M} (number of stages), {3/6} (number of regions), and {18/64/128/256} (sample size). The {H/B} (sampling method) is neglected here because it does not impact the scale of the instances. The numerical results in Table 1, as well as theoretical analysis, reveal these three factors all contribute to the computation time (Figure 3), but in different ways. As the number of stages and sample size increase, the number of problems per iteration (forward, backward, and upper bound estimation steps) rises linearly, while the number of regions adds to the computational burden in these steps, as in the columns "$\frac{\text{Time}}{\text{Iter.}}$" and "$\frac{\text{Time'}}{\text{Iter.}}$". Figure 4 shows under all sample size settings, the lower bound decreases to 10% of the upper bound in less than 50 iterations, while the computation time differs significantly. Among forward, backward, and upper bound estimation steps, the time for forward

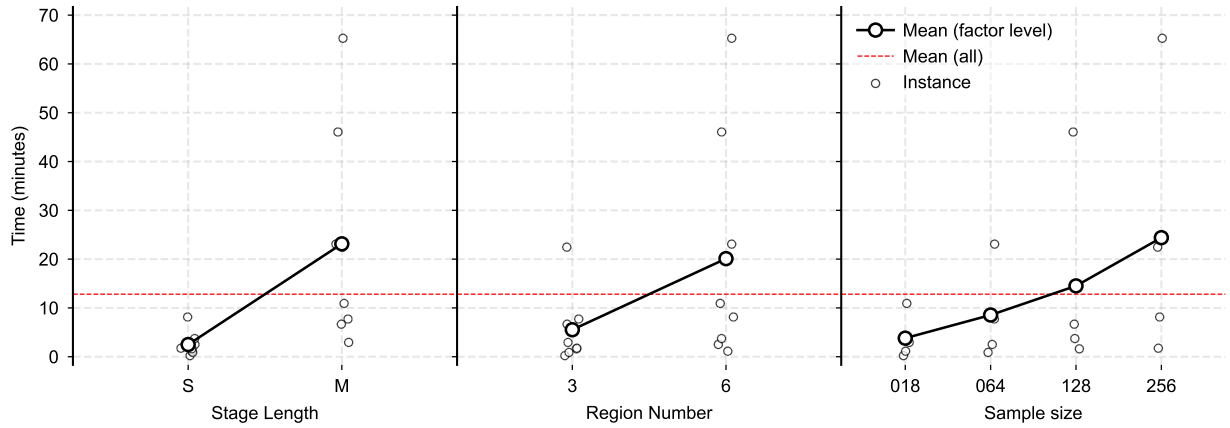


Figure 3 **Main effects on computation time.**

simulation is the least, while the time for backward recursion is the most. Moreover, backward recursion time increases more rapidly with larger sample sizes because more cuts are generated, resulting in larger subproblems in the next iteration. An interesting observation from Table 1 and Figure 4 is that the upper bound estimation can consume over half of the total computation time. This suggests that the algorithm can be accelerated without losing accuracy by further reducing the frequency of upper bound estimation, as discussed in Section 5.2.

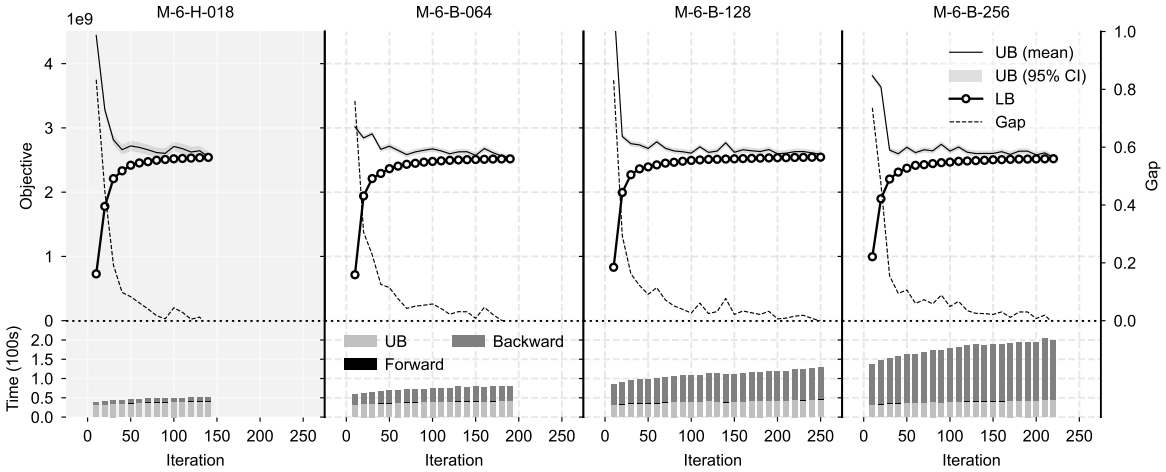


Figure 4 **Convergence curve and computation time.**

The computational results with quadratic disutility ($\beta_j^t = 2$) are presented in Table 2. The quadratic disutility alters the model structure compared to the linear case, resulting in lower LB values and increased computation times. The lower objective values in the quadratic case arise because the quadratic disutility $10^6/\rho_j^t[u_j^{'t}]^2$ (ρ_j^t represent the number of people affected, $u_j^{'t}$ denotes the shortage) is always not larger than the linear $10^6 u_j^{'t}$ when $u_j^{'t} \in [0, \rho_j^t]$. In computation times, the quadratic disutility models take longer than the linear models due to more iterations and longer per-iteration duration. The SDDP method approximates future costs using Benders cuts. Nonlinear (quadratic) disutility requires more cutting planes for accurate approximation than linear disutility. For instance, while it is possible to "approximate" $10^6 u_j^{'t}$ by a single linear cutting plane, capturing the curvature of $10^6/\rho_j^t[u_j^{'t}]^2$ requires more cutting planes. The number of cutting planes required directly correlates with the number of iterations, as each iteration

adds a fixed number of cutting planes. Another bottleneck is that Quadratic Programming — a subroutine of SDDP in the quadratic case — is generally slower than Linear Programming. In the quadratic case, the forward simulation, backward recursion, and upper bound estimation steps (which nearly form a complete SDDP method) are all more computationally intensive. The following subsections employ linear disutility for efficiency.

Table 2 In-sample computational performance (quadratic)

Instance	LB ($\times 10^6$)	UB ($\times 10^6$)	Gap (%)	Time (s)	Time' (s)	Iter.	$\frac{\text{Time}}{\text{Iter.}}$	$\frac{\text{Time'}}{\text{Iter.}}$
S-3-H-018	2315.74	2423.65 ± 126.94	$\leq$0.00	167	45	60	2.79	0.75
M-3-H-018	2219.41	2283.17 ± 79.37	$\leq$0.00	3677	965	250	14.71	3.86
S-6-H-018	2653.57	2689.94 ± 130.44	$\leq$0.00	399	102	70	5.70	1.46
M-6-H-018	2535.82	2635.52 ± 78.43	0.83	7532	1974	280	26.90	7.05
S-3-B-064	2178.97	2215.58 ± 40.36	$\leq$0.00	454	244	100	4.54	2.44
M-3-B-064	2198.41	2274.22 ± 42.07	1.51	7494	4111	280	26.76	14.68
S-6-B-064	2509.57	2542.34 ± 46.43	$\leq$0.00	2049	1117	180	11.38	6.20
M-6-B-064	2488.47	2614.08 ± 44.13	3.17	7636	4171	210	36.36	19.86
S-3-B-128	2231.43	2252.99 ± 43.20	$\leq$0.00	804	563	110	7.31	5.11
M-3-B-128	2236.62	2345.41 ± 45.78	2.74	7586	5340	220	34.48	24.27
S-6-B-128	2524.21	2567.93 ± 45.79	$\leq$0.00	2455	1723	160	15.34	10.77
M-6-B-128	2479.64	2674.61 ± 49.14	5.55	7835	5484	160	48.97	34.27
S-3-B-256	2185.47	2215.47 ± 40.98	$\leq$0.00	1401	1152	110	12.74	10.47
M-3-B-256	2175.85	2336.79 ± 40.75	5.24	7490	6168	160	46.81	38.55
S-6-B-256	2496.43	2537.89 ± 43.04	$\leq$0.00	3675	3035	130	28.27	23.35
M-6-B-256	2395.83	2737.54 ± 41.33	11.14	7572	6243	100	75.72	62.43
Mean	2364.09	-	-	4264	2652	161	24.92	16.60

[1] See the notes in Table 1. The column "Time" may exceed 7,200 seconds because the stop criteria is checked every ten iterations.

We introduce instances with larger sample sizes using the bootstrap method for two reasons. First, to analyze how the number of samples affects the solution process, such that we can anticipate the algorithm behavior when the proposed method is applied to instances with a larger number of available historical data. Second, the bootstrap method may yield better policies than directly using historical records. The first has been analyzed above, but it is unclear if increasing sample sizes will improve out-of-sample solution quality, given the significant increase in computation time. We discuss the second in Section 6.3.

6.3. Out-of-sample Performance

We use the EM-DAT (CRED 2024) from 2000 to 2017 (18 years, around 80% of 23 years) to generate the in-sample instances (Section 6.1) and evaluate the in-sample performance of the proposed method (Section 6.2). The data from 2018 to 2022 (five years, around 20% of 23 years) that are not used for policy optimization and it can be used to test the performance of the policy obtained, i.e., *out-of-sample performance* analysis. A good policy should not only have low costs and small gaps (as in Table 1) but should also can be generalized to new instances that are not used for optimization. This is known as the trade-off between *underfitting* and *overfitting*, and *generalization error* in machine learning.

We aim to answer the following questions in this subsection. First, *given the effort required for algorithm convergence, how does the policy perform on out-of-sample instances during iterations?* For larger instances, the algorithm may not converge within a reasonable time, we would like to know if it is safe to terminate the algorithm early without sacrificing the solution quality on out-of-sample instances. Second, *while larger sample sizes of in-sample instances may improve the policy obtained, how do they impact the quality of out-of-sample solutions?* This is important because, in the backward-recursion step in Algorithm 1, the number of problems solved is proportional to the sample size at each stage. A larger sample size, on the one hand, may lead to a better policy, on the other hand, will definitely result in longer running times. A balance is needed between these two contradictory aspects.

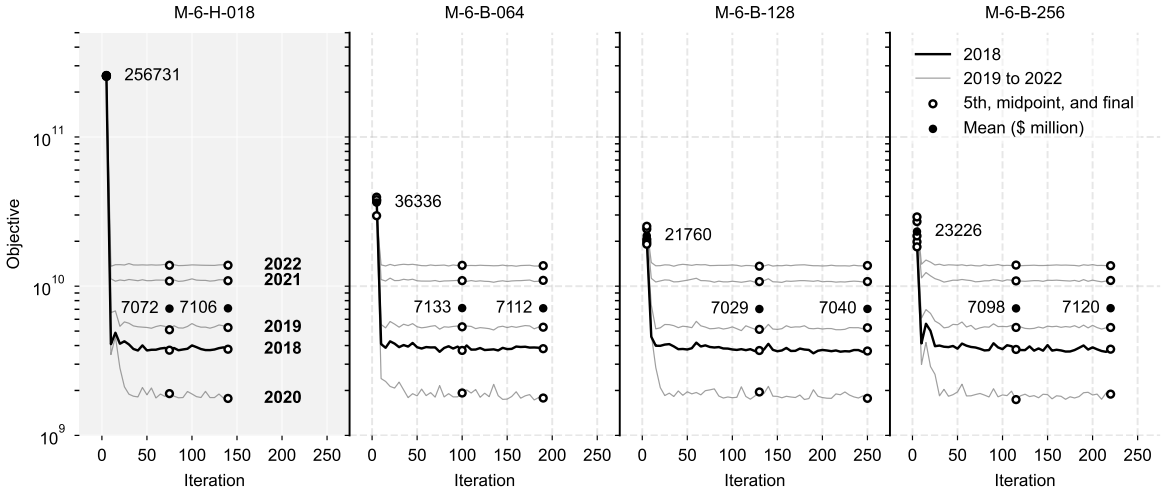


Figure 5 **Out-of-sample performance during iterations.**

The first question is answered by simulating using the out-of-sample instances (five sample paths from 2018 to 2022) and the policy with only Benders cuts (6) obtained till a certain iteration. For the family of instances M-6-*-***, this simulation is conducted every five iterations. In each simulation on the testing instances, we solve SDDP^0 to SDDP^T forwardly with the policy obtained in the μth iteration, where $\mu \bmod 5 = 0$, and record the values of objective and θ^t of SDDP^t ($t \in \mathcal{T}$). Then the actual expense at each stage is the objective value of SDDP^t minus θ^t. This is illustrated in Figure 5, where each line represents a sample path (year). In addition, we present the costs of all five sample paths and their mean at the 5th iteration, the midpoint, and the final iteration. We can observe that, under all the sample sizes (18, 64, 128, and 256), the out-of-sample costs decrease significantly from the 5th iteration to the midpoint, while the decrease from the midpoint to the final iteration is not as significant. In addition, the out-of-sample costs at less than 50 iterations have already been close to the final iteration. Similar trends appear in other instances (see the Supporting Information). This trend is resulted from the Benders cuts added. Initially, insufficient Benders cuts result in myopic decisions that incur high disutility from future inventory shortages. As more Benders cuts are added, each stage's model becomes more forward-looking due to improved approximation of future costs (via Benders cuts) in the current

stage. Out-of-sample stability requires fewer iterations than convergence (when LB meets UB in Figure 4), as a policy with only a subset of Benders cuts can perform well on a single sample path. Therefore, we can answer the first question: it is safe to terminate the algorithm early without significantly sacrificing the solution quality on out-of-sample instances.

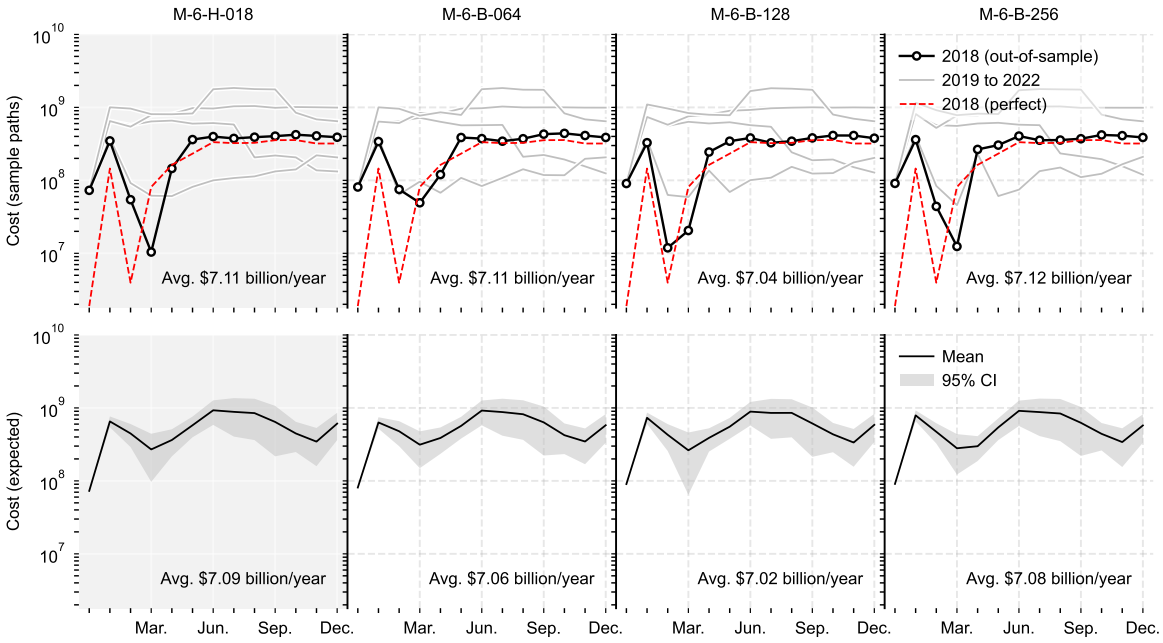


Figure 6 **Out-of-sample performance using the final policy.**

To answer the second question — how does sample size affect the out-of-sample solution quality? — we simulate the out-of-sample instances with the final policy obtained in the last iteration. The first row of Figure 6 (see the Supporting Information for other instances) applies the policy to data from 2018 to 2022. We also provide a perfection information lower bound (dashed lines) for 2018 by generating an instance with only one sample per stage (the 2018 demand) and solving it with the proposed method. While the first row of Figure 6 shows the near-future out-of-sample performance directly using sample paths from 2018 to 2022, the second row represents the long-term out-of-sample performance, assuming future demand has similar statistical properties to 2018–2022. We resample ten synthetic years, where each year's monthly (quarterly) demand is selected from the five scenarios (2018–2022) with equal probability. Applying the policy to these synthetic years evaluates its performance beyond the 2018–2022 period.

In Figure 6, the first column uses the policy obtained directly with historical records as the in-sample instances, while the following columns use the policy obtained with bootstrapped samples, with sizes ranging from 64 to 256. A counterintuitive observation is that increased sample sizes do not necessarily lead to lower out-of-sample costs (the cost differences in Figure 6 are more likely stem from stochasticity, rather than the policies obtained), despite that more samples mean more cuts in each iteration. For example, an instance with a sample size being 128 doubles the cuts generated in each iteration compared with an instance with a sample size being 64. However, an advantage of the bootstrap method compared with directly using historical records can be found in Figure 5. In the early iterations (less than 50), larger sample sizes lead to lower out-of-sample

costs. Because more cuts — coded information in the mathematical programming model from the demand distribution — are generated. The similarity in trends between the costs in 2018 using our method and the perfect information benchmark demonstrates the quality of the policy obtained, under all sample sizes in Figure 6. Therefore, we can answer the second question: in the final iteration, bootstrap and historical records lead to similar out-of-sample outcomes; while in the early iterations, bootstrap method with larger sample sizes leads to lower out-of-sample costs. For hard instances, there is a trade-off between using larger sample sizes and terminating early, and using smaller sample sizes and running till convergence.

6.4. Results Illustration and Analysis

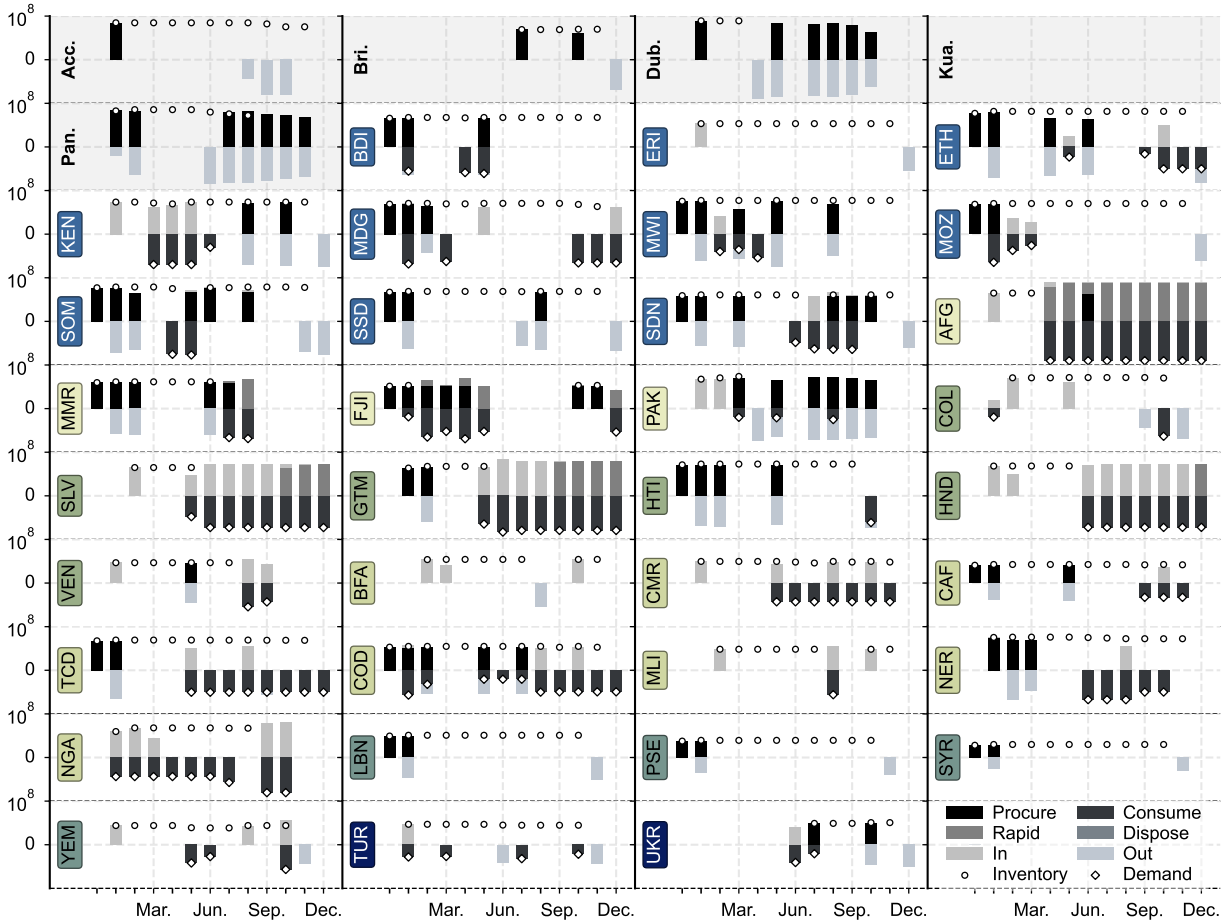


Figure 7 **Out-of-sample decisions in 2018 with the policy yielded by M-6-H-018. It shows the decisions in the five UNHRD Hubs and 34 countries or territories (represented by specific three-letter codes). In each subplot, elements above zero represent inventory and inflow (procurement, rapid response, and transport in), while those below zero indicate demand and outflow (consumption, disposal, and transport out).**

Though Section 6.2 and Section 6.3 provide insights from both in-sample and out-of-sample perspectives, examining the decisions made by the resulting policy can be more informative. This subsection analyzes the decision of 2018 with the policy obtained from M-6-H-018. The year 2018 is chosen because it is the most recent year after the years (from 2000 to 2017) in the in-sample instances. This means when making decisions for it, the only available information is the in-sample

data used for policy optimization. while for the following years (e.g., 2019), one can obtain a policy with more data (the data from 2018 can also be used). In other words, the solutions by the policy obtained in Section 6.2 and Section 6.3 for the years from 2019 have potential to be improved by using more recent data. The instance M-6-H-018 is chosen because it achieves a good efficiency and quality trade-off among the instances M-6-*-** (monthly plan for the globe), as Table 1 and Figure 6 show. It is the most likely choice for decision-makers. Therefore, the 2018 results with the policy obtained from instance M-6-H-018 can be considered representative for real-world conditions. The results are illustrated in Figure 7 (the codes of countries or territories and detail relief item flow are presented in the Supporting Information), with the procurement, consumption, transportation, and other decisions, for the five UNHRD Hubs and 34 countries or territories across six regions. We summarize the observations and insights as follows.

No shortage and zero waste. Recall that our model includes regular procurement (before disasters), rapid response (after disasters), transportation among all nodes, and disposal of relief items, offering a flexible and reliable decision-making tool for humanitarian organizations. With this tool, in the year 2018, using the policy optimized with data from 2000 to 2017, there is no shortage of relief items in any country or territory, and no relief item is wasted. In other words, for all 34 countries or territories, the consumption variable values are all equal to the demand in all 12 months, and all disposal variable values are zero. No shortage is desirable since meeting the demand is the primary goal of humanitarian organizations, while zero waste is also vital because humanitarian organizations usually rely on donations, implying the budget is usually limited and the social image is always valuable.

Procure wisely not excessively. The disaster-affected population is imbalanced spatially and temporally. During the year, the countries or territories with the highest demand are Afghanistan (AFG), El Salvador (SLV), Guatemala (GMT), and Honduras (HND), in the last three quarters. Among them, Afghanistan is located in Asia, and the others are located in Latin America and the Caribbean. Even under our assumption that the hubs have unlimited capacity and budget, the procurement of relief items is not excessive. Most of the relief items are procured by the hubs in Dubai (Dub.), and Panama (Pan.) from April to the end of the year, while the others procure less because Dubai and Panama are closer to the countries or territories with the highest demand. This observation suggests that, though procuring and prepositioning of relief items is important, it should be done wisely, not excessively, to avoid financial inflexibility and waste.

Implement preventive transportation. In some countries or territories, transportation occurs before the demand occurs. For example, in Venezuela (VEN) and Mali (MLI), relief items are transported to them at the beginning of the year before the third quarter, when demand occurs. In Lebanon (LBN), Occupied Palestinian Territory (PSE), Syrian Arab Republic (SYR), and Yemen (YEM), relief items are transported to them at the beginning of the year but are not consumed till the end of the year in most of them, except for Yemen. This decision is made because based on the historical records, which have been coded as cuts in the policy, there is a chance that demand will occur. These preventive transportation decisions are in alignment with the *anticipatory action* (OCHA 2022) spirit.

6.5. Value of Stochastic Dynamic Solutions

The previous subsections analyzed the multi-stage stochastic model and the SDDP method themselves, but it remains unclear whether implementing this complex (dynamic and stochastic) approach is worthwhile. This subsection compares the SDDP policy with alternative (simpler, static or deterministic) policies, inspired by two well-established methods — the SAA method and the Rolling Horizon method. The original SAA method builds a scenario tree and solves the deterministic equivalent model on this tree. Though being a standard for stochastic programming, the SAA method is computationally expensive and often infeasible for real-world-scale problems. Take instance M-*-*-018 as an example, the number of sample paths (leaf nodes) on the scenario tree is 18^{12}. We mentioned in Section 6.2 that it is impossible to solve even smaller models under our computation resources. Therefore, we propose a *myopic* SAA policy, which only use scenarios in the current stage, significantly reducing the computational burden by ignoring the dynamic aspect of our problem. The Rolling Horizon policy is also implemented, recurrently optimizing deterministic multi-period problems over a specified look-ahead period. It ignores the stochastic aspect. We can evaluate the value of stochastic dynamic solutions by comparing the SDDP policy with the others, detailed below (see the Supporting Information for mathematical models).

- (Non-dynamic) *Myopic SAA policy:* At each stage, we solve a two-stage stochastic programming problem. The first stage corresponds to the decision-hazard decisions (e.g., procurement, transportation, etc.) under unknown demand. The second stage includes the hazard-decision decisions (e.g., rapid response, transportation, and consumption), capturing demand uncertainty through discrete scenarios. The objective minimizes the here-and-now costs plus the expected future costs after demand realization. After solving this model, only the first-stage decisions are implemented. The system state is then updated with the realized demand, and the process is repeated for the subsequent stages. This approach is stochastic but short-sighted.
- (Non-stochastic) *Rolling Horizon policy:* At each stage t, we solve a deterministic multi-period model that spans a look-ahead window of H future stages. Within this model, the demands for all future stages are fixed at their expected values, ignoring all demand variability. The model minimizes the total costs over the entire H-period horizon. Following the standard Rolling Horizon procedure, only the decisions for the current stage t are executed. The system then advances to stage $t+1$, the state is updated with the true out-of-sample demand, and a new H-period deterministic problem is solved from the new state. This policy is dynamic, as it considers a future look-ahead period of length H, but it is not stochastic.

The SDDP policy is compared with these two baseline policies on instance M-6-H-018, as illustrated in Figure 8. The experiment is conducted by generating 100 sample paths and simulating using the SDDP policy, the myopic SAA policy, and the Rolling Horizon policy ($H = 2$ and $H = 6$). We ensure these four policies use the same set of sample paths for a fair comparison. Though all the policies are efficient enough for practical use (each simulation takes less than one second), they vary in cost performance and procurement patterns. The first row of Figure 8 displays monthly costs, covering procurement, rapid response, and transportation. The SDDP policy

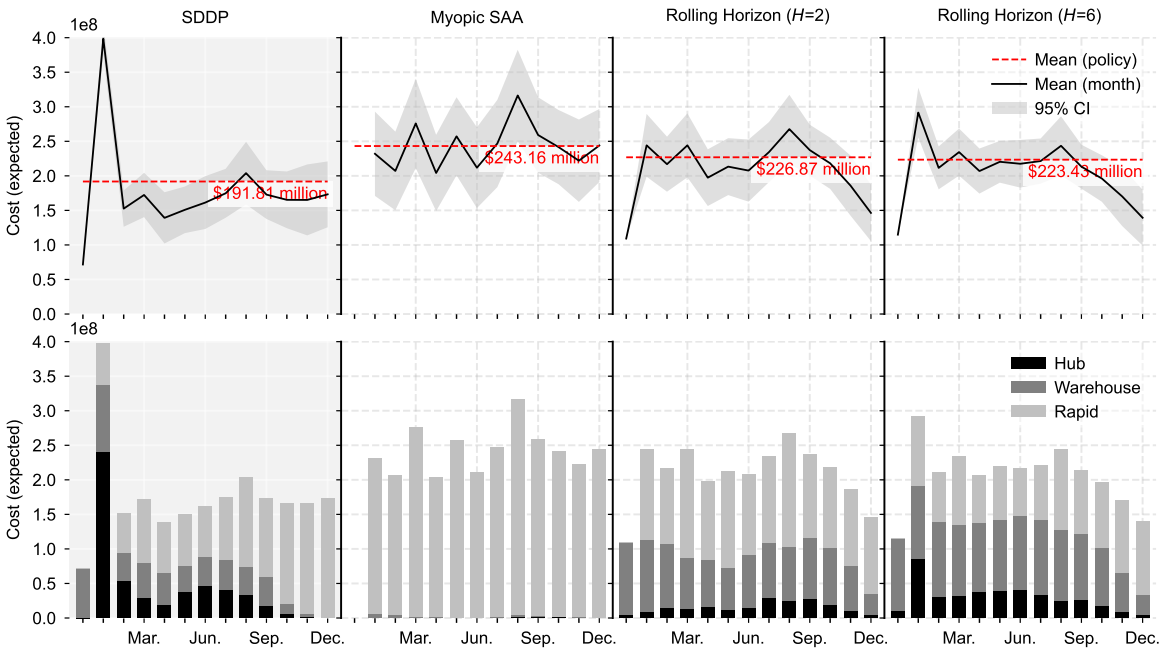


Figure 8 **Comparing SDDP with baseline policies.**

has significantly lower costs: US$51.35 billions saving per month (on average) compared with the myopic SAA policy, which means US$616.2 million (21%) saving per year. The SDDP policy also outperforms the Rolling Horizon policy, under both $H = 2$ and $H = 6$. This observation is unsurprising because the SDDP focuses on minimizing expected costs over the planning horizon, considering dynamics and uncertainties that are ignored by myopic SAA or Rolling Horizon policies. The second row of Figure 8 presents that these policies have distinct procurement patterns. In the initial stage, the SDDP policy procures high levels from lower-priced hubs because it has access to all available information coded as Benders cuts and considers these early procurement decisions safe. The myopic SAA policy aggressively procures using the rapid response mechanism, which is priced at twice the standard procurement, resulting in higher overall expenses. This behavior occurs because not all scenarios in each stage of the myopic SAA have nonzero demand. Consequently, it is reasonable to procure less until demand is confirmed, at which point rapid response is used. The Rolling Horizon policies ($H = 2$ and $H = 6$) achieve a balanced distribution of procurement across different times and methods. However, although expanded H yields slightly lower costs, they remain higher than those of the SDDP policy. The SDDP policy outperforms both the myopic SAA, which overlooks dynamics, and the Rolling Horizon policy, which ignores stochasticity, demonstrating the value of stochastic dynamic solutions.

6.6. Benefits of Coordination

As demonstrated in the title, "coordination" is one of the most important aspects of the problem of interest. In the mathematical programming model, coordination means balance of relief inventory levels among the hubs and regions considered. The coordination is achieved by relief item transportation decisions among the hubs and regions, through the planning horizon. In our initial problem settings, it is assumed that the following transportation is allowed: (i) among hubs; (ii) from hubs to countries or territories; and (iii) among countries or territories. That is

except reverse transportation, all other transportation modes — vertical and horizontal — are allowed. However, administrative boundaries and geographical limitations can restrict certain transportation modes. This subsection investigates the benefits of coordination, by comparing the out-of-sample solutions with different transportation mode settings for policy training. We conduct further experiments by putting restrictions on transportation variables: whether to allow inter-region transportation (Yes/No) and whether to allow intra-region transportation (Yes/No). There are six regions as Figure 1 shows. The inter-region transportation refers to transportation among warehouses belongs to different $\mathcal{W}_r, r \in \mathcal{R}$, whereas the intra-region transportation refers to transportation among warehouses belongs to the same $\mathcal{W}_r, r \in \mathcal{R}$. In the above subsection, our experiments use the "Yes-Yes" ("Y-Y") combination. We then generate policies using the "No-Yes" ("N-Y") combination (disallowing inter-region but allowing intra-region transportation) and the "No-No" ("N-N") combination (disallowing both inter-region and intra-region transportation). The "Y-N" combination is excluded because it is unreasonable to allow inter-region transportation but disallow intra-region transportation. These combinations are implemented by setting the upper bounds of the transportation variables to zeros, and remaining other settings unchanged.

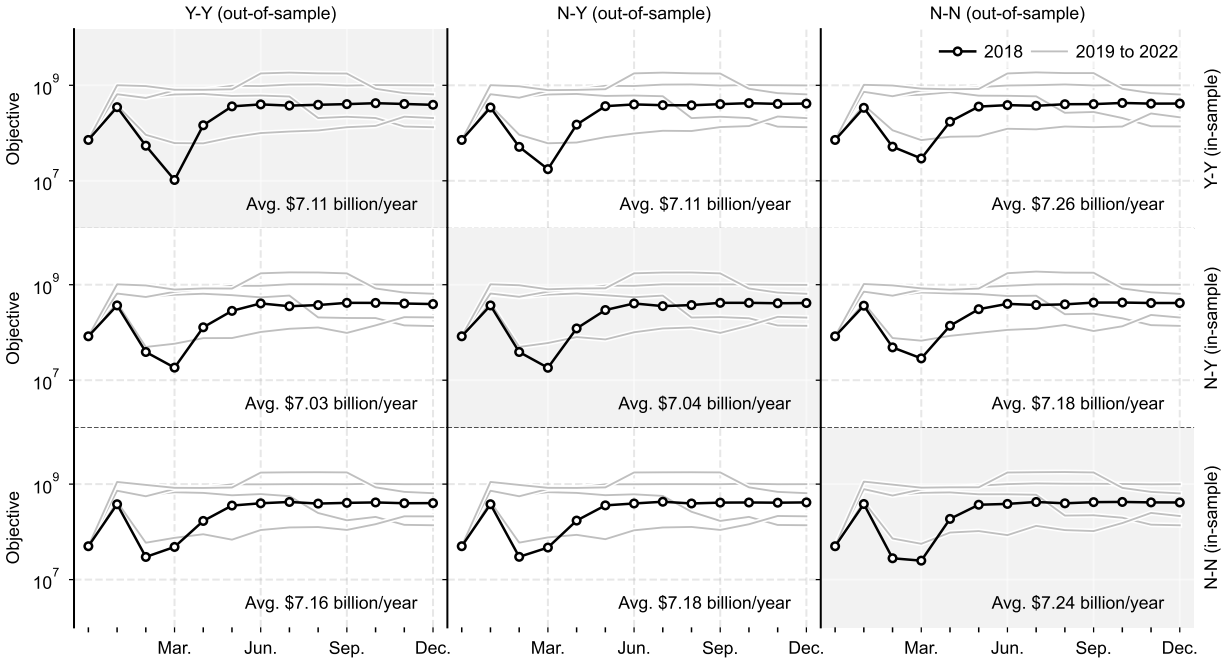


Figure 9 **Benefits of coordination on instance M-6-H-018.**

With the three policies obtained, we implement the policies with the out-of-sample data from 2018 to 2022. When simulating the out-of-sample instances, we also pose the restrictions on transportation variables as in policy training. $3 \times 3 = 9$ out-of-sample simulation results are illustrated in Figure 9. The most significant observation is that among all the rows, the first column ("Y-Y") always has the lowest costs, while the last column ("N-N") always has the highest costs. This observation demonstrates that allowing coordination broadly (inter-region and intra-region) is always beneficial, regardless of whether it is allowed when training the policies. The "Y-Y,

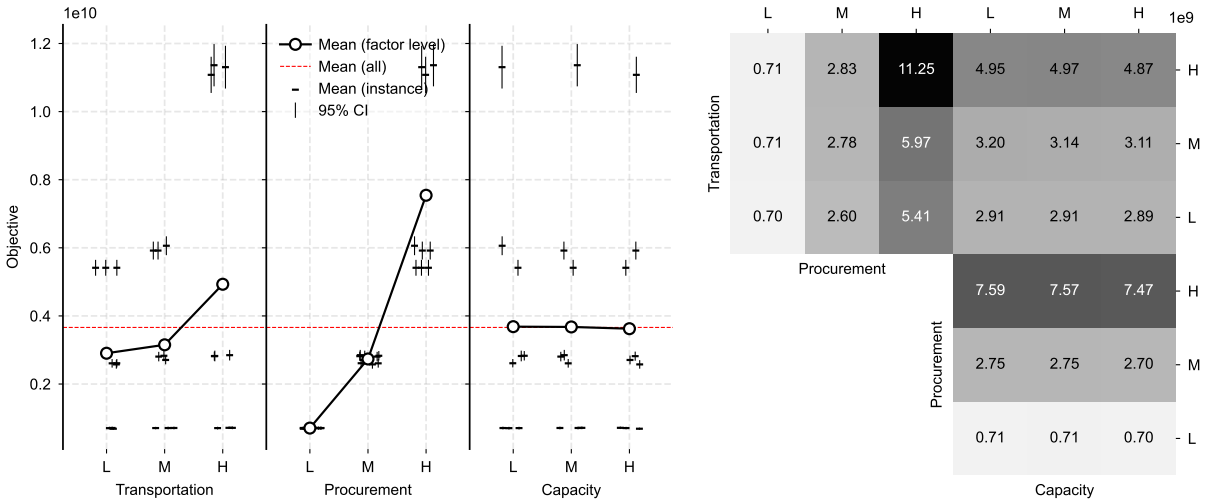
Y-Y" case results in a cost of US$7,106 million and the "N-N, N-N" case results in a cost of US$7,240 million, meaning the benefit of coordination is US$134 million, which is around 1.9% of the total cost without coordination. Another interesting observation is, in all the cases for policy training (rows), the "N-Y" cases (no inter-region transportation but have intra-region transportation) always have the lowest cost no matter whether coordination is allowed when implementing the policies. This observation suggests that the current OCHA administrative boundary settings, which are primarily based on geographical locations, are reasonable and effective. Without modifying the OCHA organizational structure, the cost can be close enough to the lowest cost, i.e., US$7044 million ("N-Y, N-Y") versus US$7025 million ("N-Y, Y-Y"), with a difference of around 0.3%. Our observations demonstrate that while broad coordination is beneficial, good performance can still be achieved within the current OCHA organizational structure.

6.7. Sensitivity to Parameters

In Section 6.1 several parameters are involved for the case study, such as transportation price, procurement (hubs, countries or territories, and rapid response) price, budget, and capacity of countries or territories. Accurately determining these parameters can be challenging, as values may vary by region and time period. Therefore, we use main effect and interactive effect analysis — statistical methods used to identify the primary factors that contribute to an observed outcome — to investigate the sensitivity of the solution to these parameters. Experiments are conducted by varying one factor at a time while holding all other factors constant, and measuring the effect on the outcome (objective). Specifically, we set three levels (low, medium, and high) for transportation-related parameters (transportation price), procurement-related parameters (hub, country or territory, and rapid response price), and capacity-related parameters (budget and capacity), and run our method on the instance S-6-H-018 with the $3^3 = 27$ sets of parameters. Other parameters (e.g., demand) are fixed to the ones in Section 6.1. The instance S-6-H-018 is chosen because the seasonal instances require less computation time compared with the monthly instances, while their objective values show similarities. Before presenting the results, we first modify the parameters as follows.

- *Transportation price:* In the baseline (the ones used in the previous subsections, and the medium-level (`M`)) instances, we used the land freight price US$0.32 $\times$16.89/1000. The low-level (`L`) and high-level (`H`) instances use the sea freight price US$0.18 $\times$16.89/1000 and air freight price US$6.24 $\times$16.89/1000, respectively. Note that the unit of these prices is the cost per kilometer for supplying the items required by one beneficiary for one month.
- *Procurement price:* In the baseline instances, procurement price by hubs is US$ 13.13, by countries or territories is US$ 13.13$\times$1.5, and by rapid response is US$ 13.13$\times$2. Factors of 0.5 and 2 are multiplied by these values for the low-level and high-level instances, respectively. Note that the unit of these prices is the cost of items to feed one beneficiary for one month.
- *Capacity:* In the baseline instances, each country or territory's budget and capacity are set to be 0.7 and 1.0 times the monthly average demand from 2000 to 2017. In the high-level instances,

the factors remain the same, but using the monthly maximum demand. In the low-level instances, the factors are 0.35 and 0, respectively, which means countries or territories cannot store relief items for future use.



(a) Main effects (b) Interactive effects

Figure 10 Main effects and interactive effects analysis with instance S-6-H-018.

We repeatedly solve the instance S-6-H-018 with the 27 sets of parameters, and record the upper bounds with their confidence intervals. The main effects plot and interactive effects plot are Figure 10(a) and Figure 10(b), respectively. From Figure 10(a), we have two observations. First, among the three factors: transportation price, procurement price, and capacity, the procurement price has the most significant impact on the objective value, followed by the transportation price, and then the capacity. This observation is not surprising, as among all the costs in humanitarian logistics, the procurement cost can not be avoided by any means, while the transportation cost can be reduced by optimizing the transportation plan, e.g., by transporting relief items from the closer hubs. The capacity factors do not have a significant impact on the objective value, on the `H`, `M`, and `L` levels, the average objective values are all close to the overall mean value. This suggests that investigating capacity building by increasing the budget and building larger warehouses may not contribute to reducing the total cost, despite the large investment required, which has not been involved in Figure 10. Second, for each factor on each level, we can observe cluster-distributed objectives. For the transportation price factor, on each level, there are three clusters; for the procurement price factor, on each level, there are one or two clusters; for the capacity factor, on each level, there are four clusters, indicating the interactive effects among the factors. Especially the interactive effects between the transportation price and the procurement price. In Figure 10(b), the highest costs are incurred when the procurement price and transportation price are both high. In contrast, the lowest costs are associated with low procurement prices. Therefore, from humanitarian organizations' perspective, the procurement price should be the primary focus, followed by the transportation price, while the capacity should be considered last, and the interactive effects between the procurement price and transportation price should be

taken into account. Humanitarian organizations should avoid purchasing expensive relief items and storing them far from disaster-affected areas, as this will lead to high expenses.

7. Conclusions

This study provides international humanitarian organizations an inventory coordination framework, which is formulated as a multi-stage stochastic programming model. This model involves convex objective functions, which are the summation of monetary costs and humanitarian costs, the latter is formulated as the *target-based disutility* proposed. The model uses historical data to minimize long-term costs and guide future decisions, which is achieved by employing the SDDP method. The SDDP method decomposes the multi-stage problem into a series of single-stage problems, each for a stage. These problems replace the constraints and objectives of future stages with Benders cuts and estimators, respectively. The Benders cuts generated by the SDDP method approximate the estimators, which are the expected values of the future costs. After a sufficient number of iterations, an optimal policy defined by the stage-specific problems with cuts is obtained. The policy guides future decisions. In addition to the mathematical programming model and the solution method, we also collect data of nodes and their locations from UNHRD (2024) and OCHA (2024), and the population affected from the EM-DAT database (CRED 2024). We generate instances based on these datasets for the case study, and conduct experiments to evaluate the performance of the proposed method. The experiments include in-sample and out-of-sample performance analysis, results illustration and analysis, benefits of coordination, and sensitivity analysis. The results provide several insights for humanitarian organizations as follows.

First, when using our method as a tool for decision-making, the frequency of probability upper bound estimation can be reduced to accelerate the algorithm without compromising the solution quality. However, even without reducing the frequency, for worldwide instances with 12 stages, six regions, 34 countries or territories, and 256 samples, the algorithm can converge in slightly longer than one hour (3,915 seconds) when using linear disutility. When using historical data without resampling, one can expect the algorithm to converge in minutes. Second, the proposed SDDP method is out-of-sample stable, one can use small sample sizes (one historical year as a sample) to obtain policies with good out-of-sample performance. Third, the out-of-sample decisions from the optimal policy offer valuable insights. That is, with proper coordination no shortage and zero waste can be achieved simultaneously, procurement should be done wisely but not excessively, and preventive transportation can be implemented as anticipatory action. Fourth, the SDDP policy achieves up to 21% cost savings compared to myopic or deterministic policies, amounting to US$616.2 million annually, highlighting the benefits of stochastic dynamic solutions. Fifth, we observe that though adding flexibility through broad coordination is beneficial, good performance can still be achieved within the current OCHA organizational structure. The relative cost difference is only 0.3%. Finally, the sensitivity analysis reveals that humanitarian organizations should pay more attention to the procurement price, followed by the transportation price, while the capacity should be considered last. This demonstrates that humanitarian organizations

should negotiate with suppliers to lower procurement costs and optimize transportation plans to reduce shipping expenses, while lowering the priority of capacity building, to ensure flexibility, adaptability, and cost efficiency.

This study can be further extended in the following directions. First, we assume that the procurement and transportation prices are static and deterministic, while in reality they can be dynamic and uncertain due to factors like inflation, fuel prices, and exchange rates. Future research can consider incorporating these factors into the model. Second, the proposed model involves only continuous decision variables, thus it is continuous convex. Future research can consider incorporating integer decision variables and employ the Stochastic Dual Dynamic Integer Programming method (Zou et al. 2019). Third, incorporating decentralized decision-making methods, e.g., the Alternating Direction Method of Multipliers, is also interesting. This allows humanitarian organizations to protect private information while maintaining coordination. Finally, incorporating multi-stage robust optimization methods is also promising.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) [Grant No. 72071106, 72571135] and the High Performance Computing Platform of Nanjing University of Aeronautics and Astronautics. We are grateful to Donghong Huang for his help with computation.

References

Acimovic, J. and Goentzel, J. (2016). Models and metrics to assess humanitarian response capacity. *Journal of Operations Management*, 45:11–29.

Adsanver, B., Balcik, B., Bélanger, V., and Rancourt, M.-È. (2024). Operations research approaches for improving coordination, cooperation, and collaboration in humanitarian relief chains: A framework and literature review. *European Journal of Operational Research*, 319(2):384–398.

Aflaki, A. and Pedraza-Martinez, A. J. (2023). Competition and Collaboration on Fundraising for Short-Term Disaster Response: The Impact on Earmarking and Performance. *Manufacturing & Service Operations Management*, 25(4):1451–1470.

Ahmed, S., Cabral, F. G., and Freitas Paulo da Costa, B. (2022). Stochastic Lipschitz dynamic programming. *Mathematical Programming*, 191(2):755–793.

Akhtar, P., Marr, N., and Garnevska, E. (2012). Coordination in humanitarian relief chains: Chain coordinators. *Journal of Humanitarian Logistics and Supply Chain Management*, 2(1):85–103.

Balcik, B., Beamon, B. M., Krejci, C. C., Muramatsu, K. M., and Ramirez, M. (2010). Coordination in humanitarian relief chains: Practices, challenges and opportunities. *International Journal of Production Economics*, 126(1):22–34.

Balcik, B., Silvestri, S., Rancourt, M.-E., and Laporte, G. (2019). Collaborative prepositioning network design for regional disaster response. *Production and Operations Management*, 28(10):2431–2455.

Besiou, M., Pedraza-Martinez, A. J., and Van Wassenhove, L. N. (2021). Humanitarian operations and the UN Sustainable Development Goals. *Production and Operations Management*, 30(12):4343–4355.

Birge, J. R. (1985). Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research*, 33(5):989–1007.

Castro, M. P., Bodur, M., and Song, Y. (2022). Markov chain-based policies for multi-stage stochastic integer linear programming with an application to disaster relief logistics. Technical Report arXiv:2207.14779, arXiv.

CRED (2024). Centre for Research on the Epidemiology of Disasters (CRED): Emergency Events Database (EM-DAT). Accessed 2024-04-01.

Ding, L. (2020). *Multistage Stochastic Programming*. PhD thesis, School of Industrial and Systems Engineering, Georgia Institute of Technology.

Ding, T., Hu, Y., and Bie, Z. (2018). Multi-stage stochastic programming with nonanticipativity constraints for expansion of combined power and natural gas systems. *IEEE Transactions on Power Systems*, 33(1):317–328.

Dore, M. H. I. and Singh, R. G. (2013). Economic Valuation of Life. In Bobrowsky, P. T., editor, *Encyclopedia of Natural Hazards*, Encyclopedia of Earth Sciences Series, pages 240–242. Springer Netherlands, Dordrecht.

Dowson, O., Morton, D. P., and Downward, A. (2022). Bi-objective multistage stochastic linear programming. *Mathematical Programming*, 196:907–933.

Dufour, E., Laporte, G., Paquette, J., and Rancourt, M.-E. (2018). Logistics service network design for humanitarian response in East Africa. *Omega*, 74:1–14.

Duque, D. and Morton, D. P. (2020). Distributionally robust stochastic dual dynamic programming. *SIAM Journal on Optimization*, 30(4):2841–2865.

EC-DRMKC (2023). INFORM Risk Index 2024. Accessed 2024-03-26.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26.

Eligüzel, İ. M., Özceylan, E., and Weber, G.-W. (2022). Location-allocation analysis of humanitarian distribution plans: A case of United Nations Humanitarian Response Depots. *Annals of Operations Research*, 324:825–854.

Fattahi, M., Keyvanshokooh, E., and Govindan, D. K. K. (2023). Resource planning strategies for healthcare systems during a pandemic. *European Journal of Operational Research*, 304(1):192–206.

FSC (2022). Guideline on Food Security and Agriculture Cluster Response Packages. Technical report, Food Security Cluster (FSC).

Füllner, C. and Rebennack, S. (2025). Stochastic Dual Dynamic Programming and Its Variants: A Review. *SIAM Review*, 67(3):415–539.

Gangammanavar, H. and Sen, S. (2021). Stochastic dynamic linear programming: A sequential sampling algorithm for multistage stochastic linear programming. *SIAM Journal on Optimization*, 31(3):2111–2140.

GDHO (2024). Global Database of Humanitarian Organisations. Accessed 2024-06-28.

Geoffrion, A. M. (1972). Generalized Benders decomposition. *Journal of Optimization Theory and Applications*, 10(4):237–260.

Girardeau, P., Leclere, V., and Philpott, A. B. (2015). On the convergence of decomposition methods for multistage stochastic convex programs. *Mathematics of Operations Research*, 40(1):130–145.

Gossen, H. H. (1983). *The Laws of Human Relations and the Rules of Human Action Derived Therefrom*. MIT Press.

Grass, E. and Fischer, K. (2016). Two-stage stochastic programming in disaster management: A literature survey. *Surveys in Operations Research and Management Science*, 21(2):85–100.

Guigues, V. (2020). Inexact cuts in stochastic dual dynamic programming. *SIAM Journal on Optimization*, 30(1):407–438.

Guigues, V., Monteiro, R., and Svaiter, B. (2021). Inexact cuts in stochastic dual dynamic programming applied to multistage stochastic nondifferentiable problems. *SIAM Journal on Optimization*, 31(3):2084–2110.

Guo, P. and Zhu, J. (2023). Capacity reservation for humanitarian relief: A logic-based Benders decomposition method with subgradient cut. *European Journal of Operational Research*, 311(3):942–970.

Helseth, A. and Mo, B. (2023). Hydropower aggregation by spatial decomposition—an SDDP approach. *IEEE Transactions on Sustainable Energy*, 14(1):381–392.

Holguin-Veras, J., Amaya-Leal, J., Cantillo, V., Van Wassenhove, L. N., Aros-Vera, F., and Jaller, M. (2016). Econometric estimation of deprivation cost functions: A contingent valuation experiment. *Journal of Operations Management*, 45:44–56.

Holguín-Veras, J., Pérez, N., Jaller, M., Van Wassenhove, L. N., and Aros-Vera, F. (2013). On the appropriate objective function for post-disaster humanitarian logistics models. *Journal of Operations Management*, 31(5):262–280.

Hu, S., Han, C., Dong, Z. S., and Meng, L. (2019). A multi-stage stochastic programming model for relief distribution considering the state of road network. *Transportation Research Part B: Methodological*, 123:64–87.

Hu, S., Hu, Q., Tao, S., and Dong, Z. S. (2023). A multi-stage stochastic programming approach for pre-positioning of relief supplies considering returns. *Socio-Economic Planning Sciences*, 88:101617.

ISO (2020). ISO 3166-1:2020 Codes for the representation of names of countries and their subdivisions - Part 1: Country codes.

John, L., Gurumurthy, A., Mateen, A., and Narayanamurthy, G. (2022). Improving the coordination in the humanitarian supply chain: Exploring the role of options contract. *Annals of Operations Research*, 319:15–40.

Kahneman, D. and Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, 47(2):263–291.

Keshvari Fard, M., Ljubić, I., and Papier, F. (2022). Budgeting in international humanitarian organizations. *Manufacturing & Service Operations Management*, 24(3):1261–1885.

Kizito, R., Liu, Z., Li, X., and Sun, K. (2022). Multi-stage stochastic optimization of islanded utility-microgrids design after natural disasters. *Operations research perspectives*, 9:100235.

Lan, G. (2022). Complexity of stochastic dual dynamic programming. *Mathematical Programming*, 191(2):717–754.

Lan, G. and Shapiro, A. (2024). Numerical Methods for Convex Multistage Stochastic Optimization. *Foundations and Trends® in Optimization*, 6(2):63–144.

Leclère, V., Carpentier, P., Chancelier, J.-P., Lenoir, A., and Pacaud, F. (2020). Exact converging bounds for stochastic dual dynamic programming via Fenchel duality. *SIAM Journal on Optimization*, 30(2):1223–1250.

Mankiw, N. G. (2024). *Principles of Economics*. Cengage Learning, Boston, Massachusetts, 10th edition.

Nikkhoo, F., Bozorgi-Amiri, A., and Heydari, J. (2018). Coordination of relief items procurement in humanitarian logistic based on quantity flexibility contract. *International Journal of Disaster Risk Reduction*, 31:331–340.

OCHA (2020). Step-by-step practical guide for humanitarian needs overviews, humanitarian response plans and updates. Technical report, United Nations Office for the Coordination of Humanitarian Affairs (OCHA).

OCHA (2021). Complementarity between CBPFs and CERF. Technical report, United Nations Office for the Coordination of Humanitarian Affairs.

OCHA (2022). Philippines: Anticipatory Action Framework, 2022 Revision. Technical report, United Nations Office for the Coordination of Humanitarian Affairs (OCHA).

OCHA (2023). Cluster Coordination. Accessed 2023-12-23.

OCHA (2023a). Humanitarian Response Plan 2023. Technical report, United Nations Office for the Coordination of Humanitarian Affairs (OCHA).

OCHA (2023b). OCHA Annual Report 2022. Annual Report 2022, United Nations Office for the Coordination of Humanitarian Affairs (OCHA).

OCHA (2023c). United Nations Office for the Coordination of Humanitarian Affairs - The Humanitarian Programme Cycle. Accessed 2023-12-24.

OCHA (2024). United Nations Office for the Coordination of Humanitarian Affairs - Where We Work. Accessed 2024-03-26.

Olanrewaju, O. G., Dong, Z. S., and Hu, S. (2020). Supplier selection decision making in disaster response. *Computers & Industrial Engineering*, 143:106412.

Pereira, M. V. F. and Pinto, L. M. V. G. (1991). Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52(1):359–375.

Peters, K., Silva, S., Gonçalves, R., Kavelj, M., Fleuren, H., den Hertog, D., Ergun, O., and Freeman, M. (2021). The nutritious supply chain: Optimizing humanitarian food assistance. *INFORMS Journal on Optimization*, 3(2):119–226.

Philpott, A. B. and Guan, Z. (2008). On the convergence of stochastic dual dynamic programming and related methods. *Operations Research Letters*, 36(4):450–455.

Quezada, F., Gicquel, C., and Kedad-Sidhoum, S. (2022). Combining Polyhedral Approaches and Stochastic Dual Dynamic Integer Programming for Solving the Uncapacitated Lot-Sizing Problem Under Uncertainty. *INFORMS Journal on Computing*, 34(2):1024–1041.

Rambha, T., Nozick, L. K., Davidson, R., Yi, W., and Yang, K. (2021). A stochastic optimization model for staged hospital evacuation during hurricanes. *Transportation Research Part E: Logistics and Transportation Review*, 151:102321.

Rodríguez-Pereira, J., Balcik, B., Rancourt, M.-È., and Laporte, G. (2021). A cost-sharing mechanism for multi-country partnerships in disaster preparedness. *Production and Operations Management*, 30(12):4541–4565.

Seranilla, B. K. and Löhndorf, N. (2024). Optimizing vaccine distribution in developing countries under natural disaster risk. *Naval Research Logistics (NRL)*, 71(1):140–157.

Shapiro, A. (2011). Analysis of stochastic dual dynamic programming method. *European Journal of Operational Research*, 209(1):63–72.

Shapiro, A., Dentcheva, D., and Ruszczynski, A. (2021). *Lectures on Stochastic Programming: Modeling and Theory*. MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 3 edition.

Siddig, M. and Song, Y. (2025). Multistage stochastic programming with a random number of stages: Applications in hurricane disaster relief logistics planning. *European Journal of Operational Research*, 321(3):925–941.

Stienen, V., Wagenaar, J., den Hertog, D., and Fleuren, H. (2021). Optimal depot locations for humanitarian logistics service providers using robust optimization. *Omega*, 104:102494.

Thevenin, S., Adulyasak, Y., and Cordeau, J.-F. (2021). Material requirements planning under demand uncertainty using stochastic optimization. *Production and Operations Management*, 30(2):475–493.

Thevenin, S., Adulyasak, Y., and Cordeau, J.-F. (2022). Stochastic dual dynamic programming for multiechelon lot sizing with component substitution. *INFORMS Journal on Computing*, 34(6):2867–3350.

Tippong, D., Petrovic, S., and Akbari, V. (2022). A review of applications of operational research in healthcare coordination in disaster management. *European Journal of Operational Research*, 301(1):1–17.

Toyasaki, F., Arikan, E., Silbermayr, L., and Sigala, I. F. (2017). Disaster relief inventory management: Horizontal cooperation between humanitarian organizations. *Production and Operations Management*, 26(6):1221–1237.

UNHRD (2024). United Nations Humanitarian Response Depot (UNHRD). Accessed 2024-04-08.

Van Slyke, R. M. and Wets, R. (1969). L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663.

Wang, X., Li, F., Liang, L., Huang, Z., and Ashley, A. (2015). Pre-purchasing with option contract and coordination in a relief supply chain. *International Journal of Production Economics*, 167:170–176.

Wankmüller, C. and Reiner, G. (2019). Coordination, cooperation and collaboration in relief supply chain management. *Journal of Business Economics*, 90:239–276.

WFP (2023). Logistics Operational Guide (LOG): Inventory Planning and Management. Accessed 2023-12-25.

Zahiri, B., Suresh, N. C., and de Jong, J. (2020). Resilient hazardous-materials network design under uncertainty and perishability. *Computers & industrial engineering*, 143:106401.

Zhu, J., Liu, H., Guo, Y., Chen, J., Zhuo, Y., and Wang, Z. (2023). Spatiotemporal decomposed dispatch of integrated electricity-gas system via stochastic dual dynamic programming-based value function approximation. *Energy*, 282:128247.

Zou, J., Ahmed, S., and Sun, X. A. (2019). Stochastic dual dynamic integer programming. *Mathematical Programming*, 175(1):461–502.

Supporting Information: Coordinating International Humanitarian Inventory by Stochastic Dual Dynamic Programming

S1. Multi-stage Stochastic Programming

Table S1 Multi-stage stochastic programming in humanitarian operations management

Paper	Decisions		Solution method		Application
	Pre-disaster	Post-disaster	Name	Exact	
Hu et al. (2019)	Location.	Vehicle and delivery.	PH	N	Yaan earthquake in China
Olanrewaju et al. (2020)	-	Supplier selection, and procurement.	CPLEX	Y	Case study in the US
Zahiri et al. (2020)	Location.	Assignment, flow, and inventory.	Gurobi	Y	Case study in Iran
Rambha et al. (2021)	-	Vehicle and transport.	CPLEX	Y	Hurricane Isabel in North Carolina
*Kizito et al. (2022)	Location.	Power supply and store.	NL	Y	Case study in Tennessee
*Castro et al. (2022)	-	Inventory	SDDP	Y	Case study for hurricane
Fattahi et al. (2023)	-	Capacity and transshipment of healthcare resource, and acceptance of patients.	RH	N	Case study in COVID-19 pandemic
Hu et al. (2023)	Location.	Procurement, delivery, inventory, and return.	SAA	Y	Case study in the US
*Seranilla and Löhndorf (2024)	Location and allocation.	Location and allocation.	SPA	Y	Case study in Philippines
*Siddig and Song (2025)	-	Network flow	NB	Y	Hurricane Florence in South Carolina
*This paper	Procurement.	Procurement, inventory, and transportation.	SDDP	Y	International inventory coordination of UNHRD and OCHA

[1] `-`: not applicable; `PH`: Progressive Hedging; `NL`: Nested L-shaped method; `SDDP`: Stochastic Dual Dynamic Programming; `RH`: Rolling horizon method; `SAA`: Sample Average Approximation; `SPA`: Shadow price approximation; `NB`: Nested Benders decomposition; `Y`: Yes; `N`: No; `*`: implementable for out-of-sample data.

S2. SAA Formulation

This section formulates the discretized SAA model, which is also called *extensive form* or *deterministic equivalent*, as a centralized benchmark. The SAA model is based on the scenario tree discretization of the multi-stage stochastic programming problem. A scenario tree is consistent with nodes and arcs, where each node represents a scenario in a stage and each arc represents a transition from one scenario to another in two stages. We let $\mathcal{N} = \{0, 1, 2, \cdots, N\}$ ($\mathcal{N}_0 = \{1, 2, \cdots, N\}$) denote the set of nodes on the scenario tree, where 0 is the root. For each node $n \in \mathcal{N}_0$, let P^n be the probabilistic of its occurrence (from the initial stage, or equivalently, the root to that node), $a(n)$ be its ancestor, and $\tau(n)$ be the stage that it belonging to. Based on the scenario tree defined, the SAA model can be formulated with minor modifications to the notations for (3). First, the sets remain the same, except that $\mathcal{T}$ and Ω_r^t are not necessary. Second, parameters and variables with superscript t are replaced with n, where $t = \tau(n)$, while other parameters remain the same. Note that though each node $n \in \mathcal{N}_0$ in the scenario tree can be mapped to exactly one stage $t \in \mathcal{T}_0$, each stage can have multiple nodes representation, implying that the number of variables is expended. This is why the SAA model is also namely *extensive form*. The SAA model is as follows.

$$\min \sum_{i\in\mathcal{H}} c_i^0 z_i^0 + \sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{W}_r} c_j^0 z_j^0 + \sum_{n\in\mathcal{N}_0} \mathsf{P}^n \mathcal{Q}^n \tag{S1.1}$$

$$\text{s.t. } c_i^t z_i^n \le b_i^n,\ \forall\ i\in\mathcal{H}, n\in\mathcal{N} \tag{S1.2}$$

$$c_j^t z_j^n \le b_j^n,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, n\in\mathcal{N} \tag{S1.3}$$

$$x_i^n + y_i^n \le x_i^{a(n)} + z_i^n + \sum_{i'\in\mathcal{H}} f_{i'i}^n - \sum_{i'\in\mathcal{H}} f_{ii'}^n - \sum_{j\in\mathcal{W}} f_{ij}^n,\ \forall\ i\in\mathcal{H}, n\in\mathcal{N}_0 \tag{S1.4}$$

$$x_j^n + y_j^n \le x_j^{a(n)} + z_j^n + v_j^n + \sum_{i\in\mathcal{H}} f_{ij}^n + \sum_{j'\in\mathcal{W}_r} f_{j'j}^n - \sum_{j'\in\mathcal{W}_r} f_{jj'}^n - u_j^n,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, n\in\mathcal{N}_0 \tag{S1.5}$$

$$x_i^n \le q_i^n,\ \forall\ i\in\mathcal{H}, n\in\mathcal{N} \tag{S1.6}$$

$$x_j^n \le q_j^n,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, n\in\mathcal{N} \tag{S1.7}$$

$$x_i^0 = z_i^0,\ \forall\ i\in\mathcal{H} \tag{S1.8}$$

$$x_j^0 = z_j^0,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \tag{S1.9}$$

$$z_i^n = z_i^{'a(n)},\ \forall\ i\in\mathcal{H}, n\in\mathcal{N}_0 \tag{S1.10}$$

$$z_j^n = z_j^{'a(n)},\ \forall\ j\in\mathcal{W}, n\in\mathcal{N}_0 \tag{S1.11}$$

$$z_i^n, f_{ii'}^n, f_{ij}^n, z_j^n, f_{jj'}^n, u_j^n, x_i^n, x_j^n, y_i^n, y_j^n, v_j^n \in \mathbb{R}_0^+,\ \forall\ i,i'\in\mathcal{H}, j,j'\in\mathcal{W}, n\in\mathcal{N} \tag{S1.12}$$

$$z_i^{'n}, z_j^{'n} \in \mathbb{R}_0^+,\ \forall\ i\in\mathcal{H}, j\in\mathcal{W}, n\in\mathcal{N} \tag{S1.13}$$

where

$$\begin{aligned}\mathcal{Q}^n = &\sum_{i\in\mathcal{H}} c_i^{\tau(n)} z_i^n + \sum_{(i,i')\in\mathcal{A}_\mathcal{H}} g_{ii'} f_{ii'}^n + \sum_{(i,j)\in\mathcal{A}} g_{ij} f_{ij}^n + \sum_{(j,j')\in\mathcal{A}_\mathcal{W}} g_{jj'} f_{jj'}^n \\ &+ \sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{W}_r} \left[c_j^{\tau(n)} z_j^n + p_j^t v_j^n + \lambda_j\ \mathcal{U}_j^{\tau(n)}(u_j^n)\right]\end{aligned} \tag{S1.14}$$

This model is similar to the model (3) except that, in (S1.10) and (S1.11), auxiliary procurement variables $z_i^{'a(n)}$ and $z_j^{'a(n)}$ are introduced to ensure the nonanticipativity constraints. These constraints ensure the procurement decisions made at the nodes that are descendants of $a(n)$ are consistent, no matter which scenario is realized at the stage $\tau(n)$, making the procurement decisions decision-hazard variables. On the contrary, the transportation decisions are hazard-decision variables. It is usually impossible to solve this model for real-life scale problems since the cardinality of $\mathcal{N}$ grows exponentially with the stages T. In other words, this model has exponentially many decision variables with respect to T. Consider a scenario tree in which each node has η descendants, the extensive form has a decision variable of size $\mathcal{O}(\eta^T)$, leading to computational intractability. Another drawback of the SAA model is the solution obtained by it is only implementable when the realized scenarios through the planning horizons are exactly the same as one of the sample paths in the scenario tree used for building the model. However, one advantage of the SAA model is its compatibility with standard mathematical programming solvers. The solutions derived can serve as a benchmark for the stochastic model.

S3. Tuning the Sample Size $|\mathcal{S}'|$

The sample $\mathcal{S}'$ determines when to terminate the SDDP method, indirectly affecting policy quality. Ideally, $\mathcal{S}'$ should balance accuracy in upper bound estimation with computational efficiency, i.e., neither too small to compromise precision nor too large to waste computational resources. We tune the sample size $|\mathcal{S}'|$ to ensure that the statistical properties of the upper bound closely approximate the "true" upper bound. Since computing the true upper bound is computationally prohibitive (e.g., 18^{12} simulations for instance M-6-H-018), we gradually increase $|\mathcal{S}'|$ from a smaller value and monitor the stability of the upper bound estimation (mean, standard deviation, etc.). The results are presented in Figure S1, using policies of instance M-6-H-018 from the 10th and 100th iterations of the SDDP method. As anticipated, as $|\mathcal{S}'|$ increases, the mean and standard deviation of the upper bound estimates stabilize, and the confidence interval narrows. We select $|\mathcal{S}'| = 500$ because it provides accurate upper bound estimates in both early (10th) and later (100th) iterations while maintaining manageable computational effort.

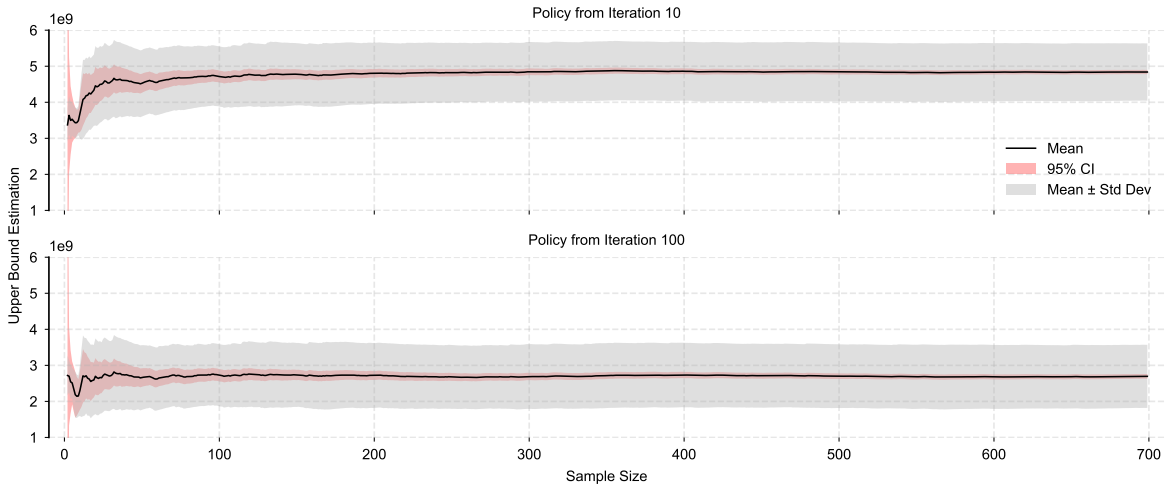


Figure S1 **Statistical properties of the upper bound with varying $|\mathcal{S}'|$**

S4. Myopic SAA Policy

The myopic SAA approach models each stage as a two-stage stochastic programming problem. For each stage, this model is built with state parameters $\bar{x}_i^{t-1}$ and $\bar{x}_j^{t-1}$ (inventory) from the previous stage. The first-stage decisions are procurement decisions with budget constraints, which is scenario-independent. This aligns with the decision-hazard variables described in the main text. The scenario-dependent decisions are included in the second stage, including transportation, rapid response, consumption, etc. Once the model is solved, the first-stage decisions are implemented, and the second-stage decisions are implemented after demand realization. The state parameters for the next myopic SAA problem are updated based on the realized scenarios and decisions. Let $\mathcal{S}$ be a set of discrete scenarios, the myopic SAA model for $t \in \mathcal{T}_0$ is formulated as follows.

$$\min \sum_{i\in\mathcal{H}} c_i^t z_i^t + \sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{W}_r} c_j^t z_j^t + \sum_{s\in\mathcal{S}} \mathsf{P}^s \mathcal{Q}^s \tag{S2.1}$$

$$\text{s.t. } c_i^t z_i^t \le b_i^t,\ \forall\ i\in\mathcal{H} \tag{S2.2}$$

$$c_j^t z_j^t \le b_j^t,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \tag{S2.3}$$

$$z_i^t, z_j^t \in \mathbb{R}_0^+,\ \forall\ i\in\mathcal{H}, j\in\mathcal{W}_r \tag{S2.4}$$

where

$$\mathcal{Q}^s =$$

$$\min \sum_{(i,i')\in\mathcal{A}_\mathcal{H}} g_{ii'} f_{ii'}^{ts} + \sum_{(i,j)\in\mathcal{A}} g_{ij} f_{ij}^{ts} + \sum_{(j,j')\in\mathcal{A}_\mathcal{W}} g_{jj'} f_{jj'}^{ts} + \sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{W}_r} (p_j^t v_j^{ts} + 10^6 u_j^{'ts}) \tag{S2.5}$$

$$\text{s.t. } x_i^{ts} + y_i^{ts} \le \bar{x}_i^{t-1} + z_i^t + \sum_{i'\in\mathcal{H}} f_{i'i}^{ts} - \sum_{i'\in\mathcal{H}} f_{ii'}^{ts} - \sum_{j\in\mathcal{W}} f_{ij}^{ts},\ \forall\ i\in\mathcal{H}, s\in\mathcal{S} \tag{S2.6}$$

$$x_j^{ts} + y_j^{ts} \le \bar{x}_j^{t-1} + z_j^t + v_j^{ts} + \sum_{i\in\mathcal{H}} f_{ij}^{ts} + \sum_{j'\in\mathcal{W}_r} f_{j'j}^{ts} - \sum_{j'\in\mathcal{W}_r} f_{jj'}^{ts} - u_j^{ts},\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, s\in\mathcal{S} \tag{S2.7}$$

$$x_i^{ts} \le q_i^t,\ \forall\ i\in\mathcal{H}, s\in\mathcal{S} \tag{S2.8}$$

$$x_j^{ts} \le q_j^t,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, s\in\mathcal{S} \tag{S2.9}$$

$$u_j^{'ts} \ge \bar{d}_j^{ts} - u_j^{ts},\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, s\in\mathcal{S} \tag{S2.10}$$

$$f_{ii'}^{ts}, f_{ij}^{ts}, f_{jj'}^{ts}, u_j^{ts}, u_j^{'ts}, x_i^{ts}, x_j^{ts}, y_i^{ts}, y_j^{ts}, v_j^{ts} \in \mathbb{R}_0^+,\ \forall\ i,i'\in\mathcal{H}, j,j'\in\mathcal{W}, s\in\mathcal{S} \tag{S2.11}$$

(S2.1) is the objective function minimizing the first-stage costs plus the expected second-stage costs. (S2.2) and (S2.3) are scenario-independent budget constraints. (S2.4) is the domain of the first-stage decision variables. (S2.5) defines the second-stage recourse function. (S2.6) and (S2.7) define the inventory dynamics for hubs and warehouses, respectively. (S2.8) and (S2.9) are capacity constraints. (S2.10) introduces shortage variables for calculating the disutility. (S2.11) is the domain of the second-stage decision variables. The myopic SAA simulation uses predetermined sample paths for the uncertain parameters, allowing us to update the next stage's state using scenario-dependent inventory variables (based on demand realization $\bar{d}_j^{ts}$). We exclude the myopic SAA simulation for stage $t=0$ because it only involves stockpiling, not resource allocation decisions, while removing all second-stage elements from the model predictably results in zero initial stockpiling.

S5. Rolling Horizon Policy

The Rolling Horizon approach solves deterministic multi-period models (where "period" is used instead of "stage" to distinguish from multi-stage stochastic programming) with a given look-ahead horizon H from an initial period t_0. For the time window $[t_0, t_0+H]$, the model uses input from the previous period: *state* from the previous period $\bar{x}_i^{t_0-1}, \bar{x}_j^{t_0-1}, \bar{z}_i^{'t_0-1}$, and $\bar{z}_j^{'t_0-1}$ (inventory and procurement); current demand realization $\bar{d}_j^{t_0}$; and expected future demands $\mathbb{E}(d_j^t)$. When solving the multi-period model, we obtain decisions for each period $t\in[t_0, t_0+H]$, but only the first-period decisions at t_0 are implemented. The subsequent periods are solved in a rolling manner,

implementing the first-period decisions of the next time window $[t_0+1, t_0+1+H]$. Note that when $t_0+H>T$, we set $H:=T-t_0$. Following this approach, the Rolling Horizon policy continuously updates its decisions using the latest information, ensuring responsiveness to demand changes. For $t_0 \in \mathcal{T}_0$, the model is formulated as follows.

$$\min \sum_{t\in[t_0,t_0+H]} \left(\mathcal{Q}^t_{\mathcal{H}} + \sum_{r\in\mathcal{R}} \mathcal{Q}^t_r \right) \tag{S3.1}$$

$$\text{s.t. } x_i^{t_0} + y_i^{t_0} \le \bar{x}_i^{t_0-1} + z_i^{t_0} + \sum_{i'\in\mathcal{H}} f_{i'i}^{t_0} - \sum_{i'\in\mathcal{H}} f_{ii'}^{t_0} - \sum_{j\in\mathcal{W}} f_{ij}^{t_0},\ \forall\ i\in\mathcal{H} \tag{S3.2}$$

$$x_j^{t_0} + y_j^{t_0} \le \bar{x}_j^{t_0-1} + z_j^{t_0} + v_j^{t_0} + \sum_{i\in\mathcal{H}} f_{ij}^{t_0} + \sum_{j'\in\mathcal{W}_r} f_{j'j}^{t_0} - \sum_{j'\in\mathcal{W}_r} f_{jj'}^{t_0} - u_j^{t_0},\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \tag{S3.3}$$

$$x_i^t + y_i^t \le x_i^{t-1} + z_i^t + \sum_{i'\in\mathcal{H}} f_{i'i}^t - \sum_{i'\in\mathcal{H}} f_{ii'}^t - \sum_{j\in\mathcal{W}} f_{ij}^t,\ \forall\ i\in\mathcal{H}, t\in[t_0+1, t_0+H] \tag{S3.4}$$

$$\begin{aligned} x_j^t + y_j^t \le x_j^{t-1} + z_j^t + v_j^t + \sum_{i\in\mathcal{H}} f_{ij}^t + \sum_{j'\in\mathcal{W}_r} f_{j'j}^t - \sum_{j'\in\mathcal{W}_r} f_{jj'}^t - u_j^t \\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in[t_0+1, t_0+H] \end{aligned} \tag{S3.5}$$

$$z_i^{t_0} = \bar{z}_i^{'t_0-1},\ \forall\ i\in\mathcal{H} \tag{S3.6}$$

$$z_j^{t_0} = \bar{z}_j^{'t_0-1},\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \tag{S3.7}$$

$$z_i^t = z_i^{'t-1},\ \forall\ i\in\mathcal{H}, t\in[t_0+1, t_0+H] \tag{S3.8}$$

$$z_j^t = z_j^{'t-1},\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in[t_0+1, t_0+H] \tag{S3.9}$$

$$x_i^t \le q_i^t,\ \forall\ i\in\mathcal{H}, t\in[t_0, t_0+H] \tag{S3.10}$$

$$x_j^t \le q_j^t,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in[t_0, t_0+H] \tag{S3.11}$$

$$c_i^{t+1} z_i^{'t} \le b_i^{t+1},\ \forall\ i\in\mathcal{H}, t\in[t_0, t_0+H] \tag{S3.12}$$

$$c_j^{t+1} z_j^{'t} \le b_j^{t+1},\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in[t_0, t_0+H] \tag{S3.13}$$

$$u_j^{'t_0} \ge \bar{d}_j^{t_0} - u_j^{t_0},\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r \tag{S3.14}$$

$$u_j^{'t} \ge \mathbb{E}(d_j^t) - u_j^t,\ \forall\ r\in\mathcal{R}, j\in\mathcal{W}_r, t\in[t_0+1, t_0+H] \tag{S3.15}$$

$$z_i^{'t}, z_j^{'t} \in \mathbb{R}_0^+,\ \forall\ i\in\mathcal{H}, j\in\mathcal{W}, t\in[t_0, t_0+H] \tag{S3.16}$$

$$x_i^t, z_i^t, f_{ii'}^t, f_{ij}^t, y_i^t \in \mathbb{R}_0^+,\ \forall\ i, i'\in\mathcal{H}, j\in\mathcal{W}, t\in[t_0, t_0+H] \tag{S3.17}$$

$$x_j^t, z_j^t, f_{jj'}^t, u_j^t, u_j^{'t}, y_j^t, v_j^t \in \mathbb{R}_0^+,\ \forall\ j, j'\in\mathcal{W}, t\in[t_0, t_0+H] \tag{S3.18}$$

where

$$\mathcal{Q}^t_{\mathcal{H}} = \sum_{i\in\mathcal{H}} c_i^t z_i^t + \sum_{(i,i')\in\mathcal{A}_{\mathcal{H}}} g_{ii'} f_{ii'}^t + \sum_{(i,j)\in\mathcal{A}} g_{ij} f_{ij}^t + \sum_{(j,j')\in\mathcal{A}_{\mathcal{W}}} g_{jj'} f_{jj'}^t \tag{S3.19}$$

$$\mathcal{Q}^t_r = \sum_{j\in\mathcal{W}_r} \left(c_j^t z_j^t + p_j^t v_j^t + 10^6 u_j^{'t} \right) \tag{S3.20}$$

(S3.1) is the objective function defined similar to previous sections, but minimizing the total cost over the window $[t_0, t_0+H]$. (S3.2)–(S3.5) define the inventory dynamics. For the initial period t_0, the inventories are determined by state parameters $\bar{x}_i^{t_0-1}$ and $\bar{x}_j^{t_0-1}$ in (S3.2) and (S3.3), respectively. (S3.6)–(S3.9) are nonanticipativity constraints that ensure procurement decisions at each period t_0 are made without knowing exact demand realizations. This aligns with our assumption

in the main text that procurement decisions are decision-hazard type, while others (e.g., transportation and rapid response) are hazard-decision type determined after demand realizations. (S3.10)–(S3.13) are capacity and budget constraints, ensuring that the procurement decisions do not exceed the available resources. (S3.14) and (S3.15) introduce auxiliary variables to represent shortage. For the initial period t_0, the consumption decisions (which determines shortage) are made with realized demand $\bar{d}_j^{t_0}$ for all $j \in \mathcal{W}$. For subsequent periods $t \in [t_0+1, t_0+H]$, only the expected demand $\mathbb{E}(d_j^t)$ is available and used. (S3.16)–(S3.18) are domains of the decision variables. For period $t_0 = 0$, all variables except $z_i^0, z_j^0, x_i^0, x_j^0, z_i^{'0}$, and $z_j^{'0}$ are fixed to zero, as they are unnecessary. The initial inventory for $t_0 = 0$ is equal to the procurement quantities: $x_i^0 = z_i^0$ and $x_j^0 = z_j^0$ for every $i \in \mathcal{H}$ and $j \in \mathcal{W}_r$, where $r \in \mathcal{R}$. After this model for $[t_0, t_0+H]$ is solved, decisions $\bar{x}_i^{t_0}, \bar{x}_j^{t_0}, \bar{z}_i^{'t_0}$, and $\bar{z}_j^{'t_0}$ are used as state parameters in the next window $[t_0+1, t_0+1+H]$. This process repeats until $t_0 = T$.

S6. Out-of-sample Performance

Unlike Figure 5, which forward simulates every five iterations, Figure S2 does so every two iterations. This is because these simpler instances can reach out-of-sample stability sooner than five iterations, and a lower simulation frequency would obscure performance trends.

S7. Results Illustration

Among the codes (ISO 2020) used in Figure 7 and Figure S4, we use "FJI" to denote the Pacific Islands (Cook Islands, Federated States of Micronesia (FSM), Fiji, Kiribati, Republic of the Marshall Islands (RMI), Nauru, Niue, Palau, Samoa, Solomon Islands, Tokelau, Tonga, Tuvalu, and Vanuatu) instead of solely Fiji, where one of OCHA's regional offices is located.

Other codes are as follows: Acc.: Accra, Bri.: Brindisi, Dub.: Dubai, Kua.: Kuala Lumpur, Pan.: Panama, BDI: Burundi, ERI: Eritrea, ETH: Ethiopia, KEN: Kenya, MDG: Madagascar, MWI: Malawi, MOZ: Mozambique, SOM: Somalia, SSD: South Sudan, SDN: Sudan, AFG: Afghanistan, MMR: Myanmar, FJI: Pacific Islands, PAK: Pakistan, COL: Colombia, SLV: El Salvador, GTM: Guatemala, HTI: Haiti, HND: Honduras, VEN: Venezuela, BFA: Burkina Faso, CMR: Cameroon, CAF: Central African Republic, TCD: Chad, COD: Democratic Republic of the Congo, MLI: Mali, NER: Niger, NGA: Nigeria, LBN: Lebanon, PSE: Occupied Palestinian Territory, SYR: Syrian Arab Republic, YEM: Yemen, TUR: Türkiye, UKR: Ukraine.

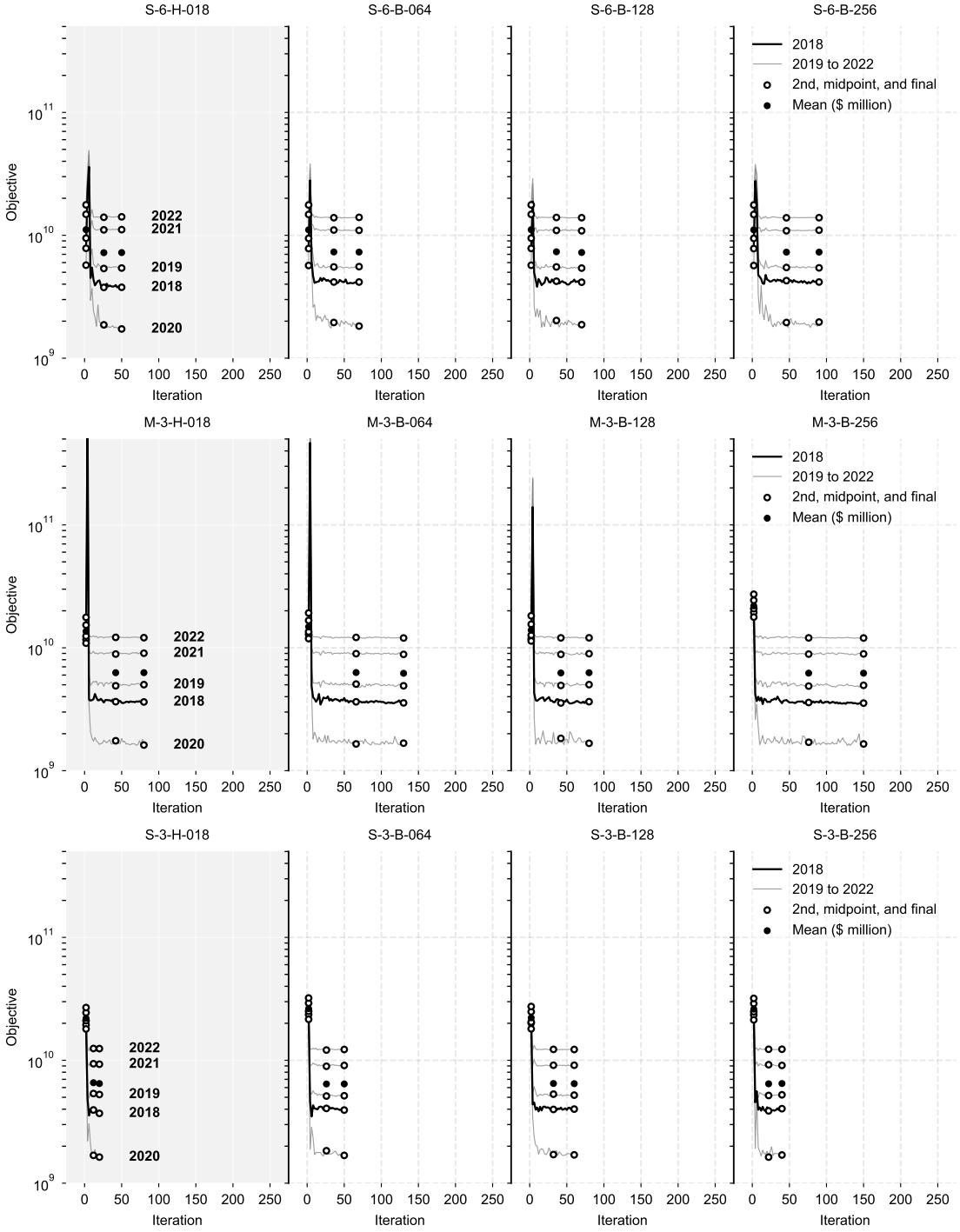


Figure S2 **Out-of-sample performance during iterations.**

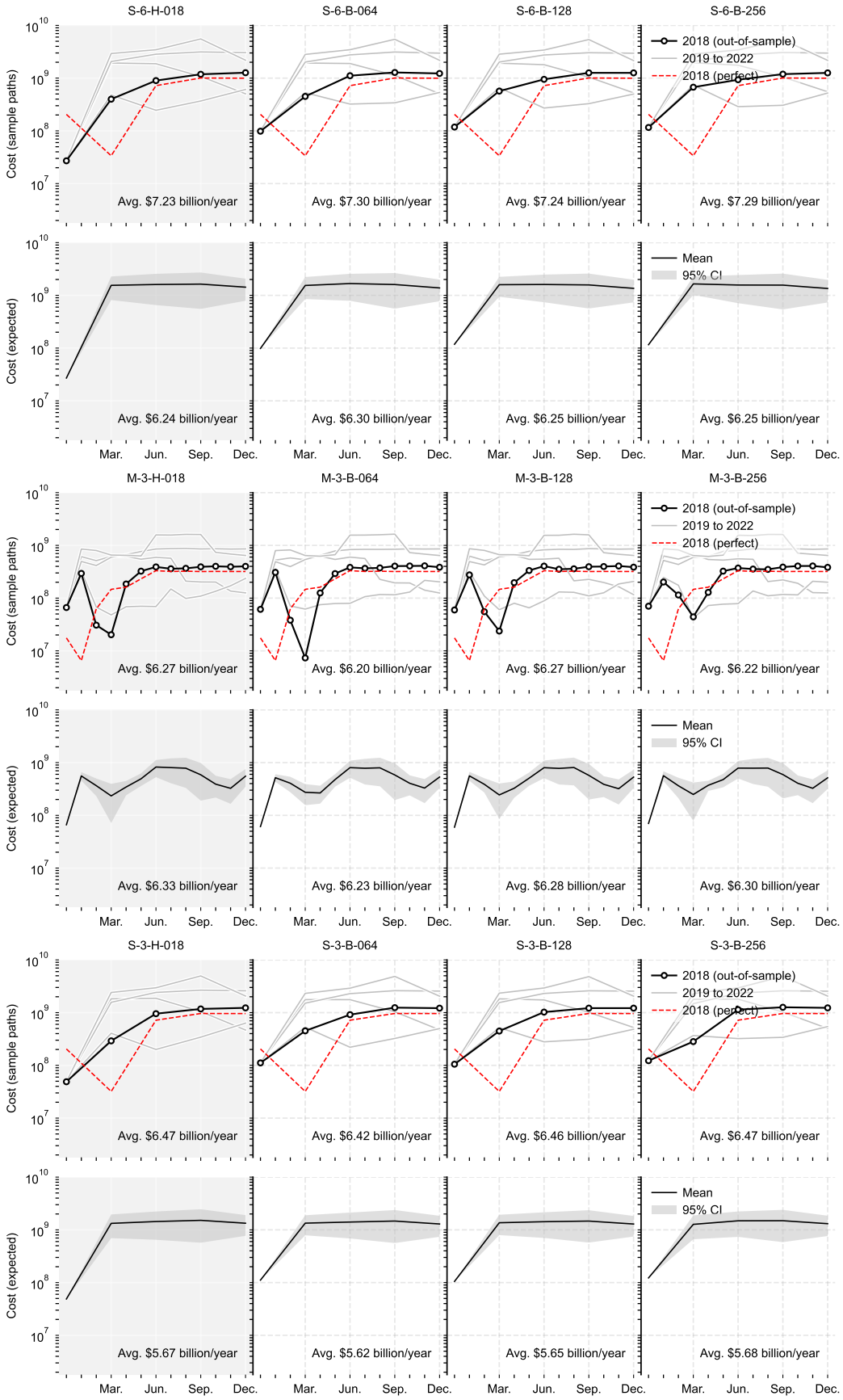


Figure S3 **Out-of-sample performance using the final policy.**

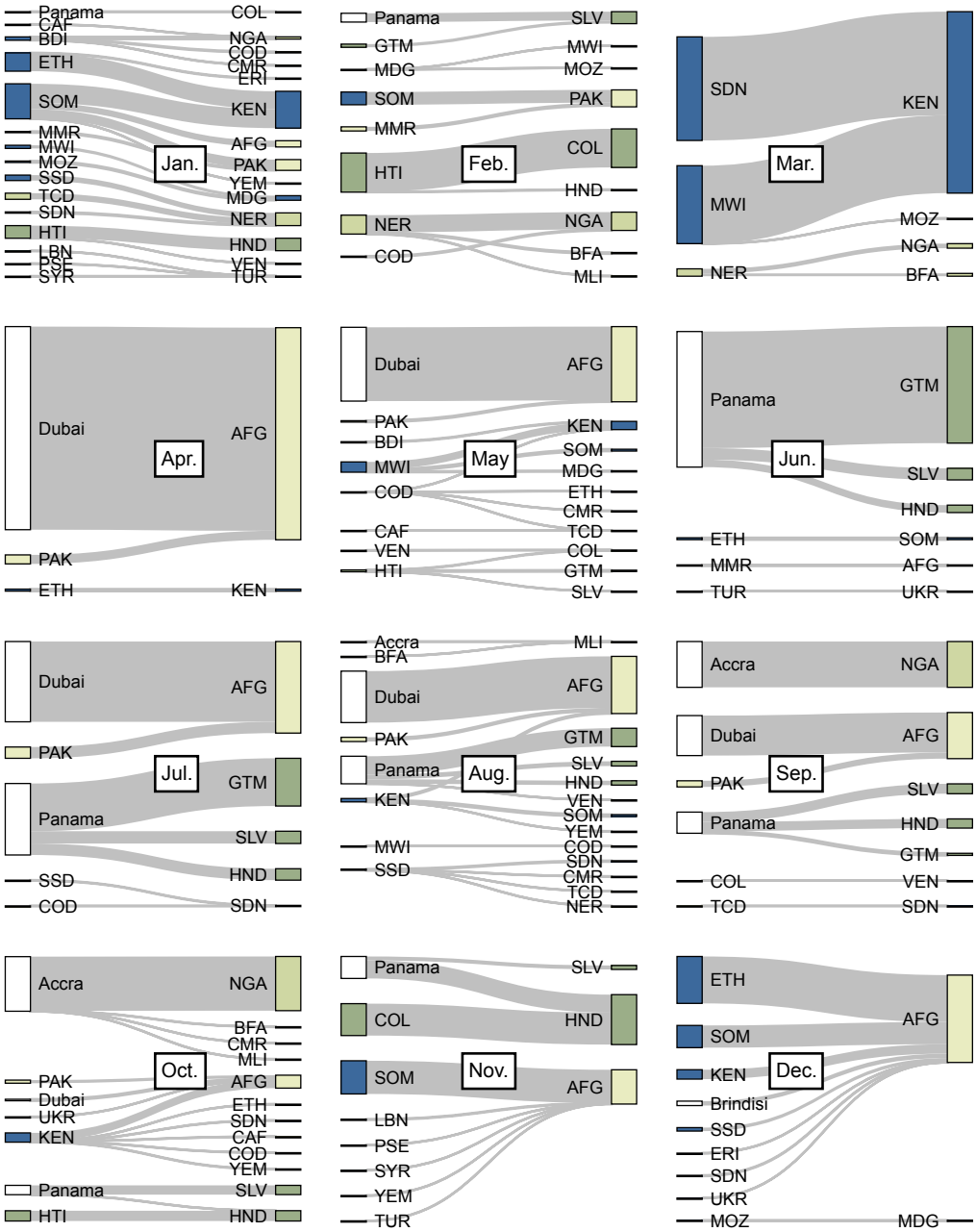


Figure S4 **Sankey diagram showing relief items flow among regions (2018)**