

## Using 3D-printing in disaster response

**Citation for published version (APA):**

Tönissen, D. D., & Schlicher, L. (2020). Using 3D-printing in disaster response: the two-stage stochastic 3D-printing knapsack problem. *arXiv*, 2020, 1-25.

**Document status and date:**

Published: 18/04/2020

**Document Version:**

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

**Please check the document version of this publication:**

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

**General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

[www.tue.nl/taverne](http://www.tue.nl/taverne)

**Take down policy**

If you believe that this document breaches copyright please contact us at:

[openaccess@tue.nl](mailto:openaccess@tue.nl)

providing details and we will investigate your claim.

# Using 3D-printing in disaster response: The two-stage stochastic 3D-printing knapsack problem

Denise D. Tönissen

School of Business and Economics, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands

d.d.tonissen@vu.nl

Loe Schlicher

School of Industrial Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

l.p.j.schlicher@tue.nl

April 21, 2020

## Abstract

In this paper, we will shed light on when to pack and use 3D-printers in disaster response operations. For that, we introduce a new type of problem, which we call the two-stage stochastic 3D-printing knapsack problem. We provide a two-stage stochastic programming formulation for this problem, for which both the first and the second stage are NP-hard integer linear programs. We reformulate this formulation to an equivalent integer linear program, which can be efficiently solved by standard solvers. Our numerical results illustrate that for most situations using a 3D-printer is beneficial. Only in extreme circumstances, where the quality of printed items is extremely low, the size of the 3D-printer is extremely large compared to the knapsack size, when there is no time to print the items, or when demand for items is low, packing no 3D-printers is the best option.

**Keywords:** disaster response operations, 3D-printing, two-stage stochastic programming, knapsack problems

## 1 Introduction

Natural disasters, like floods, hurricanes, tornadoes, earthquakes or volcanic eruptions, do have a significant impact on humanity. They cause tens of thousands of deaths, hundreds of thousands of injuries, and billions of dollars in economic losses each year around the world (Dilley et al., 2005). The International Federation of Red Cross and Red Crescent Societies (2005) show that a significant

part hereof is caused in the aftermath of such natural disasters. Considering this fact, it is important that the allocation of basic survival resources, such as clean water, food, medicines, cooking utensils, and shelter, as well as the allocation of supportive resources, such as spare parts and repair tools is executed in a fast and proper way (Özdamar et al., 2004; Perry, 2007).

An aspect that complicates the allocation of basic survival and supportive resources is the level of uncertainty. Typically, the impact of a natural disaster is hard to predict (Berkes, 2007; Van Wassenhove, 2006) and, as a consequence, the actual number of basic survival and supportive resources needed, is hard to predict as well. A new, emerging technique that can deal with this form of uncertainty is the technique of 3D-printing, (see, e.g., Savonen et al. (2018); Tatham et al. (2015) and the references therein). With this technique, also known as additive manufacturing, one can create objects (e.g., cooking utensils, bottles, screwdrivers, shelter, pipe clamps, or water purification kits) locally. This means that, if one decides to transport 3D-printers to a disaster area, one introduces the possibility to anticipate on actual (i.e., local) demand. For instance, in Nepal, this technique has been successfully used for printing medical supplies and restoring water supply in the aftermath of the earthquake in 2015 (Hall, 2016; Saunders, 2017) and in Haïti, this technique has been successfully used for printing screwdrivers, pipe clamps and bottles in the aftermath of the earthquake in 2010 (Goulding, 2017).

The benefits of using 3D-printing are not limited to the possibility to deal with demand uncertainty. For instance, in literature, it is believed that printing material, which is in liquid or powder form, can be packed much more efficient than physical items. Consequently, more printing material (and subsequently more (future) printed items) can be packed for a disaster response mission. However, 3D-printing also has some drawbacks, which makes it a less attractive option. For instance, the weight and volume of 3D-printers is still significant nowadays. This implies that, if one plans to transport some 3D-printers to a disaster area, this weight and volume cannot be used to pack other critical items. Moreover, the quality, and usability, of printed items is still, inferior to non-printed items. And, finally, the 3D-printing time of items is still in terms of hours –time that is crucial during a disaster response mission. Indeed, with these advantages and disadvantages of 3D-printing (see Table 1 for a summary), is not clear immediately whether we should pack 3D-printers for a disaster response mission, or not.

Positive effects	Negative effects
<i>Possibility to deal with demand uncertainty</i>	<i>3D-printing time</i>
<i>Efficient packing of printing material</i>	<i>Weight and volume of 3D-printers</i>
	<i>Reduced quality of printed items</i>

Table 1: Positive and negative effects of using 3D-printing.

In this paper, we will shed light on when to bring 3D-printers or not, which amount of printing material to take and which items to bring physically, to a disaster area. For that, we introduce a new

type of problem, which we call the two-stage stochastic 3D-Printing knapsack problem (TSS-3DKP). In the first stage of this problem, a decision maker has to fill a multidimensional knapsack (e.g., a cargo airplane at a home location), with the special consideration of taking 3D-printers with an associated amount of printing material, or not. Then, in the second stage (e.g., in the disaster area), demand for basic survival requirements as well as supportive resources is revealed, and the physically brought items that match demand are allocated. If at least one 3D-printer is taken, the next consideration is how to use the printing material for the remaining demand. Based on a maximum of printing time per 3D-printer - which resembles the urgency of delivering items fast (e.g., within 24 or 48 hours) - a decision on the number and type of printed items (per 3D-printer) is made. Meeting demand, via a physically brought item or a printed item, results in an item-specific reward, which may depend on its nature (i.e., whether it is printed or not). This reward may represent the (relative) importance of having such an item. Aim is to make, a priori, a decision upon the number of physical items to take, whether to bring 3D-printers or not, the amount of printing material to take, and how to use it (i.e., which items to print), in order to maximize the expected total reward.

From a practical perspective, it is important that our TSS-3DKP can be solved within reasonable computation time (e.g., within hours). This may be challenging, since our problem is, in contrast to standard two-stage stochastic programming problems (Birge and Louveaux, 2011), NP-hard in both the first and the second-stage problem. We tackle our TSS-3DKP, by transforming it into a deterministic equivalent (i.e., into a large integer linear programming problem). This allows us to solve our problem by standard solvers within reasonable computation time. We want to emphasize that this transformation is not straightforward, because the number of constraints and variables of our second-stage problem depend on the first-stage decisions. We overcome this dependency by introducing several dummy variables and a smart upper bound on the possible number of packed 3D-printers.

We use our deterministic equivalent to execute several numerical experiments. In particular, we will investigate how the advantages and disadvantages of Table 1 (i.e., quality of printed items, the weight and volume of 3D-printers, the storage efficiency of printing material, the printing time, and the demand uncertainty) effect the decision of when to use 3D-printing, and when not.

Now, we summarize the main contributions of this paper:

1. We formulate a new two-stage stochastic knapsack problem, inspired by disaster response operations, in which one determines the numbers of physical items and 3D-printers to pack, the amount of printing material to pack, and how to use this printing material.
2. We are able to provide a deterministic equivalent for our two-stage stochastic 3D-printing knapsack problem, which allows us to solve the problem within reasonable computation time.
3. Via numerical experiments, we are able to identify under which circumstances it is beneficial to bring 3D-printers, and by how much, to a disaster area.

Finally, it is worth mentioning that our model formulation of the TSS-3DKP and associated (numerical) results may find future applications in other domains such as the military (e.g. a military mission on a remote location) and aerospace (e.g., a long-term mission to Mars).

The remainder of this paper is organized as follows. We start in Section 2 with a literature review. In Section 3, we introduce our two-stage stochastic programming problem. In Section 4, we focus on solving our problem, by presenting and testing a deterministic equivalent. Then, in Section 5, we identify, via numerical experiments, under which circumstances it is beneficial to bring 3D-printers, and by how much, to a disaster area. We close with a conclusion in Section 6.

## 2 Literature review

In this section, we will identify how our TSS-3DKP contributes to the literature. We do so by providing an overview of the (OM/OR) literature on two-stage stochastic knapsack problems, disaster operations management, and 3D-printing, and for each of them, identify how our TSS-3DKP contributes to it.

### 2.1 Two-stage stochastic knapsack problems

A classical combinatorial optimization problem is the knapsack problem (Kolesar (1967)). In this problem, a decision maker has to fill a knapsack with items, each with an associated weight and reward. The aim of the decision maker is to find the best layout, i.e., a feasible combination of items, according to the size of the knapsack, that maximizes the sum of the rewards. Many variations of this classical problem have been studied. For some of these variations, the decision maker has to fill a knapsack based on incomplete information (e.g., the reward of items is uncertain) and has recovery options once complete information becomes available. The aim of the decision maker is then to maximize the expected reward. In literature, these variations are classified as two-stage stochastic knapsack problems. The first stage refers to the setting with partial information and the second stage refers to the setting with complete information and recovery options.

The literature on two-stage stochastic knapsack problems is relative young and consists of a few papers only. Kosuch and Lisser (2011) study a two-stage stochastic knapsack problem with normally distributed item weights and a recovery option that limits to either the addition or the removal of items. To restrict the probability that the item weights exceed the knapsack size (in the second stage) they include a chance constraint to the first-stage problem. A variant, with discretely distributed item weights, is studied by Kosuch (2014). Knapsack size uncertainty is studied by Akker et al. (2016). They model this uncertainty by discrete scenarios and introduce a recovery option that consists of the removal of items. This problem is extended to multiple knapsacks by Tönissen et al. (2017). Finally, a quadratic version of the two-stage stochastic knapsack problem with item weight uncertainty has been studied by Lisser et al. (2010) and with both item weight and reward uncertainty by Song et al. (2018).

Our paper contributes to this new stream of literature, by studying a two-stage stochastic knapsack problem with uncertainty in (item) demand and a recovery option that consists of 'creating' new items out of another item (i.e., the printing material). To the best of our knowledge, this paper is the first that studies a two-stage stochastic knapsack problem with such type of uncertainty and recovery option.

## 2.2 Disaster operations management

Disaster operations represent the set of activities performed before, during and after a disaster in order to diminish its impact (Altay and Green (2006)). In general, it is hard to prepare for, and consequently to manage such type of activities. This is mainly due to the unpredictable nature of disasters. In the OM/OR literature on disaster operations management, this uncertainty plays a predominant role (see, e.g., the extensive literature reviews of Galindo and Batta (2013) and Leiras et al. (2014)). Common uncertainties in this literature include demand uncertainty (Alem et al., 2016; Tofighi et al., 2016; Tricoire et al., 2012; Verma and Gaukler, 2015), uncertainty of the condition of facilities and/or roads (Chang et al., 2007; Fan and Liu, 2010; Hu et al., 2019; Li et al., 2012; Sanci and Daskin, 2019), travel time uncertainty (Ahmadi et al., 2015; Bayram and Yaman, 2018; Döyen et al., 2012), and cost uncertainty (Bozorgi-Amiri et al., 2012; Li et al., 2011; Paul and Zhang, 2019).

Two-stage stochastic programming is an appropriate method to deal with such uncertainties. Instead of using standard cost minimizing objectives, in disaster operations management, meeting expected demand is often preferred due to significant consequences with respect to human lives (Barbarosolu and Arda (2004) and Rawls and Turnquist (2010)). In Balcik and Beamon (2008) the fulfillment of demand is modelled by maximizing the expected satisfied demand, in Salmerón and Apte (2010) by minimizing the expected casualties, and in Noyan et al. (2016) by maximizing the expected accessibility. A common first-stage decisions is the in-advance storage of relief items (Davis et al. (2013); Lodree Jr et al. (2012)) or locating facilities (Elçi and Noyan (2018); Li et al. (2011)). A few authors use other first-stage decisions such as the retrofitting of roads (Peeta et al. (2010)), buildings (Zolfaghari and Peyghaleh (2015)) or bridges (Liu et al. (2009)). The most prevalent second-stage decisions are the transport of commodities in the aftermath of a disaster (Rezaei-Malek et al. (2016); Tofighi et al. (2016)) or an evacuation plan (Li et al. (2012, 2011)). We refer to Grass and Fischer (2016) for an overview of two-stage stochastic programming problems for disaster operations management.

In this paper, we also use two-stage stochastic programming to deal with uncertainty (namely, demand uncertainty). In particular, we will use this method in a disaster response setting. To the best of our knowledge, we are the first who use two-stage stochastic programming to investigate whether bringing 3D-printers to a disaster area is useful or not. We want to emphasize that there exist some qualitative papers on 3D-printing for disaster response missions (see, e.g., Savonen et al. (2018) and Rodríguez-Espíndola and Beltagui (2018)). These studies focus on the necessary requirements and specifications 3D-printers should have for being successfully used in a disaster area.

## 2.3 3D-printing

Although the technique of 3D-printing, also known as additive manufacturing, has already been applied in practice for years, the development of quantitative models studying the impact of 3D-printing in OM/OR literature is rather limited. To the best of our knowledge, there exist a few published works in this domain only. Westerweel et al. (2018) investigate the impact of 3D-printing on component design. In particular, they characterise under which conditions a component should be produced with traditional technology and under which conditions a component should be produced via 3D-printers. Song and Zhang (2019) present a general framework to study the design of spare parts logistics in the presence of 3D-printing technology. In particular, they formulate a model that determines which parts to stock and which to print. Khajavi et al. (2014) and Liu et al. (2014) both model a multi-echelon spare parts supply chain and numerically investigate the effect of centralized versus decentralized 3D-printing capacity. Dong et al. (2016) evaluate the choice between traditional technology and 3D-printing related to assortment planning for general inventories in a manufacturing setting. An overlap in all these papers is the presence of a 3D-printer with a, possibly unlimited, amount of printing material. However, in our problem, the presence of a 3D-printer and associated printing material is not guaranteed: it is an essential decision in our optimization problem. This indicates that we do study a new type of 3D-printing (OM/OR) problem and so contribute to this rather new stream of literature.

## 3 Two-stage stochastic 3D-printer knapsack problem

In this section, the two-stage stochastic 3D-printing knapsack problem (shortly TSS-3DKP) will be formulated. We will do so by first describing the two stages (i.e., the first and second stage) and thereafter presenting the mathematical formulation of our TSS-3DKP.

### 3.1 The first stage

In the first stage, a decision maker has to fill a multidimensional knapsack (e.g., a cargo airplane at a home location) with physical items, units of printing material and 3D-printers such that it does not exceed the weight capacity  $W$  and volume capacity  $V$  of the knapsack. Each physical item  $i$ , from the set of physical items  $N$ , has a weight  $w_i$  and a volume  $v_i$ . In addition, each unit of printing material has weight  $w_b$  and volume  $v_b$  and each 3D-printer has weight  $w_p$  and volume  $v_p$ . The **first-stage decision of the decision maker** is denoted by vector  $\mathbf{a} = ((a_i)_{i \in N}, a_p, a_b)$  with  $a_i$  the number of times item  $i \in N$  is added to the knapsack,  $a_b$  the units of printing material included into the knapsack, and  $a_p$  the number of 3D-printers added to the knapsack.

### 3.2 The second stage

In the second stage (e.g., a disaster area) the decision maker has to allocate the  $a_b$  units of printing material to the  $a_p$  3D-printers. Before the decision maker does so, demand for items is revealed and the physically brought items that match (this) demand are allocated. Demand is modelled by a discrete set of scenarios  $S$ , where each scenario  $s \in S$  specifies the demand  $d_i^s$  for all items  $i \in N$  and occurs with probability  $q_s \in [0, 1]$  such that  $\sum_{s \in S} q_s = 1$ . We want to stress that, in literature on disaster operations management, discrete scenarios are often used to model uncertainty. This is due to the uniqueness of disasters, which makes the determination of probability distributions problematic (see, e.g., Grass and Fischer (2016)). For each scenario  $s \in S$ , we denote the number of physically brought items that match demand by  $a_i^s = \min\{a_i, d_i^s\}$  for all  $i \in N$  and consequently, denote remaining demand by  $d_i^s - a_i^s$  for all  $i \in N$ . Based on this remaining demand, the decision maker has to allocate the  $a_b$  units of printing material to the  $a_p$  3D-printers. For this second-stage decision, the decision maker has to take into account the maximum amount of printing time  $T$  per 3D-printer, and the fact that each printable item  $i \in N^p$ , with  $N^p \subseteq N$ , requires  $m_i$  units of printing material and time  $t_i$  to print. Finally, given that  $P(x) = \{i \in \mathbb{N}_+ | i \leq x\}$  is defined as the set of 3D-printers for any  $x \in \mathbb{N}_0$ , the **second-stage decision of the decision maker** for each scenario  $s \in S$  is denoted by  $\mathbf{p}^s = (p_{ij}^s)_{i \in N, j \in P(a_p)}$  with  $p_{ij}^s$  the number of times item  $i$  is printed on 3D-printer  $j$  in scenario  $s$ .

Meeting demand via a physically brought item  $i \in N$  or printed item  $i \in N^p$  results in an item-specific reward, which may depend on its nature (i.e., whether it is printed or not). This reward may represent the (relative) importance of having such an item. The reward for a physically brought and matched item  $i \in N$  is  $r_i$  and its printed reward is  $\alpha r_i$  with  $\alpha \in [0, 1]$  for all  $i \in N^p$ . This factor  $\alpha$  resembles the lower quality of the printed items. The total reward for a given first-stage decision  $\mathbf{a}$  and a second-stage decision  $\mathbf{p}^s$  for scenario  $s \in S$  is given by  $\sum_{i \in N} a_i^s r_i + \alpha \sum_{i \in N^p} \sum_{j \in P(a_p)} p_{ij}^s r_i$ . And, subsequently, the expected total reward, based on all possible demand scenarios, is given by  $\sum_{s \in S} q_s \left[ \sum_{i \in N} a_i^s r_i + \alpha \sum_{i \in N^p} \sum_{j \in P(a_p)} p_{ij}^s r_i \right]$ . Aim of the decision maker is to make a first-stage decision  $\mathbf{a}$  and a second-stage decision  $\mathbf{p}^s$  for all  $s \in S$  that maximizes the expected total reward<sup>1</sup>.

### 3.3 Mathematical formulation of the TSS-3DKP

Now we give a mathematical formulation of our two-stage stochastic 3D-printing knapsack problem.

$$\begin{aligned} \text{TSS-3DKP} \quad & \max \sum_{s \in S} q_s Q(\mathbf{a}, s) \\ \text{s.t.} \quad & \sum_{i \in N} a_i w_i + a_p w_p + a_b w_b \leq W, \end{aligned} \tag{1}$$

---

<sup>1</sup>For the specific setting with  $r_i = 1$  for all  $i \in N$ , our objective can be recognized as the (expected) amount of satisfied demand. This objective is used frequently in the literature on disaster operations management.



$$\sum_{i \in N} a_i v_i + a_p v_p + a_b v_b \leq V, \quad (2)$$

$$a_b \leq a_p \mathcal{M}, \quad (3)$$

$$a_p, a_b, a_i \in \mathbb{N}_0 \quad \forall i \in N, \quad (4)$$

where,

$$Q(\mathbf{a}, s) := \max \sum_{i \in N} a_i^s r_i + \sum_{i \in N^p} \sum_{j \in P(a_p)} \alpha p_{ij}^s r_i$$

$$\text{s.t.} \quad a_i^s \leq d_i^s \quad \forall i \in N \setminus N^p, \quad (5)$$

$$a_i^s + \sum_{j \in P(a_p)} p_{ij}^s \leq d_i^s \quad \forall i \in N^p, \quad (6)$$

$$a_i^s \leq a_i \quad \forall i \in N, \quad (7)$$

$$\sum_{i \in N^p} \sum_{j \in P(a_p)} p_{ij}^s m_i \leq a_b, \quad (8)$$

$$\sum_{i \in N^p} p_{ij}^s t_i \leq T \quad \forall j \in P(a_p), \quad (9)$$

$$a_i^s \in \mathbb{N}_0 \quad \forall i \in N, \quad (10)$$

$$p_{ij}^s \in \mathbb{N}_0 \quad \forall i \in N^p, \quad \forall j \in P(a_p). \quad (11)$$

Constraints (1) and (2) ensure that no more number of items, units of printing material and 3D-printers will be packed, according to the weight and volume restrictions of the knapsack. Constraint (3), which is a big- $\mathcal{M}$  formulation, guarantees that it is only allowed to pack printing material if at least one 3D-printer is taken. Constraint (4) ensures integrality of the decision variables. Constraints (5), (6), and (7) ensure that reward can only be gained for those physical items and printed items that meet demand. Constraint (8) guarantees that the amount of printing material used does not exceed the amount of printing material packed. Constraints (9) ensure that each 3D-printer can be used for at most  $T$  time units. Constraints (10) and (11) ensure integrality of the decision variables. Note that we model  $a_i^s (= \min\{a_i, d_i^s\})$  as a decision variable in our second-stage problem which takes, due to our formulation, the minimum of the values  $a_i$  and  $d_i^s$  for all  $i \in N$  and all  $s \in S$ . Also note that the number of constraints (9) and (11) depend on the first-stage decision  $a_p$ . Hence, the number of constraints and variables of the second-stage problem depend on the decisions in the first-stage problem.

We want to emphasize that both the first-stage and the second-stage problem of our TSS-3DKP are NP-hard integer programming problems. Both problems are NP-hard because they are generalisations of the unbounded knapsack problem (Martello and Toth, 1990), which is a classical NP-hard combinatorial optimization problem. We will support these claims, by showing that special cases of our first and second-stage problems can be recognized as unbounded knapsack problems. For the first-stage, we consider the special case with no 3D-printers, no printing material, and exactly one scenario for

which demand for each item is infinite. Furthermore, we set  $V = 0$  and  $v_i = 0$  for all  $i \in N$ . For this situation, our first-stage problem can be recognized as a classical unbounded knapsack with size (i.e., weight capacity)  $W$  and items with weight  $w_i$  and reward  $r_i$  for all  $i \in N$ . For the second-stage, we consider the special case with no physical items, exactly one 3D-printer, infinite demand, and  $m_i = 0$  for all  $i \in N^p$ . For this situation, our second-stage problem can be recognized as a classical unbounded knapsack with size  $T$  and items with weight  $t_i$  and reward  $\alpha r_i$  for all  $i \in N^p$ .

We will now illustrate our two-stage stochastic 3D-printing problem by means of a small example.

**Example 1.** Consider a knapsack with weight capacity  $W = 4$  and volume capacity  $V = 4$ , that can be filled with two types of printable items (i.e.,  $N = N^p = \{1, 2\}$ ). For the items, we have  $w_1 = v_1 = w_2 = v_2 = 4$  with  $r_1 = 1$  and  $r_2 = 2$ . The items require  $m_1 = m_2 = 2$  units of printing material and each unit of printing material has weight  $w_b = 1$  and volume  $v_b = 1$ . In addition, each 3D-printer has weight  $w_p = 2$  and volume  $v_p = 2$ , and, in total, there are two scenarios (i.e.,  $S = \{s_1, s_2\}$ ), with  $q_{s_1} = 0.7$  and  $q_{s_2} = 0.3$ . Demand is given by  $d_1^{s_1} = 1, d_2^{s_1} = 0, d_1^{s_2} = 0, d_2^{s_2} = 1$ . Finally, the maximum amount of printing time per 3D-printer is given by  $T = 1$ , the printing time per item by  $t_1 = t_2 = 1$ , and the quality factor (per printed item) by  $\alpha = 0.8$ .

For this specific situation, there exist three candidate solutions, namely: (i) taking one physical item of type 1, (ii) taking one physical item of type 2, or (iii) taking one 3D-printer with 2 units of printing material. For the first two candidate solutions, no 3D-printer is taken, and consequently, the expected total reward for the first strategy equals  $(0.7 \cdot 1 =) 0.7$  and for the second strategy equals  $(0.3 \cdot 2 =) 0.6$ . For determining the expected reward for the third strategy, we first need to identify how the units of printing material should be used in the two demand scenarios. For the first demand scenario (i.e.,  $s_1$ ), it is optimal to print item 1, resulting in  $(0.7 \cdot 0.8 \cdot 1 =) 0.56$ , and for the second demand scenario (i.e.,  $s_2$ ), it is optimal to print item 2, resulting in  $(0.3 \cdot 0.8 \cdot 2 =) 0.48$ . Hence, the expected total reward under the third strategy equals  $(0.56 + 0.48 =) 1.04$ , implying that bringing a 3D-printer and allocating the printing material according to demand is optimal.

To the best of our knowledge, there exist no standard (commercial) solvers that can directly solve two-stage stochastic programming problems. For that reason, we will transform our TSS-3DKP into another, but equivalent, formulation that is suitable for standard solvers (e.g., CPLEX).

## 4 Solving the TSS-3DKP

In this section we will present an equivalent formulation of our TSS-3DKP that can be solved by standard solvers. Moreover we will show, via some numerical experiments, that this equivalent formulation can be solved (with standard solvers) within reasonable computation time.

### 4.1 An equivalent formulation for solving the TSS-3DKP

It is well-known that a standard two-stage stochastic programming problem can be modelled as a large integer linear programming problem with variables and constraints for each of the scenarios. Such an

integer linear programming formulation is called the deterministic equivalent. The idea is to transform our TSS-3DKP, which is also a two-stage stochastic programming problem, into its deterministic equivalent. However, the TSS-3DKP has the non-standard feature that the number of constraints and variables in the second-stage depends on the first-stage decisions, implying that the deterministic equivalent cannot be given directly. We overcome this dependency, by identifying an upper bound on the total number of 3D-printers that can be packed, and include printed-related constraints and variables of the second-stage problem into our deterministic equivalent as if the knapsack would be filled with this upper bound of 3D-printers. Although this approach may lead to an increase in the total number of constraints and variables, it allows for a reformulation of our TSS-3DKP<sup>2</sup>.

Before we present the deterministic equivalent formulation, we first explain how we determine an upper bound on the maximum number of packed 3D-printers. For each scenario  $s \in S$ , we determine how many 3D-printers are needed to print all demand  $d^s$ , assuming that items are allocated one by one in order of their indices to the 3D-printers. Hence, we first try to allocate all demand of item 1 to the first 3D-printer, followed by item 2, 3,  $\dots$ ,  $|N^p|$ , and we go to the next 3D-printer if adding another item would exceed the printing  $T$  of the 3D-printer. Taking the maximum number of 3D-printers over all possible scenarios then gives us our first upper bound  $U$ . Sometimes, this upper bound exceeds the total knapsack weight or volume. In those cases, we select the maximum number of 3D-printers that fit the knapsack as upper bound. Formally, we define our upper bound as  $Z = \min \{ \lfloor \frac{W}{w_p} \rfloor, \lfloor \frac{V}{v_p} \rfloor, U \}$ .

We will now illustrate our upper bound by means of a small example.

**Example 2.** Consider a situation with two types of printable items (i.e.,  $N = N_p = \{1, 2\}$ ), two scenarios (i.e.,  $S = \{s_1, s_2\}$ ) with associated demands  $d_1^{s_1} = 3, d_2^{s_1} = 1, d_1^{s_2} = 1, d_2^{s_2} = 2$ , a knapsack with weight capacity  $W = 10$  and volume capacity  $V = 12$ , a 3D-printer with  $w_p = 5, v_p = 3, T = 5$ , and printing times  $t_1 = 2$  and  $t_2 = 3$ . We now allocate the demand of the items to the 3D-printers in order of their indices. Hence, for scenario  $s_1$ , we allocate two items of type 1 to the first 3D-printer, and one item of type 1 and one item of type 2 to the second 3D-printer. For scenario  $s_2$ , we allocate one item of type 1 and one item of type 2 to the first 3D-printer, one item of type 2 to the second 3D-printer, and one item of type 2 to the third 3D-printer. The maximum number of 3D-printers needed (over all scenarios) is thus  $U = 3$ . Moreover,  $\lfloor \frac{W}{w_p} \rfloor = 2$  and  $\lfloor \frac{V}{v_p} \rfloor = 4$ , implying an upper bound on the total number of 3D-printers of  $Z = 2$ .

For the deterministic equivalent formulation, we also need to introduce some new notation. For the given upper bound  $Z$ , we define binary variables  $y_j \in \{0, 1\}$  for all  $j \in P(Z)$ , with  $y_j = 1$  if 3D-printer  $j$  is packed and 0 otherwise.

Now, we are ready to present the deterministic equivalent of our TSS-3DKP.

---

<sup>2</sup>In Section 4.2, we will also show that, although the total number of constraints and variables grows under this approach, our deterministic equivalent formulation can still be solved in reasonable computation time.

$$\text{ILP-3DKP}(Z) \quad \max \sum_{s \in S} q_s \left[ \sum_{i \in N} a_i^s r_i + \sum_{i \in N^p} \sum_{j \in P(Z)} \alpha p_{ij}^s r_i \right]$$

$$\text{s.t.} \quad \sum_{i \in N} a_i w_i + a_b w_b + \sum_{j \in P(Z)} y_j w_p \leq W, \quad (12)$$

$$\sum_{i \in N} a_i v_i + a_b v_b + \sum_{j \in P(Z)} y_j v_p \leq V, \quad (13)$$

$$a_b \leq \sum_{j \in P(Z)} y_j M, \quad (14)$$

$$a_i^s \leq d_i^s \quad \forall i \in N \setminus N^p, \quad \forall s \in S, \quad (15)$$

$$a_i^s + \sum_{j \in P(Z)} p_{ij}^s \leq d_i^s \quad \forall i \in N^p, \quad \forall s \in S, \quad (16)$$

$$a_i^s \leq a_i \quad \forall i \in N, \quad \forall s \in S, \quad (17)$$

$$\sum_{i \in N^p} \sum_{j \in P(Z)} p_{ij}^s m_i \leq a_b \quad \forall s \in S, \quad (18)$$

$$\sum_{i \in N^p} p_{ij}^s t_i \leq T y_j \quad \forall j \in P(Z), \quad \forall s \in S, \quad (19)$$

$$y_j \leq y_{j-1} \quad \forall j \in P(Z) \setminus \{1\}, \quad (20)$$

$$y_j \in \{0, 1\} \quad \forall j \in P(Z), \quad (21)$$

$$a_b \in \mathbb{N}_0 \quad (22)$$

$$a_i \in \mathbb{N}_0 \quad \forall i \in N, \quad (23)$$

$$a_i^s \in \mathbb{N}_0 \quad \forall i \in N, \quad (24)$$

$$p_{ij}^s \in \mathbb{N}_0 \quad \forall i \in N^p, \quad \forall j \in P(Z), \quad \forall s \in S. \quad (25)$$

The objective coincides with the original objective of the TSS-3DKP: it is the expected total reward obtained from meeting demand by physical and 3D-printed items over all possible scenarios. Constraints (12) and (13) limit the total item, 3D-printer and printing material weight and volume to the weight and volume restrictions of the knapsack. The big- $\mathcal{M}$  Constraint (14) guarantees that printing material is only taken when at least one 3D-printer is taken. This constraint guarantees indirectly that we can only print when we have a 3D-printer. Constraints (15), (16) and (17) ensure that reward can only be gained for those physical items and 3D printed items that meet demand. Constraints (18) guarantee that the amount of printing material used does not exceed the amount of printing material packed. Constraints (19) ensure that each 3D-printer can be used for at most  $T$  time units. Note that this amount (of time) becomes zero if the 3D-printer is not taken. Constraints (20) guarantee that 3D-printer  $j$  is taken if and only if 3D-printer  $j - 1$  is taken. This excludes many symmetric solutions and consequently decreases the solution time of the CPLEX solver. Constraints (22), (23), (24), and (25) ensure integrality of the decision variables.

## 4.2 Computational experiments

In this section, we will discuss the outcomes of our computational experiments to provide insights in the solvability of our deterministic equivalent formulation. We do so by first explaining how we generate our instances and then continue by discussing our computational experiments.

### 4.2.1 Instance generation

For every instance, we first specify the following parameters: the number of items  $|N|$ , an upper limit to demand  $D$ , the number of scenario  $|S|$ , the (printed item) quality factor  $\alpha$ , and the 3D-printer weight  $w_p$  and volume  $v_p$ . Thereafter, we will generate the items. Remember that every item has a weight  $w_i$ , a volume  $v_i$ , a reward  $r_i$ , a required printing material  $m_i$ , and printing time  $t_i$ . We generate the weight  $w_i$  and the reward  $r_i$  according to the uncorrelated instance class of Pisinger (Pisinger, 2005). This class is realistic for disaster response missions: it considers many different items for which the weight, volume, and rewards vary heavily. For the volume, printing time and required printing material per item, we use the following distributions for generation:  $v_i \sim \lfloor U[0.2, 5]w_i \rfloor$ ,  $t_i \sim \lfloor U[0, 10] \rfloor$  and  $m_i \sim \lfloor U[0.5, 0.9] \cdot \min(w_i, v_i) \rfloor$ . Note that we use  $\lfloor \cdot \rfloor$  to denote that we round our values to the nearest integer. Moreover, we set  $w_b = v_b = 1$ . In combination with the generation of  $m_i$ , this implies that printed items can always be packed more efficiently than physical items. Thereafter, we generate the demand for the  $|S|$  scenarios. First, we generate the maximum demand for each item  $i \in N$  from uniform distribution  $U_i \sim \lfloor U[1, D] \rfloor$ . Then, for each scenario  $s \in S$  we generate a demand  $d_i^s \sim \lfloor U[0, U_i] \rfloor$  for all  $i \in N$  and the probability  $q_s = \frac{1}{|S|}$ . We continue by generating printing time  $T \sim \lfloor U[0.2, 1] \cdot \sum_{s \in S} q_s \sum_{i \in N} t_i d_i^s \rfloor$ . Note that this printing time  $T$  is a fraction of the average total time to print all items. Finally, we generate the knapsack weight capacity  $W \sim \lfloor U[0.5, 1] \cdot \sum_{s \in S} q_s \sum_{i \in N} w_i d_i^s \rfloor$ , which can be seen as a fraction of the average total weight of all items, and volume capacity  $V \sim \lfloor U[0.5, 2] \cdot W \rfloor$ , which depends on the knapsack weight capacity.

We will study various instance sets with varying number of items, upper limit demand and number of scenarios, but with the same  $\alpha = 0.8$  and  $w_p = v_p = 5000$ . We will refer to such an instance set by  $NxDySz$  with  $x$  the number of items,  $y$  the upper limit to the demand and  $z$  the number of scenarios.

### 4.2.2 Computational results

The experiments are programmed in Java with the CPLEX library version 12.8.0, and run on a laptop with an Intel Core i7-4710MQ Quad Core 2.5 GHz processor with 32 GB of RAM. We use CPLEX standard settings and the reported solution times include everything that is required to solve our instances, such as the time required for reading the instances, the time to determine upper bound  $Z$ , the time required for building the model, and the time to solve the model. The maximum computation time to solve a single instance is set equal to 3600 seconds.

We will test our ILP-3DKP(Z) on five different instance sets, each consisting of 100 instances. The first instance set, which we use as a base case, consists of 100 items, an upper limit of demand of 100, and 50 scenarios. Recall that we (can) refer to this instance set as N100D100S50. In addition, we will study instance set N200D100S50, N100D200S50, N100D100S100, and N200D200S100.

We will study each instance set on four performance criteria: the average number of cuts and nodes used by the branch-and-bound tree of CPLEX, the number of instances that cannot be solved within 3600 seconds, and the average computation time of those instances that can be solved within 3600 seconds. For the first two criteria (i.e., cuts and nodes) we want to stress that if an instance is not solved within 3600 seconds, a fail is registered, and the number of cuts and nodes used so far is reported. The performance criteria of the five instance sets can be found in Table 2.

Instance set	cuts (#)	nodes (#)	fails (#)	time (seconds)
N100D100S50	1297	38822	1	12
N200D100S50	2870	0	0	3
N100D200S50	1488	1	0	1
N100D100S100	2638	19921	1	50
N200D200S100	5616	5	0	16

Table 2: The performance criteria of the 5 instance sets.

Table 2 indicates that almost all instances (except for the two failed ones) can be solved within minutes. In particular, we observe, based on individual results not reported here, that most of these instances do not require branching (i.e., generate no nodes) and are solved entirely by adding cuts. However, the two failed instances (one from N100D100S50 and one from N100D100S100) turn out to be hard to solve (i.e., they cannot be solved to optimality within 3600 seconds). These particular instances have a high number of generated nodes (namely 3017763 and 1158576) and consequently require extensive branching. Moreover, it turns out that, if we increase the maximum computation time to 86,400 seconds (i.e., 24 hours) these two instances cannot be solved to optimality either.

Since finding an optimal solution seems to be a problem for some instances, the next step is to investigate the effect of the stopping criteria on the branch-and-cut algorithm (in CPLEX). The standard stopping criteria of this algorithm is the relative gap of 0.01%. We will study the effect by solving a specific instance set for various relative gaps, namely 0.01%, 0.1%, 1.0%, and 10%. The specific instance set combines the five, previously introduced, instance sets. We present our results in Table 3.

The results from Table 3 indicate that for some instances, a lot of time is spent on improving the solution slightly, or on proving that the optimal solution has been found. This becomes most apparent

Relative gap	cuts (#)	nodes (#)	fails (#)	time (seconds)
0.01%	2782	11750	2	31
0.1%	2832	1	0	4
1%	2942	0	0	4
10%	2777	0	0	3

Table 3: Comparison of the ILP-3DKP(Z) method for different relative gaps.

when increasing the gap from 0.01% to 0.1%. In that specific case, the number of nodes decreases from 11750 to 1, which illustrates that branching is not required anymore for most instances. Moreover, from Table 3, we can learn that increasing the relative gap from 0.1% to 10% has a rather limited effect on the average computation time (i.e., computation time reduces from 4 to 3 seconds, on average).

We believe that for practical applications (e.g., a disaster response mission), it is important to obtain sufficiently good solutions within hours. Solving our TSS-3DKP with a relative gap of 0.1% will exactly do that. Consequently, we will apply this relative gap in the remainder of this paper.

## 5 To print or not to print?

In Section 4, we presented a deterministic equivalent of our two-stage stochastic 3D-printing knapsack problem. We have shown that, with this formulation, it is possible to solve our original problem with standard solvers in reasonable computation time. Consequently, this formulation can be used to identify whether bringing 3D-printers to a disaster area is useful, or not. As discussed in the introduction of this paper (i.e., in Section 1), this decision depends, amongst others, on the following five aspects: (i) the quality of printed items, (ii) the volume and weight of 3D-printers, (iii) the storage efficiency of printing material, (iv) the 3D-printing time, and (v) demand uncertainty. In this section, we will study the effect of these five aspects on the outcomes of our TSS-3DKP. For that, we will, for each aspect, construct a number of instance sets. These instance sets will be constructed such that they vary in the associated aspect only (e.g., a varying quality of the printed items  $\alpha$ , or a varying weight  $w_p$  and volume  $v_p$  of the 3D-printer). Per instance set, we then answer the following two questions:

1. Should we bring 3D-printers, and if so, how many?
2. What is the added value of using 3D-printers?

For answering the first question, we will return, per instance set, the median, minimum, maximum, and average number of 3D-printers taken. For answering the second question, we will first determine, per instance, the percentage increase in total reward if bringing 3D-printers is allowed. We then answer our second question by taking the median, minimum, maximum, and average of this percentage over

all instances of the instance set. Since we construct several instance sets per aspect, we will answer these questions several times per aspect. We now continue by discussing these outcomes per aspect.

## 5.1 The quality of printed items

As discussed in the introduction of this paper, the quality of printed items is still inferior to physical items. This could prevent a decision maker from packing 3D-printers. In this section, we will study the effect of this quality on our two questions. We model the quality of the printed items by solving instance set N100D100S50 for different values of  $\alpha$ . In particular, we will solve this instance set for  $\alpha \in \{0.1, 0.2, \dots, 1.0\}$ . The results are presented in Table 4.

$\alpha$	3D-printers (#)		reward (%)	
	median	min-max (average)	median	min-max (average)
0.1	0	0-0 (0.0)	0.0	0-0 (0.0)
0.2	0	0-1 (0.0)	0.0	0-0 (0.0)
0.3	1	0-1 (0.5)	0.1	0-3 (0.5)
0.4	1	0-1 (0.9)	2.2	0-8 (2.5)
0.5	1	1-2 (1.0)	6.0	0-13 (5.7)
0.6	1	1-2 (1.1)	9.6	2-21 (9.7)
0.7	1	1-3 (1.2)	14.0	4-32 (14.3)
0.8	1	1-3 (1.3)	19.2	7-45 (19.5)
0.9	1	1-3 (1.4)	25.0	9-58 (25.5)
1.0	1	1-4 (1.6)	31.8	13-73 (32.5)

Table 4: An overview of the median, minimum, maximum and average number of both the packed 3D-printers and the percentage increase in reward, for varying values of  $\alpha$ .

From Table 4 we can learn that an increase in  $\alpha$  (i.e., a higher quality of printed items) coincides with an increase in (i) the number of packed 3D-printers and (ii) the percentage increase in total reward. Next, we observe that for values of  $\alpha$  larger or equal than 30%, it is beneficial to bring 3D-printers, implying that printing items, even with a fairly low quality, payoffs rapidly.

## 5.2 The effect of the weight and volume of 3D-printers

The weight and volume of a 3D-printer is still significant nowadays. This weight and volume can also be used for transporting physical items. In this section, we will study the effect of the 3D-printers weight and volume on our two questions. We will do so by introducing a factor  $k$ . This factor represents the number of times a 3D-printer fits (in terms of weight and volume) within the knapsack.



In particular, we study instance set N100D100S50 with  $k \in \{2, 3, 5, 10, 20, 30, 40, 50, \infty\}$ . If  $k = \infty$ , we set  $w_p = v_p = 0$  and otherwise we set  $w_p = \lfloor \frac{W}{k} \rfloor$  and  $v_p = \lfloor \frac{V}{k} \rfloor$ . The results are presented in Table 5.

$k$	3D-printers (#)		reward (%)	
	median	min-max (average)	median	min-max (average)
2	0	0-0 (0.0)	0	0-0 (0.0)
3	1	0-1 (0.7)	5.1	0-21 (5.7)
5	1	0-1 (1.0)	11.4	0-32 (12.2)
10	1	1-2 (1.1)	16.1	3-39 (16.9)
20	1	1-2 (1.2)	18.8	4-43 (19.4)
30	1	1-3 (1.4)	19.8	5-44 (20.3)
40	1	1-3 (1.4)	20.3	5-45 (20.7)
50	1	1-3 (1.5)	20.8	5-45 (21.0)
$\infty$	3	2-7 (3.1)	22.7	6-47 (22.2)

Table 5: An overview of the median, minimum, maximum and average number of both the packed 3D-printers and the percentage increase in reward, for varying values of  $k$ .

The results of Table 5 show that the number of packed 3D-printers and the percentage increase in total reward increases in  $k$ . We observe that even for fairly low values of  $k$ , it is already beneficial to pack a 3D-printer. The weight and volume of a 3D-printer should exceed 33% ( $k < 3$ ) of the total knapsack weight and volume before packing physical items only (and thus no 3D-printer) is more beneficial.

### 5.3 The storage efficiency of printing material

As discussed in the introduction of this paper, the storage efficiency of physical items is inferior to the storage efficiency of printing material. In this section, we study the effect of the storage efficiency of printing material on our two questions. We do this by introducing factor  $l$ . This factor indicates the fraction of printing material needed in comparison to physical items, in terms of both weight as volume. In particular, we will study instance set N100D100S50 with  $l \in \{0, 0.1, 0.2, \dots, 1\}$  and set  $w_i = v_i$ ,  $m_i = \lfloor l \cdot w_i \rfloor$ , and  $w_b = v_b = 1$ . The results are presented in Table 6.

The results of Table 6 show that the number of 3D-printers and the percentage increase in total reward decreases for an increasing  $l$  (i.e. if the material needed to print items takes up more space). Even when there is no benefit of the storage efficiency of the printing material (i.e.,  $l = 1$ ) a 3D-printer is still often packed. This illustrates that the uncertainty in demand can be enough to bring 3D-printers.

$l$	3D-printers (#)		reward (%)	
	median	min-max (average)	median	min-max (average)
0	1	1-3 (1.3)	17.9	6-51 (19.5)
0.1	1	1-3 (1.3)	16.1	6-45 (17.4)
0.2	1	1-2 (1.2)	14.3	5-39 (15.3)
0.3	1	1-2 (1.2)	12.4	5-32 (13.1)
0.4	1	1-2 (1.1)	10.6	4-25 (10.9)
0.5	1	1-2 (1.1)	8.6	3-18 (8.7)
0.6	1	1-1 (1.0)	6.7	3-12 (6.8)
0.7	1	1-1 (1.0)	5.0	2-9 (4.9)
0.8	1	0-1 (1.0)	3.3	0-7 (3.3)
0.9	1	0-1 (0.9)	2.1	0-5 (2.0)
1	1	0-1 (0.8)	1.3	0-3 (1.2)

Table 6: An overview of the median, minimum, maximum and average number of both the packed 3D-printers and the percentage increase in reward, for varying values of  $l$ .

#### 5.4 3D-printing time

Time is crucial during a disaster response mission. However, the time to print a single item may still require hours. This may result in a relative low number of printed items. This number will increase if the time to print a single item reduces (e.g., due to improvements in printing technology). In this section, we will investigate how a change in this printing time affects our two questions. In particular, we will model this by introducing a factor  $m \in \{0, 0.1, 0.2, \dots, 1.0, \infty\}$ . This factor indicates the fraction of items that can be printed on one 3D-printer, on average. Given this setup, we will use maximum amount of printing time  $T = \lfloor m \cdot \sum_{s \in S} q_s \sum_{i \in N^p} t_i d_i^s \rfloor$ . We will present our results in Table 7.

From Table 7, we observe that, except for the case  $m = 0$ , in which no 3D-printers are taken, the number of packed 3D-printers decreases in  $m$  and converges for large  $m$  to exactly one 3D-printer. Note that for  $m = \infty$  all items can be printed on one 3D-printer and consequently there is no reason to pack more than one 3D-printer. From Table 7, we also observe that for  $0.6 \leq m \leq 1$  at most one 3D-printer is packed, although not all items can be printed on one 3D-printer. It turns out that in this interval, on average, more than half of the printing time of the demand is utilized on the first 3D-printer. Consequently, the second 3D-printer is, on average, not fully utilized, which makes it likely that this second 3D-printer is not being packed at all. Furthermore, we observe from Table 7 that the percentage increase in total reward increases until  $m = 0.8$  and stabilizes afterwards. This indicates that for large  $m$ , some items are preferred being packed physically (e.g. to get a better reward), although these items could be printed without packing any additional 3D-printers.

$m$	3D-printers (#)		reward (%)	
	median	min-max (average)	median	min-max (average)
0	0	0-0 (0.0)	0	0-0 (0.0)
0.1	3	1-5 (2.8)	10.3	1-21 (10.3)
0.2	2	1-3 (2.2)	14.7	4-28 (15.1)
0.3	2	1-2 (1.8)	16.8	5-33 (16.9)
0.4	1	1-2 (1.4)	17.4	5-35 (17.7)
0.5	1	1-2 (1.1)	18.2	5-35 (18.5)
0.6	1	1-1 (1.0)	18.7	5-37 (19.1)
0.7	1	1-1 (1.0)	19.0	5-38 (19.4)
0.8	1	1-1 (1.0)	19.2	5-39 (19.5)
0.9	1	1-1 (1.0)	19.2	5-39 (19.5)
1	1	1-1 (1.0)	19.2	5-39 (19.5)
$\infty$	1	1-1 (1.0)	19.2	5-39 (19.5)

Table 7: An overview of the median, minimum, maximum and average number of both the packed 3D-printers and the percentage increase in reward, for varying values of  $m$ .

## 5.5 The effect of demand uncertainty

One of main advantages of using 3D-printers is the possibility to deal with demand uncertainty. Consequently, it is likely that the number of packed 3D-printers increases for an increasing degree of uncertainty. In this section, we will investigate the effect of this degree of uncertainty on our two questions. We will model the degree of demand uncertainty by solving our instance sets for different values of the upper limit of demand  $D$ . In particular, we will study instance set N100D100S50 with  $D \in \{2^0, 2^1, 2^2, \dots, 2^{17}\}$ . Recall that, in our instance generator,  $D$  identifies the domain of the uniform distributions that generate the demand for all items. Consequently, an increasing  $D$  will coincide with a higher degree of uncertainty. Moreover, recall that in our instance generator, the knapsack weight and volume capacity, and the printing time are based on the total average demand. In this section, we are only interested in the effect of changing  $D$ . For that reason, we fix the following parameters  $W = V = 100,000$  and  $T = 4000$ . We report our results in Table 8.

From Table 8, we observe that both the number of 3D-printers as the percentage increase in reward increase up to a certain degree of uncertainty (i.e.,  $D = 2048$  and  $D = 256$ , respectively), and decrease afterwards. The increasing part confirms our intuition that more uncertainty coincides with packing more 3D-printers. The decreasing parts, however, is counterintuitive. This outcome, which is a result of our way to model uncertainty, can be explained as follows. For sufficiently large degrees of uncertainty, demand of items will exceed the knapsack weight and volume capacity for a majority of the scenarios.

$D$	3D-printers (#)		reward (%)	
	median	min-max (average)	median	min-max (average)
1	0	0-0 (0.0)	0.0	0-0 (0.0)
2	0	0-0 (0.0)	0.0	0-0 (0.0)
4	0	0-0 (0.0)	0.0	0-0 (0.0)
8	0	0-0 (0.0)	0.0	0-0 (0.0)
16	1	0-1 (0.6)	0.1	0-2 (0.3)
32	1	1-1 (1.0)	5.9	2-15 (6.4)
64	1	1-1 (1.0)	14.7	6-31 (15.6)
128	1	1-2 (1.5)	23.6	12-47 (24.3)
256	2	1-3 (2.1)	25.1	15-51 (26.5)
512	3	2-4 (2.6)	22.5	8-43 (23.1)
1024	3	2-5 (3.1)	18.0	5-60 (20.5)
2048	3	1-7 (3.3)	14.5	3-37 (15.4)
4096	3	0-7 (3.2)	10.5	0-51 (12.9)
8192	3	0-8 (3.1)	6.7	0-51 (9.4)
16384	2	0-8 (2.4)	3.0	0-67 (7.3)
32768	0	0-12 (1.5)	0.0	0-55 (4.8)
65536	0	0-11 (1.0)	0.0	0-31 (2.2)
131072	0	0-11 (1.1)	0.0	0-59 (3.5)

Table 8: An overview of the median, minimum, maximum and average number of both the packed 3D-printers and the percentage increase in reward, for varying values of  $D$ .

As a consequence, many of these scenarios can be recognized as if there was no restriction on demand for these items. This means that we can recognize our knapsack problem as a deterministic one. Consequently, there is no reason to pack 3D-printers to adapt for demand uncertainty. The only remaining reason to pack 3D-printers is the better packing efficiency of printing material. Hence, the number of 3D-printers packed will reduce significantly and this is exactly what we see in Table 8.

## 6 Conclusion

In this paper, we shed light on when to bring 3D-printers or not, which amount of printing material to pack and which items to bring physically, to a disaster area. For that, we introduce a new type of problem, which we call the two-stage stochastic 3D-printing knapsack problem. We present a deterministic equivalent of our problem, which allows us to solve the problem with standard solver techniques. Our numerical results illustrates how the quality of printed items, the weight and volume of 3D-printers,

the storage efficiency of printing material, the printing time, and the demand uncertainty effect the outcomes of our model. It turns out that for most situations packing a 3D-printer is beneficial. Only in extreme circumstances, where the quality of printed items is extremely low, the size of the 3D-printer is extremely large compared to the knapsack size, when there is no time to print the items, or when demand for items is low, taking no 3D-printers is the best option.

For future research, we identify four interesting research directions. The first one relates to the modelling assumptions of the 3D-printers. In our paper, we consider a sort of "ideal" 3D-printer: it can print all items with the same type of printing material and it can be set-up, and used easily in a disaster response mission. It would be of interest to see how robust our model is against these type of assumptions (e.g., what happens if we have different types of 3D-printers that can print specific types of items only?). As a second research direction, one could extend our TSS-3DKP by including dependencies between the items. With this modelling feature, it is possible to include composed items that consist of several printed or physical subitems. As a third research direction, one could extend our TSS-3DKP by introducing decision dependent uncertainty (cf. Goel and Grossmann (2006)). In such a setting, one could invest time to get more accurate demand information. Clearly, this creates a trade-off between reducing demand uncertainty and arriving late(r) at a disaster area. As a final research direction, it is of interest to see how our TSS-3DKP could be applied to other application domains (e.g., a military or aerospace setting), which each may have their own, unique, characteristics.

One of the strengths of our model is that sufficiently large instances can be solved by commercial solvers within reasonable computation time. However, this may no longer be the case for (the aforementioned) future research directions. Consequently, the need for (other) efficient solution methods becomes apparent. A candidate for this is logic-based Benders decomposition (Hooker and Ottosson, 2003), a generalization of Benders decomposition (Benders, 1962) that allows the second-stage problem to be any optimization problem rather than precisely a linear or nonlinear programming problem.

## References

- M. Ahmadi, A. Seifi, and B. Tootooni. A humanitarian logistics model for disaster relief operation considering network failure and standard relief time: a case study on san francisco district. *Transportation Research Part E: Logistics and Transportation Review*, 75:145–163, 2015.
- J. v. d. Akker, P. Bouman, J. Hoogeveen, and D. Tönissen. Decomposition approaches for recoverable robust optimization problems. *European Journal of Operational Research*, 251(3):739750, 2016.
- D. Alem, A. Clark, and A. Moreno. Stochastic network models for logistics planning in disaster relief. *European Journal of Operational Research*, 255(1):187–206, 2016.

- N. Altay and W. G. Green, I. OR/MS research in disaster operations management. *European Journal of Operational Research*, 175(1):475–493, 2006.
- B. Balcik and B. M. Beamon. Facility location in humanitarian relief. *International Journal of Logistics*, 11(2):101–121, 2008.
- G. Barbarosolu and Y. Arda. A two-stage stochastic programming framework for transportation planning in disaster response. *Journal of the Operational Research Society*, 55(1):43–53, 2004.
- V. Bayram and H. Yaman. Shelter location and evacuation route assignment under uncertainty: a benders decomposition approach. *Transportation Science*, 52(2):416–436, 2018.
- J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.
- F. Berkes. Understanding uncertainty and reducing vulnerability: lessons from resilience thinking. *Natural Hazards*, 41(2):283–295, 2007.
- J. R. Birge and F. Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.
- A. Bozorgi-Amiri, M. S. Jabalameli, M. Alinaghian, and M. Heydari. A modified particle swarm optimization for disaster relief logistics under uncertain environment. *The International Journal of Advanced Manufacturing Technology*, 60(1-4):357–371, 2012.
- M. Chang, Y. Tseng, and J. Chen. A scenario planning approach for the flood emergency logistics preparation problem under uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, 43(6):737–754, 2007.
- L. B. Davis, F. Samanlıoğlu, X. Qu, and S. Root. Inventory planning and coordination in disaster relief efforts. *International Journal of Production Economics*, 141(2):561–573, 2013.
- M. Dilley, R. Chen, U. Deichmann, A. Lerner-Lam, and M. Arnold. *Natural disaster hotspots: a global risk analysis*. The World Bank, 2005.
- L. Dong, D. Shi, and F. Zhang. 3D printing vs. traditional flexible technology: implications for manufacturing strategy. *Available at SSRN 2847731*, 2016.
- A. Döyen, N. Aras, and G. Barbarosoğlu. A two-echelon stochastic facility location model for humanitarian relief logistics. *Optimization Letters*, 6(6):1123–1145, 2012.
- Ö. Elçi and N. Noyan. A chance-constrained two-stage stochastic programming model for humanitarian relief network design. *Transportation Research Part B: Methodological*, 108:55–83, 2018.
- Y. Fan and C. Liu. Solving stochastic transportation network protection problems using the progressive hedging-based method. *Networks and Spatial Economics*, 10(2):193–208, 2010.

- G. Galindo and R. Batta. Review of recent developments in OR/MS research in disaster operations management. *European Journal of Operational Research*, 230(2):201–211, 2013.
- V. Goel and I. E. Grossmann. A class of stochastic programs with decision dependent uncertainty. *Mathematical Programming*, 108(2-3):355–394, 2006.
- C. Goulding. 3D printing of disaster relief tools and shelters, and R&D tax credits, October 2017. URL <https://3dprint.com/192352/3dp-disaster-relief-rd-credit/>.
- E. Grass and K. Fischer. Two-stage stochastic programming in disaster management: a literature survey. *Surveys in Operations Research and Management Science*, 21(2):85–100, 2016.
- N. Hall. 3D printers can change humanitarian aid, August 2016. URL <https://3dprintingindustry.com/news/3d-printers-can-change-humanitarian-aid-91658/>.
- J. N. Hooker and G. Ottosson. Logic-based benders decomposition. *Mathematical Programming*, 96(1):33–60, 2003.
- S. Hu, C. Han, Z. S. Dong, and L. Meng. A multi-stage stochastic programming model for relief distribution considering the state of road network. *Transportation Research Part B: Methodological*, 123:64–87, 2019.
- International Federation of Red Cross and Red Crescent Societies. World disasters report, December 2005. URL <https://www.ifrc.org/Global/Publications/disasters/WDR/69001-WDR2005-english-LR.pdf>.
- S. Khajavi, J. Partanen, and J. Holmström. Additive manufacturing in the spare parts supply chain. *Computers in Industry*, 65(1):50–63, 2014.
- P. Kolesar. A branch and bound algorithm for the knapsack problem. *Management Science*, 13(9):723–735, 1967.
- S. Kosuch. Approximability of the two-stage stochastic knapsack problem with discretely distributed weights. *Discrete Applied Mathematics*, 165:192–204, 2014.
- S. Kosuch and A. Lisser. On two-stage stochastic knapsack problems. *Discrete Applied Mathematics*, 159(16):1827–1841, 2011.
- A. Leiras, I. de Brito Jr, E. Q. Peres, T. R. Bertazzo, and H. T. Y. Yoshizaki. Literature review of humanitarian logistics research: trends and challenges. *Journal of Humanitarian Logistics and Supply Chain Management*, 2014.
- A. C. Li, L. Nozick, N. Xu, and R. Davidson. Shelter location and transportation planning under hurricane conditions. *Transportation Research Part E: Logistics and Transportation Review*, 48(4):715–729, 2012.

- L. Li, M. Jin, and L. Zhang. Sheltering network planning and management with a case in the gulf coast region. *International Journal of Production Economics*, 131(2):431–440, 2011.
- A. Lisser, R. Lopez, and X. Hu. Stochastic quadratic knapsack with recourse. *Electronic Notes in Discrete Mathematics*, 36:97–104, 2010.
- C. Liu, Y. Fan, and F. Ordóñez. A two-stage stochastic programming model for transportation network protection. *Computers & Operations Research*, 36(5):1582–1590, 2009.
- P. Liu, S. Huang, A. Mokasdar, H. Zhou, and L. Hou. The impact of additive manufacturing in the aircraft spare parts supply chain: supply chain operation reference (Scor) model based analysis. *Production Planning & Control*, 25(13-14):1169–1181, 2014.
- E. J. Lodree Jr, K. N. Ballard, and C. H. Song. Pre-positioning hurricane supplies in a commercial supply chain. *Socio-Economic Planning Sciences*, 46(4):291–305, 2012.
- S. Martello and P. Toth. An exact algorithm for large unbounded knapsack problems. *Operations Research Letters*, 9(1):15–20, 1990.
- N. Noyan, B. Balcik, and S. Atakan. A stochastic optimization model for designing last mile relief networks. *Transportation Science*, 50(3):1092–1113, 2016.
- L. Özdamar, E. Ekinci, and B. Küçükyazici. Emergency logistics planning in natural disasters. *Annals of Operations Research*, 129(1-4):217–245, 2004.
- J. A. Paul and M. Zhang. Supply location and transportation planning for hurricanes: a two-stage stochastic programming framework. *European Journal of Operational Research*, 274(1):108–125, 2019.
- S. Peeta, F. S. Salman, D. Gunneç, and K. Viswanath. Pre-disaster investment decisions for strengthening a highway network. *Computers & Operations Research*, 37(10):1708–1719, 2010.
- M. Perry. Natural disaster management planning: a study of logistics managers responding to the tsunami. *International Journal of Physical Distribution & Logistics Management*, 37(5):409–433, 2007.
- D. Pisinger. Where are the hard knapsack problems? *Computers & Operations Research*, 32(9):2271–2284, 2005.
- C. G. Rawls and M. A. Turnquist. Pre-positioning of emergency supplies for disaster response. *Transportation Research Part B: Methodological*, 44(4):521–534, 2010.
- M. Rezaei-Malek, R. Tavakkoli-Moghaddam, N. Cheikhrouhou, and A. Taheri-Moghaddam. An approximation approach to a trade-off among efficiency, efficacy, and balance for relief pre-positioning in disaster management. *Transportation Research Part E: Logistics and Transportation Review*, 93:485–509, 2016.



- O. Rodríguez-Espíndola and A. Beltagui. Can 3D printing address operations challenges in disaster management? In *25th Annual Euroma Conference*, 2018.
- J. Salmerón and A. Apte. Stochastic optimization for natural disaster asset prepositioning. *Production and Operations Management*, 19(5):561–574, 2010.
- E. Sanci and M. S. Daskin. Integrating location and network restoration decisions in relief networks under uncertainty. *European Journal of Operational Research*, 279(2):335–350, 2019.
- S. Saunders. Field ready 3D printing much-needed medical supplies for a nepal health clinic, March 2017. URL <https://3dprint.com/166818/3d-printed-otoscope-nepal-clinic/>.
- B. Savonen, T. Mahan, M. Curtis, J. Schreier, J. Gershenson, and J. Pearce. Development of a resilient 3-D printer for humanitarian crisis response. *Technologies*, 6(1):30, 2018.
- B. Song, Y. Li, Y. Chen, F. Yao, and Y. Chen. A repair-based approach for stochastic quadratic multiple knapsack problem. *Knowledge-Based Systems*, 145:145–155, 2018.
- J. Song and Y. Zhang. Stock or print? Impact of 3-D printing on spare parts logistics. *Management Science (forthcoming)*, 2019.
- P. Tatham, J. Loy, and U. Peretti. Three dimensional printing—a key tool for the humanitarian logistician? *Journal of Humanitarian Logistics and Supply Chain Management*, 5(2):188–208, 2015.
- S. Tofighi, S. A. Torabi, and S. A. Mansouri. Humanitarian logistics network design under mixed uncertainty. *European Journal of Operational Research*, 250(1):239–250, 2016.
- D. D. Tönissen, J. M. van den Akker, and J. A. Hoogeveen. Column generation strategies and decomposition approaches for the two-stage stochastic multiple knapsack problem. *Computers & Operations Research*, 83:125–139, 2017.
- F. Tricoire, A. Graf, and W. J. Gutjahr. The bi-objective stochastic covering tour problem. *Computers & Operations Research*, 39(7):1582–1592, 2012.
- L. Van Wassenhove. Humanitarian aid logistics: supply chain management in high gear. *Journal of the Operational Research Society*, 57(5):475–489, 2006.
- A. Verma and G. M. Gaukler. Pre-positioning disaster response facilities at safe locations: an evaluation of deterministic and stochastic modeling approaches. *Computers & Operations Research*, 62:197–209, 2015.
- B. Westerweel, R. Basten, and G. van Houtum. Traditional or additive manufacturing? Assessing component design options through lifecycle cost analysis. *European Journal of Operational Research*, 270(2):570–585, 2018.

M. R. Zolfaghari and E. Peyghaleh. Implementation of equity in resource allocation for regional earthquake risk mitigation using two-stage stochastic programming. *Risk Analysis*, 35(3):434–458, 2015.