CrossMark

An accelerated L-shaped method for solving two-stage stochastic programs in disaster management

Emilia Grass[1] · Kathrin Fischer[1] · Antonia Rams[1]

Abstract Mitigating the disastrous effects of natural disasters by performing preparation activities is one of the main purposes of relief organizations. However, the high degree of uncertainty associated with disasters impedes the work of aid agencies considerably. In this regard, two-stage stochastic programs are often used in the relevant literature to support decision making in these situations. An accelerated L-shaped method is proposed in this work, which solves realistic large-scale two-stage stochastic problems within a reasonable time-frame, allowing relief organizations to react to short-term forecasts, as e.g. available in case of hurricanes or floods. In particular, computation times needed for solving the resulting sub-problems via a specialized interior-point method are significantly reduced by exploiting the specific structure of second-stage constraints. To show the superiority of this approach with respect to solution times, a realistic large-scale case study is developed for America's hurricane-prone south-east coast. The accelerated L-shaped method outperforms the standard L-shaped method significantly whereas a commercial solver failed to solve the case study within an acceptable time-frame.

1 Introduction

Disasters represent a serious problem for affected communities not only because of their destructive force but also because of their unpredictable nature. To be able to support decision makers like relief organizations, two-stage stochastic programs are often proposed in the

✉ Emilia Grass
e.grass@tuhh.de

Kathrin Fischer
kathrin.fischer@tuhh.de

[1] Institute for Operations Research and Information Systems, Hamburg University of Technology, Am Schwarzenberg-Campus 4, 21073 Hamburg, Germany

disaster management related literature (Grass and Fischer 2016). This modeling approach is popular since decisions made before an uncertain event like a disaster, so-called *first-stage decisions*, can be improved in terms of, e.g., demand satisfaction by taking possible future developments and interim *second-stage decisions* into account. For instance, humanitarian organizations often have to purchase and store relief items, e.g. water or medicine, even before a disaster strikes. These first-stage decisions have to be made before information concerning demand as well as with respect to the actual network conditions becomes available. In the aftermath of a disaster, decisions regarding, e.g., the distribution of previously stored relief items to affected or injured people have to be made which belong to the second stage. Note that roads could become impassable and storage decisions have to take such possible circumstances into account in advance.

In order to make the application of the two-stage stochastic models valuable for humanitarian organizations, efficient solution methods are required. However, the relevant literature focuses more on model presentation than on proposing suitable solution algorithms. This constitutes a major obstacle to the use of two-stage stochastic programs in practice. Especially solution times of commercial solvers like CPLEX or Gurobi are much too long for realistic large-scale problems (Zheng et al. 2013; Pay and Song 2017).

A practical example where computation time is crucial is the forecast of hurricanes. In general, hurricanes can be predicted three to five days in advance. Based on these forecasts aid agencies often use existing buildings like hospitals and gyms as temporary warehouses or shelters. However, weather conditions can change the path and strength of a hurricane within few hours as it happened in 2005. Hurricane Wilma strengthened from the lowest category 1 (113 km/h) to the highest 5 (295 km/h) just within 24 h. To ensure fast response also in such cases, efficient solution methods are essential. Therefore, the aim of this paper is to propose an algorithm which solves a whole class of two-stage stochastic programs in disaster management within a reasonable time-frame and with a desired accuracy level. A representative of this class is the model proposed by Rawls and Turnquist (2010) for locating and pre-positioning relief items in preparation for a disaster. Such two-stage programs are commonly solved by the *L-Shaped method* (Slyke and Wets 1969) which is also known as the *stochastic Benders decomposition*. Since realistic problems are constantly increasing in size and complexity, decomposition methods are becoming increasingly popular as shown in the recent survey of Rahmaniani et al. (2017). The main idea of decomposition methods is to split up the original problem into smaller sub-problems which are easier to solve. In case of the L-shaped method, the two-stage structure is exploited by separating first and second stage. For the two-stage stochastic programs considered here, the resulting second-stage problems are often very large, requiring an efficient solution algorithm themselves. To solve these sub-problems efficiently, a *specialized interior-point method* (SIPM) is proposed in this work which exploits the specific structure of second-stage constraints in a new way leading to significant computational time savings.

Interior-point methods are often preferred to solve sub-problems within the L-shaped method in parallel, i.e. simultaneously, see e.g. Nielsen and Zenios (1997) and Fragnière et al. (2000). As the number of iterations remains nearly unaffected by an increased problem size, interior-point methods are particularly suitable for very large linear sub-problems (Gondzio and Sarkissian 2003). In addition, computations within interior-point methods can be executed in parallel, accelerating the solution process further. For instance, De Silva and Abramson (1998) show that their parallel interior-point method outperforms the simplex

algorithm despite of its better warm-starting capabilities.[1] Interior-point methods are also applied by Naoum-Sawaya and Elhedhli (2013) where Benders decomposition is embedded in a Branch-and-Bound algorithm for the multicommodity capacitated fixed charge network design problem and the capacitated facility location problem. The latter is also investigated in Castro et al. (2016) for multiple periods. The approach proposed by these authors is related to the solution method presented in this paper. However, the problem considered there is not a two-stage stochastic program and hence, does not posses the specific structure of second-stage constraints which is discussed in this work.

In the humanitarian literature examples or case studies of only moderate size are usually chosen to ensure the solvability of the proposed model by means of commercial solvers or to evaluate the quality of the applied heuristic. In addition, the lack of adequate data impedes the design of genuine examples. However, large scale problems of pre-positioning relief items become more and more relevant for practical applications and more realistic examples have to be considered to validate methodological developments in realistic settings.

Therefore, a large-scale case study on pre-positioning relief items in the Gulf of Mexico area is developed in this work. Decisions have to be made concerning the location of facilities and the storage quantities of relief items prior to the beginning of the American hurricane season. The case study is based on the one presented by Rawls and Turnquist (2010) which is extended in different ways to make it more realistic. For instance, information from various sources like the *National Oceanic and Atmospheric Administration* (NOAA) and the *International Federation of Red Cross and Red Crescent Societies* (IFRC) is integrated. The resulting large-scale case study is solved by the accelerated L-shaped method showing significant advantages compared to commercial solvers and heuristics.

The main contribution of this work is to develop the specialized interior-point method (SIMP) which exploits the specific structure of second-stage constraints inherent in two-stage stochastic programs. Using this algorithm to solve the sub-problems within the standard L-shaped method leads to its accelerated version. Numerical tests show that the newly proposed accelerated L-shaped method can provide fast and near-optimal solutions to realistic large-scale problems. Obtaining solutions fast is especially useful for humanitarian organizations as they have to react as quickly as possible to fast-approaching disasters.

This paper is organized as follows: The two-stage stochastic model for pre-positioning relief items (Rawls and Turnquist 2010) is described in Sect. 2 and used as a reference model for two-stage stochastic programs in disaster management. After highlighting the important second-stage constraint structure of the model, the accelerated L-shaped method is proposed in Sect. 3. For this purpose, the classical L-shaped method is described first in Sect. 3.1. The specialized interior-point method (SIPM) is introduced in Sect. 3.2, leading to the accelerated L-shaped method. This algorithm is then applied to solve the large-scale case study, presented in Sect. 4.1. The numerical results discussed in Sect. 4.2 illustrate the benefits of the newly proposed solution method. Finally, conclusions and future research directions are given.

---

[1] Warm-starting is to use the solution to the previous problem as a starting point for the next problem. Especially simplex algorithms benefit from such warm-start points and can find the next solution after only few iterations. However, developing efficient warm-start strategies for interior-point methods is an active research area nowadays, see e.g. Cay et al. (2017).

2 A two-stage stochastic program for pre-positioning relief items

Pre-positioning relief items especially nearby disaster-prone areas can be decisive in mitigating negative impacts of catastrophic events (Turkeš et al. 2017). In addition, storing first-aid items in advance may lead to purchase cost savings.

One of the most influential models in this regard is the one proposed by Rawls and Turnquist in 2010. In addition, it is a classical two-stage stochastic program and therefore used as the reference model in the following. In this model, location and storage decisions are made on the first stage, i.e. prior to a disaster. In the aftermath of a disaster, second-stage decisions are made concerning, e.g. the distribution of these items to areas where they are needed. The network considered in this model contains nodes to represent potential locations where storage facilities can be opened and demand for aid supplies can arise. In case of a disaster, pre-positioned items can be transported to these who are in need on roads, represented as arcs in the respective network. Demand as well as possible damages to storage facilities and roads depend on the location and the magnitude of a disaster which is not known a priori. Such uncertainties are modeled by defining appropriate scenarios with corresponding probabilities of occurrence.

In order to make the description of the newly proposed solution method more self-contained, the model and its notation are given below (Rawls and Turnquist 2010).

Sets:

I: set of demand nodes
K: set of relief items
L: set of facility size categories
N: set of nodes, $I \subseteq N$
R: set of arcs in the network
S: set of possible scenarios

Scenario-independent parameters:

b^k: space requirement per unit for relief item $k \in K$
$c_{(i, j)}^k$: costs for transporting one unit of relief item $k \in K$ on link $(i, j) \in R$
F_l: costs for opening facility of size $l \in L$
h^k: penalty cost per unit for unused relief item $k \in K$, e.g. holding or spoilage costs
M_l: capacity of a facility of size $l \in L$
p^k: penalty cost per unit for unsatisfied demand of relief item $k \in K$
q^k: acquisition cost per unit for relief item $k \in K$
u^k: required transport capacity per unit of relief item $k \in K$

Scenario-dependent parameters:

P_s: probability of scenario $s \in S$
ρ_i^{ks}: proportion of relief item $k \in K$ at node $i \in I$ in scenario $s \in S$ which is still usable
$U_{(i, j)}^s$: maximal available arc capacity of link $(i, j) \in R$ in scenario $s \in S$
v_i^{ks}: forecasted demand for relief item $k \in K$ at node $i \in I$ in scenario $s \in S$

Scenario-independent decision variables (first stage):

r_i^k: pre-positioned quantity of relief item $k \in K$ at node $i \in I$
y_{il}: is 1, if facility of size $l \in L$ is opened at node $i \in I$; 0 otherwise

Scenario-dependent decision variables (second stage):

w_i^{ks}: unsatisfied demand for relief item $k \in K$ at node $i \in I$ in scenario $s \in S$

$x_{(i,j)}^{ks}$: quantity of relief item $k \in K$ transported from i to j in scenario $s \in S$

z_i^{ks}: unused quantity of relief item $k \in K$ at node $i \in I$ in scenario $s \in S$

The model formulation is as follows:

$$\min \sum_{i \in I} \sum_{l \in L} F_l y_{il} + \sum_{k \in K} \sum_{i \in I} q^k r_i^k$$
$$+ \sum_{s \in S} P_s \left[\sum_{(i,j) \in R} \sum_{k \in K} c_{(i,j)}^k x_{(i,j)}^{ks} + \sum_{i \in I} \sum_{k \in K} (h^k z_i^{ks} + p^k w_i^{ks}) \right] \tag{1}$$

s.t.

$$\sum_{l \in L} y_{il} \leq 1, \ \forall i \in I \tag{2}$$

$$\sum_{k \in K} b^k r_i^k \leq \sum_{l \in L} M_l y_{il}, \ \forall i \in I \tag{3}$$

$$\sum_{k \in K} u^k x_{(i,j)}^{ks} \leq U_{(i,j)}^s, \ \forall (i,j) \in R, s \in S \tag{4}$$

$$\sum_{j \neq i \in N} x_{(j,i)}^{ks} + \rho_i^{ks} r_i^k - z_i^{ks} = \sum_{j \neq i \in N} x_{(i,j)}^{ks} + v_i^{ks} - w_i^{ks}, \ \forall i \in N, k \in K, s \in S \tag{5}$$

$$y_{il} \in \{0, 1\}, \ \forall i \in I, l \in L, \tag{6}$$

$$r_i^k \geq 0, \ \forall i \in I, k \in K, \ x_{(i,j)}^{ks} \geq 0, \ \forall (i,j) \in R, k \in K, s \in S, \tag{7}$$

$$z_i^{ks}, w_i^{ks} \geq 0, \ \forall i \in I, k \in K, s \in S \tag{8}$$

The objective (1) is to minimize costs which are divided into first- and second-stage costs. Opening facilities (y_{il}) and pre-positioning relief items (r_i^k) causes costs on the first stage, i.e. before the disaster occurs. Second-stage costs contain transportation costs $c_{(i,j)}^k$ and penalties for unused items h^k and unsatisfied demand p^k, respectively. These costs are weighted by the probability of occurrence P_s for each scenario. Although these second-stage costs are not incurred de facto,[2] they are useful for supporting decisions on the first stage. Neglecting possible future developments of disasters when deciding where and how much to pre-position in advance (first stage), can have devastating consequences in terms of demand satisfaction later on. Therefore, possible second-stage actions are taken into account already before the catastrophe. In the aftermath of a disaster, actual demand and network conditions become known such that an updated distribution plan can be used in practice to supply affected people.

The first constraint group (2) ensures that not more than one facility is opened at node $i \in I$. An open facility has a specific capacity M_l, according to the chosen size, which cannot be exceeded by the pre-positioned amount of relief items, see (3). Both constraint groups (2) and (3) belong to the first stage and are therefore scenario independent. However, facility and pre-positioning decisions directly influence second-stage decisions concerning, e.g. transport quantities. According to (4), it is not allowed to transport more items along an arc than the available arc capacity $U_{(i,j)}^s$. In case of road destructions, where $U_{(i,j)}^s = 0$ in (4) for a specific scenario, delivering relief items is impossible. The possibility of such road network conditions and the related risk has to

---

[2] Note that it is very unlikely that a scenario will occur exactly as predefined. Moreover, penalty costs for unsatisfied demand are only fictitious and used to minimize unmet demand.

be considered when location decisions are made. Moreover, facilities can be damaged or even destroyed such that the amount of usable items $\rho_i^k r_i^k$ can be reduced significantly after a disaster. This has a direct implication for the amount of unsatisfied demand w_i^{ks} in (5). These flow conservation constraints balance, as usual, the incoming and outgoing flows. In particular, demand v_i^{ks} at node i is satisfied either by pre-positioned and undamaged items $\rho_i^k r_i^k$ or by delivered quantities $x_{(j,i)}^{ks}$. In case of a surplus, relief items remain unused, i.e. $z_i^{ks} > 0$. Constraints (5) are also referred to as *linking constraints* since the second stage is linked to the first stage through the use of first-stage variables r_i^k.

As stated above, this model can be seen as a representative for two-stage stochastic programs in disaster management, especially in preparation for a disaster. For a unified presentation of the model and the corresponding solution method, a more general notation[3] of (1)–(8) is used in the following:

$$\min \; a^T \chi + \sum_{s \in S} P_s f^T \gamma_s \tag{9}$$

$$\text{s.t. } G\chi \leq d \tag{10}$$

$$T_s \chi + W \gamma_s = t_s, \; \forall s \in S \tag{11}$$

$$\chi \geq 0, \tag{12}$$

$$\gamma_s \geq 0, \; \forall s \in S. \tag{13}$$

For the Rawls and Turnquist (2010) model first-stage vectors and matrix are denoted by $a \in \mathbb{R}^{|I|(|K|+|L|)}$, $\chi = (y, r) \in \{0, 1\}^{|I||L|} \times \mathbb{R}_+^{|K||I|}$, $d \in \mathbb{R}^{2|I|}$ and $G \in \mathbb{R}^{2|I| \times |I|(|K|+|L|)}$, respectively. Constraints (11) and (13) belong to the second stage with $f, \gamma_s \in \mathbb{R}_+^{|R||K|+2|N||K|+|R|}$, right hand side vector $t_s \in \mathbb{R}^{|R|+|N||K|}$ and so-called *technology matrix* $T_s \in \mathbb{R}^{|R|+|N||K| \times |I|(|K|+|L|)}$ for each scenario s. The most important matrix with regard to the design of an efficient solution method is the *recourse matrix* $W \in \mathbb{R}^{|R|+|N||K| \times |R||K|+2|N||K|+|R|}$. If W is independent of s, as it is the case here, the two-stage program is said to have *fixed* recourse. For the underlying problem (1)–(8), the recourse is even *complete*, meaning that any first-stage decision χ leads to a feasible second-stage decision γ_s. This is due to the presence of variables w_i^{ks} and z_i^{ks} in (5). For instance, even if no facilities are opened and no relief items are stored, the corresponding second-stage solution is still feasible. In this case, demand remains completely unsatisfied such that the total costs in (9) contain only penalty costs for unsatisfied demand.

In order to get a better idea of the respective matrix structures, a small example is given below:

Example:

For $|S| = 3$, $|N| = |I| = 3$, $|K| = 2$ and $|L| = 2$, the first-stage vector χ and matrix G in (10) look as follows:

$$\chi = \begin{pmatrix} y_{11} \\ y_{21} \\ y_{31} \\ y_{12} \\ y_{22} \\ y_{32} \\ r_1^1 \\ r_2^1 \\ r_3^1 \\ r_1^2 \\ r_2^2 \\ r_3^2 \end{pmatrix}, G = \begin{pmatrix} 1 & & & 1 & & & & & \\ & 1 & & & 1 & & & & \\ & & 1 & & & 1 & & & \\ -M_1 & & & -M_2 & & & b_1 & b_2 & \\ & -M_1 & & & -M_2 & & & b_1 & b_2 \\ & & -M_1 & & & -M_2 & & b_1 & b_2 \end{pmatrix}.$$

The first three rows in G belong to constraints (2) and the last three rows to (3).

In case of the Rawls' and Turnquist's model, the second-stage decision vector γ_s for each scenario s is:

$$\gamma_s = \left(x_{(1,1)}^{1s}\ x_{(1,1)}^{2s}\ g_{(1,1)}\ x_{(1,2)}^{1s}\ x_{(1,2)}^{2s}\ g_{(1,2)}\ \cdots\ x_{(3,3)}^{2s}\ g_{(3,3)}\ z_1^1\ z_2^1\ z_3^1\ z_1^2\ z_2^2\ z_3^2\ w_1^1\ w_2^1\ w_3^1\ w_1^2\ w_2^2\ w_3^2 \right),$$

where slack variables $g_{(1,1)}, \ldots, g_{|R|}$ for each arc $(i, j) \in R$ are added to γ_s to convert inequality constraints (4) into equalities. The technology matrix T_s contains only entries ρ_i^{ks} which are connected to the first-stage pre-positioning variables r_i^k in χ:

$$T_s = \begin{pmatrix} 0 & & & & & \\ & \ddots & & & & \\ & & 0 & & & \\ & & & \rho_1^{1s} & & \\ & & & & \ddots & \\ & & & & & \rho_3^{2s} \end{pmatrix}.$$

As will be shown later, the formulation of the recourse matrix W is decisive for an efficient solution algorithm in terms of fast computation times. The specific form is:

$$W = \begin{pmatrix} u & & & \\ & u & & \\ & & \ddots & \\ & & & u \\ H_1 & H_2 & \cdots & H_{|R|}\ Q \end{pmatrix}, \tag{14}$$

with $u = [u^1 \cdots u^K\ 1] \in \mathbb{R}^{1 \times (|K|+1)}$ and $Q = \begin{bmatrix} -1 & & 1 & \\ & \ddots & & \ddots \\ & & -1 & & 1 \end{bmatrix} \in \mathbb{R}^{|N||K| \times 2|N||K|}.$

The recourse matrix W is a special *lower block triangular matrix*, where all entries are zero except on the main diagonal and the last block row. This structure is favorable for the solution method as will be shown in the next section.

Note that 1 in u corresponds to the slack variables $g_{(1,1)}, \ldots, g_{|R|}$ as described above.

The last block row of W belongs to the flow conservation constraints (5) and contains block matrices $H_i \in \mathbb{R}^{|N||K| \times (|K|+1)}$, $i = 1, \ldots, |R|$ such that $H = [H_1\ H_2\ \cdots H_{|R|}] \in \mathbb{R}^{|N||K| \times |R|(|K|+1)}$. For the example given above, H is as follows:

$$H = \begin{pmatrix} -1 & -1 & 1 & & & 1 & & \\ 1 & & -1 & -1 & & & 1 & \\ & 1 & & 1 & -1 & -1 & & \\ -1 & -1 & 1 & & & 1 & & \\ 1 & & -1 & -1 & & & 1 & \\ & 1 & & 1 & -1 & -1 & & \end{pmatrix}$$

For instance, the first row in H indicates if relief items of type 1 are supplied to node 1 or transported away from node 1, i.e. $-x^{1s}_{(1,2)} - x^{1s}_{(1,3)} + x^{1s}_{(2,1)} + x^{1s}_{(3,1)}$.

3 The accelerated L-shaped method

3.1 The L-shaped method

The two-stage structure in (9)–(13) is perfectly suited to split up the original problem into smaller sub-problems as it is done by the L-shaped method. Roughly speaking the idea is to solve the first- and second-stage problems separately and then to connect them through a set of *optimality cuts*; see Birge and Louveaux (2011) for a detailed description. The corresponding pseudo-code is given in Fig. 1.

In the initialization step $\iota = 0$, the so-called *master problem* containing only first-stage constraints (10) and (12) is solved without taking $\sum_{s \in S} P_s f^T \gamma_s$ in (9) into account. The corresponding first-stage solution $\overline{\chi}^0$ is included as an input parameter into each of the S sub-problems in (16), see Fig. 1.

The dual of (16) is

$$\max(t_s - T_s \overline{\chi}^\iota)^T \pi_s \quad \text{s.t.} \quad W^T \pi_s \leq f,$$

where π_s is the vector of dual variables and is also referred to as the vector of simplex multipliers for each scenario s.[4] Using slack variables z_s, an equality constrained maximization sub-problem is obtained:

$$\max(t_s - T_s \overline{\chi}^\iota)^T \pi_s \quad \text{s.t.} \quad W^T \pi_s + z_s = f \quad z_s \geq 0. \tag{17}$$

For a given first-stage solution $\overline{\chi}^0$ and the vector of simplex multipliers $\overline{\pi}^0_s$, the objective function value of the second-stage sub-problem $\vartheta^0 = \sum_{s \in S} P_s (t_s - T_s \overline{\chi}^0)^T \overline{\pi}^0_s$ is determined in the second step of the algorithm. If ϑ^0 exceeds the current θ^0, the counter of iterations ι is increased by one and a first optimality cut

$$\theta \geq \sum_{s \in S} P_s (t_s - T_s \chi)^T \overline{\pi}^0_s$$

is inserted to the master problem, see Step 3 in Fig. 1.

For $\iota = 1$, the master problem in Step 1 of the algorithm is solved resulting in an updated first-stage solution $\overline{\chi}^\iota$. Afterwards, $\overline{\chi}^\iota$ is used for solving the sub-problems, leading to an additional optimality cut (15).[5] Note that χ in (15) again is a vector of decision variables

---

[4] Note that simplex multipliers are also known as shadow prices which are available, e.g. in the final tableau of the simplex algorithm (Diwekar 2008, p. 23). However, the dual sub-problem will be important for the primal–dual interior-point method described in the next section.

[5] For two-stage stochastic models where no complete recourse is given, so-called *feasibility cuts* have to be included to the master problem in addition to the optimality cuts (Birge and Louveaux 2011, Ch 5).
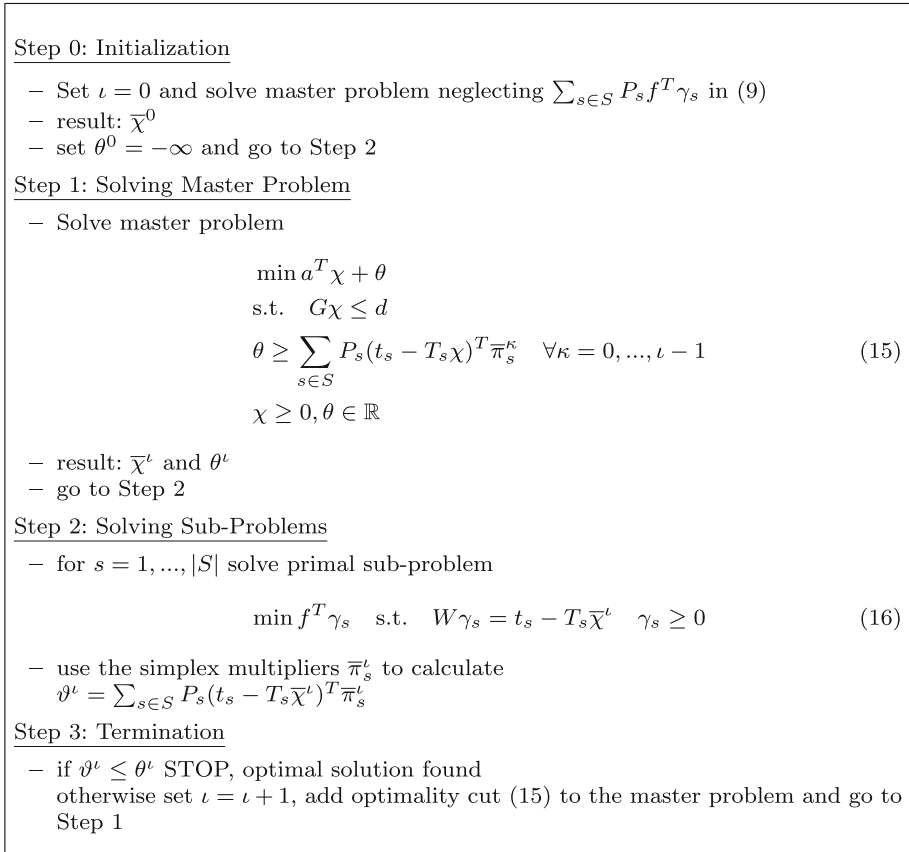
Step 0: Initialization

  – Set $\iota = 0$ and solve master problem neglecting $\sum_{s \in S} P_s f^T \gamma_s$ in (9)
  – result: $\overline{\chi}^0$
  – set $\theta^0 = -\infty$ and go to Step 2

Step 1: Solving Master Problem

  – Solve master problem

$$\min a^T \chi + \theta$$
$$\text{s.t.} \quad G\chi \leq d$$
$$\theta \geq \sum_{s \in S} P_s (t_s - T_s \chi)^T \overline{\pi}_s^\kappa \quad \forall \kappa = 0, ..., \iota - 1 \qquad (15)$$
$$\chi \geq 0, \theta \in \mathbb{R}$$

  – result: $\overline{\chi}^\iota$ and θ^ι
  – go to Step 2

Step 2: Solving Sub-Problems

  – for $s = 1, ..., |S|$ solve primal sub-problem

$$\min f^T \gamma_s \quad \text{s.t.} \quad W\gamma_s = t_s - T_s \overline{\chi}^\iota \quad \gamma_s \geq 0 \qquad (16)$$

  – use the simplex multipliers $\overline{\pi}_s^\iota$ to calculate
    $\vartheta^\iota = \sum_{s \in S} P_s (t_s - T_s \overline{\chi}^\iota)^T \overline{\pi}_s^\iota$

Step 3: Termination

  – if $\vartheta^\iota \leq \theta^\iota$ STOP, optimal solution found
    otherwise set $\iota = \iota + 1$, add optimality cut (15) to the master problem and go to
    Step 1

Fig. 1 The L-shaped method [based on Birge and Louveaux (2011, Ch. 5)]

whereas $\overline{\pi}_s^\kappa$ are the simplex multipliers found in previous iterations $\kappa = 0, \ldots, \iota - 1$. As long as the solution is not optimal, i.e. $\theta^\iota < \vartheta^\iota$ for the current $\overline{\chi}^\iota$, this process has to be repeated. The main challenge is the efficient solution of the master problem in Step 1 and of the sub-problems in Step 2 of the L-shaped method. Often, integer or binary decision variables occur only on the first stage in the relevant literature, e.g. decisions concerning locations as in Rawls and Turnquist (2010), such that second-stage sub-problems are 'simple' LP problems. However, these sub-problems can be very large in comparison to the master problem. For instance, assuming the entire American southeast coast with its 900 counties has to be prepared for the upcoming hurricane season, i.e. $|N| = |I| = 900$, such that the network contains potentially 810,000 shortest paths between each pair of nodes. While the actual network will likely be sparser, $|R| = 810,000$ arcs may exist in the worst case. If the number of items and possible facility sizes are set to $|K| = |L| = 3$, the master problem contains $|I|(|K| + |L|) = 5400$ decision variables (of which 2700 are binary) and $2|I| = 1800$ constraints. In contrast, each
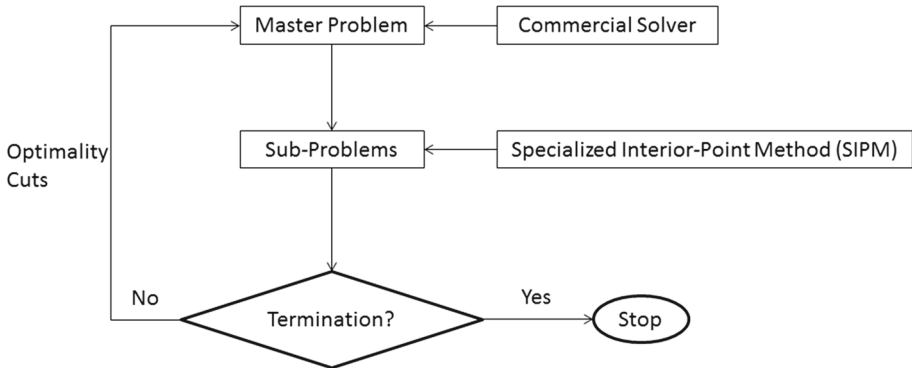
Fig. 2 The accelerated L-shaped method

sub-problem has $|R||K|+2|N||K|+|R| = 3{,}245{,}400$ variables and $|R|+|N||K| = 812{,}700$ constraints. Note that the respective sub-problem has to be solved S times, i.e. for each scenario, such that solving these problems efficiently is crucial for the overall performance of the L-shaped method. Usually, the linear sub-problems are solved via the simplex algorithm (Rahmaniani et al. 2017). However, as the problem size increases the computational effort of such exact methods becomes prohibitive. Therefore, here an iterative algorithm, namely the specialized interior-point method SIPM, is applied for solving the sub-problems as shown in the flow diagram of the accelerated L-shaped procedure in Fig. 2. In contrast, the master problem is solved via a commercial solver like CPLEX or Gurobi. The motivation behind this approach is that the master problem is of moderate size, e.g. 5400 variables as in the example above, and therefore does not present a significant hurdle for today's optimization software.

Note that the only difference between the standard L-shaped method and its accelerated counterpart in Fig. 2 is the use of SIPM for the sub-problems.

3.2 A primal–dual interior-point method

In the following, it will be shown how the specific structure of W in (14) is exploited to improve a primal–dual interior-point method for solving sub-problems (16) and (17).[6]

Especially for large scale problems, interior-point methods often outperform the simplex algorithm (Ferris et al. 2007, p. 212). Instead of evaluating vertices of the feasible region until the optimal solution is found, interior-point methods reach the optimal vertex from the interior. As a result, the algorithm can be terminated when the solution is close enough to the boundary, resulting in an approximate solution. This is particularly beneficial for the L-shaped method, since exactness is not required for the construction of valid optimality cuts (15) (Rahmaniani et al. 2017).

---

[6] The approach is presented for one sub-problem such that the subscript s is omitted for now.

From the theory of linear programming it is well-known that a solution of a linear problem as presented in (16) is optimal if the following conditions are satisfied (Ferris et al. 2007, p. 108):

$$W^T \pi + z = f \tag{18}$$

$$W \gamma = \tilde{t} \tag{19}$$

$$\Gamma Z e = 0 \tag{20}$$

$$\gamma, z \geq 0 \tag{21}$$

with $\tilde{t} = t - T \overline{\chi}^{\iota}$, $e = (1, \ldots, 1)^T$, $Z = diag(z_1, \ldots, z_n)$ and $\Gamma = diag(\gamma_1, \ldots, \gamma_n)$. Equation (19) states the primal and (18) the dual feasibility. Note that (20) represents the standard complementarity conditions, i.e. at optimum either γ_j or z_j has to be zero (for $j = 1, \ldots, n$). In this case the *primal–dual solution* vector is (γ^*, π^*, z^*) if γ^* is the solution of the primal problem (16) and (π^*, z^*) the solution of the dual problem (17). The main idea of interior-point methods is to perturb the complementarity condition (20) by introducing the barrier parameter $\mu > 0$:

$$\Gamma Z e \geq \mu e. \tag{22}$$

During the solution process μ is successively reduced such that the optimal solution is approached from the interior of the feasible set. In each iteration of interior-point methods, the system (18)–(21) has to be solved, where (20) is replaced by (22). Due to the nonlinearity of (22) the Newton method[7] is applied to find the direction $(\triangle \gamma, \triangle \pi, \triangle z)$:

$$\begin{bmatrix} 0 & W^T & I \\ W & 0 & 0 \\ Z & 0 & \Gamma \end{bmatrix} \begin{bmatrix} \triangle \gamma \\ \triangle \pi \\ \triangle z \end{bmatrix} = \begin{bmatrix} f - W^T \pi - z \\ \tilde{t} - W \gamma \\ -\Gamma Z e + \mu e \end{bmatrix}, \tag{23}$$

where I is an identity matrix of appropriate size. The right hand side of (23) represents the residuals of the corresponding constraints in (18), (19) and (22), respectively. In Fig. 3, the pseudo code of a general primal–dual interior-point method is given (Ferris et al. 2007).

After choosing an interior starting point (γ^0, π^0, z^0), i.e. γ^0, z^0 are strictly positive, the search direction has to be determined in which the next step is taken. However, the new solution should not be outside the feasible region. This is ensured through a suitable choice of the step length parameter α, see Step 2 in Fig. 3.[8] If a predefined tolerance level is not reached in the last step of the algorithm (Step 3), the barrier parameter $\mu^{\mathbf{k}}$ is reduced and a new search direction is calculated.[9]

---

[7] See Ferris et al. (2007) for a general description of the Newton method for interior-point algorithms.

[8] An approach for setting the step length α can be found in Ferris et al. (2007).

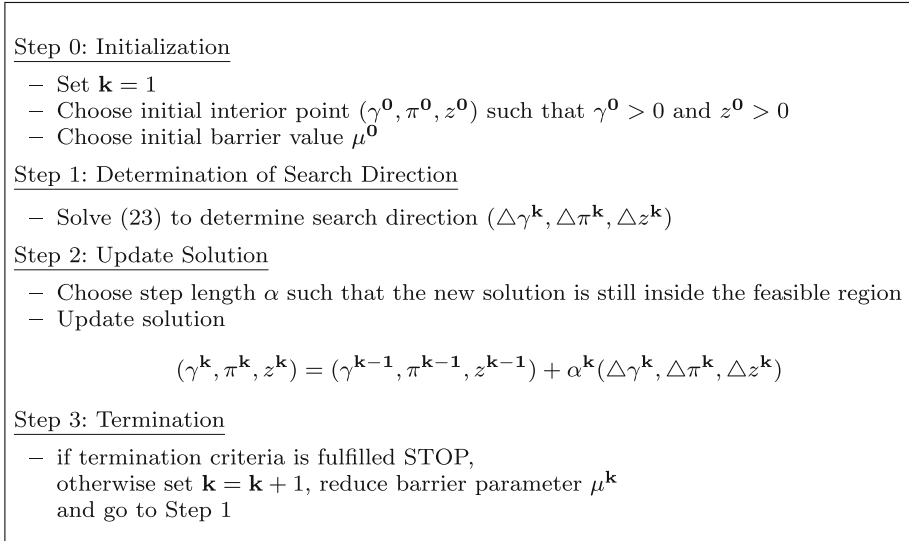[9] A detailed description of the algorithm and how to reduce $\mu^{\mathbf{k}}$, can be found in Ferris et al. (2007) or Wright (1997).

Step 0: Initialization

- Set $\mathbf{k} = 1$
- Choose initial interior point $(\gamma^{\mathbf{0}}, \pi^{\mathbf{0}}, z^{\mathbf{0}})$ such that $\gamma^{\mathbf{0}} > 0$ and $z^{\mathbf{0}} > 0$
- Choose initial barrier value $\mu^{\mathbf{0}}$

Step 1: Determination of Search Direction

- Solve (23) to determine search direction $(\triangle\gamma^{\mathbf{k}}, \triangle\pi^{\mathbf{k}}, \triangle z^{\mathbf{k}})$

Step 2: Update Solution

- Choose step length α such that the new solution is still inside the feasible region
- Update solution

$$(\gamma^{\mathbf{k}}, \pi^{\mathbf{k}}, z^{\mathbf{k}}) = (\gamma^{\mathbf{k-1}}, \pi^{\mathbf{k-1}}, z^{\mathbf{k-1}}) + \alpha^{\mathbf{k}}(\triangle\gamma^{\mathbf{k}}, \triangle\pi^{\mathbf{k}}, \triangle z^{\mathbf{k}})$$

Step 3: Termination

- if termination criteria is fulfilled STOP,
  otherwise set $\mathbf{k} = \mathbf{k} + 1$, reduce barrier parameter $\mu^{\mathbf{k}}$
  and go to Step 1

Fig. 3 A primal–dual interior-point method

3.3 Determining the search direction

The most expensive step within interior-point methods is the determination of search direction $(\triangle\gamma^{\mathbf{k}}, \triangle\pi^{\mathbf{k}}, \triangle z^{\mathbf{k}})$ by solving (23) in each iteration $\mathbf{k}$. Now, the idea is to exploit the specific structure of the recourse matrix W in (14) in order to reduce the size of system (23). This leads to the specialized interior-point method (SIPM). To simplify the presentation, the iteration counter $\mathbf{k}$ is omitted for now.

Eliminating $\triangle z$ from (23) by setting

$$\triangle z = \Gamma^{-1}(\mu e - \Gamma Z e - Z \triangle \gamma), \tag{24}$$

leads to the so called *reduced KKT-system*:

$$\underbrace{\begin{bmatrix} -D & W^T \\ W & 0 \end{bmatrix}}_{S} \begin{bmatrix} \triangle\gamma \\ \triangle\pi \end{bmatrix} = \begin{bmatrix} rhs_d \\ rhs_p \end{bmatrix}, \tag{25}$$

with $S \in \mathbb{R}^{(2|R||K|+5|N||K|+3|R|) \times (2|R||K|+5|N||K|+3|R|)}$ for the Rawls and Turnquist (2010) model, the diagonal matrix D

$$D = \Gamma^{-1} Z = \begin{bmatrix} \dfrac{z_1}{\gamma_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \dfrac{z_n}{\gamma_n} \end{bmatrix},$$

with $n = |R||K| + 2|N||K| + |R|$ and right hand sides

$$rhs_d = f - W^T \pi - \mu \Gamma^{-1} e$$
$$rhs_p = \tilde{t} - W\gamma.$$

One can even further reduce the KKT-system in (25) by eliminating $\triangle \gamma$:

$$\triangle \gamma = -D^{-1}(rhs_d - W^T \triangle \pi), \tag{26}$$

such that

$$W(-D^{-1}(rhs_d - W^T \triangle \pi)) = rhs_p,$$

and therefore the system of *normal equations* has to be solved:

$$WD^{-1}W^T \triangle \pi = rhs, \tag{27}$$

with $rhs = rhs_p + WD^{-1}rhs_d$. Solving (27) instead of (25) has disadvantages, e.g. if dense columns exist in W as it is often the case in practice.[10] However, due to the specific structure of W in (14), the so-called *Schur complement* $WD^{-1}W^T$ of matrix $\mathcal{S}$ in (25) is a *block arrowhead matrix*:

$$
WD^{-1}W^T =
\begin{bmatrix} u & & \\ & \ddots & \\ & & u \\ H_1 & \cdots & H_{|R|} \; Q \end{bmatrix}
\begin{bmatrix} D_1^{-1} & & \\ & \ddots & \\ & & D_{|R|}^{-1} \\ & & \quad D_Q^{-1} \end{bmatrix}
\begin{bmatrix} u^T & & H_1^T \\ & \ddots & \vdots \\ & u^T & H_{|R|}^T \\ & & Q^T \end{bmatrix}
$$

$$
=
\begin{bmatrix}
uD_1^{-1}u^T & & & uD_1^{-1}H_1^T \\
& \ddots & & \vdots \\
& & uD_{|R|}^{-1}u^T & uD_{|R|}^{-1}H_{|R|}^T \\
H_1 D_1^{-1}u^T & \cdots & H_{|R|}D_{|R|}^{-1}u^T & \sum_{r \in R} H_r D_r^{-1}H_r^T + QD_Q^{-1}Q^T
\end{bmatrix}, \tag{28}
$$

where only entries occur on the diagonal, in the last block row and last block column. The pattern of the arrowhead matrix allows the definition of block matrices in the following way:

$$WD^{-1}W^T = \begin{bmatrix} A & C^T \\ C & B \end{bmatrix} \in \mathbb{R}^{|R|+|N||K| \times |R|+|N||K|}, \tag{29}$$

with diagonal matrix A:

$$A = \begin{bmatrix} uD_1^{-1}u^T & & \\ & \ddots & \\ & & uD_{|R|}^{-1}u^T \end{bmatrix} \in \mathbb{R}^{|R| \times |R|},$$

$$C = \begin{bmatrix} H_1 D_1^{-1}u^T & \cdots & H_{|R|}D_{|R|}^{-1}u^T \end{bmatrix} \in \mathbb{R}^{|N||K| \times |R|},$$

and

$$B = \begin{bmatrix} \sum_{r \in R} H_r D_r^{-1}H_r^T + QD_Q^{-1}Q^T \end{bmatrix} \in \mathbb{R}^{|N||K| \times |N||K|}.$$

Then the system of normal equations (27) can be rewritten as

$$\begin{bmatrix} A & C^T \\ C & B \end{bmatrix} \begin{bmatrix} \triangle \pi_1 \\ \triangle \pi_2 \end{bmatrix} = \begin{bmatrix} rhs_1 \\ rhs_2 \end{bmatrix}, \tag{30}$$

where $\triangle \pi$ and the right-hand side rhs in (27) are partitioned accordingly.

---

[10] In such cases the matrix $WD^{-1}W^T$ is often dense leading to an increased effort when computing its inverse.

Finally, eliminating $\triangle \pi_1$ from (30)

$$\triangle \pi_1 = A^{-1}(rhs_1 - C^T \triangle \pi_2), \tag{31}$$

leads to the equation:

$$\tilde{\mathcal{S}} \triangle \pi_2 = rhs_2 - CA^{-1}rhs_1, \tag{32}$$

where $\tilde{\mathcal{S}} = (B - CA^{-1}C^T) \in \mathbb{R}^{|N||K| \times |N||K|}$ represents the Schur complement of $WD^{-1}W^T$ in (29). As the recourse matrix W has full row rank, $WD^{-1}W^T$ and thus its Schur complement $\tilde{\mathcal{S}}$ is positive definite. After obtaining $\triangle \pi_2$, one can determine the complete search direction $(\triangle \gamma, \triangle \pi, \triangle z)$ just by back-solving the remaining Eqs. (31), (26) and (24). Note that these equations just require the inverses of diagonal matrices A, D and Γ which are computationally inexpensive. Now, the actual task is to solve (32) efficiently. However, this is not problematic since the major matrix size has been reduced significantly from $2|R||K| + 5|N||K| + 3|R| \times 2|R||K| + 5|N||K| + 3|R|$ in (25) to $|N||K| \times |N||K|$ in (32). In addition, one can exploit the symmetry and positive definiteness of $\tilde{\mathcal{S}}$ by using the *Cholesky factorization* (Golub and Van Loan 2013, p. 163). In this way, $\tilde{\mathcal{S}}$ is factorized such that

$$\tilde{\mathcal{S}} = \mathcal{L} \cdot \mathcal{L}^T,$$

where $\mathcal{L}$ is a lower triangular matrix.

Now, Eq. (32) can be solved through simple forward- and backward-substitutions:

$$\mathcal{L}y = rhs_2 - CA^{-1}rhs_1$$
$$\mathcal{L}^T \triangle \pi_2 = y.$$

Here, the product $|N||K|$ (locations multiplied by the number of relief items) and therefore the size of $\tilde{\mathcal{S}}$ will not be very large in practical applications. Castro (2000) propose a similar solution approach for multi-commodity network problems where $WD^{-1}W^T$ is also a block arrowhead matrix. However, W does not represent a recourse matrix in his application. In addition, the resulting Schur complement $\tilde{\mathcal{S}}$ can reach a high degree of density and depends on the number of arcs which is very large in the multi-commodity network problem considered by Castro (2000). Therefore, the author suggests an iterative algorithm, namely the preconditioned conjugate gradient method, instead of Cholesky factorization.

The accelerated L-shaped procedure described in Fig. 2 can be applied to a whole class of two-stage stochastic programs as they are often proposed in disaster management. It is required that the underlying problem fulfills the required conditions for using the standard L-shaped method, i.e. the recourse should be fixed and integer variables occur only on the first stage. This is often the case for models proposed in humanitarian literature where, e.g. location decisions are made on the first stage (binary variables) and transport decisions on the second stage (continuous variables).

The second requirement for applying the accelerated L-shaped method is that the size of the master problem is relatively small. As indicated in Fig. 2, a commercial solver can be used in this case. Indeed, master problems in the humanitarian context are often of manageable size. For instance, first-stage decisions concerning relief locations are most frequently proposed in disaster management (Grass and Fischer 2016). As natural disasters hit usually a specific region with limited coverage, the corresponding locational master problem is therefore of moderate size.

Finally, the block triangular structure of the recourse matrix W in (14) is essential for the applicability of the specialized interior-point method. Various ordering techniques can be found in the literature which can reorder a matrix according to a block triangular form

similar to (14). For instance, the reordering approach of Hu et al. (2000) can be applied to any sparse matrix independent of, e.g. symmetry. In this sense, the accelerated L-shaped method can be used to any two-stage stochastic programs fulfilling these conditions even beyond the humanitarian setting.

4 Realistic large-scale case study

In this section a realistic case study is designed to test the efficiency of the solution method proposed in the previous section. For reasons of comparability, this case study is based on the one used in Rawls and Turnquist (2010) considering the hurricane prone Gulf and Atlantic coast, but it is supplemented by more realistic data. For this case study solely hurricanes, i.e. storms with wind speeds of at least 119 km/h, are considered which can be classified into five categories depending on their wind speed, see Table 1.

4.1 Input data

In total, 900 counties at the American southeast coast have been hit by storms or hurricanes in the past. These counties are grouped to 90 nodes where each node represents the geographical center of the respective neighborhood, see Fig. 4. Hence, the number of nodes in the network,

Table 1 Hurricane categories

Category	Wind speed	
1	119–153 km/h	74–95 mi/h
2	154–177 km/h	96–109 mi/h
3	178–208 km/h	110–129 mi/h
4	209–251 km/h	130–155 mi/h
5	> 252 km/h	> 155 mi/h
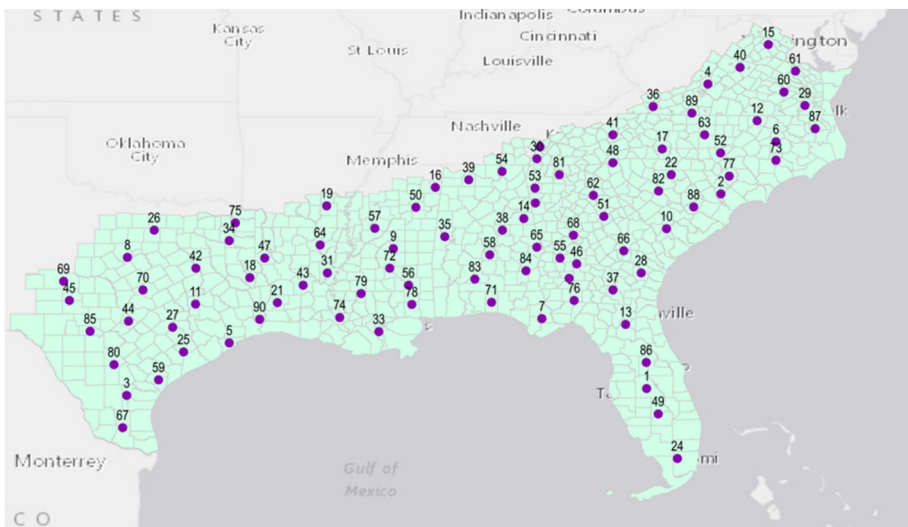
Fig. 4 Network containing 90 nodes at the Gulf and Atlantic coast

Table 2 Opening costs and storage capacities (Rawls and Turnquist 2010)

Size category l	Fixed costs F_l (in \$)	Capacity M_l (in ft³)
1 Small	19,600	36,400
2 Medium	188,400	408,200
3 Large	300,000	780,000

Table 3 Relief items

Item k	Unit	Purchase price q^k (in \$/unit)	Weight u^k (in kg/unit)	Space b^k (in ft³/unit)
1 Water	One liter	0.68	1.13	0.093
2 MRE	One package	3.36	0.76	0.1029
3 Tent	One piece	310	55	9.9499
4 Medical set	One box	770	45	6.3566
5 FAK	One box	870	10.5	2.2434

i.e. $|I| = |N|$, has been increased from 30 to 90 in contrast to the case study of Rawls and Turnquist (2010). This allows a more detailed analysis since the area covered by each node and the distances between them are reduced. In this way, more precise estimations of affected people and the differentiation between areas which are actually hit by a hurricane and those which are close but less affected, are possible.

4.1.1 Storage facilities

Three different sizes of facilities can be opened with the following costs and capacities which are taken from Rawls and Turnquist (2010) for comparability reasons, see Table 2.

4.1.2 Specifications of relief items

For this case study five common types of relief items are considered with specifications given in Table 3. Water, MREs (meals-ready-to-eat) and tents belong to basic equipment of evacuated persons. Medical sets contain essential medicines for preventing epidemics and treating chronically ill persons whereas first aid kits (FAK) are used for injured people. The specific purchase costs q^k, the consumption share per unit of transport capacity u^k and space requirement b^k are based on the *South Carolina Emergency Operations Plan* (SCEMD 2007) as well as the *items catalogue* of IFRC (2016).

Based on MSF (2016), costs for transportation $c_{(i,j)}$ are set to \$0.50 per kg and increase linearly with the respective distance.[11] Relief items can be transported only in case of undamaged arcs, i.e. where arc capacity is large enough. Here, $U^s_{(i,j)} = 100{,}000$ kg should be sufficient and is used for hurricane category 1–3 and for all arcs (i, j). Major hurricanes 4 and 5 cause complete destruction in specific scenarios s such that $U^s_{(i,j)} = 0$ for particular arcs (i, j). According to Rawls and Turnquist (2010), penalty costs for unused items h^k are one quarter of the purchase costs q^k and for unsatisfied demand, $p^k = 10 \cdot q^k$.

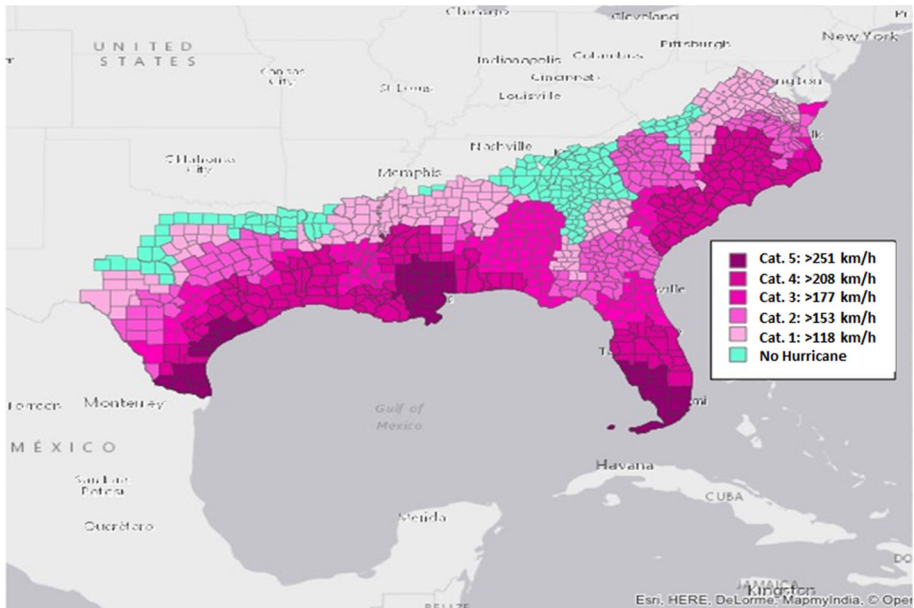[11] Distances are determined by the ArcGIS software.

Fig. 5 Counties hit by hurricanes from 1900 to 2014

4.1.3 Scenario generation

In order to determine realistic scenarios, the database *International Best Track Archive for Climate Stewardship* (IBTrACS) of the *National Centers for Environmental Information* is used, where all storms have been registered since 1900. In this time-frame, the landfall of 203 hurricanes can be identified with a diameter of 161 km on average (URI/GSO 2016). Hence, counties are assumed to be affected if they are located within 80.5 km of a hurricane path. Figure 5 shows areas which were hit by hurricanes of different categories from 1900 to 2014. Especially three regions were hit by hurricanes of the highest category (dark violet areas). One can deduce that these areas were also affected by weaker hurricanes and are therefore under particular threat. In contrast, counties far away from the coast were hit by weaker hurricanes (light pink areas) or by storms which are not classified as hurricanes (green areas).

The database IBTrACS provides information concerning locations, frequencies and strengths of hurricanes of the last 115 years. Hurricanes with similar characteristics, e.g. category and path, are clustered together allowing the definition of 118 scenarios. These scenarios differ in terms of hurricane location, demand for relief items and the destruction level of storage facilities and roads. In contrast, only 15 historical hurricanes are used for the case study proposed by Rawls and Turnquist (2010).

It is assumed that each of the 203 hurricanes which made landfall between 1900 and 2014 had the same probability to occur. As each scenario consists of several hurricanes these probabilities are summed up leading to the corresponding probability of scenario P_s.

4.1.4 Demand

Since demand can be influenced by several factors, like hurricane category, additional floods or the degree of injuries, determining demand is a challenging task. It is assumed that the

Table 4 Percentage of population to be supplied depending on the damage potential of hurricanes (NOAA/AOML 2016)

Category	Wind speed (km/h)	Damage potential	Percentage of affected population
1	119–153	1	0.1
2	154–177	10	1
3	178–208	50	5
4	209–251	250	25
5	> 252	500	50

Table 5 Coverage of relief items and required amounts per person

No.	k	Unit	Coverage pp (1 day)	Required items pp (2 days)
1	Water	One liter	1/3	6
2	MRE	One package	1/2	4
3	Tent	One piece	5	1/5
4	Medical set	One box	1000	1/1000
5	FAK	One box	5	1/5

damage potential of a hurricane, as defined in Table 4, influences the actual demand for each scenario. For instance, 50% of the population in affected counties have to be supplied with pre-positioned relief items in case of a category-5-hurricane.

In addition, aid items should last for two days such that relief organizations have time to determine demand more exactly and can order additional supply. Following the emergency plan of South Carolina, e.g. three liters of water and two meals (MREs) per day and person are required (SCEMD 2007), i.e. one liter of water covers one-third and one meal covers one-half of the recommended daily amount, see Table 5. Hence, six liter of water and 4 MREs for each person have to be pre-positioned for the entire period of two days, as stated in the last column of Table 5.

The resulting case study has a problem size of 4,885,920 decision variables, of which 270 are binary, and over one million constraints (1,009,080). In contrast, a problem with 147,060 (90 binary) variables and 7608 constraints is studied and solved in Rawls and Turnquist (2010).

4.2 Computational results

4.2.1 Implementation details

All numerical experiments in this section were carried out on a Dell Latitude E7470 laptop with a dual 2.60 GHz Intel Core i7 processor and 8 GB of memory.

The commercial optimization solver Gurobi 7.0.0 was used to solve the original problem (1)–(8) directly. In addition, the standard and the accelerated L-shaped method, introduced in the previous section, were implemented in MATLAB 2016b. Note that the algorithm in Fig. 1 is a basic version of the L-shaped method and many acceleration techniques exist in the literature (Rahmaniani et al. 2017). For instance, the one-tree approach allows the

implementation of a single search tree where solving the master problem to optimality in each iteration can be avoided (Naoum-Sawaya and Elhedhli 2013). In addition, advanced cut generation techniques, as proposed by, e.g. Saharidis and Ierapetritou (2013) and Sherali and Lunday (2013), can reduce the number of L-shaped iterations. However, the focus of this work is to highlight the superior behavior of SIPM concerning solution times for the sub-problems. As a result, and as will be shown in the following, the L-shaped method can be accelerated significantly. The advantage of the method proposed in Fig. 2 is that acceleration techniques like the one-tree approach and cut improvements can be applied in addition to SIPM. Such further developments of the L-shaped method seem very promising and are left for future research.

The only difference between the algorithm presented in Fig. 1 and the implementation in MATLAB is that a *multi-cut* approach (Birge and Louveaux 2011, 199) is used. In the classical version of the L-shaped method, a single optimality cut (15) is generated by summing up the multipliers $\overline{\pi}_s^\kappa$ over all scenarios, potentially loosing valuable information. In the multi-cut approach one cut per scenario s is added to the master problem such that θ is replaced by θ_s:

$$\min a^T \chi + \sum_{s \in S} \theta_s.$$

Hence, optimality cuts (15) are changed to

$$\theta_s \geq P_s (t_s - T_s \chi)^T \overline{\pi}_s^\kappa \quad \forall \kappa = 0, \ldots, \iota - 1; \forall s \in S.$$

In this way, several optimality cuts are added to the master problem in every iteration, decreasing the required number of iterations, but also leading to an increased size of the master problem. As stated by Birge and Louveaux (2011), the multi-cut approach is recommended if the number of scenarios does not significantly exceed the number of first-stage constraints, as it is the case here.

According to Fig. 2, the Gurobi interface was used to solve the master problem in every iteration whereas each sub-problem was solved by either MATLAB's built-in function `linprog`[12] (leading to the standard L-shaped method) or by SIPM introduced in Sect. 3.2 (leading to the accelerated L-shaped method). In the latter case a standard primal–dual path-following interior-point method was implemented in MATLAB (Ferris et al. 2007, Ch 8.3) but where the search direction was determined by the procedure described in the previous section. Solving (32) by the Cholesky factorization is the key issue within SIPM. For this purpose MATLAB's built-in function `chol` was used to decompose matrix $\tilde{S}$ into matrices of lower and upper triangular form.

Both interior-point methods, i.e. `linprog` and SIPM, were terminated if the sum of relative residuals and the optimality gap

$$\frac{\|\tilde{t} - W\gamma^{\mathbf{k}}\|}{\max(1, \|\tilde{t}\|)} + \frac{\|f - W^T\pi^{\mathbf{k}} - z^{\mathbf{k}}\|}{\max(1, \|f\|)} + \frac{\left|f^T\gamma^{\mathbf{k}} - \tilde{t}^T\pi^{\mathbf{k}}\right|}{\max\left(1, \left|f^T\gamma^{\mathbf{k}}\right|\right)},$$

was smaller than 1e−06.[13]

In the following these solution approaches are compared for the realistic large-scale case study.

---

[12] The built-in function `linprog` uses a *predictor-corrector primal–dual* interior-point method by default.

[13] Such termination criteria are common for interior-point methods (Wright 1997, p. 226).

Table 6 Comparison of computational results

	Gurobi	L-shaped	Accelerated L-shaped
Overall time	233,637 s (64.9 h)	7541 s (2.09 h)	3968 s (1.1 h)
Rel. opt. gap	0.004	0.009	0.008
No iterations		28	27

4.2.2 Solution times for realistic large-scale case study

First, Gurobi 7.0.0 was used to solve the original model (1)–(8), requiring almost 65 h (see column 'Gurobi' in Table 6).[14] A time-frame of almost three days is unacceptable especially when new short-term forecasts become available and disaster preparation has to take place in a rather short time-frame. Hence, the standard L-shaped method, i.e. where `linprog` was used for the solution of the sub-problems, was analyzed (see column 'L-Shaped' in Table 6). According to Table 6 this approach led to a significant time reduction with an overall solution time of 7541 seconds (about 2 h). Applying the accelerated version of the L-shaped method resulted in further time savings. The solution time of 3968 s (about 1 h) was nearly twice as fast as the standard L-shaped method.

Numerical tests showed that after the relative optimality gap $\frac{\text{upper bound} - \text{lower bound}}{\text{upper bound}}$ fell below 0.008 it remained almost unchanged in the following L-shaped iterations.[15] Such a *tailing-off effect* is a well-known drawback of Benders decomposition methods and several techniques are presented in the literature to overcome this unfavorable behavior (Rahmaniani et al. 2017). However, most of these approaches are problem-dependent and require adjustments for each problem formulation. As already mentioned, the main purpose of this work is to show the efficiency of the specialized interior-point method with respect to computation times and therefore no further acceleration techniques for the L-shaped method were used.

The good performance of the accelerated L-shaped method indicated in Table 6 is due to SIPM. As shown in Table 7, SIPM required on average 0.98 seconds for one sub-problem, i.e. for one scenario. Hence, 115.6 seconds (1.9 min) were needed on average to solve 118 scenarios in each iteration of the accelerated L-shaped method. Almost 80% of the overall time (3121 s of 3968 s) accounted for solving the sub-problems whereas the master problem is computationally inexpensive. In contrast, MATLAB's built-in function `linprog` solved each of these 118 sub-problems in 2 seconds on average, i.e. double as much time as the new approach. As a result the standard L-shaped method in Table 6 needed twice the time of its accelerated counterpart.[16]

Although a MATLAB implementation is not competitive compared with codes written in C/C++, it is intuitively easier to understand. Due to MATLAB's user-friendly interface the accelerated L-shaped method is transparent and can be readily verified. Comparing `linprog`

---

[14] By default, Gurobi solves the MIP root node via the dual simplex method and run the barrier and simplex method on multiple threads concurrently for LP problems (Gurobi Optimization 2017).

[15] For comparison reasons Gurobi was terminated if the relative optimality gap fell below 0.009. However, the relative optimality gap jumped from 0.173 to 0.004 in the last iteration of Gurobi.

[16] As mentioned above, column 'L-Shaped' in Table 6 refers to the method where MATLAB's `linprog` and 'Accelerated L-Shaped' where SIPM is used for solving the sub-problems. Since computation times for solving the master problem in each iteration are independent of the sub-problems and insignificant in comparison to the solution times of the sub-problems, the overall time for the L-shaped method consists mainly of computation times for the sub-problems. Therefore, solution times given in Table 7 have a major influence on the overall solution times of the L-shaped method as given in Table 6.

Table 7 Comparison of LP solvers for sub-problems

	SIPM (s)	linprog (s)	Dual simplex (s)	Barrier (s)
Average time (per sub-problem)	0.98	2	1.63	0.93
Total time (per L-shaped iteration)	115.6	236	192.3	109.7

with the specialized interior-point method is reasonable as both approaches are implemented in MATLAB with the exception that the calculation of the search direction within linprog is coded in FORTRAN.

However, additional tests were carried out where Gurobi was used to solve the sub-problems. With Gurobi a solution time of 1.49 s per sub-problem was achieved on average when the dual simplex was chosen (see 'Dual Simplex' in Table 7). Using Gurobi's barrier method instead led to an average time of 0.93 s for each sub-problem. It can be concluded that if a more sophisticated programming language like Fortran or C++ was used, SIPM could outperform Gurobi further. It should be mentioned that calculation times could be reduced even more if the sub-problems were solved in parallel.

4.2.3 Location results for realistic large-scale case study

In this section the results for the specific planning problem are to be presented. Location decisions as determined by the proposed solution method are shown in Fig. 6, where the heuristically found solution of Rawls and Turnquist (2010) is added for comparison.

It is obvious that the solution structures concerning the location of facilities differ significantly. This is mainly due to the way in which scenarios have been defined, namely which areas have been identified as particularly vulnerable. In case of Rawls and Turnquist, two large-sized facilities are opened in areas which were hit by major hurricanes in the past and are therefore under particular threat. One can deduce that these facilities, and relief items stored there, are likely to be destroyed in a future hurricane setting. In this case items have to be supplied from the facility located at Memphis (blue triangle in the North). However, some serious injuries have to be treated with medicine within the first 4 h which cannot be guaranteed in this case. Even if some items are not time-critical, sufficient quantities may be not available due to the small size of this facility. In contrast, the solution of the large-scale case study suggests to locate much more small-sized facilities along the coast and the northern border. The latter can be used as backup storages if coastal facilities are destroyed.

It becomes clear that scenarios which only insufficiently reflect reality can have far-reaching implications with regard to the results and hence to the performance of humanitarian operations. When planning is based on more realistic scenario definitions, as in the proposed case study, storage facilities are located rather away from vulnerable areas, but still close enough to provide first aid. The results can then be used as decision support by humanitarian organizations
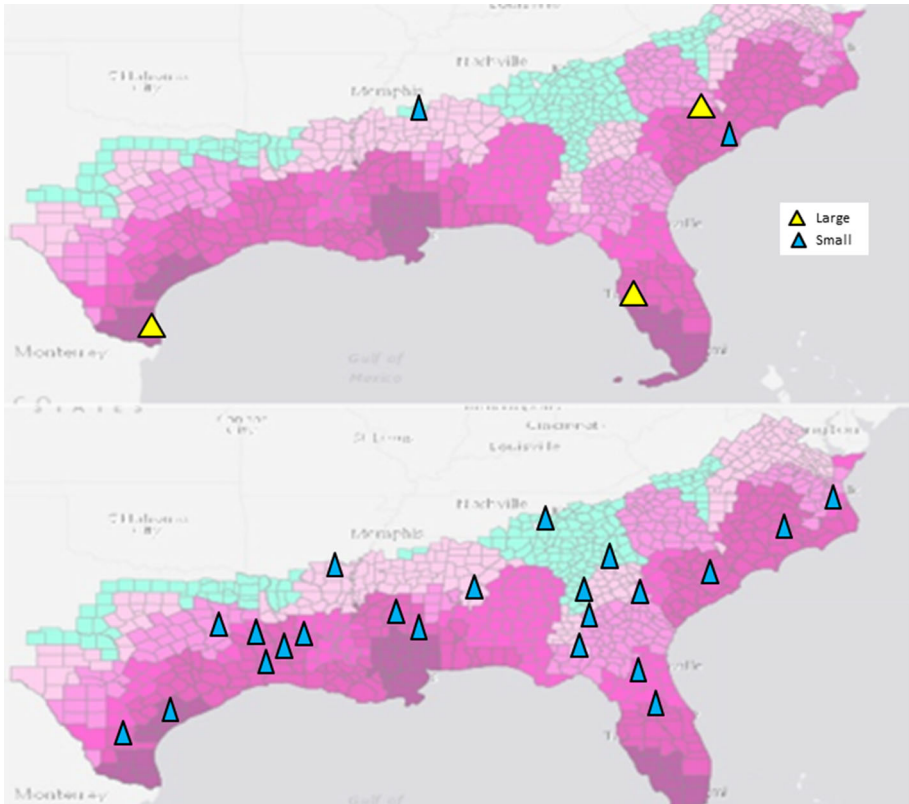
Fig. 6 Facility locations as suggested by Rawls and Turnquist ([2010]) (top) and solution for the large-scale case study found by newly proposed method (bottom)

4.2.4 Additional tests

In order to verify the competitiveness of the accelerated L-shaped method additional tests were carried out. In particular, three case studies have been designed with problem sizes shown in Table 8.

The small-scale case study has the same size and data as the one proposed by Rawls and Turnquist ([2010]). Missing parameter values like distances and arc capacities were completed in the same way as described in Sect. 4.1.

Table 8 Problem size of a small-, medium- and large-scale case study

Number of	Case study		
	Small	Medium	Large
Nodes	30	90	90
Types of relief items	3	3	5
Facility size categories	3	3	3
Scenarios	51	80	118
Constraints	7608	669, 780	1, 009, 080
Decision variables	147, 150	1, 988, 010	4, 886, 190

Table 9 Comparison of computational results

Case study	Gurobi	L-shaped	Accelerated L-shaped
Small			
Overall time	7.8 s	244.5 s (4.1 min)	221.3 s (3.7 min)
Rel. opt. gap	0.0001	0.0002	0.0005
No iterations		29	29
Medium			
Overall time	3462 s (57.7 min)	1348 s (22.5 min)	631 s (10.5 min)
Rel. opt. gap	0.02	0.06	0.05
No iterations		16	16
Large			
Overall time	191,851 s (53.29 h)	5486 s (1.52 h)	2939 s (49 min)
Rel. opt. gap	0.003	0.006	0.006
No iterations		24	23

The medium-scale case study is about half the size of the realistic case study with almost 700,000 constraints and two million decision variables.

Finally, the large-scale case study is the same size as the realistic case study but some scenario-dependent parameters were randomly altered. The purpose of this case study is to analyze the performance of the accelerated L-shaped method, and especially of SIPM, in the case of data changes.

According to Table 9 the L-shaped method cannot be recommended for small-sized problems as both versions were much slower than Gurobi, requiring several minutes instead of only seconds. The reason for the poor performance of the L-shaped method is that the master problem has to be solved in each iteration. Overall this is more time-consuming than solving the original problem directly. However, as the problem size increases the computational benefits of the L-shaped method become apparent.

As can be seen in Table 9, Gurobi needed almost 1 h to solve the medium-scale problem whereas the standard L-shaped method could reduce the overall solution time to about 22 min. The accelerated L-shaped method even halved this time again to about 10 min. Note that both versions of the L-shaped method were terminated after 16 iterations to avoid the tailing-off effect. As mentioned before the relative optimality gap showed little change in the following iterations. For reasons of comparability Gurobi was terminated after reaching a relative optimality gap of 0.06. However, as for the realistic case study, an actual gap of 0.02 was achieved.

For the large-scale problem, similar performance results as in Table 6 can be obtained. Moreover, the solution times are similar to the large-scale case study, i.e. almost 1 h for the accelerated method and 2 h for the standard method. Hence, changes in the data do not seem to have a big influence on the performance. In particular, both versions of the L-shaped method were much faster than Gurobi again reducing solution times significantly. Again this is mainly due to the use of SIPM for solving the sub-problems.

In Table 10 different solution approaches for sub-problems are compared.

The table reveals that SIPM is twice as fast as `linprog` independent of the problem size and input data. In addition, SIPM outperforms Gurobi's dual simplex for the medium- and large-scale problem. Although written in MATLAB the specialized interior-point method can even compete with the barrier method of Gurobi. Based on these observations it can be

Table 10 Comparison of LP solvers for sub-problems

Case study	SIPM (s)	`linprog` (s)	Dual simplex (s)	Barrier (s)
Small				
Average time (per sub-problem)	0.04	0.07	0.04	0.12
Total time (per L-shaped iteration)	2.04	3.57	2.04	6.12
Medium				
Average time (per sub-problem)	0.4	0.7	0.7	0.36
Total time (per L-shaped iteration)	32	56	56	28.8
Large				
Average time (per sub-problem)	0.87	1.58	1.55	0.88
Total time (per L-shaped iteration)	103	186	183	104

recommended to implement SIPM into, e.g. CPLEX's Benders decomposition to achieve solutions even faster.

As illustrated by the case study, more realistic problems are harder to solve even with state-of-the-art solvers. Here it was shown that exploiting the two-stage structure via the L-shaped method and making use of the special structure of the recourse matrix W in (14) leads to significant time savings. Of course, solution times may be not crucial for strategic location decisions. However, in practice most relief organizations use public buildings like gyms or town halls as provisional warehouses or shelters for evacuees and do not establish new buildings. Therefore, decisions where to store relief items are often made on a short-term basis and as soon as, e.g. a hurricane forecast is available. Hence, fast solution methods are essential to ensure responsiveness.

5 Conclusion and future research directions

Due to their unpredictable and devastating nature, disasters represent a serious threat for those who are affected. One of the main activities to mitigate human suffering is to take appropriate precautions like storing relief items at selected locations before a disaster occurs. Such decisions have to be made under a high degree of uncertainty and two-stage stochastic programming is a preferred approach in this regard. Especially the model proposed by Rawls and Turnquist (2010) is seen as a benchmark in the relevant literature and therefore used here as a representative for two-stage stochastic programs in disaster management. Obtaining solutions of these programs fast, especially in case of short-term forecasts, can be decisive in terms of rapid aid response and live-savings. Unfortunately, commercial solvers may be too memory- and time-consuming for realistic large-scale problems. As a remedy, heuristics are often suggested which have, however, several disadvantages. First, optimality of the resulting solution usually is not guaranteed and the optimality gap is often not measurable. Second, heuristics are usually problem-dependent such that relief organizations need the necessary know-how to use and adjust these heuristics for their applications.

Therefore, the L-shaped method is suggested in this paper which is an exact method but can be terminated when the desired accuracy level is achieved. In order to accelerate the standard L-shaped algorithm, the specialized interior-point method SIPM is used to solve the sub-problems efficiently by exploiting the specific structure of the second-stage constraints. In this way, the most expensive step within interior-point methods, namely the determination

of search directions, is facilitated. To illustrate the advantages of this approach, a realistic large-scale case study for the hurricane-prone American south-east coast was designed. Time for solving this problem was reduced significantly in comparison to the Gurobi solver and to the standard L-shaped method. Additional tests confirmed the competitiveness of the newly proposed SIPM against Gurobi's simplex and barrier method.

An important advantage of the accelerated L-shaped method is that it can be applied to a whole class of two-stage stochastic programs often proposed in the humanitarian literature. In order to extend this class of two-stage stochastic models even more, further developments are needed. For instance, difficult or large-scale master problems can be solved by constructing a single search tree as it is done by, e.g. the *Branch-and-Benders-cut* algorithm (Naoum-Sawaya and Elhedhli 2013) instead of repeatedly solving the master problem to optimality. Moreover, it would be useful to analyze what kind of cut-generation and branching strategies are particularly suitable for two-stage stochastic programs in disaster management. Extending the accelerated L-shaped method by both approaches represents a valuable research direction. In addition, future research should focus further on parallelized computations of sub-problems to generate several optimality cuts at the same time.

In conclusion, as solution methods which solve the underlying problem within a reasonable time-frame allow the humanitarian organization to react especially to short-term forecasts leading to less supply surplus, reduced costs, higher demand satisfaction and therefore reduced numbers of casualties, they are particularity useful for aid organizations. The L-shaped method and its modifications offer interesting future research opportunities in this regard.

References

Birge, J., & Louveaux, F. (2011). *Introduction to stochastic programming*. Berlin: Springer.

Castro, J. (2000). A specialized interior-point algorithm for multicommodity network flows. *SIAM Journal on Optimization*, *10*(3), 852–877. https://doi.org/10.1137/S1052623498341879.

Castro, J., Nasini, S., & Saldanha-da Gama, F. (2016). A cutting-plane approach for large-scale capacitated multi-period facility location using a specialized interior-point method. *Mathematical Programming*,. https://doi.org/10.1007/s10107-016-1067-6.

Cay, S. B, Pólik, I., & Terlaky, T. (2017). *Warm-start of interior point methods for second order cone optimization via rounding over optimal Jordan frames*. Technical report, ISE technical report 17T-006, Lehigh University, 2017. http://www.optimization-online.org/DB_HTML/2017/05/5998.html.

De Silva, A., & Abramson, D. (1998). A parallel interior point method and its application to facility location problems. *Computational Optimization and Applications*, *9*(3), 249–273.

Diwekar, U. (2008). *Introduction to applied optimization*. Springer. http://www.ebook.de/de/product/7522360/urmila_diwekar_introduction_to_applied_optimization.html.

Ferris, M., Mangasarian, O., & Wright, S. (2007). Linear programming with MATLAB. *Society for Industrial and Applied Mathematics*,. https://doi.org/10.1137/1.9780898718775.

Fragnière, E., Gondzio, J., & Vial, J. P. (2000). Building and solving large-scale stochastic programs on an affordable distributed computing system. *Annals of Operations Research*, *99*(1), 167–187. https://doi.org/10.1023/A:1019245101545.

Golub, G. H., & Van Loan, C. F. (2013). *Matrix computations* (4th ed.). Baltimore: John Hopkins.

Gondzio, J., & Sarkissian, R. (2003). Parallel interior-point solver for structured linear programs. *Mathematical Programming*, *96*(3), 561–584.

Grass, E., & Fischer, K. (2016). Two-stage stochastic programming in disaster management: A literature survey. *Surveys in Operations Research and Management Science*, *21*(2), 85–100. https://doi.org/10.1016/j.sorms.2016.11.002.

Gurobi Optimization. (2017). *Parameter documentation*. http://www.gurobi.com/documentation/7.5/refman/method.html#parameter:Method. Accessed December 05, 2017.

Hu, Y., Maguire, K., & Blake, R. (2000). A multilevel unsymmetric matrix ordering algorithm for parallel process simulation. *Computers & Chemical Engineering*, *23*(11), 1631–1647. https://doi.org/10.1016/S0098-1354(99)00314-2.

IFRC. (2016). *International federation of red cross and red crescent societies—Items catalogue*. http://procurement.ifrc.org/catalogue/detail.aspx. Accessed January 01, 2017.

MSF. (2016). Aerzte ohne Grenzen e.V.-Private Communication.

Naoum-Sawaya, J., & Elhedhli, S. (2013). An interior-point Benders based branch-and-cut algorithm for mixed integer programs. *Annals of Operations Research*, *210*(1), 33–55. https://doi.org/10.1007/s10479-010-0806-y.

Nielsen, S. S., & Zenios, S. A. (1997). Scalable parallel Benders decomposition for stochastic linear programming. *Parallel Computing*, *23*(8), 1069–1088. https://doi.org/10.1016/S0167-8191(97)00044-6.

NOAA/AOML. (2016). *National oceanic and atmospheric administration/Atlantic oceanographic and meteorological laboratory*. http://www.aoml.noaa.gov/hrd/tcfaq/D5.html. Accessed August 18, 2017.

Pay, B. S., & Song, Y. (2017). Partition-based decomposition algorithms for two-stage stochastic integer programs with continuous recourse. *Annals of Operations Research*,. https://doi.org/10.1007/s10479-017-2689-7.

Rahmaniani, R., Crainic, T. G., Gendreau, M., & Rei, W. (2017). The Benders decomposition algorithm: A literature review. *European Journal of Operational Research*, *259*(3), 801–817. https://doi.org/10.1016/j.ejor.2016.12.005.

Rawls, C. G., & Turnquist, M. A. (2010). Pre-positioning of emergency supplies for disaster response. *Transportation Research Part B: Methodological*, *44*(4), 521–534. https://doi.org/10.1016/j.trb.2009.08.003.

Saharidis, G. K. D., & Ierapetritou, M. G. (2013). Speed-up benders decomposition using maximum density cut (mdc) generation. *Annals of Operations Research*, *210*(1), 101–123. https://doi.org/10.1007/s10479-012-1237-8.

SCEMD. (2007). *South Carolina emergency management division-South Carolina logistical operations plan: Appendix 7*. http://dc.statelibrary.sc.gov/handle/10827/20614. Accessed August 18, 2017.

Sherali, H. D., & Lunday, B. J. (2013). On generating maximal nondominated benders cuts. *Annals of Operations Research*, *210*(1), 57–72. https://doi.org/10.1007/s10479-011-0883-6.

Slyke, R. M. V., & Wets, R. (1969). L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, *17*(4), 638–663. https://doi.org/10.1137/0117061.

Turkeš, R., Cuervo, D. P., & Sörensen, K. (2017). Pre-positioning of emergency supplies: Does putting a price on human life help to save lives? *Annals of Operations Research*,. https://doi.org/10.1007/s10479-017-2702-1.

URI/GSO. (2016). University of Rhode Island and Graduate School of Oceanography-Hurricanes: Science and Society. http://www.hurricanescience.org/science/science/hurricanestructure/. Accessed August 18, 2017.

Wright, S. J. (1997). *Primal–dual interior-point methods*. Philadelphia: SIAM. https://doi.org/10.1137/1.9781611971453.bm.

Zheng, Q. P., Wang, J., Pardalos, P. M., & Guan, Y. (2013). A decomposition approach to the two-stage stochastic unit commitment problem. *Annals of Operations Research*, *210*(1), 387–410. https://doi.org/10.1007/s10479-012-1092-7.