Stochastic Optimization for the Allocation of Medical Assets in Steady-state Combat Operations II

Alex Kwan[1[1001559057]] and Victor Lo[2[1001262732]]

University of Toronto, Toronto ON M5S 1A1, CAN
alexkwan@mail.utoronto.ca, victorlo@mie.utoronto.ca

Abstract. In this paper we present a Benders decomposition and constraint programming model to solve a two-stage stochastic optimization problem which determines the placement of medical and evacuation facilities to minimize the time traveled, weighted by casualty severity, from an evacuation site to injury site to medical facility. The first-stage decisions determine the placement of these medical and evacuation facilities. The second-stage determines the optimal routing of casualties to medical facilities, given uncertain scenario parameters including casualty locations, severity, and numbers. Scenarios are generated following historical data from Operation Iraqi Freedom.

Keywords: Emergency services · Medical evacuation · Location facility · Mixed-integer-programming · Constraint programming · Benders decomposition.

1 Introduction

In military operations, militaries often rely on mobile hospitals and evacuation assets to treat casualties during a military campaign. Thus, to minimize the severity of casualties, it is important to consider the location of placing these facilities. This paper presents a two-stage stochastic optimization model where the first stage decision is the placement of evacuation and hospital sites. The second stage decision is the number of patients transported from injury locations to hospital sites by a crew and vehicle stationed at evacuation sites. Thus, the model presented follows a network flow model. The model's objective is to minimize the total time traveled, weighted by patient severity, from the evacuation site to the point of injury and onward to the hospital location. Stochasticity arises from casualty locations, numbers, and severity. Additionally, transportation speed, capacity, and method proportion (e.g. a maximum of 15% air evacuations allowed), and hospital capacity can vary between scenarios. This paper builds on *Two-stage Stochastic Optimization for the Allocation of Medical Assets in Steady-state Combat Operations*, which presents an extensive form model [1]. In this paper, we introduce two new optimization models: a Benders decomposition and a constraint programming model. We benchmark these models against the original extensive form model over a series of generated data instances.

2 Problem Description

In this section, we provide a mathematical description of this medical evacuation problem.

Model sets:

- $\mathcal{T}$, scenarios for evaluation with index $t \in \mathcal{T}$
- $\mathcal{I}$, set of locations where injuries may occur with index $i \in \mathcal{I}$
- $\mathcal{J}$, set of candidate evacuation sites with index $j \in \mathcal{J}$
- $\mathcal{K}$, set of candidate hospital sites with index $k \in \mathcal{K}$
- $\mathcal{L}$, type of bed requirement for a patient (minimal, intermediate, and intensive care) with $l \in \mathcal{L}$
- $\mathcal{M}$, type of evacuation assets, ground or air with $m \in \mathcal{M}$

Parameters:

- p_t, probability that scenario t occurs (non-negative and summing to 1)
- a_{it}, injury severity weight of most critically injured patient departing from location i for scenario t
- b_{ijt}, distance between injury location i and evacuation site j for scenario t
- c_{ikt}, distance between injury location i and hospital location k for scenario t
- d_{jkmt}, speed of transport from j to k by vehicle type m for scenario t
- e_{it}, number of total patients injured at location i for scenario t
- f_{lt}, the number of patients with type l bed requirements in scenario t
- $ecap_{mt}$, capacity during scenario t for patient evacuation by m-type evacuation assets
- $hcap_{lt}$, capacity during scenario t for hospital acceptance of l-type patients
- $enod_{jt}$, evacuation node capacity for each node j in scenario t
- $hnod_{kt}$, hospital node capacity for each node k in scenario t
- u, maximum number of hospitalization sites to be occupied
- v, maximum number of evacuation sites to be occupied
- wx_{jt}, percentage of ground ambulances required for each evacuation site j based on optimal mix factors in scenario t

3 Models

In this section, we discuss the three models implemented: the original extensive form MIP, and our two methods: Benders decomposition and constriant programming.

3.1 Extensive Form MIP

The extensive form MIP uses two first stage and one second stage decision variable:

- $y_j = \begin{cases} 1, & \text{if air evacuation site } j \text{ is chosen} \\ 0, & \text{otherwise} \end{cases}$
- $z_k = \begin{cases} 1, & \text{if hospitalization site } l \text{ is chosen} \\ 0, & \text{otherwise} \end{cases}$
- $x_{ijklmt} \geq 0$, number of patients traveling from injury location i, by a vehicle and crew located at evacuation site j, to hospital site k with bed-type requirement l on vehicle m for scenario t

Then, the model is as follows:

$$\min \sum_{t\in\mathcal{T}} p_t \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{l\in\mathcal{L}} \sum_{m\in\mathcal{M}} a_{it} \left(\frac{b_{ijt} + c_{ikt}}{d_{ijkmt}} \right) x_{ijklmt} \tag{1}$$

$$s.t. \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{l\in\mathcal{L}} \sum_{m\in\mathcal{M}} x_{ijklmt} = e_{it} \qquad \forall i \in \mathcal{I}, \forall t \in \mathcal{T} \tag{2}$$

$$\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{l\in\mathcal{L}} x_{ijklmt} \leq ecap_{mt} \qquad \forall m \in \mathcal{M}, \forall t \in \mathcal{T} \tag{3}$$

$$\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{m\in\mathcal{M}} x_{ijklmt} \leq hcap_{lt} \qquad \forall l \in \mathcal{L}, \forall t \in \mathcal{T} \tag{4}$$

$$\sum_{i\in\mathcal{I}} \sum_{k\in\mathcal{K}} \sum_{l\in\mathcal{L}} \sum_{m\in\mathcal{M}} x_{ijklmt} \leq enod_{jt} y_j \qquad \forall j \in \mathcal{J}, \forall t \in \mathcal{T} \tag{5}$$

$$\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \sum_{l\in\mathcal{L}} \sum_{m\in\mathcal{M}} x_{ijklmt} \leq hnod_{kt} z_k \qquad \forall k \in \mathcal{K}, \forall t \in \mathcal{T} \tag{6}$$

$$\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{l\in\mathcal{L}} \sum_{m\in\mathcal{M}} x_{ijklm='G't} \geq \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{l\in\mathcal{L}} \sum_{m\in\mathcal{M}} wx_{it} x_{ijklmt} \qquad \forall t \in \mathcal{T} \tag{7}$$

$$\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{m\in\mathcal{M}} x_{ijklmt} = f_{lt} \qquad \forall l \in \mathcal{L}, \forall t \in \mathcal{T} \tag{8}$$

$$\sum_{j\in\mathcal{J}} y_j = v \tag{9}$$

$$\sum_{k\in\mathcal{K}} z_k = u \tag{10}$$

$$x_{ijklmt} \geq 0 \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \forall k \in \mathcal{K}, \forall l \in \mathcal{L}, \forall t \in \mathcal{T} \tag{11}$$

$$y_j \in \{0, 1\} \qquad \forall j \in \mathcal{J} \tag{12}$$

$$z_k \in \{0, 1\} \qquad \forall k \in \mathcal{K} \tag{13}$$

Objective (1) minimizes the number the penalty-weighted time for evacuating patients via helicopter or ground ambulance along all routes for all scenarios. There are no costs incurred during the first stage. Constraint (2) ensures that all patients are evacuated. Constraint (3) prevents over-capacity of evacuation vehicles. Constraint (4) prevents over-capacity of hospitalization sites. Constraints (5) and (6) set the maximum network flow at evacuation and hospitalization nodes. Constraint (7) sets the proportion of ground based evacuations. Note G refers to ground transportation. This constraint represents real-world limits on aircraft evacuation. Constraint (8) forces the distribution of patient types at the hospitals equals the distribution of patient types in flow. Constraints (9) and (10) limit the total number of evacuation and hospitalization sites to open out of the all possible locations. Constraints (11) - (13) set the decision variable types.

3.2 Constraint Programming

The constraint programming model follows the same form as the extensive form model (1). No global constraints were added to the model. Thus, the constraint programming model's main difference from the extensive form MIP is that it is solved using a constraint programming solver.

3.3 Benders Decomposition

Benders Multi-cut - Relaxed Master Problem

$$\min \sum_{t\in\mathcal{T}} p_t\eta_t \tag{14}$$
$$s.t. \sum_{j\in\mathcal{J}} y_j = v \tag{15}$$
$$\sum_{k\in\mathcal{K}} z_k = u \tag{16}$$
$$y_j \in \{0,1\} \qquad \forall j\in\mathcal{J} \tag{17}$$
$$z_k \in \{0,1\} \qquad \forall k\in\mathcal{K} \tag{18}$$

Benders Single-cut - Relaxed Master Problem

$$\min \theta \tag{19}$$
$$s.t. \sum_{j\in\mathcal{J}} y_j = v \tag{20}$$
$$\sum_{k\in\mathcal{K}} z_k = u \tag{21}$$
$$y_j \in \{0,1\} \qquad \forall j\in\mathcal{J} \tag{22}$$
$$z_k \in \{0,1\} \qquad \forall k\in\mathcal{K} \tag{23}$$

Sub-problems

$$Q_t(x) = \min \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} a_{it}\left(\frac{b_{ijt}+c_{ikt}}{d_{ijkmt}}\right)x_{ijklmt} \tag{24}$$
$$s.t. \sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} x_{ijklmt} = e_{it} \qquad \forall i\in\mathcal{I} \tag{25}$$
$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}} x_{ijklmt} \le ecap_{mt} \qquad \forall m\in\mathcal{M} \tag{26}$$
$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{m\in\mathcal{M}} x_{ijklmt} \le hcap_{lt} \qquad \forall l\in\mathcal{L} \tag{27}$$
$$\sum_{i\in\mathcal{I}}\sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} x_{ijklmt} \le enod_{jt}y_j \qquad \forall j\in\mathcal{J} \tag{28}$$
$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} x_{ijklmt} \le hnod_{kt}z_k \qquad \forall k\in\mathcal{K} \tag{29}$$
$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} x_{ijklm='G't} \ge wx_{it}\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} x_{ijklmt} \tag{30}$$
$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{m\in\mathcal{M}} x_{ijklmt} = f_{lt} \qquad \forall l\in\mathcal{L} \tag{31}$$
$$x_{jikt} \ge 0 \qquad \forall j\in\mathcal{J},\forall i\in\mathcal{I},\forall k\in\mathcal{K} \tag{32}$$

Note that the $\forall t\in\mathcal{T}$ disappears for each subproblem.

Benders Cuts Optimality cuts added to the Master Problem for each scenario using the objective from the Dual of the Sub-Problem (6 dual variables since 6 primal constraints, RHS of Primal constraints become coefficients of Dual Objective). Note that wx_{it} does not appear in the dual objective as a coefficient since constraint (30)'s RHS is 0. No feasibility cuts are required since the sub-problems were always feasible.
Multi-cut Benders Cut:

$$\eta_t > Q_t(x) = max\sum_{i\in\mathcal{I}}(e_{it})\pi_{it} + \sum_{m\in\mathcal{M}}(ecap_{mt})\gamma_{mt} + \sum_{l\in\mathcal{L}}(hcap_{lt})\alpha_{lt} + \sum_{j\in\mathcal{J}}(enod_{jt}y_j)\lambda_{jt} + \sum_{k\in\mathcal{K}}(hnod_{kt}z_k)\delta_{kt} + \sum_{l\in\mathcal{L}}(f_{lt})\psi_{lt} \tag{33}$$

Single-cut Benders Cut:

$$\theta > Q(x) = \sum_{t\in\mathcal{T}} p_t(max \sum_{i\in\mathcal{I}}(e_{it})\pi_{it} + \sum_{m\in\mathcal{M}}(ecap_{mt})\gamma_{mt} + \sum_{l\in\mathcal{L}}(hcap_{lt})\alpha_{lt} + \sum_{j\in\mathcal{J}}(enod_{jt}y_j)\lambda_{jt} + \sum_{k\in\mathcal{K}}(hnod_{kt}z_k)\delta_{kt} + \sum_{l\in\mathcal{L}}(f_{lt})\psi_{lt}) \tag{34}$$

4 Computational Results

This section discusses the computational results for the three solution methods implemented, which were tested using a variety of problem instances. We define an instance as a randomly generated problem structure at a selected number of scenarios. For each instance, 10 individual randomly generated trials were run. The instances tested are located in Table 1. For example, all of the 10 trials from instance 2 have 50 scenarios. All trials were ran with a time limit of 30 minutes and

Instance	1	2	3	4	5	6
# Scenarios	25	50	75	100	125	150

Table 1. Data Instances

were limited to use one thread. The experiments were run on a Ubuntu 18.04.3 LTS computer with a Intel Core i7-9700 CPU @ 3.00GHz $\times$ 8 CPU and 31.2 GB of RAM. The extensive form and Benders models were coded in Python and solved using Gurobi 9.0. The constraint programming model was coded in Java and solved using IBM ILOG CP Optimizer 12.9.

4.1 Original Paper Re-implementation

Before experimenting with Benders Decomposition and constraint programming using the scenarios outlined in Table 1, the extensive model with the same nine scenarios as the original paper was formulated and solved. Additionally the locations of the same 10 potential hospital sites and same 20 potential evacuation sites were also used. These nine scenarios (T0 to T8) follow a factorial experiment design with four factors as provided in Table 2:

Factor	lethality multiplier —	casualty radius (NM) —	min % of ground ambulances(wx) —	days
Scenario				
0	1	50	0.2	7
1	1.5	50	0.4	14
2	2.0	50	0.3	21
3	1.0	100	0.3	14
4	1.5	100	0.2	21
5	2.0	100	0.4	7
6	1.0	150	0.4	21
7	1.5	150	0.3	7
8	2.0	150	0.2	14

Table 2. Factorial Design Parameters

The model was solved in 195 seconds with simplex compared to the original paper's 19.3 minutes. A visual plot of the nine scenario casualties and the chosen sites are in Figure 1 below. Note that nearly the exact same 5 hospital sites and 10 evacuation sites are chosen as the original paper, with the minor difference in chosen sites likely being attributed to the random distribution sampling for casualty locations. A optimal objective value of 375 was found with a 0.0000% optimality gap.

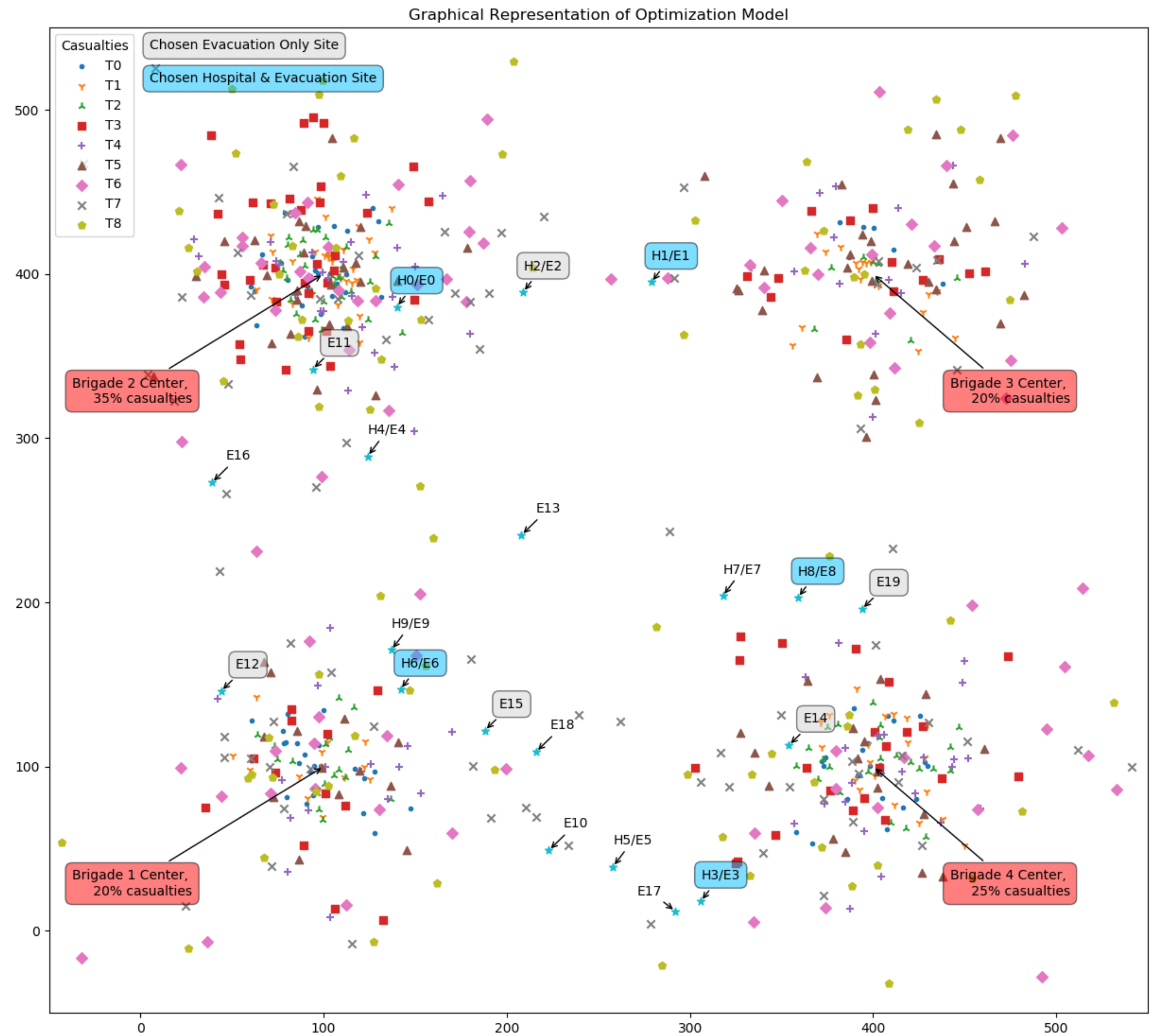

Figure 1. Extensive model's selected hospital and evacuation sites based on nine scenarios from the original paper.

4.2 Data Generation

As this problem is based off of past military historical data, each trial's parameters were sampled following the probability distributions in the original problem paper [1]. The random seed was set based on a function of the instance and trial number such that each solution method was run using the same trial data for a given trial. All of the data generation was completed in Python.

4.3 Benders Decomposition and Constraint Programming Issues

Unfortunately, we were unable to run the Benders decomposition method against the generated test instances as it was taking a long time to initialize the model. MIPs with Big-M constraints are hard to solve [2] so the values of $enod_{kt}$ and $hnod_{jt}$ (evacuation site and hospital site capacities) were changed from 1 million to tight scenario-specific values to avoid numerical issues. Therefore for each scenario, t the $enod_{kt}$ and $hnod_{jt}$ values chosen were the sum of all casualties in each scenario = $\sum(e_i t) \forall i \in \mathcal{I}$. Even still, the Single-cut Benders failed to initialize (lower bound did not change after 100 iterations) and while the Multi-cut Benders did initialize (lower bound did change) it crashed before converging on an optimal solution. The constraint programming model was unable to find any feasible solutions to the given trials in the allowed time limit. In fact, when the model was run without a time limit, it took approximately three hours to find a feasible solution. Thus, the results sections below only showcase the extensive form MIP model.

4.4 Optimality Gap

Figure 1 shows the average optimality gap for the extensive form model across all instances. At instances with less scenarios, the extensive form is able to solve all trials to optimality. However, by instance 2, consisting of trials with 50 scenarios, the model begins to have issues finding optimal solutions in the given time limit. This trend continues as the extensive form optimality gap increases as the number of scenarios increases.

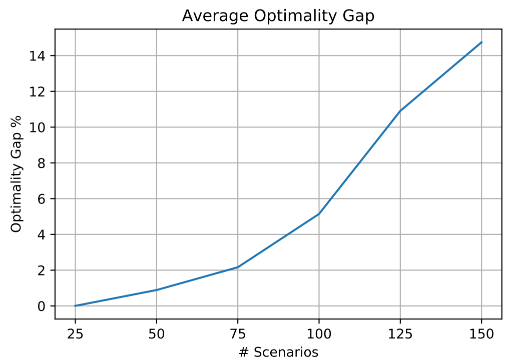

Fig. 1. The optimality gaps for the extensive form MIP across all instances

4.5 Run Time

Figure 2 shows the average run time for the extensive form model across all instances. At instances with less scenarios, the extensive form runs quite quickly; however, its run time quickly reaches the max limit. For the remaining instances, it uses the full time limit.

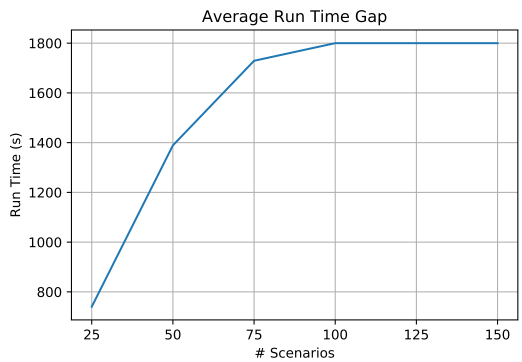

Fig. 2. Run time for the extensive form MIP across all instances

4.6 Discussion

From the extensive form MIP results, it appears that it is an efficient way to solve problems with a smaller number of scenarios. In these situations it is able to find optimal solutions quickly. However, as the number of scenarios increases, the extensive form model's solution quality begins to worsen. After this point, it would be more appropriate to solve the trials with a more sophisticated technique, such as Benders Decomposition. However our Benders Decomposition implementation was unable to converge on an optimal solution. One implementation issue is that the RHS of constraint 30 was not able to be changed, and therefore had to be removed and re-added for each subproblem. Also the single-cut Benders failing to initialize is likely due to the need for all sub-problems having to be solved and the enormous number of constraints. The constraint programming model's poor performance was likely do to the number of variables and following a domain pruning approach. Even for the smaller sized instances, they had approximately 80,000 variables. As such, constraint programming faced difficulty in determining the correct domains for these, and further had difficulty in assigning them values.

5 Conclusions and Future Work

This paper presented two new solution approaches to minimize severity-time weighted patient evacuation in military combat operations whilst determining the facility location of medical assets. The Benders and constraint programming models were tested against the previous literature extensive form model. While the extensive form MIP model outperforms the Benders Decomposition model for smaller numbers of scenarios, we would expect the Benders model to exceed the extensive form model for larger problem instances due to the benefit of decomposing the large master problem. Unfortunately, the constraint programming model showed very poor performance, likely due to the large number of variables in the extensive form model and the poor domain inferencing that could a result of a lack of any global constraints. The results from this paper can be applied to better protect against uncertainty with medical evacuations as Benders decomposition models can consider a larger number of possible scenarios.

As this paper considers a two-stage stochastic optimization problem, a promising future direction could consider a multistage approach. This would apply more realism to the optimization model as you can consider the sequential occurrence of military operations in a given region, considering the impact of casualty routing in later stages. Additionally, it would be beneficial to change the model from a routing model to a scheduling based model, which would have a fleet of evacuation vehicles and crews that you must schedule and route between evacuation sites, casualty locations, and medical sites. After evacuating casualties, these vehicles and crews could be rescheduled to serve other casualty locations. This problem could be seen as a combination of vehicle routing problems for each evacuation site.

References

1. Fulton, L.V., Lasdon, L.S., Reuben R. McDaniel, J., Coppola, M.N.: Two-stage stochastic optimization for the allocation of medical assets in steady-state combat operations. The Journal of Defense Modeling and Simulation **7**(2), 89–102 (2010). https://doi.org/10.1177/1548512910364390
2. Ragheb Rahmaniani, Teodor Gabriel Crainic, M.G.W.R.: The benders decomposition algorithm: A literature review. European Journal of Operational Research **259**, 17 (2017)