Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23)

Simulation-Assisted Optimization for Large-Scale Evacuation Planning with Congestion-Dependent Delays

Kazi Ashik Islam, **Da Qi Chen**, **Madhav Marathe**, **Henning Mortveit**, **Samarth Swarup** and **Anil Vullikanti**
Biocomplexity Institute, University of Virginia
{ki5hd, wny7gj, marathe, henning.mortveit, swarup, vsakumar}@virginia.edu

Abstract

Evacuation planning is a crucial part of disaster management. However, joint optimization of its two essential components, routing and scheduling, with objectives such as minimizing average evacuation time or evacuation completion time, is a computationally hard problem. To approach it, we present MIP-LNS, a scalable optimization method that utilizes heuristic search with mathematical optimization and can optimize a variety of objective functions. We also present the method MIP-LNS-SIM, where we combine agent-based simulation with MIP-LNS to estimate delays due to congestion, as well as, find optimized plans considering such delays. We use Harris County in Houston, Texas, as our study area. We show that, within a given time limit, MIP-LNS finds better solutions than existing methods in terms of three different metrics. However, when congestion dependent delay is considered, MIP-LNS-SIM outperforms MIP-LNS in multiple performance metrics. In addition, MIP-LNS-SIM has a significantly lower percent error in estimated evacuation completion time compared to MIP-LNS.

1 Introduction

Evacuation plans are essential to ensure the safety of people living in areas that are prone to disasters such as floods, hurricanes, tsunamis and wildfires. Large-scale evacuations have been carried out during the past hurricane seasons in Florida, Texas, Louisiana, and Mississippi. Examples of hurricanes when such evacuations were carried out include, Katrina & Rita (2005), Ike & Gustav (2008), Irma & Harvey (2017), Laura (2020), Ida (2021), and Ian (2022). For instance, about 2.5 million individuals were evacuated from the coastal areas of Texas [Carpender *et al.*, 2006] before the landfall of Hurricane Rita. The most recent category four hurricane, Ian, caused 119 deaths in the state of Florida alone [of Law Enforcement, 2022]. To ensure people can evacuate in a safe and orderly manner, a good evacuation plan needs to have two essential components: (i) Evacuation Routes, i.e. what roads to take, and (ii) Evacuation Schedule i.e. when to leave.

The focus of our paper is on *jointly optimizing the routes and schedules*. Informally, the idea is to find a schedule of when individuals can begin evacuation (within a given time window) and a route that would be used to evacuate, so as to minimize the objective functions capturing the system level evacuation time (see Section 3 for formal definition of the problems). Jointly optimizing over the routes and schedule is significantly harder from a computational standpoint (See Section 4 for hardness results). Existing methods, even those designed to find bounded sub-optimal solutions, do not scale to city or county level planning problems. Thus, finding good evacuation routes and schedule within a reasonable amount of time, for a city or county with a large population, remains an open problem.

Moreover, during evacuations, large number of people try to egress out of an area in a relatively small amount of time. This results in traffic congestion and huge delays in the evacuation process. It is crucial to consider and model such delays during the planning phase. However, most of the existing works on finding optimized evacuation plans do not consider the slowdown of traffic caused by high traffic density. For instance, [Even *et al.*, 2015; Romanski and Van Hentenryck, 2016; Hafiz Hasan and Van Hentenryck, 2021] all consider a constant travel time on each road, no matter how high (or low) the traffic density is on those roads. To overcome this, we treat the travel time on each road, as a parameter. We then utilize agent-based simulation, which is capable of modeling the complex relationship among the traffic density and speed on different roads, to learn the parameter values.

Our Contributions

First, we present MIP-LNS, a scalable optimization method that can find solutions to a class of evacuation planning problems, while optimizing for a variety of objectives (Section 5). It is designed based on the Large Neighborhood Search (LNS) framework. In this paper, we work with *three objectives*: minimizing (i) average evacuation time, (ii) evacuation completion time, and (iii) average evacuation time of 'non-outlier' evacuees. We show that all of these three optimization problems are hard to approximate within a logarithmic factor. In MIP-LNS, we model the problems as Mixed Integer Programs (MIP) and then find solutions to these programs using a combination of heuristic search and mathematical optimization. A key technical challenge involves modeling *flows over time*; this is best achieved using time expanded graphs. But it also leads to substantial increase in the size of the MIP and results in a significant increase in computing resources (time and space). This necessitates the need to combine heuristic search methods developed in the AI literature with MIP techniques developed in the OR literature.

Second, we illustrate how our approach can scale to large problem sizes and can be applied to realistic real-world problems. We choose Harris county in Houston, Texas as our study area and apply MIP-LNS. The county has about 1.5 million households, spans an area of $1,778$ square miles, and has been affected by several hurricanes in the past. We use real-world road network data and a synthetic population data to construct a realistic problem instance. *It is ten times larger (in terms of the size of the time expanded graph, and the number of evacuating vehicles) than the problem instance of our baseline [Hafiz Hasan and Van Hentenryck, 2021].* We show that, within a given time limit, MIP-LNS finds solutions for our problem instance that are on average 13%, 20.7% and 58.43% better than the baseline method in terms of average evacuation time, evacuation completion time and optimality guarantee of the solution, respectively (Section 7.2).

Finally, we present MIP-LNS-SIM, where we treat the travel time on each edge as a delay parameter and utilize agent-based simulation to learn the parameter values (Section 6). The MIP model, with the learned parameter values, is then solved to find optimized evacuation plans. Agent-based simulations provide a natural approach to capture the delays one incurs due to congestion – *dynamic flow problems cannot capture these delays.* Our approach is an example of methods that combine simulation and optimization methods (SO) [Gosavi and others, 2015; Amaran *et al.*, 2016] considered in OR and has become increasingly popular in AI [Van Hentenryck, 2013; Kambhampati, 2020; Doppa, 2021]. Through our experiments, we show that MIP-LNS-SIM outperforms MIP-LNS in terms of average evacuation time, evacuation completion time, and average time spent on the road ($10\%, 17\%, 77\%$ improvement respectively) when delay due to congestion is considered (Section 7.3). In addition, MIP-LNS-SIM has a significantly lower percent error (6%) in estimated evacuation completion time compared to MIP-LNS (76%), demonstrating the efficacy of MIP-LNS-SIM in evacuation planning subject to congestion constraints.

2 Related Work

Researchers have approached the evacuation planning problem in different ways in the past. [Hamacher and Tjandra, 2002] formulated it as a dynamic network flow optimization problem and introduced the idea of time expanded graphs to solve it using mathematical optimization methods. However, their method had prohibitively high computational cost, which paved the way to several heuristic methods [Lu *et al.*, 2005; Kim *et al.*, 2007; Shahabi and Wilson, 2014]. These methods solve the routing problem only – they either do not consider the scheduling problem at all or propose simple schemes such as letting evacuees leave at a constant rate. On the other hand, [Even *et al.*, 2015; Romanski and Van Hentenryck, 2016; Hafiz Hasan and Van Hentenryck, 2021] have considered the joint optimization problem of routing and scheduling. They formulated the problem as Mixed Integer Programs and used decomposition techniques [Benders, 1962; Magnanti and Wong, 1981] to separate the route selection and scheduling process. However, none of these works consider the slowdown of traffic at high traffic densities. A review of existing works on evacuation planning can be found in the survey paper [Bayram, 2016].

The use of convergent evacuation routes has been explored in the literature [Even *et al.*, 2015; Romanski and Van Hentenryck, 2016; Hafiz Hasan and Van Hentenryck, 2021], where all evacuees coming to an intersection follow the same path afterwards. This is also known as confluent flow [Chen *et al.*, 2006]. [Golin *et al.*, 2017] investigated the single-sink confluent quickest flow problem where the goal is minimizing the time required to send supplies from sources to a single sink. They showed that the problem is hard to approximate within a logarithm factor. We prove that all the planning problems considered in this paper are also hard to approximate.

We use the most recent method (Benders Convergent or BC) by [Hafiz Hasan and Van Hentenryck, 2021] as our baseline and show that MIP-LNS finds better solutions in terms of three different metrics. In addition, we provide direct MIP formulations for three different objectives, all of which can be optimized using MIP-LNS (as well as MIP-LNS-SIM) without needing any modifications.

Heuristic search methods are generally applied to problems that are computationally intractable. The goal is to find good solutions in a reasonable amount of time. The Large Neighborhood Search (LNS) framework [Shaw, 1998] has been successfully applied to various hard combinatorial optimization problems in the literature [Pisinger and Ropke, 2018]. Recently, [Li *et al.*, 2021] applied the LNS framework to find solutions for the Multi-Agent Path Finding Problem where the goal is to find collision free paths for multiple agents.

Simulation models have been used in the literature for finding optimal decision variables for a given objective function [Dangelmaier *et al.*, 2006; Sajedinejad *et al.*, 2011; Osorio and Bierlaire, 2013; Teufl *et al.*, 2018]. This is useful especially when the objective function's closed form is unknown or is too complex, but the function's value can be evaluated through (possibly time-expensive) simulation. In such cases, simulation models have been utilized as fitness functions within heuristic search and meta-heuristic algorithms [Sajedinejad *et al.*, 2011; Teufl *et al.*, 2018]. Existing research works have also proposed constructing a representative function, often termed as a *metamodel*, of the actual objective function by using its calculated values from the simulation model [Osorio and Bierlaire, 2013; do Amaral *et al.*, 2022]. Parameters of the metamodel are learned by methods such as regression, artificial neural networks [Gosavi and others, 2015]. Metamodels are mainly constructed because it is possible or easier to optimize these models. To fulfill our goal to find optimized evacuation plans considering congestion-dependent delays, we use our MIP model as a metamodel and use agent-based simulation to learn its parameter values. Both parameter learning and metamodel optimization are performed within MIP-LNS-SIM.

3 Problem Formulation

In this section, we introduce some preliminary terms that we use in our problem formulation.

Definition 3.1. A *road network* is a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ where every edge $e \in \mathcal{A}$ has (i) a capacity c_e, representing the number of vehicles that can enter the edge at a given time and (ii) a travel time T_e representing the time it takes to traverse the edge.

Definition 3.2. Given a road network, a *single dynamic flow* is a flow f along a single path with timestamps a_v, representing the arrival time of the flow at vertex v that obeys the travel times. In other words, $a_v - a_u \geq T_{uv}$ for edge (u, v). A *valid dynamic flow* is a collection of single dynamic flows where no edge at any point in time exceeds its edge capacity.

Definition 3.3. An *evacuation network* is a road network that specifies $\mathcal{E}, \mathcal{S}, \mathcal{T} \subset \mathcal{N}$, representing a set of source, safe and transit nodes respectively. Furthermore, for each source node $k \in \mathcal{E}$, let $W(k)$ and d_k represent the set of evacuees and the number of evacuees at source k respectively. Let W denote the set of all evacuees.

For scheduling an evacuation, we observe that once an evacuee has left their home, it is difficult for them to pause until they reach their destination. We also assume that people from the same location evacuate to the same destination. Similarly, we assume that if two evacuation routes meet, they should both be directed to continue to the same location.

Definition 3.4. Given an evacuation network, we say a valid dynamic flow is an *evacuation schedule* if the following are satisfied:

- all evacuees end up at some safe node,
- no single dynamic flow has any intermediary wait-time (i.e. $a_v - a_u = T_{uv}$ and,
- the underlying flow (without considering time) is confluent, where if two single dynamic flows use the same vertex (possibly at different times), their underlying path afterwards is identical.

Two natural objectives to minimize are the average evacuation time of the evacuees and the evacuation completion time. To define these formally, let t_i denote the evacuation time of evacuee i. We then formally define the following problems:

Problem 1. Average Dynamic Confluent Flow Problem (A-DCFP). Given an evacuation network, let T_{max} represent an upper bound on evacuation time. Find an evacuation schedule such that all evacuees arrive at some safe node before time T_{max} while minimizing $\frac{1}{|W|} \sum_{i \in W} t_i$.

Problem 2. Completion Time Dynamic Confluent FLow Problem (CT-DCFP). Given an evacuation network, find an evacuation schedule such that all evacuees arrive at some safe node while minimizing $\max_{i \in W} t_i$.

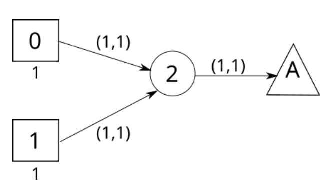
We define a third problem, Outlier Average Dynamic Confluent Flow Problem (O-DCFP), where the goal is to minimize average evacuation time of 'non-outlier' evacuees. For brevity, its formal definition is provided in the supplementary materials [Islam *et al.*, 2023].

3.1 Time Expanded Graph for Capturing Flow Over Time

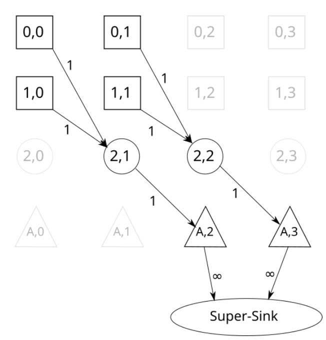
Joint routing and scheduling over networks requires one to study *flows over time*, as static flows make the unrealistic assumption that flows travel instantaneously (detailed discussion in the supplementary materials [Islam *et al.*, 2023]). For this purpose, researchers have defined dynamic flows ([Skutella, 2009; Ford and Fulkerson, 2015]) and used time expanded graphs to solve dynamic flow problems ([Romanski and Van Hentenryck, 2016; Hafiz Hasan and Van Hentenryck, 2021]). In this paper, we also use a time expanded graph (**TEG**) to capture the flow of evacuees over time.

Time expanded graph is a directed graph denoted by $\mathcal{G}^x = (\mathcal{N}^x = \mathcal{E}^x \cup \mathcal{T}^x \cup \mathcal{S}^x, \mathcal{A}^x)$. To construct it, we first fix a time horizon $\mathcal{H}$ and discretize the temporal domain into discrete timesteps of equal length. Then we create copies of each node at each timestep within $\mathcal{H}$. After that, for each edge $e(u, v)$ in the road network, we create edges in the TEG as $e_t(u_t, v_{t+T_e})$ for each $t \leq \mathcal{H} - T_e$ where the edges e_t have the same flow capacity as e. Finally, we add a super sink node v_t that connects to the nodes u_t for each $u \in \mathcal{S}$ and each $t \leq \mathcal{H}$. Edges to the super sink node are assigned an infinite amount of capacity. Note that, when creating the time expanded graph, we are adding an additional dimension (i.e. time) to the road network. *The size of the TEG is about $\mathcal{H}$ times as large as the road network in terms of number of nodes and edges.* – yielding a substantially larger problem representation.

A sample evacuation network and its corresponding TEG with time horizon $\mathcal{H} = 3$ are shown in Figure (1a-1b). The source, safe and transit nodes are denoted by squares, triangles, and circles respectively. In the TEG, there may be some nodes that are (i) not reachable from the source nodes, or (ii) no safe node can be reached from these nodes within the time horizon. These nodes are greyed out in Figure 1b. An optimal solution of A-DCFP (and CT-DCFP) for this sample problem instance is to use the routes $0 \rightarrow 2 \rightarrow A$ from source node 0 and $1 \rightarrow 2 \rightarrow A$ from source node 1, where the evacuee at source node 0 and 1 leave at timestep 1 and 0 respectively.



(a) Sample Evacuation Network. Edges are labeled with travel time and flow capacity respectively. Source, safe and transit nodes are denoted by squares, triangles, and circles respectively. Source nodes are labeled with number of evacuees.



(b) Time Expanded Graph (TEG) for the Sample Network. Edges are labeled with capacity. Construction of this TEG sets an upper bound of 3 time units for evacuation completion.

Figure 1: Sample Problem Instance

3.2 Mixed Integer Program (MIP) Model

Now, we present the Mixed Integer Program (1–8) that we use to represent a class of evacuation planning problems. We can have different objectives in the program (Objective 1), each representing a certain planning problem. We use two types of variables: (i) Binary variable $x_e, \forall e \in \mathcal{A}$, which will be equal to one if and only if the edge e is used for evacuation. Otherwise, it will be zero. (ii) Continuous variable $\phi_{e_t}, \forall e_t \in \mathcal{A}^x$, which denotes the flow of evacuees on edge e at timestep t.

Constraint (2) ensures that there is exactly one outgoing edge from each evacuation node. Constraint (3) ensures that at each transit node, there is at most one outgoing edge. Constraint (4) enforces that the total flow coming out of every evacuation node is equal to the number of evacuees at the corresponding node. Constraint (5) ensures flow conservation in the time-expanded graph; here, $\delta^-(i)$ and $\delta^+(i)$ denote the set of incoming and outgoing edges to/from node i, respectively. Constraint (6) allows flow on chosen edges only; it also enforces flow capacity on each edge of the time-expanded graph. Constraint (7) defines ϕ as continuous and non-negative variable; constraint (8) defines x as binary variable. The constraint that evacuation completion time needs to be less than the given upper bound is implicit in the model, as we set the time horizon of the TEG to this upper bound.

$$\text{Objective to Optimize} \tag{1}$$
$$s.t. \sum_{e \in \delta^+(k)} x_e = 1 \qquad \forall k \in \mathcal{E} \tag{2}$$
$$\sum_{e \in \delta^+(i)} x_e \leq 1 \qquad \forall i \in \mathcal{T} \tag{3}$$
$$\sum_{e \in \delta^+(k)} \sum_{t \leq \mathcal{H}} \phi_{e_t} = d_k \qquad \forall k \in \mathcal{E} \tag{4}$$
$$\sum_{e \in \delta^-(i)} \phi_e = \sum_{e \in \delta^+(i)} \phi_e \qquad \forall i \in \mathcal{T}^x \cup \mathcal{S}^x \tag{5}$$
$$\phi_{e_t} \leq x_e c_{e_t} \qquad \forall e \in \mathcal{A}, t \leq \mathcal{H} \tag{6}$$
$$\phi_e \geq 0 \qquad \forall e \in \mathcal{A}^x \tag{7}$$
$$x_e \in \{0, 1\} \qquad \forall e \in \mathcal{A} \tag{8}$$

To solve A-DCFP using model (1–8), we represent the total evacuation time using the variables x and ϕ as follows:

$$\text{Total Evacuation Time} = \sum_{e \in \delta^-(v_t)} \phi_e t_s(e) \tag{9}$$

Here, $t_s(e)$ denotes the timestep of the starting node of edge e. Note that, minimizing the average evacuation time and the total evacuation time are equivalent. So, the A-DCFP objective would be: $\min_{x,\phi} \sum_{e \in \delta^-(v_t)} \phi_e t_s(e)$.

We have just provided details on how to formulate A-DCFP as a MIP. Details on CT-DCFP and O-DCFP are provided in the supplementary materials [Islam *et al.*, 2023].

4 Inapproximability Results

In this section, we show that the problems we consider are not only NP-hard but also hard to approximate. A summary of the hardness results is found in Table 1.

Hardness	Problems		
	A-DCFP	CT-DCFP	O-DCFP
$O(\log n)$-hard to approx.	Thm. 1	See [Golin *et al.*, 2017]	Thm. 1

Table 1: Summary of Hardness

Theorem 1. *For* A-DCFP *and* O-DCFP *with many sources and one safe node, it is* NP-hard *to approximate within a factor of* $O(\log n)$.

The proof of Theorem 1 is provided in the supplementary materials [Islam *et al.*, 2023]. *In addition, we show that all the three problems remain* NP-hard *even when we consider the road network G to be a sub-graph of a grid and all destinations are along the border.* Street networks in several city neighborhoods resemble such networks.

5 Heuristic Optimization

In this section, we present the method MIP-LNS where we use MIP solvers in conjunction with heuristic search.

Within MIP-LNS, we first calculate an initial feasible solution in two steps: (i) calculating an initial convergent route set, and (ii) calculating the schedule that minimizes the target objective using the initial route set. For (i), we take a shortest path from each source to its nearest safe node by road. This is done using Algorithm 3 (in the supplementary materials [Islam *et al.*, 2023]) to make sure the route set is convergent. For (ii), we use the just calculated route set to fix the binary variables x_e in model (1-8). This gives us a linear program that can be solved optimally to get the schedule.

Next, we search for better solutions in the neighborhood of the solution at hand (Algorithm 1). Here, we run n iterations. In each iteration, we select $q = (100 - p)\%$ of sources uniformly at random and keep their routes fixed. This reduces

Algorithm 1: MIP-LNS Method

Input: Initial solution: *sol*, Time Expanded Graph: TEG, Time horizon: T, Model to optimize: $model$, (%) of routes to update: p, Number of Iterations: n, Positive number: p_{inc}
Output: Solution of $model$

```
for 1 to n do
    Select (100 − p)% of the source locations
      uniformly at random. Let their set be S.
    Fix the routes from the source locations in S. Set
      x_e = 1 if e is on any of the routes from S in sol.
    sol ← Solution of model from a MIP solver
    T′ ← evacuation completion time for solution sol
    if T − T′ > +threshold then
        Update the model by setting the time horizon
          to T′. Prune TEG and model by removing:
        (i) nodes that are unreachable from the
          evacuation nodes within time horizon T′, and
        (ii) nodes from which none of the safe nodes
          can be reached within time horizon T′
    p ← p + p_inc
return sol
```

the size of the MIP as we have fixed values for a subset of the variables. We then optimize the 'reduced' MIP model using a MIP solver [Gurobi Optimization, LLC, 2023]. Essentially, we are searching for a better solution in the neighborhood where the selected $q\%$ routes are already decided. Any solution found in the process will also be a feasible solution for the original problem. If we find a better solution with an evacuation completion time T' that is less than the current time horizon (T), then we also update the model by setting the time horizon to T'. When resetting the time horizon, we prune the TEG and the MIP model (lines 7–9). This reduces the number of variables in the MIP model and simplifies the constraints. At the end of each iteration, we increase the value of p by p_{inc} amount. Note that, when $p = 100$, we will be solving the original optimization problem. In our experiments, we start with $p = 75$ and set $p_{inc} = 0.5$.

When solving the reduced problem in each iteration (line 4), we use (i) a time limit, and (ii) a parameter $threshold\_gap$ to decide when to stop. MIP solvers keep track of an upper bound (Z_U) (provided by the current best solution) and a lower bound (Z_L) (obtained by solving relaxed LP problems) of the objective value. We stop the optimization when the relative gap $(Z_U - Z_L)/Z_U$ becomes smaller than the $threshold\_gap$. In our experiments, we set this to 5%. In total, MIP-LNS has four parameters: n, p, p_{inc}, and $threshold\_gap$.

6 Simulation-Assisted Model for Optimization (MIP-LNS-SIM)

In our formulation (Section 3), we assume the travel time on each edge to be a constant. However, in practice, travel time on a road is affected by the number of vehicles on it (i.e. traffic density). Moreover, travel time on an edge also affects how many cars can enter it in a given amount of time (i.e. the flow capacity) [Mannering and Washburn, 2020, Chapter 5]. We, therefore, treat the edge travel times as 'parameters' and aim to learn suitable values of these parameters to realistically model congestion-dependent delays.

To estimate the parameters, we use the agent-based queuing network simulation system QueST [Islam *et al.*, 2020] with the logistic traffic model. The simulator is able to model the complex relationship between traffic density and effective speed of vehicles on the road. Given the routes and schedule, we simulate the evacuation process using QueST and determine the average travel time on each edge used during evacuation. This provides us a reasonable estimate of travel time on the edges when certain routes and schedule are used. However, simulating the evacuation of the entire population is a time consuming task. Therefore, we only simulate the evacuation of a certain percentage (p_e) of the evacuees at each source. To be more precise, we simulate the departure of the first $p_e\%$ of the evacuees from each source, following the evacuation schedule. Our intuition is: congestion faced by the first $p_e\%$ of evacuees provides us a good estimation of the overall congestion faced by all evacuees throughout the entire evacuation. This is because people who leave first should not overlap too much with people who leave much later.

Based on the above idea, we present the method MIP-LNS-SIM (Algorithm 2). Initially, we assume that vehicles travel on each edge at the maximum speed allowed and calculate the travel time (i.e. the parameters) and flow capacity accordingly (line 1–3). We then create the time-expanded graph based on these values and construct the MIP (i.e. our *metamodel*). Next, We solve the MIP using MIP-LNS. We use the routes and schedule given by the solution to simulate the evacuation of first $p_e\%$ of the evacuees at each source. From the simulation results, we calculate the average travel time on each edge used in the solution and update the travel time as well as the flow capacity of these edges (details in supplementary materials [Islam *et al.*, 2023]). We do this iteratively for m times. In our experiments, we have used $p_e = 5, 10$ and $m = 10$. Note that, both parameter estimation (line 10) and the metamodel optimization (line 7) are performed within MIP-LNS-SIM.

Algorithm 2: MIP-LNS-SIM Method

Input: Evacuation network: $\mathcal{G}$, Initial solution: *sol*, Number of iterations: m, Percentage of Evacuees to simulate: p_e

Output: Evacuation routes and schedule.

1 **for** *each edge* $e \in \mathcal{A}$ **do**
2     $T_e \leftarrow$ Time it takes to traverse e at speed limit.
3     $c_e \leftarrow$ Updated flow capacity of e.
4 **for** 1 *to* m **do**
5     $TEG \leftarrow$ Time expanded graph of $\mathcal{G}$ with current travel time and capacity values of the edges.
6     $model \leftarrow$ MIP model (1–8) from $\mathcal{G}$ and TEG.
7     $sol \leftarrow$ Solution of $model$ from MIP-LNS.
8   Simulate evacuation of first $p_e\%$ evacuees at each source with routes and schedule from sol.
9   **for** *each edge* $e \in \mathcal{A}$ *used in sol* **do**
10         $T_e \leftarrow$ Avg. travel time on e from simulation.
11         $c_e \leftarrow$ Updated flow capacity of e.
12 **return** *sol*

7 Experiments

In this section, we present details of our problem instance and our experiment results.

7.1 Problem Instance

We use Harris County in Houston, Texas as our study area. We have used data from HERE maps [HERE, 2023] to construct its road network. The network contains roads of five (1 to 5) different function classes, which correspond to different types of roads. For instance, function class 1 roads are major highways, and function class 5 roads are residential roads.

For our experiments, we consider the nodes which connect and lead from function class 3/4 roads to function class 1/2 roads as the start/source locations of the evacuees. We then consider the problem of (i) when should evacuees target to enter the function class 1/2 roads and (ii) how to route them through the function class 1/2 roads to safely. As safe locations, we selected six locations at the periphery of Harris County which are on major roads. A visualization of the dataset is presented in Figure 2. Additional details regarding the problem instance are provided in Table 2.

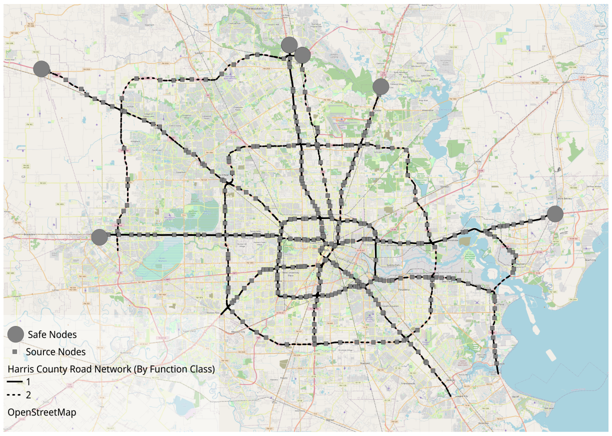


Figure 2: Harris County Problem Instance

# of nodes, edges in the road network	1338, 1751
# of (evacuee) source locations	374
# of Households in the study area	$\sim$ 1.5M
Time Horizon	15 Hours
Length of one time unit	2 minutes
# of nodes, edges in the TEG	684.7K, 841.6K
# of binary, continuous variables in A-DCFP MIP	1751, 843.7K
# of Constraints in A-DCFP MIP	1.4M

Table 2: Problem Instance Details

We use a synthetic population [Adiga *et al.*, 2015] to find the location of the households. We then assign the nearest exit ramp to each household as their source location. We assume that one vehicle is used per household for evacuation.

7.2 MIP-LNS Results and Baseline Comparisons

We performed all our experiments and subsequent analyses on a high-performance computing cluster, with 128GB RAM and 4 CPU cores allocated to our tasks. In addition to MIP-LNS, we used two more methods to solve A-DCFP. We used a time limit of one hour for each method and compared the best solutions found within this time. The three methods we experimented with are described here.

1. **Gurobi** In this experiment, we used Gurobi to directly solve model (1-8) with the A-DCFP objective. Gurobi was not able to find any feasible solution within the time limit. However, Gurobi was able to come up with a lower bound for the objective value. We used the lower bound to calculate the optimality guarantee of the solutions.
2. **Benders Decomposition** [Hafiz Hasan and Van Hentenryck, 2021] presented Benders Convergent (BC) method to solve the 'Convergent Evacuation Planning' problem. Their problem is similar to A-DCFP, differing in the objective function, which is maximizing flow of evacuees instead of minimizing average evacuation time. We repurposed their method and used it as our baseline.

Baseline	MIP-LNS		Improvement
	Avg.	Std. Dev.	Over Baseline (%)
Average evacuation time (hours)			
2.54	2.21	0.06	13
Evacuation completion time (hours)			
7.83	6.21	0.35	20.69
Optimality guarantee (%)			
20.47	8.51	2.43	58.43

Table 3: MIP-LNS results for A-DCFP over ten experiment runs and comparison with the baseline method in terms of three metrics: average evacuation time, evacuation completion time and optimality guarantee. On average, we see a $\sim 13\%$, $\sim 21\%$, and $\sim 58\%$ improvement in the three metrics respectively.

3. **MIP-LNS** In our experiments with MIP-LNS, for A-DCFP, we used thirty iterations (i.e. $n = 30$ in Algorithm 1 line 1). Also, since we have a random selection process within MIP-LNS, we ran ten experiment runs with different seeds.

To compare the quality of our solutions with the baseline, we use three metrics: average evacuation time, evacuation completion time, and optimality guarantee. *Optimality guarantee* is defined as follows: let the objective value of the solution *sol* be z_{sol} and the optimal objective be z_{opt}. Then, the optimality guarantee of *sol* is $(z_{sol} - z_{opt})/z_{sol}$, i.e. the smaller the value of optimality guarantee, the better. If z_{opt} is unknown, we can use a lower bound of it. Table 3 shows a comparison of our solutions with the baseline in terms of the three metrics.

Let the value of a metric m for the baseline and the MIP-LNS solution be m_{base} and m_{lns} respectively. Then, we quantify the improvement over the baseline as $(m_{base} - m_{lns})/m_{base}$. On average, we see an improvement of 13%, 21%, and 58% over the baseline in the three above-mentioned metrics respectively. This indicates that MIP-LNS finds better solutions than the baseline within the given time limit.

We also applied MIP-LNS to find solutions of CT-DCFP and O-DCFP for our problem instance. Due to limited space, we provide the results in the supplementary materials [Islam *et al.*, 2023]. In general, the experiment results show that MIP-LNS can effectively solve the problems with different objectives.

7.3 MIP-LNS-SIM Results

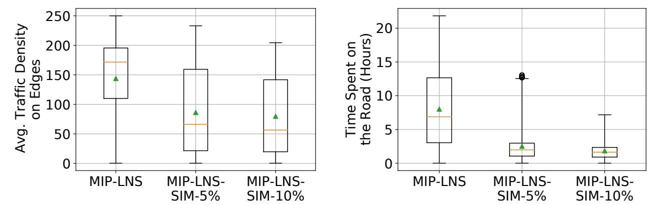
Within MIP-LNS-SIM, we set the parameter $m = 10$. We then experimented with two values for the parameter p_e, which are $5, 10$. We refer to these two settings as MIP-LNS-SIM-5% and MIP-LNS-SIM-10%. We ran MIP-LNS-SIM with the two settings and found two different solutions. We then performed agent-based simulations of the entire evacuation process (i.e. evacuate 100% of the evacuees) using a solution of MIP-LNS and then also using solutions from MIP-LNS-SIM-5% and MIP-LNS-SIM-10%. We used the

QueST simulator with the logistic traffic model here. We now compare the simulation results.

Figure (3a) shows the departure rate of the evacuees from their initial locations, in the final solution of the three settings. We observe that, MIP-LNS-SIM-5%, and MIP-LNS-SIM-10% regulates the departure of evacuees to a significant extent (compared to MIP-LNS). As evacuees leave late in these solutions, we might expect the evacuation completion time to be higher in these solutions compared to MIP-LNS. Surprisingly, we observe in Figure (3b) that the evacuation completion time is actually smaller in MIP-LNS-SIM-5%, and MIP-LNS-SIM-10% compared to MIP-LNS. This implies that although evacuees left early in the MIP-LNS solution, they could not reach safety early due to the resulting congestion on the roads. In the MIP-LNS-SIM-5% and MIP-LNS-SIM-10% solutions, evacuees departed from their initial location over a longer period of time. This way there was less congestion on the road and the evacuation was completed early even though many people started late.

Figure 4 verifies our last statement. We see that traffic density on the edges (i.e. number vehicles per lane and per km) is higher in the MIP-LNS solution (than MIP-LNS-SIM-5%, MIP-LNS-SIM-10%) throughout the evacuation time period. The higher traffic density then causes the evacuees to spend more time on the road. In summary, MIP-LNS-SIM is better than MIP-LNS in terms of average evacuation time, evacuation completion time, and average time spent on the road $(10\%, 17\%, 77\%$ improvement respectively, detailed results in the supplementary materials [Islam *et al.*, 2023]). These results indicate that MIP-LNS-SIM is better at evacuation planning than MIP-LNS in terms of reducing congestion on the roads.

Finally, Table 4 shows the estimated and the simulated evacuation completion time for Baseline [Hafiz Hasan and Van Hentenryck, 2021], MIP-LNS, MIP-LNS-SIM-5%, and MIP-LNS-SIM-10%. For instance, MIP-LNS predicts that when using the routes and schedule provided by its solution, the evacuation will be completed within 5.77 hours. However, when simulated, it actually took 24.29 hours to evacuate everyone. We also observe that the percent error in estimation decreases considerably in MIP-LNS-SIM-5% and MIP-LNS-SIM-10% where we have used $p_e = 5$ and 10 respectively. The lower percent error is earned at a cost of higher algorithm run time (MIP-LNS-SIM-5%: $\sim$ 6.5 hours, and MIP-LNS-SIM-10%: $\sim$ 7.55 hours).



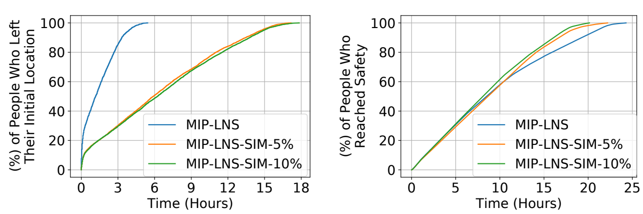
(a) Boxplots showing average traffic density (NO. of vehicles per lane per km) on edges over the evacuation time period. Average traffic density is considerably low in the MIP-LNS-SIM-10% and MIP-LNS-SIM-5% solutions compared to MIP-LNS.

(b) Boxplots showing time spent on the road by evacuees. Due to congestion (as in seen Figure 4a), evacuees spend a significantly larger amount of time on the road in the MIP-LNS solution, compared to MIP-LNS-SIM-10%, MIP-LNS-SIM-5% solutions.

Figure 4: Congestion on the roads in terms of traffic density and time spent on the road by evacuees.

Algorithm	Estimated ECT (hrs)	Simulated ECT (hrs)	Percent Error
Baseline	7.83	30.28	74.14
MIP-LNS	5.77	24.29	76.25
MIP-LNS-SIM-5%	18.43	22.2	16.98
MIP-LNS-SIM-10%	18.97	20.15	5.86

Table 4: Estimated and simulated Evacuation Completion Time (ECT) in hours for the three settings and the baseline. The percent error of the estimated ECT decreases significantly with $p_e = 5, 10$. This shows the effectiveness of MIP-LNS-SIM in capturing delays due to congestion.



(a) Departure rate of evacuees. We observe that in the MIP-LNS solution, evacuees leave very early compared to the solutions from MIP-LNS-SIM-5% and MIP-LNS-SIM-10%.

(b) Arrival rate of evacuees at safe locations. MIP-LNS-SIM-10% has the best evacuation rate and evacuation completion time, followed by MIP-LNS-SIM-5% and then MIP-LNS.

Figure 3: Comparison of MIP-LNS, MIP-LNS-SIM-5%, and MIP-LNS-SIM-10% in terms of departure rate from sources and arrival rate at safe nodes. Even though MIP-LNS-SIM-5%, and MIP-LNS-SIM-10% regulates the departure of evacuees, they evacuate everyone faster than MIP-LNS.

8 Conclusion

In this paper, we have presented an optimization method MIP-LNS to solve a class of evacuation planning problems. We demonstrated its efficacy by applying it on our study area of Harris county, Houston, Texas. We showed that, for our problem instance, MIP-LNS finds better solutions than the baseline method in terms of three different metrics. We have also presented MIP-LNS-SIM to capture congestion-dependent delays. Through our experiments we have showed that MIP-LNS-SIM outperforms MIP-LNS in terms of multiple metrics when congestion-dependent delay is considered. Our method can be incorporated into disaster management systems for effective evacuation planning. Additionally, it can help assess social vulnerability[1] of regions.

[1]https://www.atsdr.cdc.gov/placeandhealth/svi/index.html

Acknowledgments

This work was partially supported by University of Virginia Strategic Investment Fund SIF160, the NSF Grants: CCF-1918656, OAC-1916805, RISE-2053013, and the NASA Grant 80NSSC22K1048.

References

[Adiga *et al.*, 2015] Abhijin Adiga, Aditya Agashe, Shaikh Arifuzzaman, Christopher L. Barrett, Richard J. Beckman, Keith R. Bisset, Jiangzhuo Chen, Youngyun Chungbaek, Stephen G. Eubank, Sandeep Gupta, Maleq Khan, Christopher J. Kuhlman, Eric Lofgren, Bryan L. Lewis, Achla Marathe, Madhav V. Marathe, Henning S. Mortveit, Eric Nordberg, Caitlin Rivers, Paula Stretz, Samarth Swarup, Amanda Wilson, and Dawen Xie. Generating a synthetic population of the United States. Technical Report NDSSL 15-009, Network Dynamics and Simulation Science Laboratory, 2015.

[Amaran *et al.*, 2016] Satyajith Amaran, Nikolaos V Sahinidis, Bikram Sharda, and Scott J Bury. Simulation optimization: a review of algorithms and applications. *Annals of Operations Research*, 240(1):351–380, 2016.

[Bayram, 2016] Vedat Bayram. Optimization models for large scale network evacuation planning and management: A literature review. *Surveys in Operations Research and Management Science*, 21(2):63–84, December 2016.

[Benders, 1962] Jacques F Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik*, 4(1):238–252, 1962.

[Carpender *et al.*, 2006] S Kay Carpender, Paul H Campbell, Barbara J Quiram, Joshua Frances, and Jill J Artzberger. Urban evacuations and rural america: lessons learned from hurricane rita. *Public Health Reports*, 121(6):775–779, 2006.

[Chen *et al.*, 2006] Jiangzhuo Chen, Rajmohan Rajaraman, and Ravi Sundaram. Meet and merge: Approximation algorithms for confluent flows. *Journal of Computer and System Sciences*, 72(3):468–489, May 2006.

[Dangelmaier *et al.*, 2006] Wilhelm Dangelmaier, Kiran R Mahajan, Thomas Seeger, Benjamin Klopper, and Mark Aufenanger. Simulation assisted optimization and real-time control aspects of flexible production systems subject to disturbances. In *Proceedings of the 2006 winter simulation conference*, pages 1785–1795. IEEE, 2006.

[do Amaral *et al.*, 2022] João Victor Soares do Amaral, José Arnaldo Barra Montevechi, Rafael de Carvalho Miranda, and Wilson Trigueiro de Sousa Junior. Metamodel-based simulation optimization: A systematic literature review. *Simulation Modelling Practice and Theory*, 114:102403, January 2022.

[Doppa, 2021] Janardhan Rao Doppa. Adaptive experimental design for optimizing combinatorial structures. In *IJCAI*, pages 4940–4945, 2021.

[Even *et al.*, 2015] Caroline Even, Victor Pillac, and Pascal Van Hentenryck. Convergent plans for large-scale evacuations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[Ford and Fulkerson, 2015] Lester Randolph Ford and Delbert Ray Fulkerson. Flows in networks. In *Flows in Networks*. Princeton university press, 2015.

[Golin *et al.*, 2017] Mordecai J. Golin, Hadi Khodabande, and Bo Qin. Non-approximability and polylogarithmic approximations of the single-sink unsplittable and confluent dynamic flow problems. In *ISAAC*, 2017.

[Gosavi and others, 2015] Abhijit Gosavi et al. *Simulation-based optimization*. Springer, 2015.

[Gurobi Optimization, LLC, 2023] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual. https://www.gurobi.com, 2023. Accessed: 2023-05-25.

[Hafiz Hasan and Van Hentenryck, 2021] Mohd Hafiz Hasan and Pascal Van Hentenryck. Large-scale zone-based evacuation planning—Part I: Models and algorithms. *Networks*, 77(1):127–145, 2021.

[Hamacher and Tjandra, 2002] H. Hamacher and S. Tjandra. Mathematical modeling of evacuation problems: A state of the art. *Pedestrian and Evacuation Dynamics*, 2002, 01 2002.

[HERE, 2023] HERE. HERE Map Data. https://www.here.com/platform/map-data, 2023. Accessed: 2023-05-25.

[Islam *et al.*, 2020] Kazi Ashik Islam, Madhav Marathe, Henning Mortveit, Samarth Swarup, and Anil Vullikanti. A simulation-based approach for large-scale evacuation planning. In *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, December 2020.

[Islam *et al.*, 2023] Kazi Ashik Islam, Da Qi Chen, Madhav Marathe, Henning Mortveit, Samarth Swarup, and Anil Vullikanti. Simulation-assisted optimization for large-scale evacuation planning with congestion-dependent delays, 2023. arXiv:2209.01535.

[Kambhampati, 2020] Subbarao Kambhampati. Challenges of human-aware AI systems: AAAI presidential address. *AI Magazine*, 41(3):3–17, 2020.

[Kim *et al.*, 2007] Sangho Kim, Betsy George, and Shashi Shekhar. Evacuation route planning: Scalable heuristics. In *Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems*, GIS '07, pages 20:1–20:8, New York, NY, USA, 2007. ACM.

[Li *et al.*, 2021] Jiaoyang Li, Zhe Chen, Daniel Harabor, Peter J. Stuckey, and Sven Koenig. Anytime multi-agent path finding via large neighborhood search. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, August 2021.

[Lu *et al.*, 2005] Qingsong Lu, Betsy George, and Shashi Shekhar. Capacity constrained routing algorithms for evacuation planning: A summary of results. In *Proceedings of the 9th International Conference on Advances in*

Spatial and Temporal Databases, SSTD'05, pages 291–307, Berlin, Heidelberg, 2005. Springer-Verlag.

[Magnanti and Wong, 1981] T. L. Magnanti and R. T. Wong. Accelerating benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, 29(3):464–484, June 1981.

[Mannering and Washburn, 2020] Fred L Mannering and Scott S Washburn. *Principles of highway engineering and traffic analysis*. John Wiley & Sons, 2020.

[of Law Enforcement, 2022] Florida Department of Law Enforcement. Update: Florida medical examiners commission hurricane ian deaths, Oct 2022. https://www.fdle.state.fl.us/News/2022/October/Update-Florida-Medical-Examiners-Commission-H-(15), [Online; Accessed 28 Oct 2022].

[Osorio and Bierlaire, 2013] Carolina Osorio and Michel Bierlaire. A simulation-based optimization framework for urban transportation problems. *Operations Research*, 61(6):1333–1345, December 2013.

[Pisinger and Ropke, 2018] David Pisinger and Stefan Ropke. Large neighborhood search. In *Handbook of Metaheuristics*, pages 99–127. Springer International Publishing, September 2018.

[Romanski and Van Hentenryck, 2016] Julia Romanski and Pascal Van Hentenryck. Benders decomposition for large-scale prescriptive evacuations. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[Sajedinejad *et al.*, 2011] Arman Sajedinejad, Soheil Mardani, Erfan Hasannayebi, S. Ahmad Reza Mir Mohammadi K., and Alireza Kabirian. SIMARAIL: Simulation based optimization software for scheduling railway network. In *Proceedings of the 2011 Winter Simulation Conference (WSC)*. IEEE, December 2011.

[Shahabi and Wilson, 2014] Kaveh Shahabi and John P. Wilson. CASPER: Intelligent capacity-aware evacuation routing. *Computers, Environment and Urban Systems*, 46:12–24, July 2014.

[Shaw, 1998] Paul Shaw. Using constraint programming and local search methods to solve vehicle routing problems. In *Principles and Practice of Constraint Programming — CP98*, pages 417–431. Springer Berlin Heidelberg, 1998.

[Skutella, 2009] Martin Skutella. An introduction to network flows over time. In *Research trends in combinatorial optimization*, pages 451–482. Springer, 2009.

[Teufl *et al.*, 2018] Sabine Teufl, Kayode Owa, Dirk Steinhauer, Elkin Castro, Graham Herries, Robert John, and Svetan Ratchev. Optimised–developing a state of the art system for production planning for industry 4.0 in the construction industry using simulation-based optimisation. *Transdisciplinary Engineering Methods for Social Innovation of Industry 4.0*, pages 731–740, 2018.

[Van Hentenryck, 2013] Pascal Van Hentenryck. Computational disaster management. In *Twenty-third international joint conference on artificial intelligence*, 2013.