# Ambulance Routing Problem with Dynamic Traffic

Sarthak Bhagat
(2016189)

# Objective

Minimize the total number of deaths out of a total of 1000 accidents on 100 nodes subject to numerous constraints:

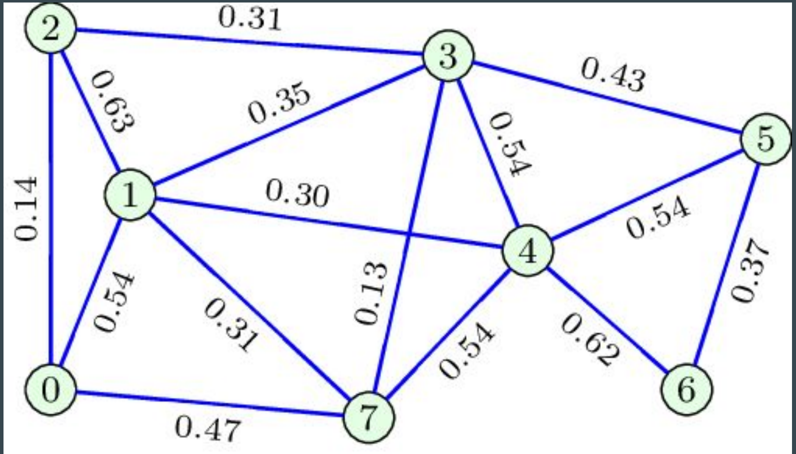
- Maximum capacity of each hospital is 10 patients and treatment time is 1800 seconds.
- Maximum capacity of each ambulance is one patient.
- The traffic on each edge (road) is dynamic and hence, we have a cap on the maximum speed ambulance can attain at each edge.
- Maximum number of ambulances available are 5.
- There are accident(s) occurring at each second on nodes only.
- Each patient has a certain time of survival (we have taken this time as) after accident and dies once it is not treated in that time.

# Procedure Followed

1. Construct a weighted graph using all nodes (and hospitals) and edges.
2. At each time step, find the shortest paths between each ambulance and all active casualties subject to various constraints using Bellman-Ford’s algorithm.
3. Assign casualties to ambulances using
   a. Naive Approach of assign closest accident to each ambulance (assigned to each hospital)
   b. An Advanced approach where all active casualties are divided among all ambulances to minimise cost of assignment.
4. Repeat process for all time steps.

# Constructing the Graph

- We create a weighted bidirectional graph by connecting all nodes between which the ambulance can navigate.
- We weigh the edges of the graph using the time taken to travel between them (and not the distance between them) in order to cater to the dynamic traffic problem as the shortest route between two nodes on the graph may vary depending on the traffic at that moment.
- Adjacency matrix is a matrix that has a 1 at all edges that have a connection while a 0 at ones that don't.



# Shortest path finding using Bellman-Ford Algorithm

- This a single source shortest path algorithm using a graph.
- This takes input as the graph and the source and returns shortest distance to all vertices from that source.
- This can also handle negative edges in the graph, in that case it doesn’t return the shortest path rather it reports the negative edge.
- Works in the case closed loops in the graph as well, and hence, this algorithm is suited for this kind of a traffic routing problems wherein there can be multiple routes between two different points. Also, the fact that we are applying this algorithm at each second helps us cater to the dynamic traffic problem as well, as at each step the shortest path may be different due to varying traffic constraints.

# Assigning accident nodes to each ambulance

- Naive Approach
  - In this approach, we only assign one casualty to each ambulance rather than dividing all active collisions among ambulances
  - At each step, we calculate the closest (distance to hospital included) accident to each ambulance using Bellman-Ford algorithm and assign it.
  - If two ambulances have the same closest ambulance then we pick one that minimises the total cost.
  - We have assigned each ambulance to each hospital
  - ***Drawback:***
    - This is not an optimal policy of assigning closest accident as it is dependant on the order of ambulance we pick and doesn't minimise the total cost of travel.
    - This assigns only one active accident to each ambulance at each time step and so doesn't give an optimal results as assigning multiple ambulances might be better in some cases.

- Advanced Approach
    - Assign each unassigned accident to all ambulances even if they already have been assigned one accident and check the cost of this assignment.
    - Choose the assignment that reduces the cost after this assignment.
    - Do this for all the active accident at time step.
    - Here, we don’t assign each hospital an ambulance.
    - *Drawback:* High computation cost over the naive approach.

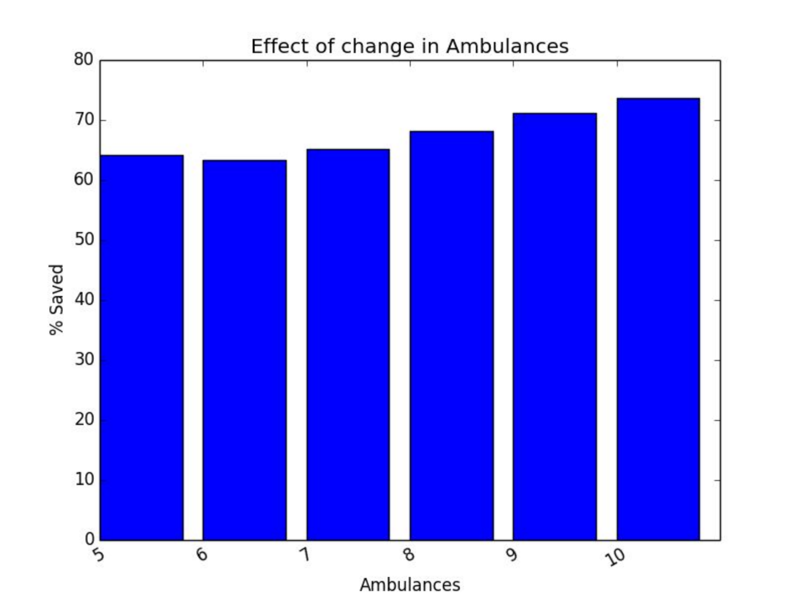
# Saved output

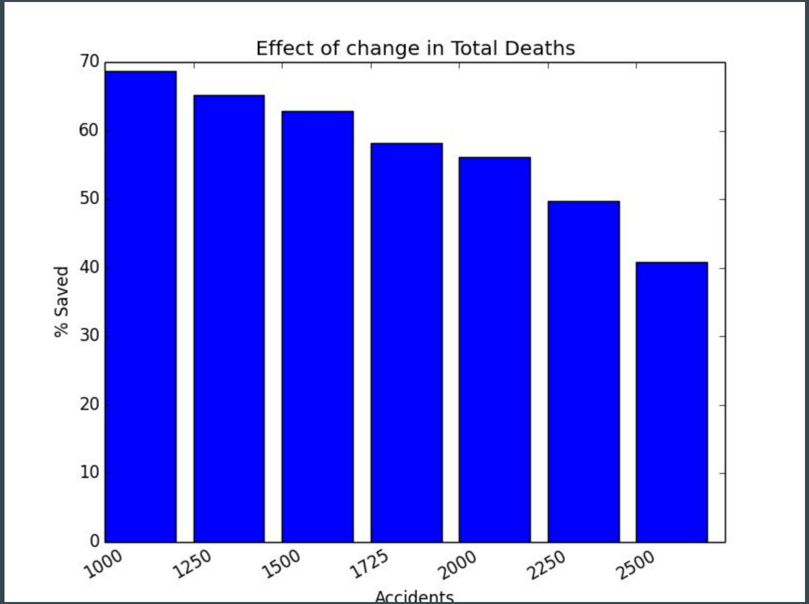
- Time step
- Active collisions at that time
- Trajectory for ambulance
- Ambulance assignment dictionary

Link to the text file

# Simulation Result

# Analysis Plots





These plots are generated using only a portion of the data rather than entire data of 2 days as computation took a lot of time.