Efficient Logistics for Minimizing Costs and Spoilage

AgriRoute

Optimized Multi-Stage Agriculture Supply Chain Design

Group - AG12

PROBLEM STATEMEMT

Data Structures and Inputs

Farm

- id: Unique identifier.
- location: Coordinates.
- produce_quantity
- perishability_window

StorageHub

- id: Unique identifier.
- location: Coordinates
- capacity
- storage_cost_per_unit
- fixed_usage_cost

Distribution Center

- id: Unique identifier.
- location: Coordinates
- demand
- deadline

Data Structures and Inputs

Network Routes

- Polyline : dictionary of route coordinates
- Cost matrix (optional or derived): If fuel cost or tolls vary per route.

Vehicle

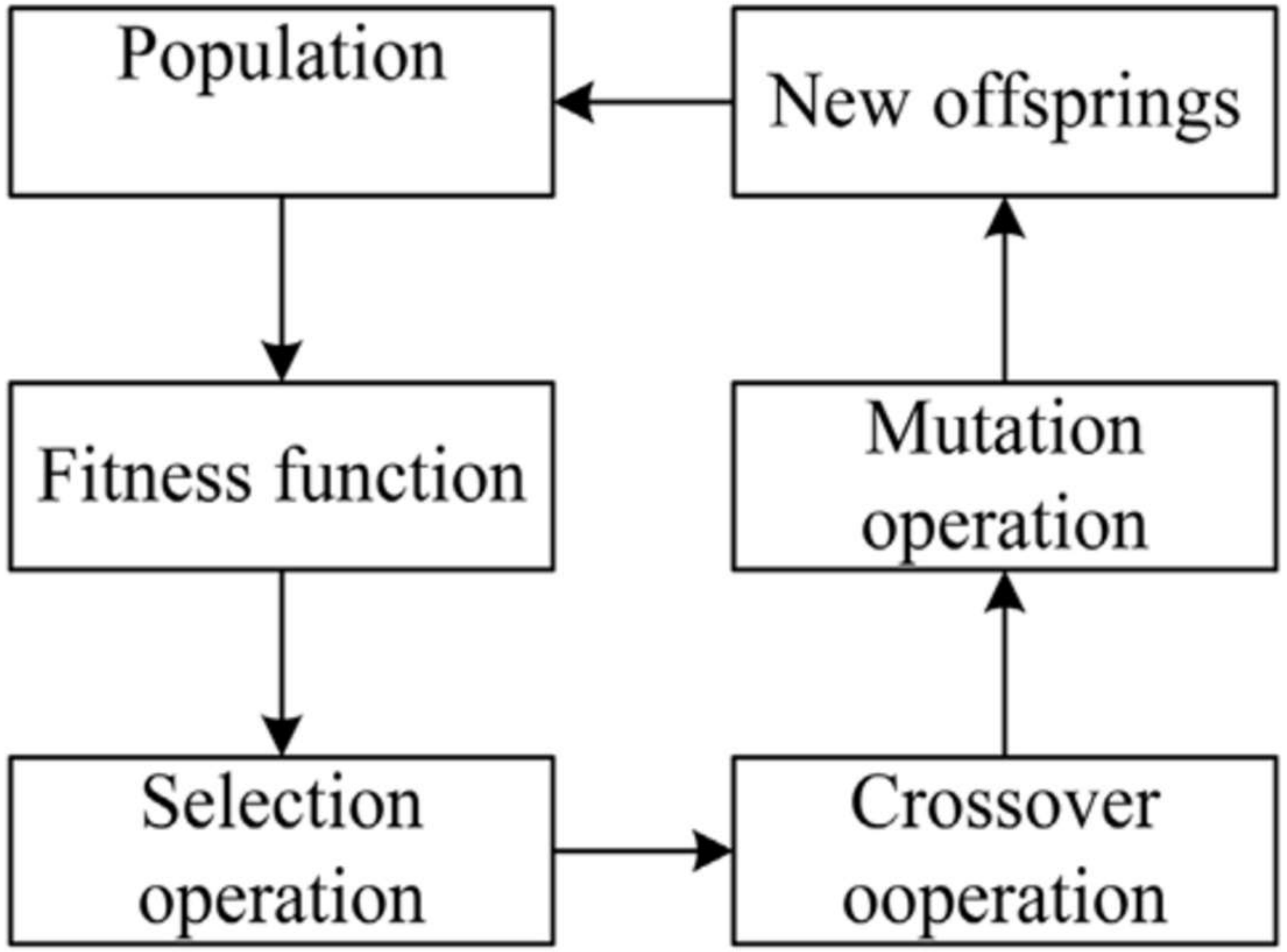
- type: "small" or "large"
- capacity
- variable_cost_per_distance:

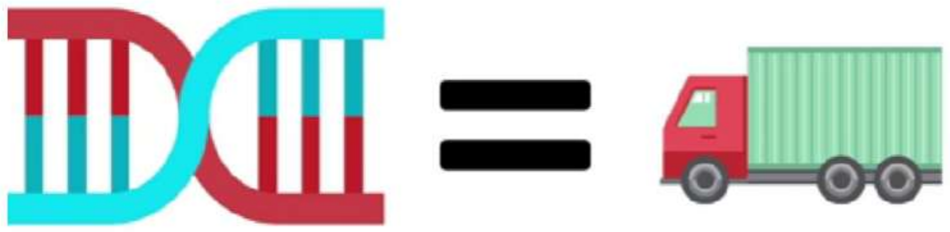
Dynamic Disruptions

- Road closures
- Traffic conditions
- Variable fuel costs

Glimpse of Genetic Algorithm

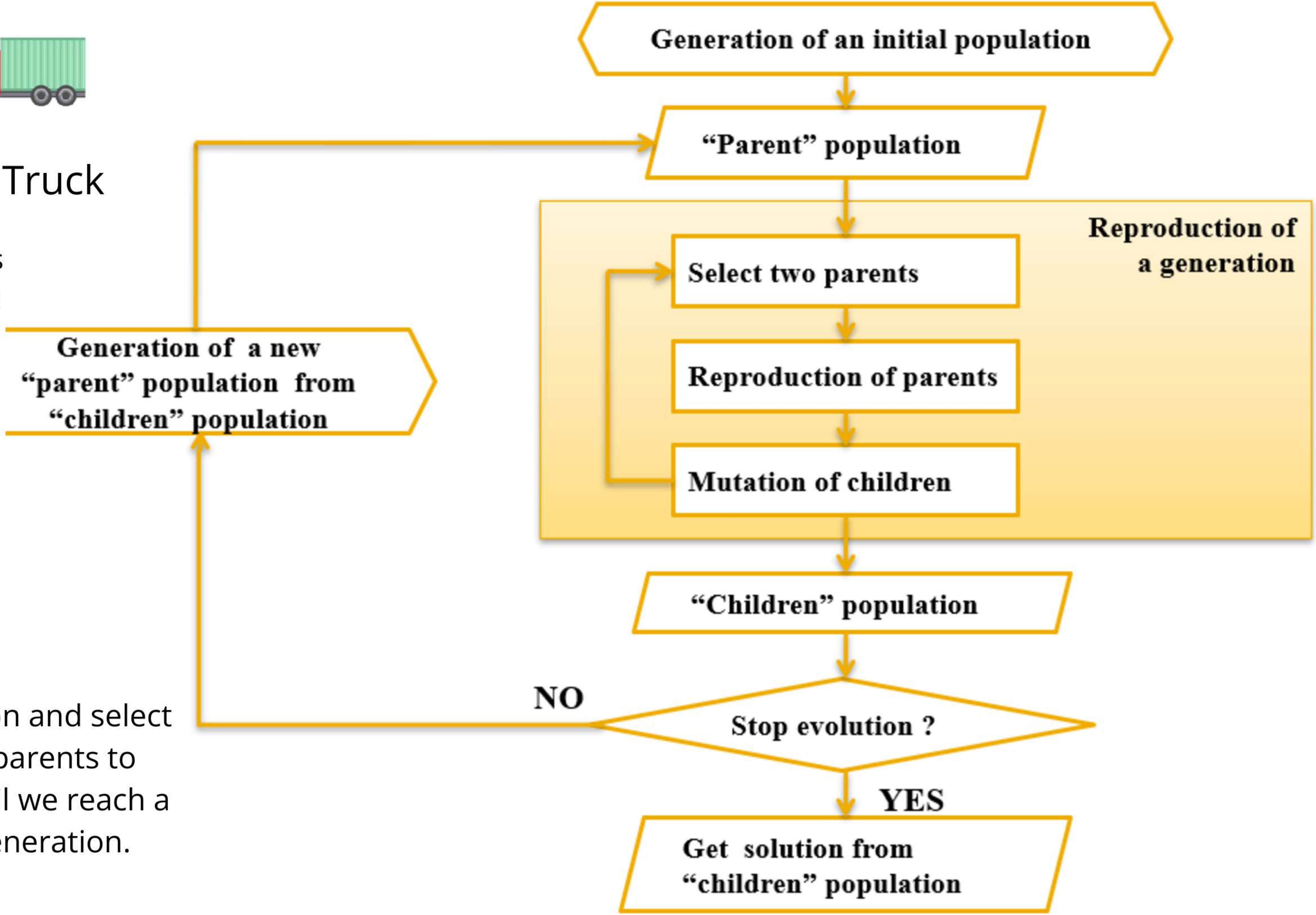
Introduction





chromosome ~ A Truck

..And the Genes in the chromosome is suppliers to be visited by the truck



We Initialize a population and select few of the strongest parents to generate offspring. Until we reach a certain number of generation.

Adaptive Genetic Algorithm

- Adaptive Crossover and Mutation Rates
- Roulette Wheel Selection
- Encoding & Population Initialization
- Constraint Handling

Adaptive Crossover and Mutation Rates

- **If an individual’s fitness is above average (a good solution):**
  A lower mutation rate is used to preserve its promising structure.
- **If an individual’s fitness is below average:**
  A higher mutation rate is employed to encourage exploration of new possibilities

Roulette Wheel Selection Based on Reciprocal Fitness

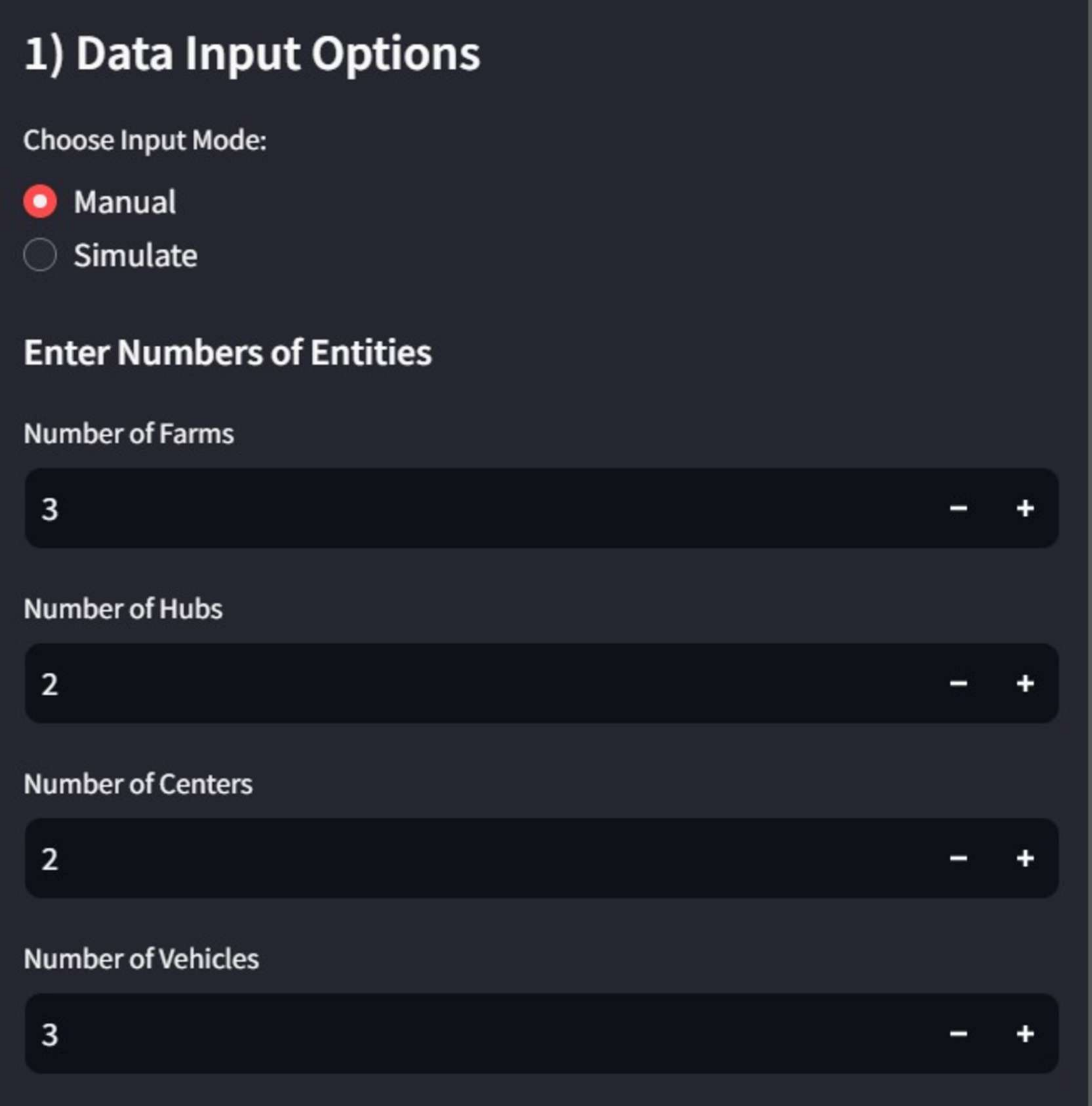
Improved GA:
Uses a reciprocal construction of the cost function, ensuring that lower–cost (higher–quality) solutions have higher fitness values and hence a greater chance of being selected. This method improves convergence toward lower-cost solutions.

Encoding & Population Initialization

- The improved GA uses natural number encoding for each shipment
- The initial population is generated randomly to cover a wide range of possibilities, which is crucial for diverse search in the solution space.

Simulation and Dataset Generator


```
4 : {
  "farm_id" : 6
  "hub_id" : 1
  "vehicle_id" : 3
  "assigned_qty" : 532
  "distance" : 0.13697732724140071
  "cost_multiplier" : 1.2
  "transport_cost" : 0.16437279268968086
}
5 : {
  "farm_id" : 7
  "hub_id" : 1
  "vehicle_id" : 0
  "assigned_qty" : 130
```



Purpose:

Simulate real-world agricultural supply chain scenarios.

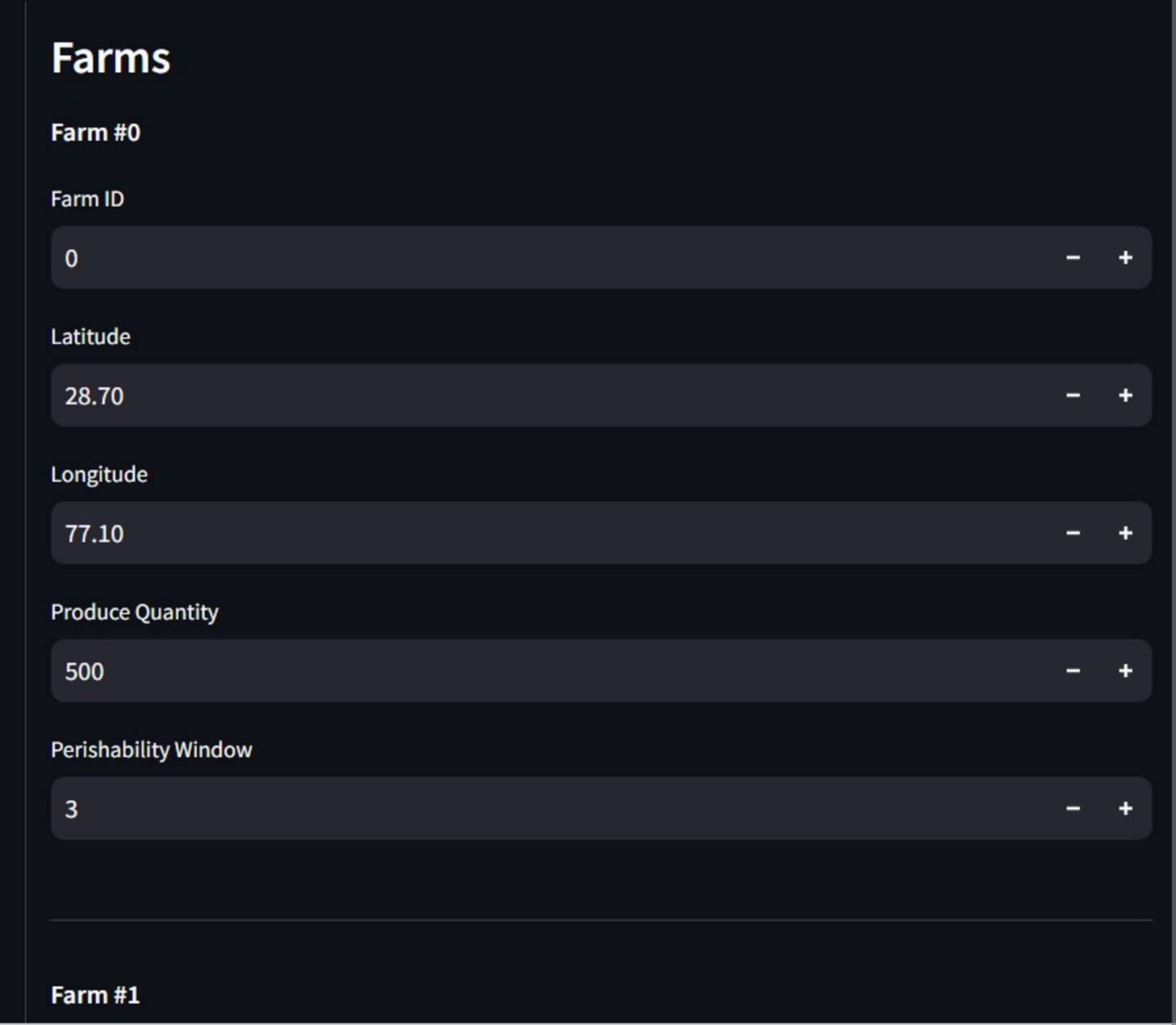
Features:

Adjustable parameters: Farm locations, vehicle capacities, perishability rates, demand levels.

Dynamic disruptions: Traffic conditions, road closures.

Deliverable:

Flexible tool for generating test datasets of varying complexity.



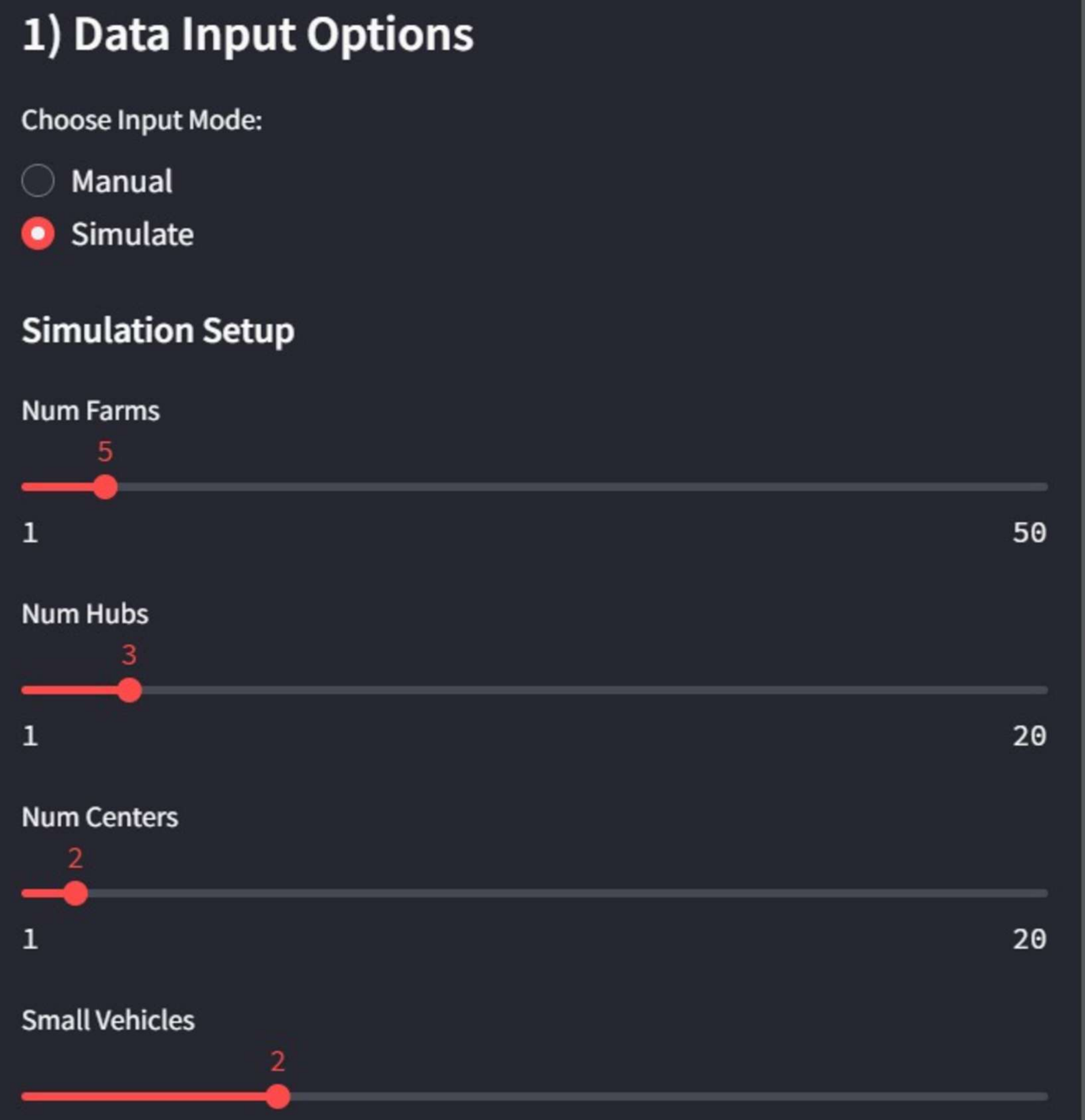
Manual Mode

In this mode, users can input data manually by specifying the following parameters:

Entities: Number of farms, hubs, distribution centers, and vehicles.

Farm Details: Farm ID, latitude, longitude, produce quantity, and perishable window.

The provided data is utilized to generate a comprehensive map of farms, hubs, and centers, alongside constructing a distance matrix for further analysis.



Simulate Mode

This mode allows users to generate realistic data sets using adjustable parameters:

Sliders: Define the number of farms, hubs, distribution centers, small vehicles, and large vehicles.

Data Generation: On selecting the "Simulate Data" option, the system generates realistic data sets, including farm details, distribution centers, storage hubs, and vehicle assignments.

The simulation mode simplifies the process of creating large-scale scenarios for testing and analysis, ensuring realistic and actionable data generation.

2) Current Data

Farms

	id	location	produ
0	0	28.7 77.1	
1	1	28.71 77.11	
2	2	28.72 77.11999999999999	

Distribution Centers

	id	location	dema
0	0	28.9 77.2	8
1	1	28.91 77.21000000000001	8

Vehicles

	id	type	capacity	fixed_cost	variabl
0	0	small	1,000	200	
1	1	small	1,000	200	
2	2	large	3,000	350	

Storage Hubs

	id	location	capa
0	0	28.6 77	2,
1	1	28.610000000000003 77.01	2,

Data Generation with Azure Maps
Integration of Azure Maps API:

Distance Matrix (sample):


```
[
  0 : [
    0 : [
      0 : "farm"
      1 : 0
      2 : "hub"
      3 : 0
    ]
    1 : [
      0 : {
        "route_id" : 1
        "distance_km" : 46.623
        "geometry" :
          [ 0 - 100 ]
          [ 100 - 200 ]
          [ 200 - 300 ]
          [ 300 - 400 ]
          [ 400 - 500 ]
          [ 500 - 600 ]
```


Generates coordinates for possible paths between farms, storage hubs, and distribution centers which are further fetched by our optimisation model to analyze possible routes based on dynamic constraints such as road closures and traffic conditions.

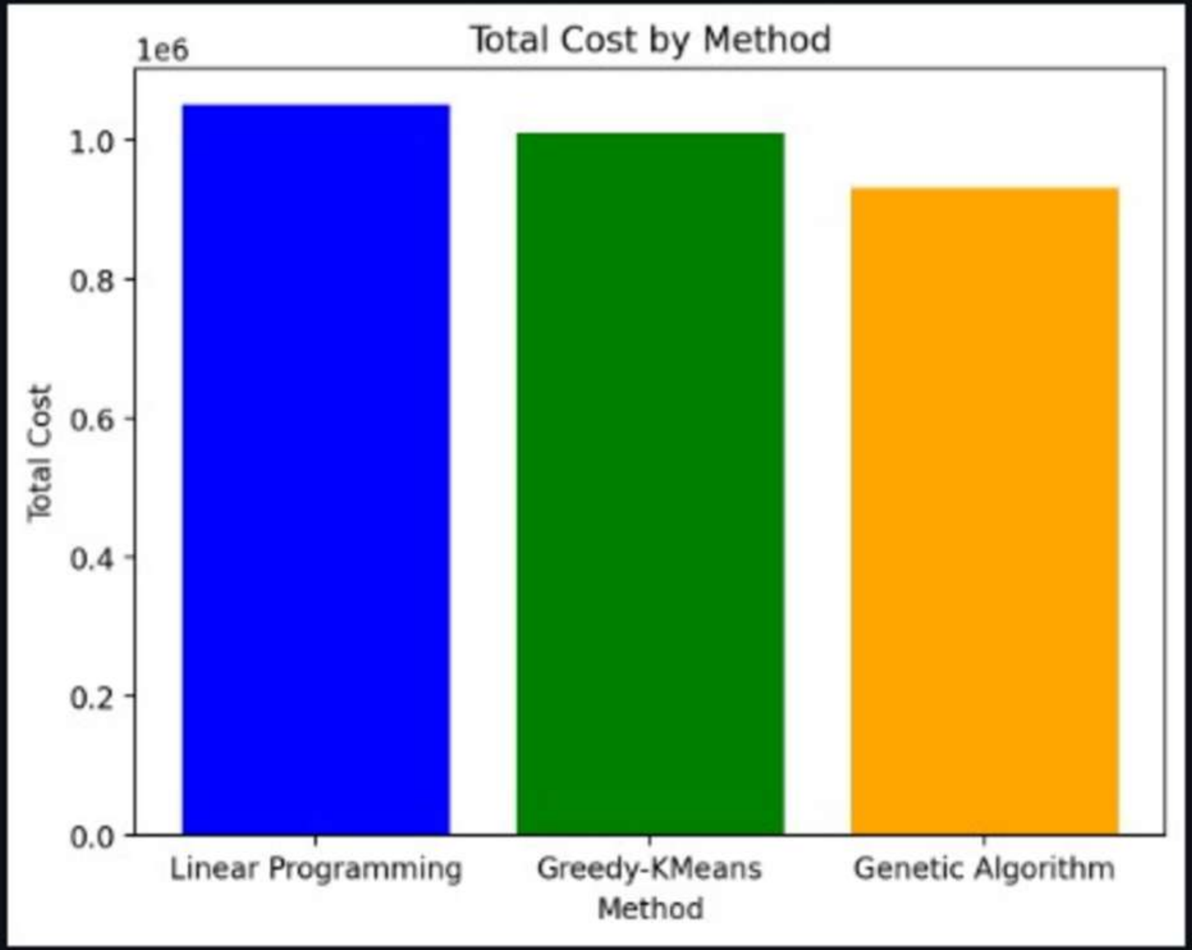
Performance Analysis

Comparison Data

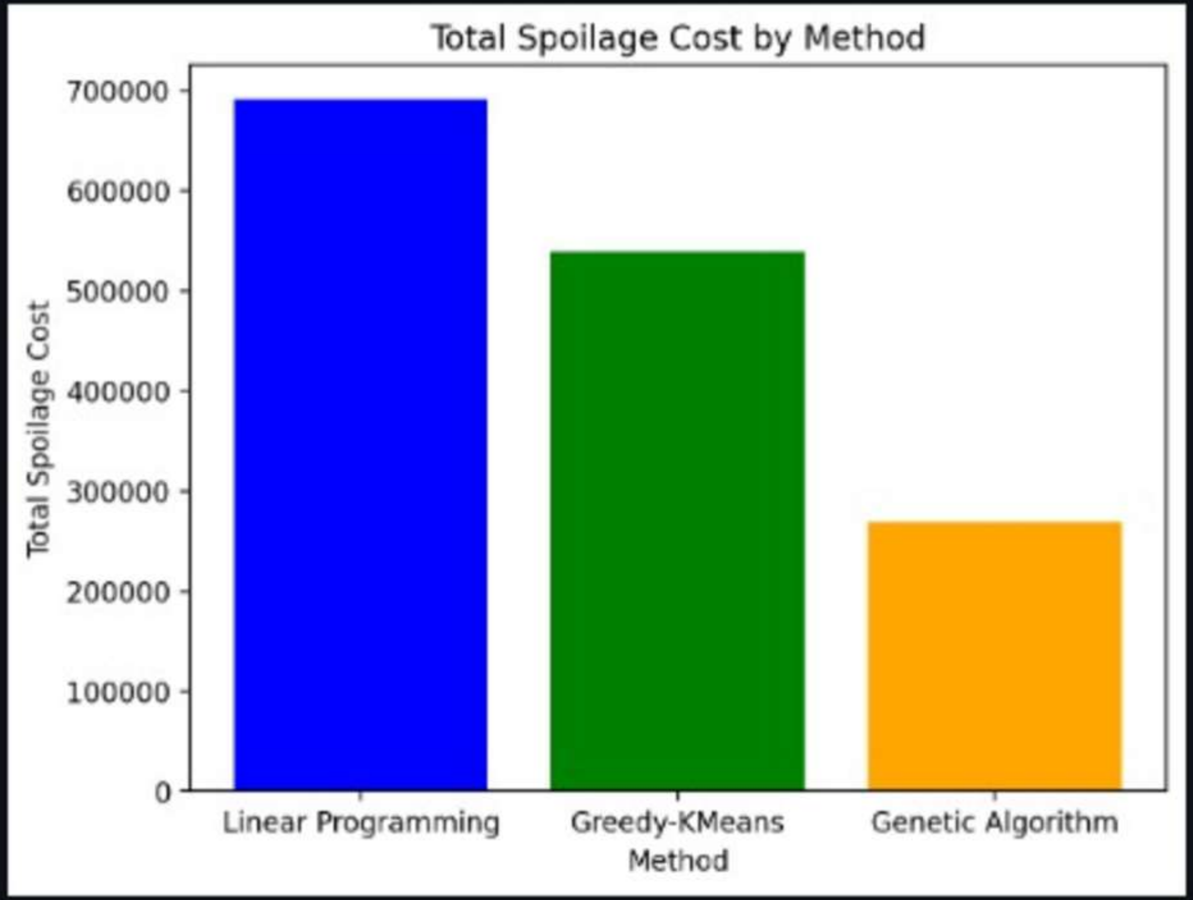
	Method	Total Cost	Total Spoilage Cost
0	Linear Programming	1,050,350	690,750
1	Greedy-KMeans	1,011,271.4	539,400
2	Genetic Algorithm	932,023.8	269,700

Cost Comparisons

Total Cost

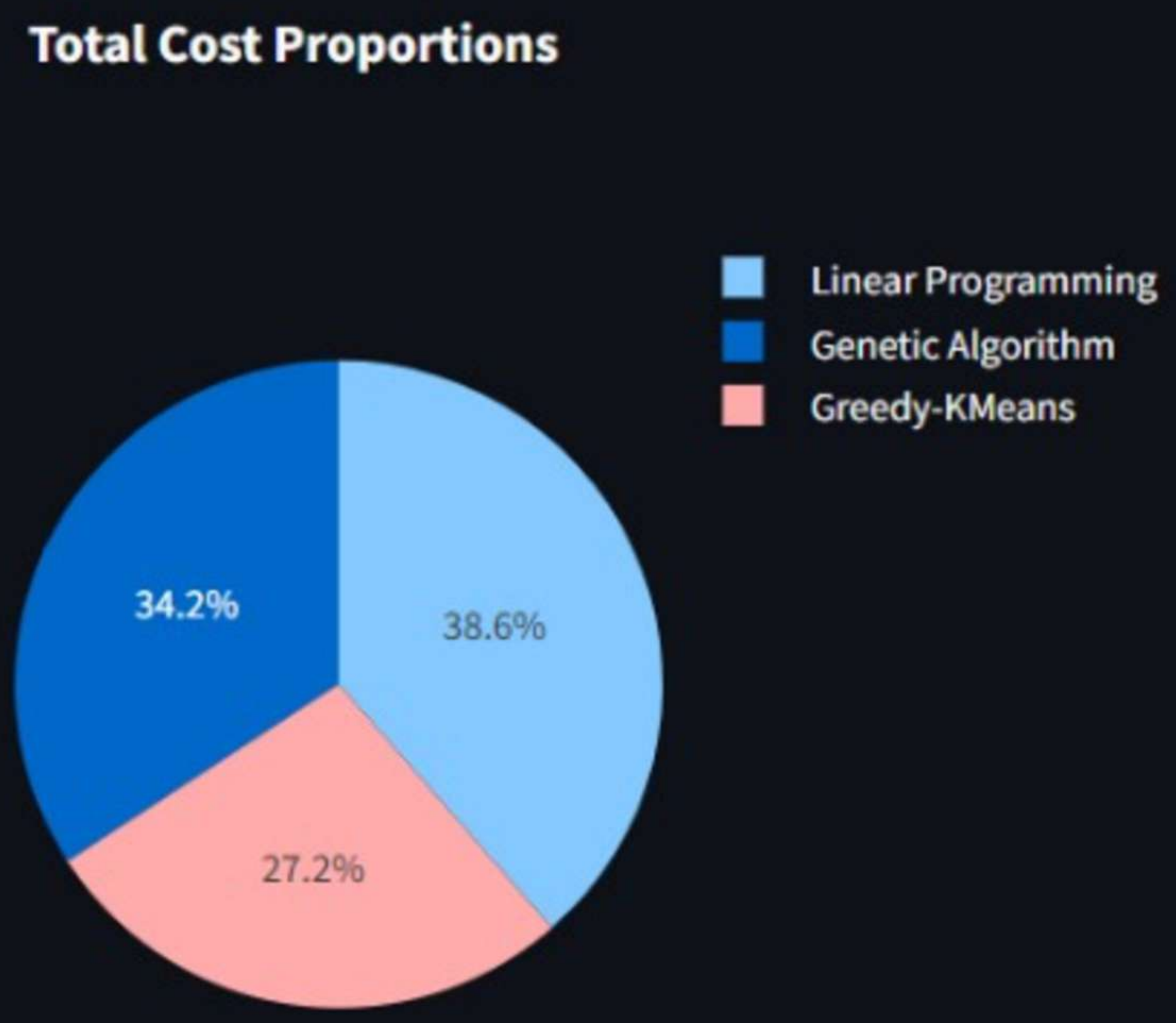


Total Spoilage Cost

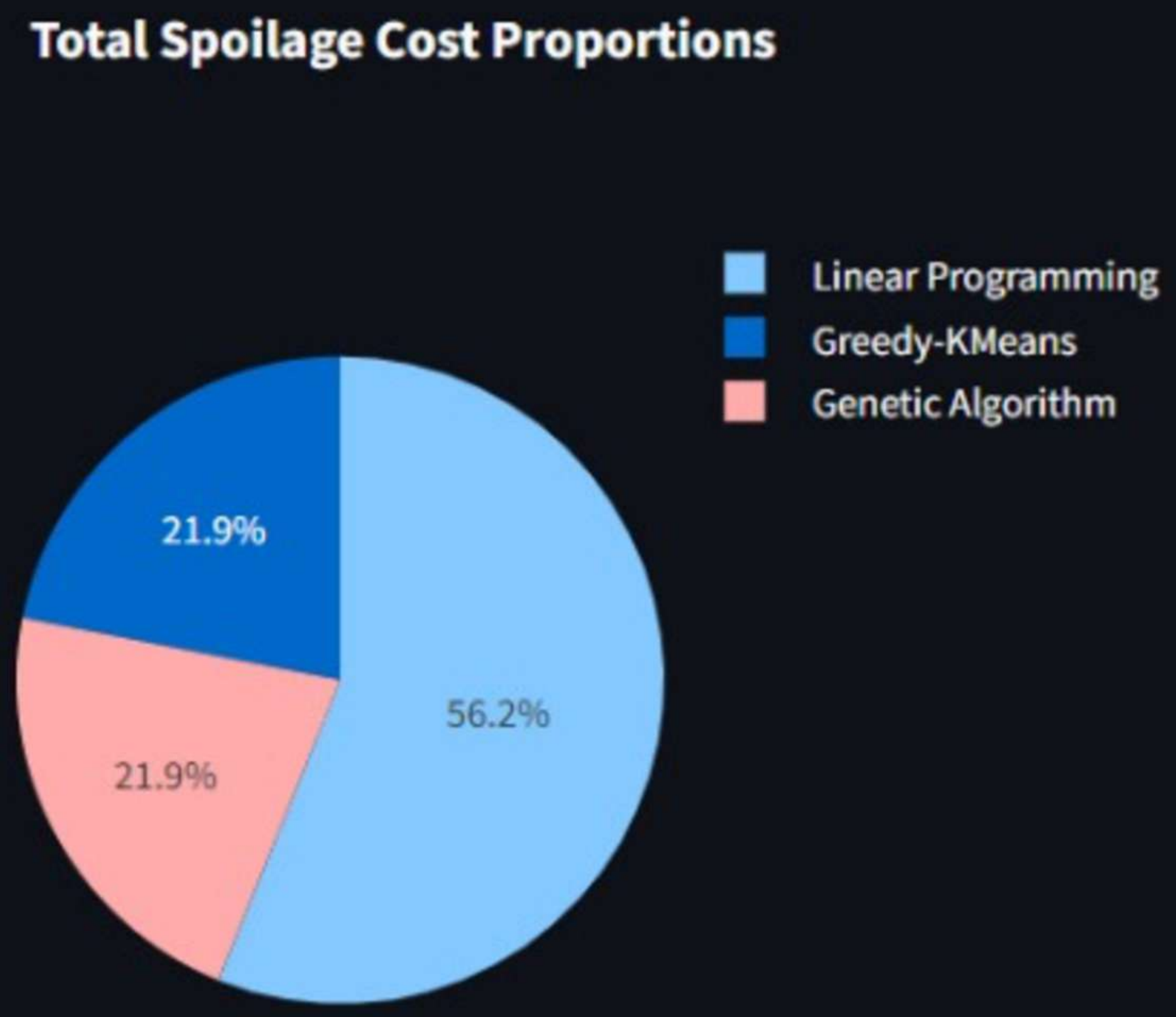


Proportional Breakdown

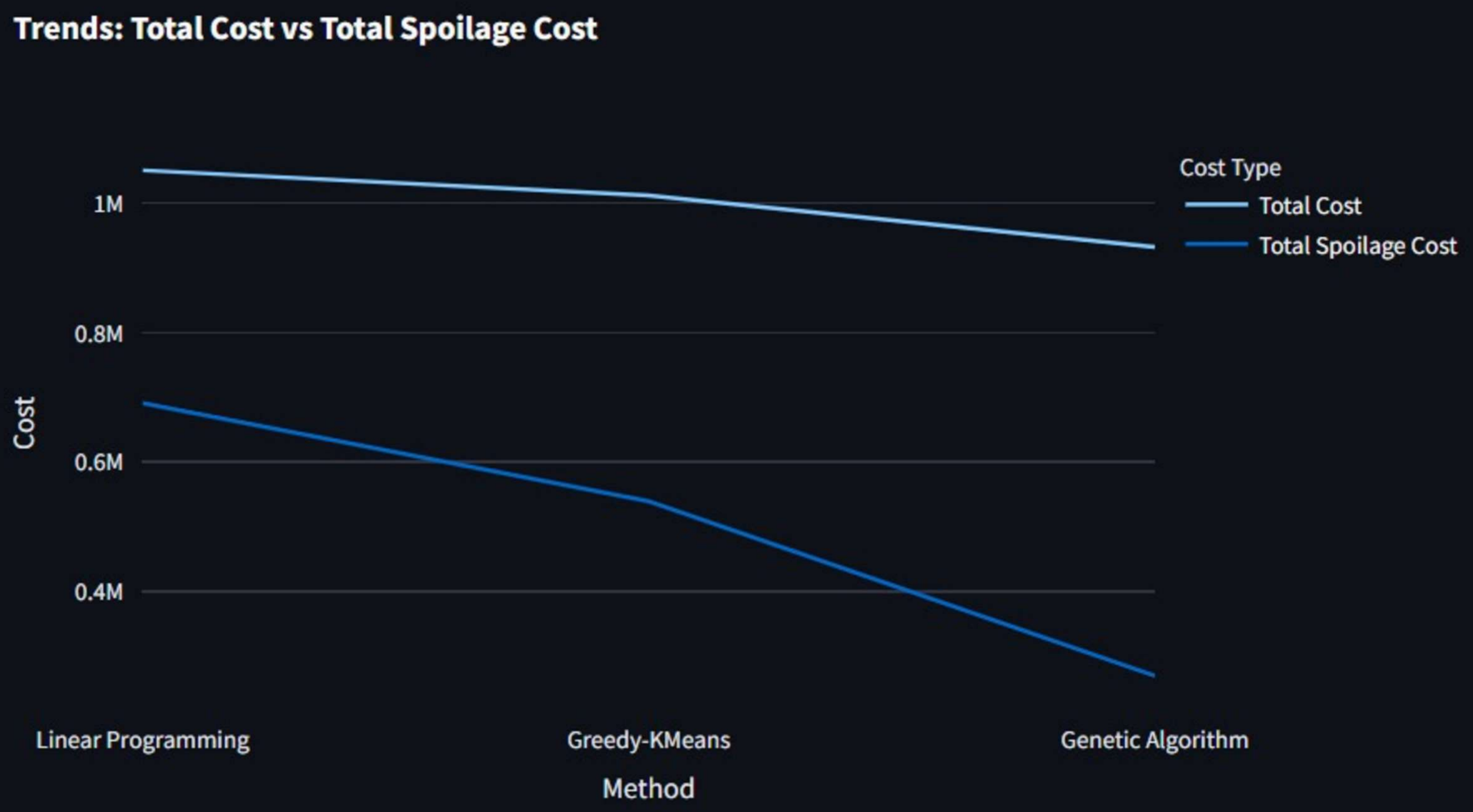
Proportion of Total Cost



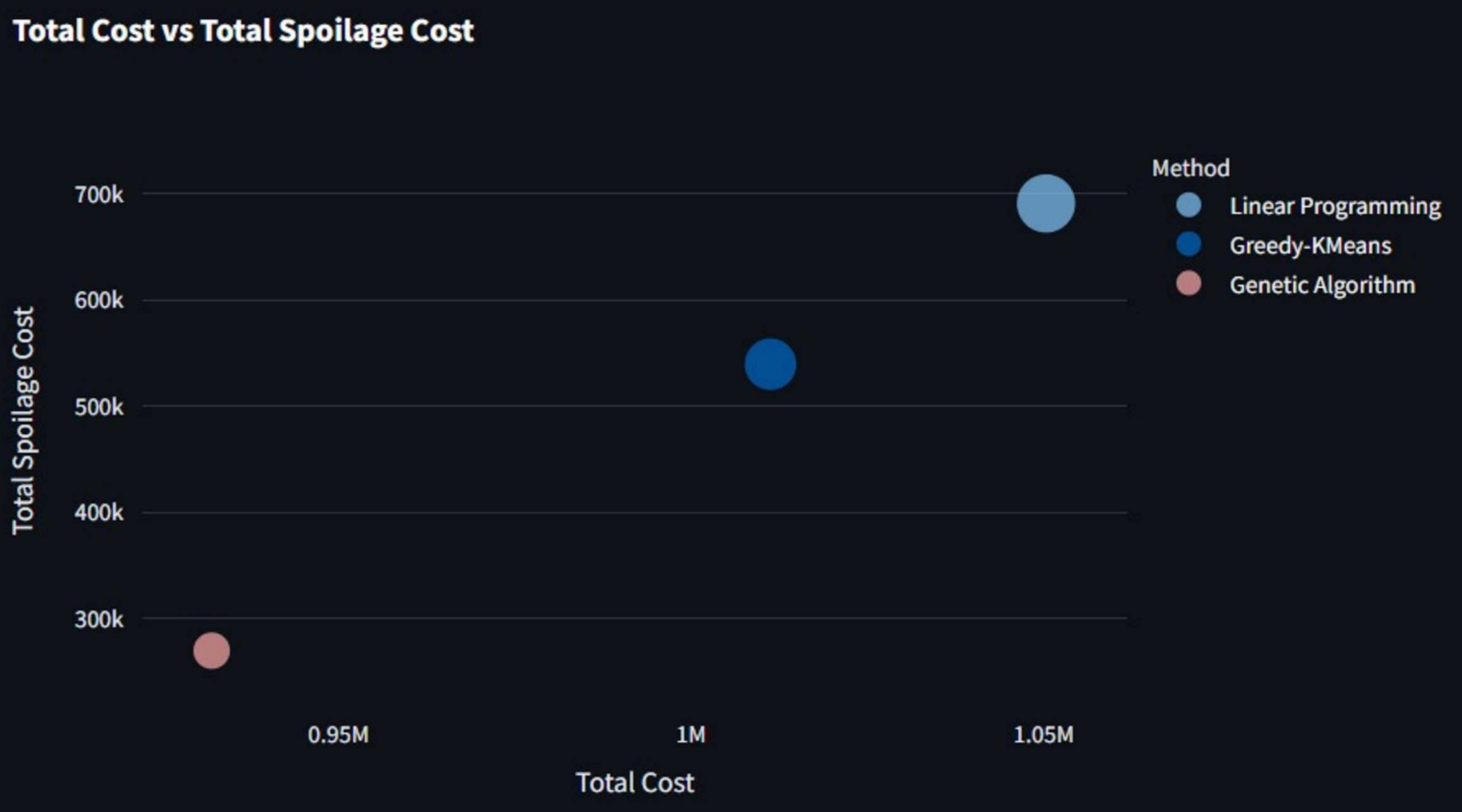
Proportion of Spoilage Cost



Trends Across Methods



Cost Relationships



- *Method:* Genetic Algorithm
- *Cost:* 932023.8

Lowest Total Spoilage Cost:

- *Method:* Genetic Algorithm
- *Spoilage Cost:* 269700.0

Highest Total Cost:

- *Method:* Linear Programming
- *Cost:* 1050350.0

Highest Total Spoilage Cost:

- *Method:* Linear Programming
- *Spoilage Cost:* 690750.0

Insights

Results & Visualisation

Greedy LP GA

Greedy Solution JSON:


```
{
  "method" : "Greedy-KMeans"
  "best_cost" : 1011271.4
  "total_spoilage_cost" : 539400
  "detailed_routes" :
    [ 0 - 100 ]
    [ 100 - 200 ]
    [ 200 - 300 ]
```

Results & Visualisation

Greedy LP GA

LP Solution JSON:


```
{
  "method" : "LinearProgramming"
  "best_cost" : 1050350
  "total_spoilage_cost" : 690750
  "detailed_routes" : []
  "runtime_sec" : 0.01
}
```

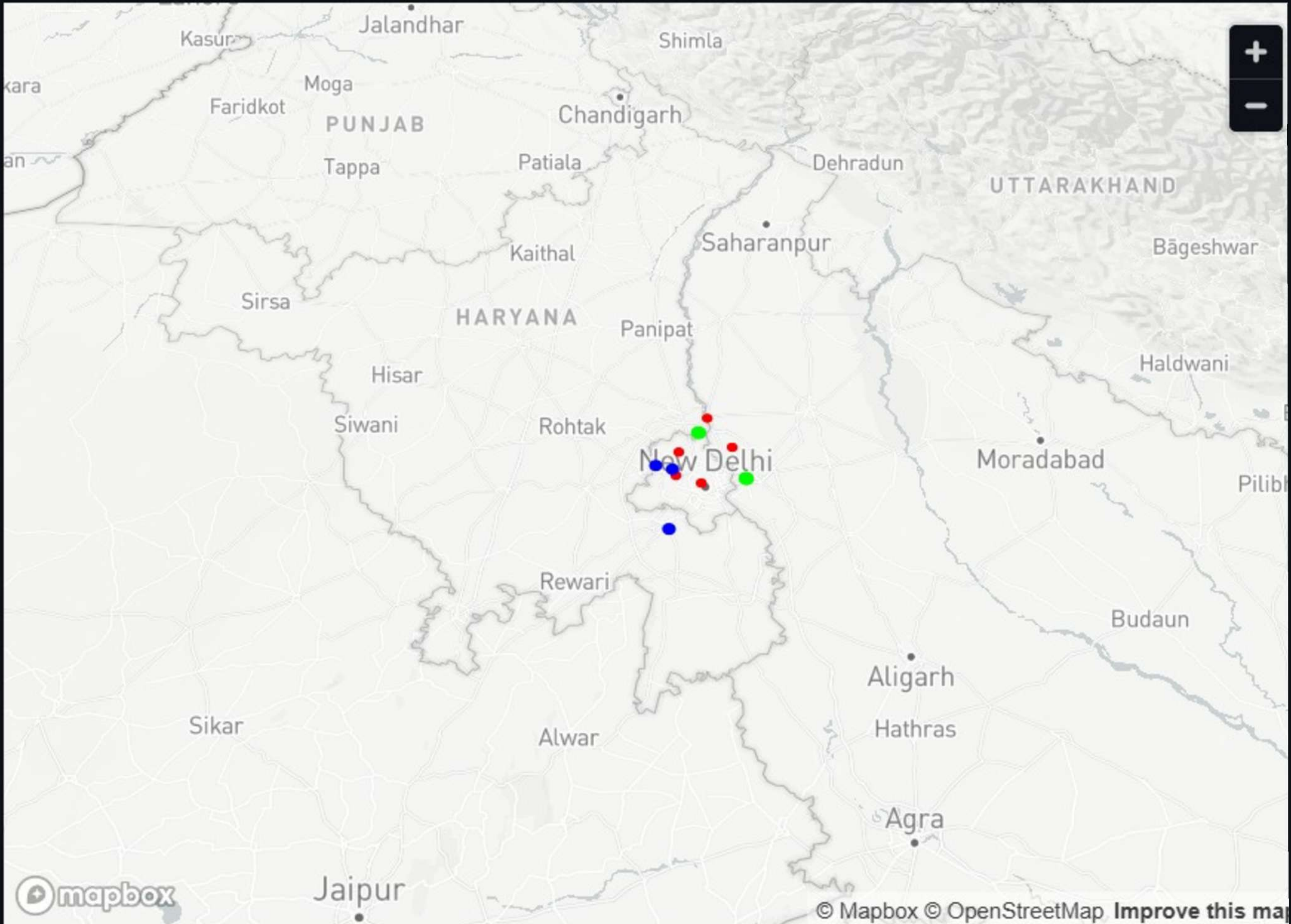
Results & Visualisation

Greedy LP GA

GA Solution JSON:


```
{
  "method" : "GeneticAlgorithm"
  "best_cost" : 932023.8
  "total_spoilage_cost" : 269700
  "detailed_routes" :
    [ 0 - 100 ]
    [ 100 - 200 ]
    [ 200 - 300 ]
    [ 300 - 400 ]
    [ 400 - 500 ]
```

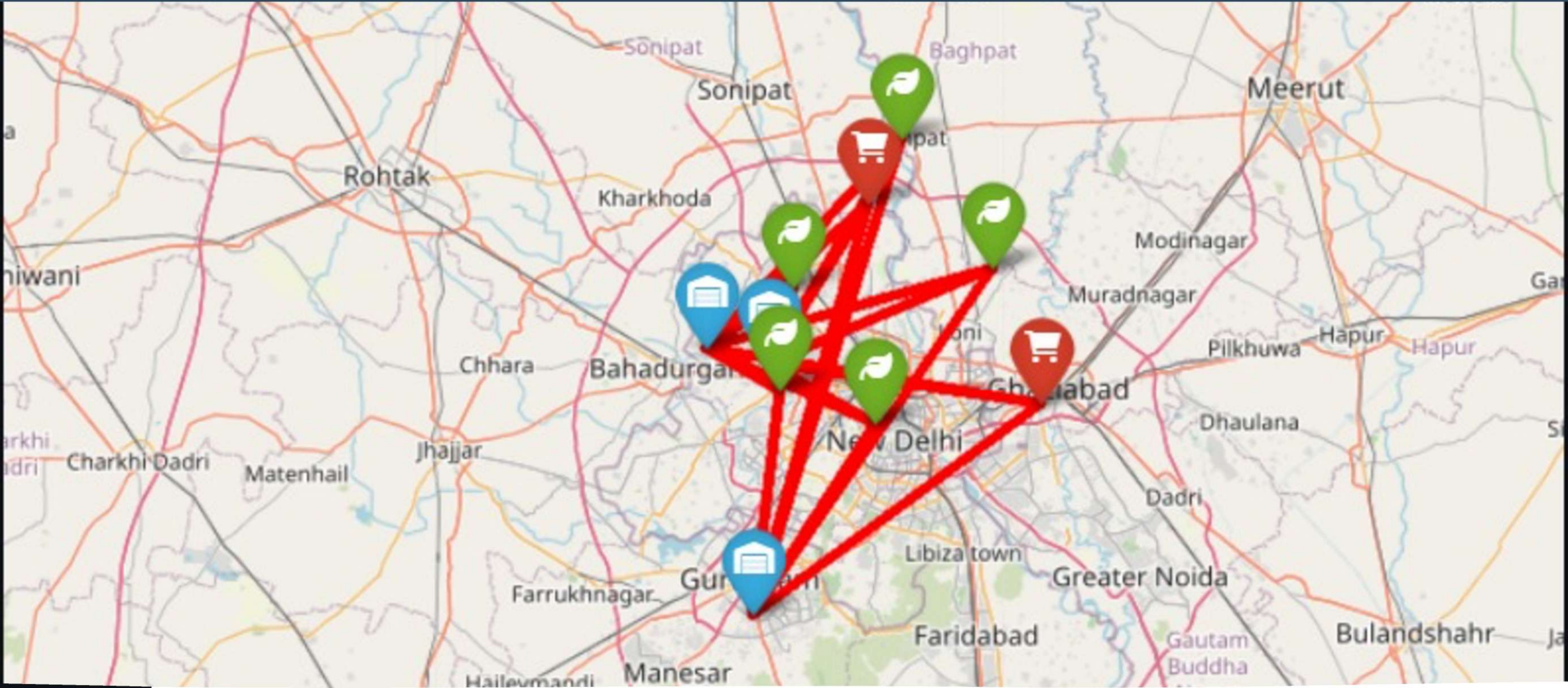
3) Map of Farms, Hubs, and Centers



Optional: Display Distance Map

Show Distances on Map

E-GA Visualisation




Thank You