Problem Definition:

We consider a **multi-modal urban transportation network** that includes **highways, railways, and subways** connecting various cities and stations. The objective is to **minimize the total transportation cost** while satisfying demand at each destination. The network has:

- **Multiple supply sources (factories, ports, warehouses)**
- **Multiple demand destinations (cities, distribution centers, retail stores)**
- **Multiple transportation modes (trucks, trains, subways)**
- **Capacity constraints for each transportation route**
- **Different transportation costs per unit for each mode**
- **Time windows for delivery requirements**
- **Congestion costs increasing with flow on each edge**

1. Nodes, Arcs, and Modes

- **Nodes (nodes)**: These represent locations in the transportation network.
  - `Factory1`, `Factory2`: Two factories where goods are produced.
  - `Port1`, `Port2`: Intermediate transportation hubs (e.g., seaports or rail terminals).
  - `City1`, `City2`, `City3`: Final destinations where demand must be met.
- **Modes (modes)**: Different transportation methods available in the network.
  - `Truck`: Road transport.
  - `Train`: Rail transport.
  - `Subway`: Urban transit system.

2. Supply and Demand Values

- **Supply (supply)**: The amount of goods each node can produce.
  - `Factory1` produces **100** units.
  - `Factory2` produces **150** units.
  - `Port1` and `Port2` have an additional supply of **200** and **250** units, respectively.
  - `City1`, `City2`, `City3` have **0** supply because they are demand centers.
- **Demand (demand)**: The amount of goods each destination needs.
  - `City1` requires **150** units.
  - `City2` requires **200** units.
  - `City3` requires **350** units.
  - The other nodes (`Factory1`, `Factory2`, `Port1`, `Port2`) have **0** demand since they are suppliers.

3. Transportation Costs (arcs)

- Each tuple (`source`, `destination`, `mode`) represents a transportation route, and the value represents the **cost per unit transported**.
- **Example Interpretations**:
  - `('Factory1', 'Port1', 'Truck'): 10` → Transporting 1 unit from `Factory1` to `Port1` via truck costs **10**.
  - `('Factory1', 'City1', 'Train'): 15` → Transporting 1 unit from `Factory1` to `City1` via train costs **15**.
  - `('City1', 'City2', 'Subway'): 5` → Moving 1 unit between `City1` and `City2` via subway costs **5**.

4. Capacity Limits (capacity)

- Defines the **maximum** number of units that can be transported along each route.
- **Example Interpretations**:
  - `('Factory1', 'Port1', 'Truck'): 250` → A **maximum of 250** units can be transported from `Factory1` to `Port1` by truck.
  - `('Port2', 'City3', 'Train'): 190` → A **maximum of 190** units can be transported from `Port2` to `City3` by train.
  - `('City1', 'City2', 'Subway'): 120` → A **maximum of 120** units can be transported from `City1` to `City2` via subway.

The goal is to **transport goods efficiently** from supply locations (`Factories` & `Ports`) to demand locations (`Cities`) **while minimizing costs** and **respecting capacity constraints**.

This is a **large-scale multi-modal transportation network problem**, where we decide **how much flow should go through each route to meet demand at the lowest cost**.

Mathematical Formulation:

1. **Decision Variables**:
   - $x_{i,j,m}$: The flow of goods from node i to node j using mode m.
2. **Objective Function**:
   - Minimize transportation cost while including a congestion penalty.
   - $$\min \sum_{(i,j,m)\in A} c_{i,j,m} x_{i,j,m} + 0.1 x_{i,j,m}^2$$

   where $c_{i,j,m}$ is the cost per unit transported.
3. **Flow Conservation Constraints**:
   - For each node k:
   $$\sum_{(i,k,m)\in A} x_{i,k,m} - \sum_{(k,j,m)\in A} x_{k,j,m} = s_k - d_k$$
   - This ensures that the incoming flow minus outgoing flow equals net supply/demand.
4. **Capacity Constraints**:
   - Each arc has a flow limit:
   $$0 \leq x_{i,j,m} \leq u_{i,j,m}$$
   - where $u_{i,j,m}$ is the capacity limit.

Brief Explanation of the Code

Libraries Used

This code uses **Gurobi** (`gurobipy`) to solve a **multi-modal transportation optimization** problem, minimizing total cost and congestion.

Libraries and Functions Used:

- **gurobipy as gp**: Imports Gurobi's optimization tools.
- **gp.Model("MultiModalTransport")**: Creates an optimization model.
- **model.addVar()**: Defines **decision variables** (flow between locations).
- **model.setObjective()**: Sets the **objective function**, minimizing cost and congestion (`0.1 * flow²`).
- **model.addConstr()**: Ensures **flow balance**, meaning supply and demand match across nodes.
- **model.optimize()**: Runs the solver to find the optimal solution.

Why It Works:

1. **Nodes & Arcs**: Define a transportation network with different transport modes.
2. **Supply & Demand**: Ensures goods move from sources (factories) to destinations (cities).
3. **Flow Balance**: Ensures no goods are lost or created.
4. **"Sink" Node**: Absorbs extra supply, preventing infeasibility.
5. **Quadratic Congestion Term (0.1 * flow^2)**: Penalizes high traffic to optimize efficiency.

Results:

Gurobi Optimizer version 12.0.1 build v12.0.1rc0 (mac64[arm] - Darwin 24.3.0 24D70)

CPU model: Apple M3

```
Thread count: 8 physical cores, 8 logical processors, using up to 8 threads

Optimize a model with 8 rows, 11 columns and 22 nonzeros
Model fingerprint: 0x22d3cf63
Model has 11 quadratic objective terms
Coefficient statistics:
  Matrix range     [1e+00, 1e+00]
  Objective range  [1e+00, 3e+01]
  QObjective range [2e-01, 2e-01]
  Bounds range     [1e+03, 1e+03]
  RHS range        [5e+01, 3e+02]
Presolve removed 2 rows and 1 columns
Presolve time: 0.00s
Presolved: 6 rows, 10 columns, 17 nonzeros
Presolved model has 10 quadratic objective terms
Ordering time: 0.00s

Barrier statistics:
 AA' NZ     : 7.000e+00
 Factor NZ  : 2.100e+01
 Factor Ops : 9.100e+01 (less than 1 second per iteration)
 Threads    : 1

                  Objective                Residual
Iter       Primal          Dual         Primal    Dual     Compl     Time
   0   1.78688132e+07 -2.38356821e+07  3.00e+03 7.18e+02  9.28e+05     0s
   1   7.39461798e+04 -5.08328360e+06  2.28e+01 5.46e+00  6.95e+04     0s
   2   5.01023841e+04 -7.81351564e+04  1.94e-01 4.63e-02  1.60e+03     0s
   3   3.62316841e+04  3.40417379e+02  2.19e-02 5.23e-03  4.49e+02     0s
   4   2.92476980e+04  1.41364799e+04  2.19e-08 5.23e-09  1.89e+02     0s
   5   2.87887486e+04  2.80503593e+04  7.60e-10 1.82e-10  9.23e+00     0s
   6   2.86954811e+04  2.86714344e+04  1.99e-13 5.03e-14  3.01e-01     0s
   7   2.86908342e+04  2.86891006e+04  0.00e+00 6.20e-15  2.17e-02     0s
   8   2.86907293e+04  2.86906955e+04  0.00e+00 7.04e-15  4.22e-04     0s
   9   2.86907292e+04  2.86907291e+04  7.11e-15 3.42e-15  4.21e-07     0s
  10   2.86907292e+04  2.86907292e+04  7.11e-15 1.11e-14  4.21e-10     0s

Barrier solved model in 10 iterations and 0.00 seconds (0.00 work units)
Optimal objective 2.86907292e+04

Flow from Factory1 to Port1 via Truck: 6.953388092334062e-12
Flow from Factory1 to City1 via Train: 99.99999999999305
Flow from Factory2 to Port2 via Truck: 4.375000013890426
Flow from Factory2 to City2 via Train: 145.62499998610957
Flow from Port1 to City1 via Truck: 78.33333333334276
Flow from Port1 to City2 via Train: 121.66666666666418
Flow from Port2 to City2 via Truck: 46.250000009263886
Flow from Port2 to City3 via Train: 208.12500000462651
Flow from City1 to City2 via Subway: 28.333333333335844
Flow from City2 to City3 via Subway: 141.87499999537349
Flow from City3 to Sink via Truck: 50.0
Total Cost: 28690.72916666674
```