ISEN 620 PROJECT

MIDDLE MILE LOGISTICS CONSULTING REPORT

TEAM #16

ARUNACHALAM SENTHILKUMAR

KARTHIKEYAN CHANDAR

PRADEEP MANOHARAN

RITWIK AJAY

04th December 2022

CONTENTS

EXECUTIVE SUMMARY:

Revenue from retail e-commerce in the United States was estimated at roughly 905 billion U.S. dollars in 2022. The Statista Digital Market Outlook forecasts that by 2027, online shopping revenue in the U.S. will exceed 1.7 trillion dollars. Therefore, it is very important that logistics are very efficient. A mixed linear optimization model was developed that solves the middle mile logistics problem. The model was solved in AMPL, an advanced optimization software to obtain optimum solutions.

- The total cost associated with shipping the packages using the original model is $ 39008.
- In the first scenario, the cost of operating plane is reduced such that usage of at least one plane becomes feasible. It has been found that when cost is decreased from $50 to $46, it becomes feasible to operate a plane.
- In the second scenario, an analysis of total no. of late packages Vs total cost was performed. It has been observed that deduction in no. of late packages causes increase in total cost incurred (Objective value). When late package no. was 25, Objective value was $39008 and when late package was 5, objective value increased to $106130
- In the third scenario.

In continuation, additional sensitivity analysis has been performed to check the optimal values and its changes due to impact of the change in the input parameter. The data and conditions have been studied are given below.

- In the first scenario, we are increasing the Loading and Unloading Capacity of the shipping center is first increased by 5% and then by 20%. So, by doing that there is a decrease in the objective value by $38434 and $38285 respectively. From this observation we can find that by increasing the capacity in the shipping center there is a reduction the total cost.
- In the second scenario, if we are decreasing the No. of operators relocating from their home location there is a considerable increase in objective value., i.e., if there are 7

operators relocating then we get an objective value of $39008, if the operator is reduced by 4 then the objective value in increased to $43157. Thus if no. of operators relocating decreased then the total cost increased

- In the third scenario, we are increasing total capacity of the vehicle from 1200 Kgs to 1400, 1500, 1600, 1700 & 1800 respectively, as a result there is a considerable decrease in the no. of truck used and increase in the total no. of trains. From this observation we can find that by increasing the capacity of the vehicle there is a reduction in the no. of truck and increase in the no. of trains.

INTRODUCTION

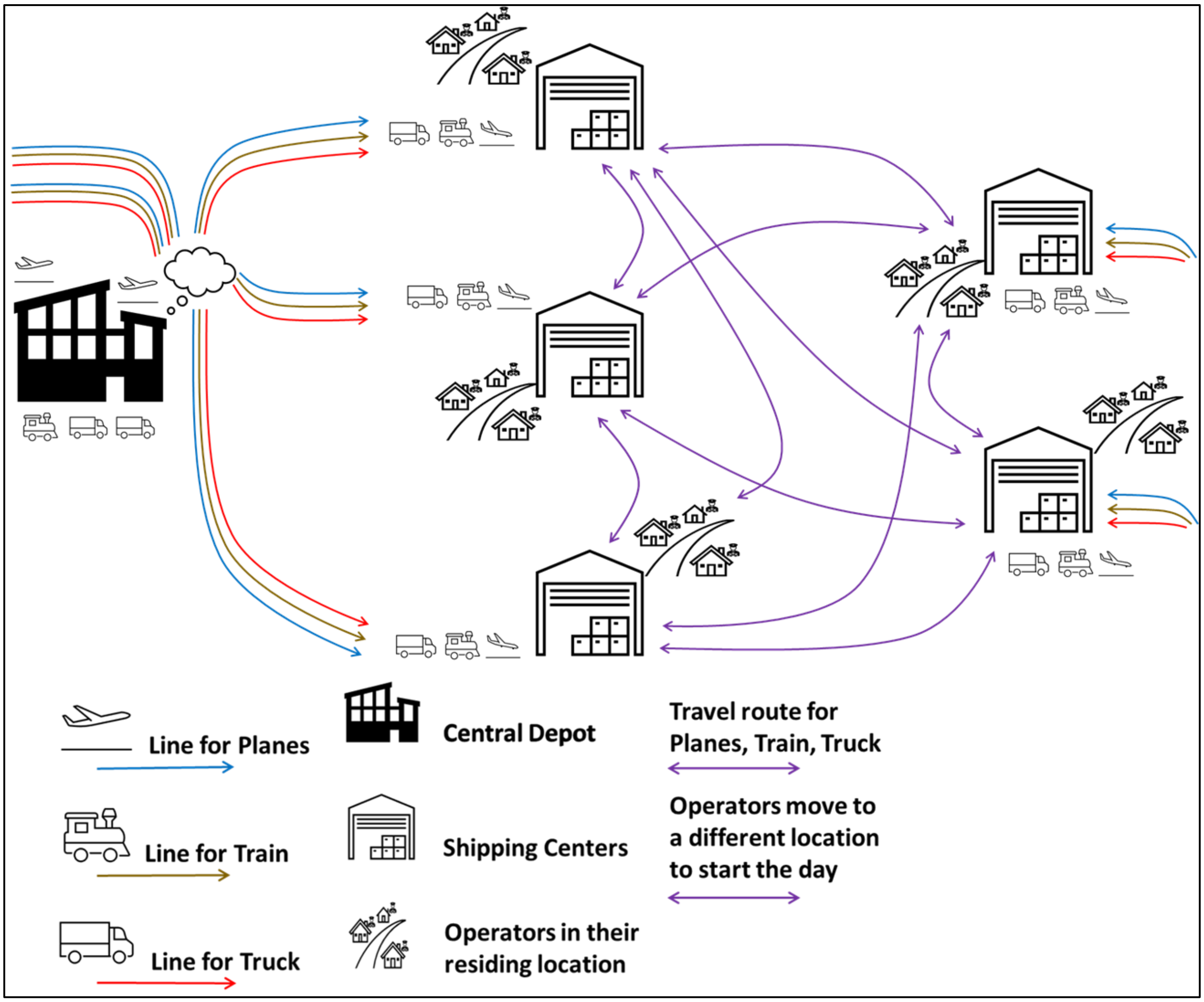
E-commerce is rapidly becoming the most popular way of consumption across the world. Gone are the days of driving to the mall for clothes, and no more sourcing the newspaper ads for the best deals on electronics or appliances. The luxury of online shopping often comes with convenience and monetary deals that are irresistible to consumers. Since Consumers spend billions of dollars through e-commerce, the shipping industry has the potential to profit immensely. At the same time, since large profits mean fierce competition, for this the shipping company must be as efficient as possible to survive. And hence “OOPS” has hired ISEN consulting firm to get all the necessary information so they can complete the shipping as cheaply as possible.

Nowadays, Supply Chain business problems are expanded across various sectors and approach like information security, business continuity, Just-in-time models, and continuous manufacturing. The before said modern approaches are trivial for Business operations and Distribution Management processes which require more flexibility with optimized solutions for new-age business problems to scale up their operations which require real-time information and intelligence. Businesses also need to consolidate data & technology in a multi-dimensional plan format and plan to lower the total cost of ownership and operation to arrive at a sustainable cost structure.

There are more complicated parameters contributing to the following uncertainty.

Uncertainty in the E-Commerce Demand: This uncertainty demand may be due to Global economic conditions, which may lead to stagnation of stocks in the OEMs. This uncertainty of demands has been the greatest business problem, leading to excess inventory, and this uncertainty in the supply leads to unpredictable ordering. Then there may be a lack of planning of the current inventory which may cause the downstream and upstream of the stock which lead to surplus and overcompensation.

The perceptibility of the Supply chain: In each organization, there are numerous technologies, formats, & key principles being followed and there are blind spots to track and trace products across multiple enterprises, business verticals, and supply chain groups internally in every organization.



Design of Supply Chain Network: In this fast-moving world design of the supply chain has made a wide network format that has increasingly become global and dispersed. Apart from this, a wide variety of factors-ranging from cost structures, tax, Human resources, and technical availability has driven companies to redesign and reconfigure their supply chains continually.

Multi-Channel Retail: On a day-to-day basis OEMs, Manufacturers, and retail distributors are facing quite a challenging task dealing with multichannel retail. Building a huge network of multiple-channel networks that can instantaneously process orders from different means and help fulfill customers' needs at the earliest possible dates.

And Hence in this report, we have developed an Integer programming model that would provide an optimal solution to this problem of shipping packages within the deadline from various shipping centers.

METHODOLOGY

PARAMETERS

A parameter is a numerical value or constant that describes the character of a system. We have the following parameters to be considered for this scenario.

1. **L - Number of shipping locations**

   The number of shipping center will be given, and the model determines the vehicle with minimal cost for transporting the package from the origin shipping 'i' center to the destination shipping center 'j'.

2. **P - Number of packages to be shipped**

   The number of packages to be shipped will be given and the model determines the optimal mode of the vehicle to be used for shipping the packages.
   Example: P = 300, indicates the total number of packages in various shipping centers is 300 which must reach the destination shipping center by optimal mode of transport.

3. **V- Number of vehicle available**

   The total number of available vehicles in the depot will be given and our linear program will decide on allocating the vehicles to various shipping center. This parameter indicates the max. capacity of vehicles to be available for shipping.
   Example: V = 100, then max. no. of vehicles available is 100.

4. **K - Number of incompatible pairs of packages**

   Total number of incompatibles pairs of packages will be given, and our linear programming model will decide and make sure that the incompatible pairs can't travel from one place to another in the same vehicles.

5. **I_{kr} - ID of package r of the k^th^ incompatible pair,**

for r =1,2 for k = 1 ,.., K

This parameter denotes the ID of the package 'r' which would be one among the incompatible pairs.

6. **w_p - Weight of package p,** **for p=1 ,.., P**

The parameter 'w_p' indicates the weight of the individual package in Kilograms.

Example: If w_{12} = 125Kg then it indicates the weight of package 12 is 100 Kg.

7. **s_p - Origin of package p,** **for p = 1 ,.., P**

The parameter 's_p' indicates the origin shipping center of the individual packages.

Example: If S_{17} = 9, then it indicates the package starts from shipping center 9.

8. **e_p - Destination of packages p,** **for p = 1 ,.., P**

The parameter 'e_p' indicates the destination shipping center of the individual packages.

Example: If S_{17} = 6, then it indicates the package is to reach destination center 6.

9. **f_p - Maximum transit time for package p,** **for p = 1 ,.., P**

The 'f_p' is the maximum time limit before which the packages should be shipped from the origin shipping center to the destination shipping center after that period penalty cost will be implemented and the package will be delivered free of cost.

Example: f_{17} = 137, indicates the package should be delivered within the 137 minutes

10. **cl_p - Penalty / cost if package p is tardy / late,** **for p = 1 ,.., P**

The 'cl_p' is the penalty cost that will be implemented if the time taken for the packages to be shipped is within the maximum transit time limit provided.

Example: If cl_{17} = 714, then this indicates a penalty cost of $714 will be added if the package is not delivered within Max. transit time for package 'f_{17} = 137 minutes.

11. h_v - Weight capacity of vehicle v, **for v = 1 ,.., V**

The parameter 'h_v' indicates the weight capacity of the vehicle in Kilograms,

Example: If h_1 = 1200Kg then it indicates that weight capacity of the vehicle 1 is 1200 Kg.

12. b_v - Vehicle type (plane, train, or truck) of vehicle v,

for v = 1 ,.., V

The parameter 'b_v' indicates the vehicle type of vehicle v.

Example: If b_v = 1 then it indicates the vehicle type used is truck.

13. a_i - Number of operators whose Homebase is location i,

for i = 1 ,.., L

The parameter 'a_i' indicates the number of operators whose homebase location is i.

Example: If a_{10} = 11 then this indicates that in location 10, there are totally 11 operators.

14. l_i - Maximum number of vehicles that can be loaded per day at location i,

for i = 1 ,.., L

The parameter 'l_i' indicates the maximum number of vehicles that can be loaded per day at the location i.

Example: If l_{10} = 15 then this indicates that in location 10, we can totally load 15 vehicles.

15. u_i - Maximum number of Vehicles that can be unloaded per day at location i,

for i = 1 ,.., L

The parameter 'u_i' indicates the maximum number of vehicles that can be unloaded per day at the location i.

Example: If u_{10} = 16 then this indicates that in location 10, we can totally unload 16 vehicles.

16.cm_{ij} - Cost of relocating one operator from location i to location j,

for i = 1 ,..., L, for j = 1 ,..., L

The parameter 'cm_{ij}' indicates the Cost of relocating one operator from location i to location j.

Example: If cm_{12} = 182, then this indicates that an operator is being relocated from location 1 to 2, then the cost would be 181*2 = $362.

17.t_{ijv} - Transit time from location i to j using vehicle v,

for v = 1 ,..., V, for i = 1 ,..., L, for j = 1 ,..., L

The parameter 't_{vij}' indicates the Transit time taken for shipping packages from location i to j using vehicle v.

Example: If t_{112} = 20, it means that we must deliver the package from location 1 to location 2 using vehicle 1 in 20 minutes.

18.co_{ijv} - Cost of operating vehicle v from location i to j,

for v = 1 ,..., V, for i = 1 ,..., L, for j = 1 ,..., L

The parameter 'co_{vij}' indicates the cost of operating vehicle v from location i to j

Example: If co_{112} = 100, means cost of operating the vehicle 1 form location 1 to location 2 is $100

DECISION VARIABLE

The variables whose values are under the control of the decision maker and influence the performance of the system are called decision variables. When the value of the decision variables is set by the decision maker, a decision is made. The decision variable is also known as Control Variable.

We consider the following decision variables out of which there is a mix of binary variables and integer variable for the given scenario.

BINARY VARIABLE:

Binary Variables can take only two values either zero or one (0 or 1).

1. x_{pv} - Binary variable to determine whether package p is shipped in vehicle v

for p = 1 ,.., P, for v = 1 ,.., V

The decision variable 'x_{pv}' will take value 1 if package p is shipped in vehicle v, 0 otherwise.

Example: If package 3 goes in vehicle 5, then x_{35} =1

2. y_{vij} - Binary variable determining whether vehicle v goes from location i to j

for v = 1 ,.., V, for i = 1 ,.., L, for j = 1 ,.., L

The decision variable 'y_{vij}' will take value 1 if vehicle v goes from location i to j, 0 otherwise.

Example: If vehicle 4 goes from location 2 to location 3, then y_{423} =1

3. T_p - Binary variable to determine whether package p is Tardy (not on time)

for p = 1 ,.., P

The decision variable 'T_p' will take value 1 if package p is late, 0 otherwise.

Example: If package 4 is late (not on time), then T_4 = 1

INTEGER VARIABLE:

Integer variables are the variables that consist of integer values.

4. z_{ij} - Number of operators relocated from location i to j, for i = 1 ,.., L, for j = 1 ,.., L

The decision variable 'z_{ij}' determines the no. of operators relocated from location 'I' to location 'j'.

Example: If z_{45} = 2, then 2 operators relocated from location 4 to location 5.

OBJECTIVE FUNCTION

In linear programming problems, there will be an objective function that is a real-valued function whose values must be minimized or maximized given the constant defined on the given linear equation over the set of feasible constraints.

Minimize Z $\sum_{p=1}^{P} cl_p * T_p + \sum_{v=1}^{V} \sum_{i=1}^{L} \sum_{j=1}^{L} co_{ijv} * y_{ijv} + \sum_{i=1}^{L} \sum_{j=1}^{L} cm_{ij} * z_{ij}$

Minimize Cost Z = (Cost of late packages) + (cost of operating vehicles) + (Cost of operator relocations)

- $\sum_{p=1}^{P} cl_p * T_p$ - (**Cost of late packages**) - In this term, total cost of late packages is found out by taking the product of cost if package is late (cl_p) and binary variable if package is late (T_p).

- $\sum_{v=1}^{V} \sum_{i=1}^{L} \sum_{j=1}^{L} co_{vij} * y_{ijv}$ - (**Cost of operating vehicles**) - In this term, total cost of operating vehicles is found out by taking the product of cost of operating vehicle v from location i to location j (co_{vij}) and binary variable determining whether vehicle v goes from location i to location j (y_{ijv}).

- $\sum_{i=1}^{L} \sum_{j=1}^{L} cm_{ij} * z_{ij}$ - (**Cost of operator relocations**) - This function defines to the objective function, the reallocation cost of operators from one destination to the other, where cm_{ij} is the relocation cost of one operator from i to j and z_{ij} is the number of operators relocated from location i to location j.

CONSTRAINTS

The constraints are the restrictions or limitations on the decision variables. They usually limit the value of the decision variables.

1. **Package can only be shipped with one vehicle, and all packages are shipped**

$$\sum_{v=1}^{V} x_{pv} = 1 \qquad \text{for p = 1 ,.., P}$$

This constraint is used to make sure that each package goes in only a single vehicle and that all packages are shipped. A package cannot be delivered through multiple delivery modes and multiple vehicles. The variable x_{pv} is binary that takes a value of 1. if the vehicle v is selected for package p and 0 otherwise. This summation for v=1,2. V defines that, only a single vehicle can take package p, i.e., it can be 1 for a given value of p.

Example: If p = 5, (i.e.,) for the package no. 5, summation of x_{5v} from v = 1 ,.., V will take value 1 for only a single term. This ensures that the package 5 goes only through a single vehicle.

2. **Vehicle can only travel on one route**

$$\sum_{i=1}^{L} \sum_{j=1}^{L} y_{vij} \leq 1 \qquad \text{for v = 1 ,.., V}$$

This constraint restricts the vehicle to travel only on one route. The summation of y_{vij}(binary variable for vehicle v going from location i to location j) over i and j for a particular v should be less than or equal to 1. The summation value is less than or equal to 1 make sure that each vehicle can only take a single route.

Example: If v = 2, (i.e.,) for vehicle number 2, summation of y_{2ij} from = 1,.., L and j = 1,.., L will take value less than or equal to 1 for a particular value of v (i.e., for v=2).

3. **Loading Packages must be shipped on a vehicle that is going from their origin to their destination**

$$x_{pv} \leq y_{vs_p e_p} \qquad \text{for p = 1 ,.., P for v = 1 ,.., V}$$

This constraint ensure that each package will only be shipped in a vehicle that goes from their origin to their destination. Binary variable to determine whether package p is shipped in vehicle v (x_{pv}) should always be less than or equal to $y_{vs_p e_p}$, i.e., binary variable determining whether vehicle v goes from origin of package p (s_p) to destination of package p (e_p).

For Example: Let us assume that package 2 goes from location 3 to location 4 and vehicle 5 goes from location 3 to location 4, then $x_{25} \leq y_{534}$. This will make sure that package 2 will be shipped on a vehicle that is only going from its origin location to its destination location.

4. **Packages must reach their destination within the time limit if they are shipped without penalty (big M)**

$$\sum_{v=1}^{V} \sum_{i=1}^{L} \sum_{j=1}^{L} t_{ijv} * \ x_{pv} \leq \ f_p + M * T_p \qquad \text{for p = 1 ,.., P}$$

This constraint limits the time and ensure that package must reach their destination within the time limit if they are shipped without penalty. Summation over product of transit time from location i to j using vehicle v (t_{vij}) and binary variable to determine whether package p is shipped in vehicle v (x_{pv}) over v=1..v , i=1..L, j=1..L should always be less than or equal to sum of maximum transit time for package p (f_p) and product of M and binary variable for package p being tardy (T_p).

For Example: Let us consider a package 2, if the package 2 does not incur penalty then it should reach the destination within the time limit. The term $\sum_{v=1}^{V} \sum_{i=1}^{L} \sum_{j=1}^{L} t_{vij} * \ x_{2v}$ should always be less than or equal to f_2. In this case T_2 will be zero as the package 2 will not meet penalty as it will reach destination within the time limit.

5. **Each location can't exceed the limit on loading per day**

$\sum_{j=1}^{L} \sum_{v=1}^{V} y_{vij} \leq l_i$ for i = 1 ,.., L

This constraint limits the loading limit per day at each location. Summation of binary variable determining whether vehicle goes from location i to location j (y_{vij}) for j = 1,.., L and v = 1 ,.., V should be less than or equal to maximum number of vehicles that can be loaded per day at location i (l_i) for a particular i. This constraint ensures that each location cannot exceed the limit on loading per day.

For Example: Let us consider a location 2, then the maximum number of vehicles that is starting from location 2, i.e., sum of binary variable (y_{v2j}) for j = 1,.., L and v =1,..V should be less than or equal to the maximum number of vehicles that can be loaded per day at location 2 (l_2). The same condition is being satisfied using this constraint for all starting location i for i = 1 ,.., L.

6. **Each location can't exceed the limit on unloading per day**

$\sum_{i=1}^{L} \sum_{v=1}^{V} y_{vij} \leq u_j$ for j = 1 ,.., L

This constraint limits the unloading limit per day at each location. Summation of binary variable determining whether vehicle goes from location i to location j(y_{vij}) for i=1..L and v=1..V should be less than or equal to maximum number of vehicles that can be unloaded per day at location j (u_j) for a particular j. This constraint ensures that each location cannot exceed the limit on loading per day.

For Example: Let us consider a location 3, then the maximum number of vehicles that is ending at location 3, i.e., sum of binary variable (y_{vi3}) for i=1..L and v=1..V should be less than or equal to the maximum number of vehicles that can be unloaded per day at location 3 (u_3). The same condition is being satisfied using this constraint for all destination location j for j=1..L.

7. **Incompatible packages each need a separate vehicle**

$$xI_{k1,v} + xI_{k2,v} \leq 1 \text{ for k = 1 ,.., k, for v = 1 ,.., V}$$

The packages that are incompatible with each other should not travel in the same vehicle. The above-mentioned constraint enforces that this criterion is satisfied. The sum of $xI_{k1,v}$(binary variable determining whether package (I_{k1}) goes in vehicle v, where I_{k1} is the ID of package 1 for kth incompatible pair) and $xI_{k2,v}$(binary variable determining whether package (I_{k2}) goes in vehicle v, where I_{k2} is the ID of package 2 for kth incompatible pair) should be less than or equal to 1, for a particular k=1..K and v=1..V. Using above logic, the maximum number of incompatible packages that can travel in same vehicle is being limited to 1 or less than 1.

For Example: Let us consider the first pair of incompatible packages, i.e., k=1, then $xI_{11,v} + xI_{12,v}$ should be less than or equal to 1 for a specific vehicle number v. This makes sure that no incompatible packages are travelling in a same vehicle v.

8. **Cannot exceed the number of operators available in each location**

$$\sum_{j=1}^{L} \sum_{v=1}^{V} y_{vij} + \sum_{j=1}^{L} z_{ij} - \sum_{j=1}^{L} z_{ji} \leq a_i \text{ for i = 1 ,.., L}$$

This constraint states that the total number of operators available at each location cannot be exceeded. In the above logic, $y_{vij}$is the binary variable determining whether vehicle v travels from location i to location j, z_{ij} is the number of operators relocated from location i to location j, z_{ji} is the number of operators relocated from location j to location i and a_i is the number of operators whose homebase is location i.

For Example: Let us consider a starting location 2 (i.e., i=2), then the term $\sum_{j=1}^{L} \sum_{v=1}^{V} y_{v2j} + \sum_{j=1}^{L} z_{2j} - \sum_{j=1}^{L} z_{j2}$ should always be less than or equal to a_2 which is the number of operators whose homebase is location 2. In the above expression, $\sum_{j=1}^{L} \sum_{v=1}^{V} y_{vij}$ gives the sum number of vehicles going from location 2 to all other locations, $\sum_{j=1}^{L} z_{2j}$ gives the total number of operators relocated from location 2 to all other locations and $\sum_{j=1}^{L} z_{j2}$ gives the total number of operators relocated from all other locations to location 2.

9. **Packages cannot exceed the weight of vehicles**

$\sum_{p=1}^{P} w_p * x_{pv} \leq h_v$ for v = 1 ,.., V

This constraint states that the weight of packages cannot exceed the weight capacity of the vehicle. The sum of individual package weight that goes in vehicle v should always be less than or equal to the weight capacity of the vehicle v. This is explained mathematically above with the use of terms w_p(weight of package p), x_{pv}(binary variable if package p goes in vehicle v) and h_v(weight capacity of vehicle v).

For Example: consider for a particular vehicle with number 4, h_4 is the weight limit defined for that vehicle 4. Then, $w_p * x_{pv}$ for all package p=1..P for that vehicle 4 should always be less than or equal to h_4 weight limit defined for that vehicle.

10. **Non negativity**

$z_{ij} \geq 0$ for i = 1 ,.., L, for j = 1 ,.., L

All Variables are greater than or equal to Zero.

All Variable >= 0

11. **Integrality**

z_{ij} is integer for i = 1,.., L, for j = 1,.., L

12. **Binary requirements**

T_p is binary for p = 1 ,.., P

x_{pv} is binary for p = 1 ,.., P, for v = 1 ,.., V

y_{vij} is binary for v = 1 ,.., V, for i = 1 ,.., L, for j = 1

SOLUTION

We have developed a model in AMPL to obtain the optimal solution for the given problem statement. For this model we have created a data file with the data provided. Below we have discussed the AMPL output solving the model and then the AMPL output for every decision variable for better understanding of the solution and better implementation of the solution in practice.

we have loaded our model file and data file into AMPL and solved the model and AMPL has returned an optimal solution for the minimum cost. This means that when considering all the parameters, decision variables, and constraints, minimum cost of shipping the packages is $39008.

PARAMETERS USED IN AMPL FROM THE GIVEN DATA

1. **Number of shipping locations (L)** given as the input in AMPL data file, A total number of 10 shipping center are currently available in the map which is denoted by L=10.

```
ampl: display L;
L = 10
```

2. **ID of package r of the kth incompatible pair (I_{kr})** given as the input in AMPL data file, and it **is** denoted by 'I_{kr}' below we have displayed the same.

```
ampl: display I;
I [*,*]
:     1     2    :=
1     41    93
2     30    79
3     76    98
4      1    15
5     77    83
6     38   126
7     77   101
8     22    54
9     27    44
10   108   178
11    53    81
12   118   162
13    75   153
14     8   117
15    34    72
16   115   173
17    66   113
18   108   124
;
```

3. **Number of vehicles available (V)** given as the input in AMPL data file, A total number of 120 vehicles are currently available in the central depot which is denoted by V=120.

```
ampl: display V;
V = 120
```

4. **Weight of packages p (w_p)** given as the input in AMPL data file, A total number of 180 packages are currently available and their weights are given in the units of Kilogram and the parameter is denoted by ‘w_p’.

```
ampl: display w;
w [*] :=
  1 587    24 252    47 830    70 291    93 807   116 748   139 803   162  63
  2 489    25 364    48 616    71  77    94 503   117 414   140 820   163 770
  3 228    26 454    49 688    72 709    95 113   118  83   141 261   164  91
  4 698    27 237    50 330    73  97    96 561   119 831   142 470   165 692
  5 722    28 405    51 240    74 435    97 455   120  74   143 104   166 181
  6 405    29  91    52 565    75 351    98  31   121 334   144 845   167 488
  7 757    30 424    53 507    76 180    99 732   122 933   145 344   168 696
  8 848    31 796    54 215    77 117   100 736   123 643   146  53   169 782
  9 465    32 304    55 108    78 995   101 339   124 844   147 846   170  14
 10 615    33 862    56 441    79 156   102 940   125 381   148 682   171 995
 11 546    34 611    57 753    80 439   103 638   126 929   149 283   172 834
 12 906    35 669    58 967    81 724   104 353   127 889   150 749   173 460
 13 304    36 425    59 656    82 106   105 876   128 711   151 414   174 127
 14 124    37 142    60 823    83 920   106 408   129 833   152 336   175 808
 15 936    38 356    61 867    84 144   107 546   130 712   153 518   176 189
 16 126    39 370    62 668    85 717   108 522   131 254   154 347   177 656
 17 339    40 251    63 958    86 755   109 912   132 444   155 834   178 922
 18 930    41  10    64 171    87 926   110 962   133 550   156 243   179 251
 19 133    42 159    65  17    88 531   111 741   134 812   157 309   180 248
 20 370    43 138    66 185    89 784   112 240   135 167   158 966
 21 434    44 575    67  30    90 654   113 380   136 723   159 289
 22 816    45   9    68 745    91 474   114 525   137 397   160 178
 23 716    46 582    69 252    92 421   115 354   138 705   161 202
;
```

5. **Origin of packages (s_p)** given as the input in AMPL data file, A total number of 180 packages are currently available and their respective origins shipping centers are displayed below It is denoted by s_p.

```
ampl: display s;
s [*] :=
  1  5    21  7    41  4    61  9    81  3   101  1   121 10   141 10   161  5
  2  7    22  9    42  2    62  3    82  9   102  1   122  4   142  7   162  4
  3  5    23  5    43  3    63  2    83  1   103  5   123  1   143  9   163  6
  4  4    24  9    44  5    64  2    84  1   104  6   124  3   144  6   164  3
  5  5    25  9    45  8    65  8    85  1   105  9   125  8   145  2   165  6
  6  1    26  2    46  5    66  4    86  4   106 10   126  4   146  6   166  4
  7  6    27  5    47  6    67  8    87  6   107  7   127  1   147 10   167  9
  8  8    28  1    48  5    68  4    88  9   108  3   128  8   148  9   168  5
  9  4    29  6    49 10    69  7    89  1   109  1   129  2   149  3   169  3
 10  7    30  7    50  8    70  7    90  9   110  4   130  7   150  3   170 10
 11  6    31  5    51  7    71  4    91  6   111  3   131  9   151 10   171  6
 12  5    32  4    52  9    72  2    92  2   112  5   132  8   152  2   172  9
 13  2    33  5    53  3    73 10    93  4   113  4   133  6   153  7   173  9
 14  5    34  2    54  9    74  5    94  7   114  1   134  1   154 10   174  3
 15  5    35 10    55  4    75  7    95  4   115  9   135  4   155  4   175 10
 16  1    36  7    56  2    76  7    96  7   116  5   136  6   156  2   176  7
 17  9    37  9    57  6    77  1    97  1   117  8   137  8   157  7   177  5
 18  8    38  4    58  3    78  6    98  7   118  4   138  1   158  2   178  3
 19  7    39  4    59  9    79  7    99  5   119  9   139  9   159  4   179  2
 20  3    40  2    60  7    80  6   100  1   120  4   140  5   160 10   180  8
;
```

6. **Number of packages to be shipped (P)** given as the input in AMPL data file, A total number of 180 packages are currently available in the shipping centers which is denoted by P=180.

```
ampl: display P;
P = 180
```

7. **Destination of packages (e_p)** given as the input in AMPL data file, A total number of 180 packages are currently available and their respective destination shipping centers are displayed below It is denoted by e_p

```
ampl: display e;
e [*] :=
  1  8    21  2    41  7    61  4    81  8   101 10   121  1   141  9   161  2
  2  6    22  8    42  3    62 10    82  3   102 10   122  8   142  9   162  1
  3  4    23  1    43 10    63  8    83 10   103  2   123  5   143  2   163  7
  4 10    24  4    44  7    64  6    84  7   104  7   124  6   144  7   164  7
  5  9    25  2    45  9    65  4    85  3   105  8   125  4   145  3   165  8
  6  6    26 10    46  3    66  9    86  6   106  7   126  8   146  9   166 10
  7  4    27  7    47  4    67  5    87  3   107 10   127  7   147  5   167  4
  8  3    28  5    48  1    68  1    88  1   108  6   128 10   148  2   168  7
  9  1    29  8    49  4    69  8    89  9   109  4   129  7   149  8   169  6
 10  6    30  3    50  1    70  1    90 10   110  6   130 10   150  2   170  5
 11  5    31  9    51  6    71  5    91  8   111 10   131  6   151  7   171  1
 12  6    32  3    52 10    72 10    92  9   112  1   132  4   152  9   172  1
 13  9    33  1    53  8    73  6    93  7   113  9   133  3   153  2   173 10
 14  1    34 10    54  8    74  3    94  4   114  9   134  7   154  6   174  4
 15  8    35  6    55  6    75  2    95  9   115 10   135  1   155  9   175  9
 16  9    36  3    56  4    76  3    96  1   116  2   136 10   156  6   176  8
 17  6    37  6    57  3    77 10    97  7   117  3   137  6   157  8   177 10
 18  3    38  8    58  1    78  3    98  3   118  1   138  6   158  5   178  6
 19 10    39  2    59  6    79  3    99 10   119  3   139  2   159  3   179  8
 20  7    40  3    60  1    80  7   100  3   120  2   140  8   160  6   180  5
;
```

8. **Maximum transit time for packages (f_p)** given as the input in AMPL data file, and it is given in the units of minutes, it is denoted by 'f_p' below we have displayed the same.

```
ampl: display f;
f [*] :=
  1 116    24 129    47 118    70 137    93 112   116 143   139 119   162 147
  2 143    25 135    48 138    71 141    94 110   117 129   140 128   163 122
  3 129    26 117    49 131    72 115    95 120   118 121   141 135   164 141
  4 134    27 119    50 134    73 142    96 129   119 148   142 145   165 128
  5 129    28 127    51 148    74 126    97 124   120 116   143 126   166 116
  6 115    29 126    52 143    75 129    98 145   121 136   144 140   167 114
  7 140    30 134    53 143    76 123    99 140   122 141   145 120   168 146
  8 134    31 115    54 143    77 144   100 118   123 143   146 125   169 112
  9 117    32 147    55 137    78 116   101 146   124 116   147 117   170 142
 10 116    33 149    56 131    79 119   102 115   125 127   148 142   171 113
 11 124    34 140    57 137    80 141   103 111   126 136   149 130   172 148
 12 139    35 119    58 118    81 123   104 128   127 119   150 134   173 137
 13 125    36 139    59 129    82 126   105 145   128 144   151 146   174 140
 14 141    37 145    60 131    83 142   106 115   129 120   152 112   175 145
 15 122    38 134    61 118    84 127   107 136   130 117   153 131   176 149
 16 136    39 139    62 138    85 120   108 140   131 117   154 146   177 110
 17 137    40 112    63 112    86 145   109 146   132 148   155 114   178 133
 18 115    41 120    64 123    87 148   110 147   133 120   156 146   179 129
 19 128    42 130    65 133    88 142   111 127   134 141   157 124   180 142
 20 130    43 115    66 126    89 140   112 145   135 116   158 129
 21 120    44 122    67 136    90 137   113 136   136 124   159 120
 22 131    45 124    68 149    91 143   114 148   137 118   160 143
 23 130    46 143    69 116    92 131   115 131   138 122   161 132
;
```

9. **Number of incompatible pairs of packages (K)** given as the input in AMPL data file, A total number of 18 incompatible pairs of packages are currently available from the total packages which is denoted by K=18.

```
ampl: display K;
K = 18
```

10. Penalty / cost incurred if package is delivered late (cl_p) given as the input in AMPL data file, and it the cost added if the package is not delivered within the transit time provided. Below we have displayed the penalty cost for all the 180 packages.

```
ampl: display cl;
cl [*] :=
  1 728    24 370    47 847    70 709    93 573   116 971   139 594   162 669
  2 915    25 207    48 536    71 588    94 540   117 295   140 375   163 956
  3 687    26 424    49 740    72 413    95 332   118 980   141 326   164 351
  4 818    27 435    50 885    73 551    96 650   119 820   142 528   165 738
  5 467    28 507    51 906    74 929    97 386   120 473   143 752   166 925
  6 665    29 310    52 985    75 899    98 598   121 405   144 455   167 838
  7 450    30 574    53 606    76 817    99 539   122 936   145 880   168 406
  8 867    31 449    54 498    77 741   100 923   123 935   146 202   169 692
  9 456    32 813    55 234    78 355   101 942   124 663   147 647   170 516
 10 423    33 590    56 716    79 801   102 279   125 578   148 471   171 476
 11 993    34 229    57 951    80 890   103 464   126 615   149 427   172 992
 12 632    35 557    58 823    81 554   104 774   127 318   150 701   173 630
 13 652    36 926    59 479    82 512   105 247   128 585   151 223   174 979
 14 586    37 805    60 272    83 637   106 518   129 417   152 641   175 730
 15 633    38 734    61 251    84 258   107 294   130 458   153 438   176 844
 16 714    39 595    62 874    85 902   108 374   131 736   154 517   177 950
 17 987    40 720    63 524    86 672   109 378   132 395   155 415   178 233
 18 884    41 719    64 932    87 675   110 964   133 356   156 927   179 561
 19 549    42 703    65 245    88 486   111 445   134 309   157 609   180 331
 20 409    43 762    66 803    89 271   112 315   135 333   158 382
 21 613    44 208    67 280    90 477   113 599   136 840   159 306
 22 755    45 314    68 436    91 713   114 243   137 222   160 312
 23 527    46 270    69 364    92 816   115 316   138 340   161 544
;
```

11. Weight Capacity of vehicle (h_v) given as the input in AMPL data file, and it is denoted by 'h_v' below we have displayed the same.

```
ampl: display h;
h [*] :=
  1 1200    19 1200    37 1200    55 1200    73 1200    91 1200   109 6000
  2 1200    20 1200    38 1200    56 1200    74 1200    92 1200   110 6000
  3 1200    21 1200    39 1200    57 1200    75 1200    93 1200   111 6000
  4 1200    22 1200    40 1200    58 1200    76 1200    94 1200   112 6000
  5 1200    23 1200    41 1200    59 1200    77 1200    95 1200   113 6000
  6 1200    24 1200    42 1200    60 1200    78 1200    96 1200   114 6000
  7 1200    25 1200    43 1200    61 1200    79 1200    97  600   115 6000
  8 1200    26 1200    44 1200    62 1200    80 1200    98  600   116 6000
  9 1200    27 1200    45 1200    63 1200    81 1200    99  600   117 6000
 10 1200    28 1200    46 1200    64 1200    82 1200   100  600   118 6000
 11 1200    29 1200    47 1200    65 1200    83 1200   101  600   119 6000
 12 1200    30 1200    48 1200    66 1200    84 1200   102  600   120 6000
 13 1200    31 1200    49 1200    67 1200    85 1200   103  600
 14 1200    32 1200    50 1200    68 1200    86 1200   104  600
 15 1200    33 1200    51 1200    69 1200    87 1200   105  600
 16 1200    34 1200    52 1200    70 1200    88 1200   106  600
 17 1200    35 1200    53 1200    71 1200    89 1200   107  600
 18 1200    36 1200    54 1200    72 1200    90 1200   108  600
;
```

12. Vehicle type (b_v) given as the input in AMPL data file, and it is denoted by 'b_v' below we have displayed the same.

```
ampl: display b;
b [*] :=
  1 1    13 1    25 1    37 1    49 1    61 1    73 1    85 1    97 2   109 3
  2 1    14 1    26 1    38 1    50 1    62 1    74 1    86 1    98 2   110 3
  3 1    15 1    27 1    39 1    51 1    63 1    75 1    87 1    99 2   111 3
  4 1    16 1    28 1    40 1    52 1    64 1    76 1    88 1   100 2   112 3
  5 1    17 1    29 1    41 1    53 1    65 1    77 1    89 1   101 2   113 3
  6 1    18 1    30 1    42 1    54 1    66 1    78 1    90 1   102 2   114 3
  7 1    19 1    31 1    43 1    55 1    67 1    79 1    91 1   103 2   115 3
  8 1    20 1    32 1    44 1    56 1    68 1    80 1    92 1   104 2   116 3
  9 1    21 1    33 1    45 1    57 1    69 1    81 1    93 1   105 2   117 3
 10 1    22 1    34 1    46 1    58 1    70 1    82 1    94 1   106 2   118 3
 11 1    23 1    35 1    47 1    59 1    71 1    83 1    95 1   107 2   119 3
 12 1    24 1    36 1    48 1    60 1    72 1    84 1    96 1   108 2   120 3
;
```

13. **Number of operators whose homebase is location i (a_i)** given as the input in AMPL data file, and it is denoted by 'b_v', below we have displayed the same.

```
ampl: display a;
a [*] :=
 1  12
 2  13
 3  10
 4  11
 5  14
 6  13
 7   8
 8  11
 9  13
10  11
;
```

14. **Maximum number of vehicles that can be loaded per day at location (l_i)** given as the input in AMPL data file, and it is denoted by 'l_i', below we have displayed the same.

```
ampl: display l;
l [*] :=
 1  16
 2  13
 3  16
 4  14
 5  15
 6  14
 7  14
 8  16
 9  13
10  15
;
```

15. **Maximum number of Vehicles that can be unloaded per day at location i (u_i)** given as the input in AMPL data file, and it is denoted by 'b_v', below we have displayed the same.

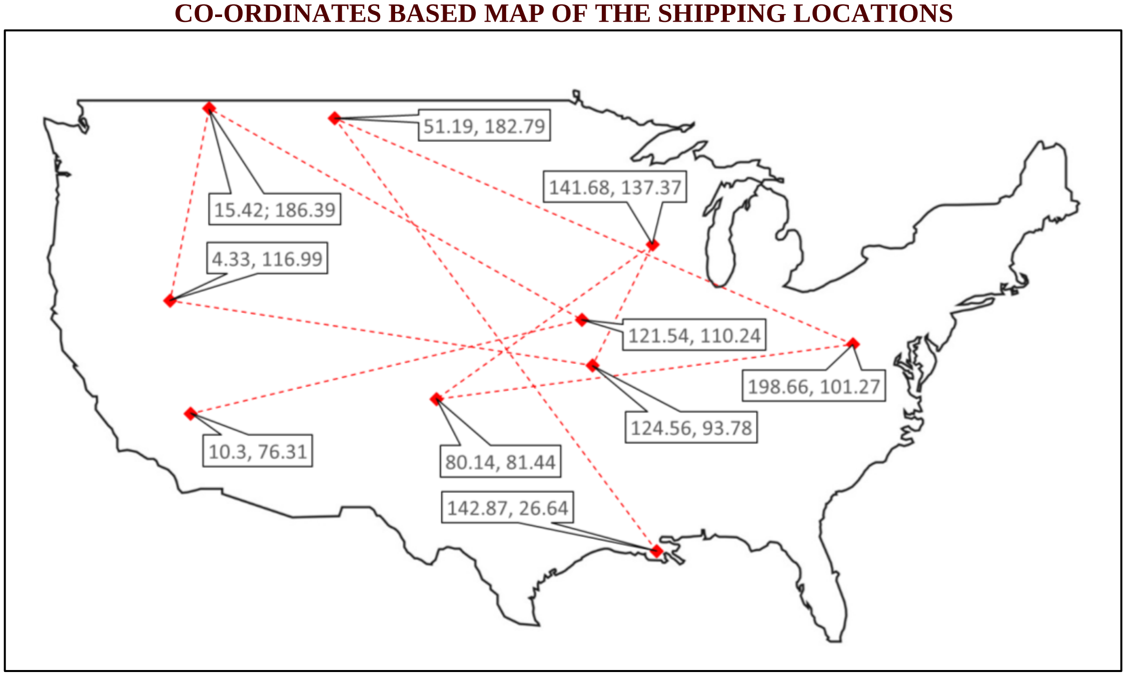
```
ampl: display u;
u [*] :=
 1  12
 2  12
 3  15
 4  14
 5  15
 6  16
 7  12
 8  12
 9  13
10  16
;
```

16. **Cost of relocating one operator from location i to location j (cm_{ij})** given as the input in AMPL data file, and it is denoted by 'cm_{ij}', Operators are paid at the rate of $2 per mile so depending on than value we have calculated the below values and entered them in the data files

```
ampl: display cm;
cm [*,*]
:     1     2     3     4     5     6     7     8     9    10    :=
1     0   362   186   166   222   140   330   408   172   284
2   362     0   338   210   202   230   162    72   202   228
3   186   338     0   240   134   148   390   404   156   380
4   166   210   240     0   166    92   168   246   100   140
5   222   202   134   166     0    94   278   270    68   290
6   140   230   148    92    94     0   244   286    34   232
7   330   162   390   168   278   244     0   140   234    82
8   408    72   404   246   270   286   140     0   262   220
9   172   202   156   100    68    34   234   262     0   232
10  284   228   380   140   290   232    82   220   232     0
;
```

Apart from the above displayed parameters we have t_{vij} and co_{vij} too large to display.

CO-ORDINATES BASED MAP OF THE SHIPPING LOCATIONS



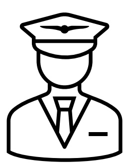
OUTPUT OBTAINED FROM AMPL WITH THE GIVEN DATA

Now all the input data which is to be considered are obtained and the AMPL software is used to solve for the given input data sets. The constraints and objective function defined are converted in the form of CODE and calculated using the AMPL software.

1. **Optimal Solution (Min Z):** For the given data, on executing the AMPL, we have arrived at the objective function that determines the minimum cost of shipping the packages under the given conditions. The optimal cost incurred are \$39008. Below I have attached the screen shot.

```
ampl: solve;
Gurobi 9.5.2: optimal solution; objective 39008
2103159 simplex iterations
14560 branch-and-cut nodes
```

2. **Decision Variable (z_{ij}) :** This is the number of operators relocated from location i to j.

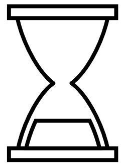


Operator relocated is defined as the operator who are travelling from their home location to another location (i.e., origin shipping center) to fulfil the needs of the problem statement. For this relocation the operators are paid a relocation allowance of \$2 per mile.

```
ampl: display z;
z [*,*]
:    1   2   3   4   5   6   7   8   9   10    :=
1    0   0   0   1   0   0   0   0   0   0
2    0   0   0   0   0   0   0   0   0   0
3    0   0   0   0   0   0   0   0   0   0
4    0   0   0   0   0   0   0   0   0   0
5    0   0   0   0   0   0   0   0   0   0
6    0   0   0   2   0   0   0   0   0   0
7    0   0   0   0   0   0   0   0   0   0
8    0   0   0   0   0   0   0   0   0   0
9    0   0   0   0   0   0   0   0   0   0
10   0   0   0   0   0   0   4   0   0   0
;
```

Totally there are 7 operators are relocating from their Homebase location

3. **Binary Variable (T_p) :** This is a binary variable for indicating if a penalty is implemented in the overall cost or not. If a penalty exists means the package has not been delivered within the time limit provided. If this happens the product should also be delivered free of cost which would ultimately result in loss. For each package there exist the separate penalty cost provided.



```
ampl: display T;
T [*] :=
  1 0    19 0    37 0    55 0    73 0    91 0   109 0   127 0   145 1   163 0
  2 0    20 1    38 0    56 0    74 0    92 0   110 0   128 0   146 0   164 1
  3 0    21 0    39 0    57 0    75 0    93 0   111 0   129 0   147 1   165 0
  4 0    22 0    40 1    58 0    76 0    94 0   112 0   130 0   148 0   166 0
  5 0    23 0    41 0    59 0    77 0    95 0   113 0   131 0   149 1   167 0
  6 0    24 0    42 1    60 0    78 0    96 0   114 0   132 0   150 1   168 0
  7 0    25 0    43 0    61 0    79 0    97 0   115 0   133 0   151 0   169 0
  8 1    26 0    44 1    62 0    80 0    98 1   116 0   134 0   152 0   170 1
  9 0    27 1    45 1    63 0    81 1    99 0   117 1   135 0   153 0   171 0
 10 1    28 0    46 0    64 0    82 0   100 0   118 0   136 0   154 0   172 0
 11 0    29 0    47 0    65 0    83 0   101 0   119 0   137 1   155 0   173 0
 12 0    30 1    48 0    66 0    84 0   102 1   120 0   138 0   156 0   174 0
 13 0    31 0    49 0    67 0    85 0   103 0   121 1   139 0   157 0   175 0
 14 0    32 0    50 1    68 0    86 0   104 0   122 0   140 0   158 0   176 0
 15 1    33 0    51 0    69 0    87 0   105 0   123 0   141 0   159 0   177 0
 16 0    34 0    52 0    70 0    88 0   106 0   124 0   142 0   160 0   178 0
 17 0    35 0    53 1    71 0    89 0   107 0   125 0   143 0   161 0   179 0
 18 1    36 0    54 0    72 0    90 0   108 0   126 0   144 0   162 0   180 0
;
```

4. **Binary Variable y_{vij} :** This is a binary variable for indicating whether vehicle v goes from shipping center i to j. Since the data is so large we have kept only the glimpse of it. We can reproduce the same using the model file and data file submitted.

```
ampl: display y;
y [*,1,*]
:     1   2   3   4   5   6   7   8   9  10    :=
1     0   0   0   0   0   0   0   0   0   0
2     0   0   0   0   0   0   0   0   0   0
3     0   0   0   0   0   0   0   0   0   0
4     0   0   0   0   0   0   0   0   0   0
5     0   0   0   0   1   0   0   0   0   0
6     0   0   0   0   0   0   0   0   0   0
7     0   0   0   0   0   0   0   0   0   0
8     0   0   0   0   0   0   0   0   0   0
9     0   0   0   0   0   0   0   0   0   0
10    0   0   0   0   0   0   0   0   0   0
11    0   0   0   0   0   0   0   0   0   0
12    0   0   0   0   0   0   0   0   0   0
13    0   0   0   0   0   0   0   0   0   0
14    0   0   0   0   0   0   0   0   0   0
15    0   0   0   0   0   0   0   0   1   0
16    0   0   0   0   0   0   0   0   0   0
17    0   0   0   0   0   0   0   0   0   0
18    0   0   0   0   0   0   0   0   0   0
19    0   0   0   0   0   0   0   0   0   1
20    0   0   0   0   0   0   0   0   0   0
21    0   0   0   0   0   0   0   0   0   0
22    0   0   0   0   0   0   0   0   0   0
23    0   0   0   0   0   0   0   0   0   0
24    0   0   0   0   0   0   0   0   0   0
25    0   0   0   0   0   0   0   0   0   0
26    0   0   0   0   0   0   0   0   0   0
27    0   0   0   0   0   0   0   0   0   0
28    0   0   0   0   0   0   0   0   0   0
29    0   0   0   0   0   0   0   0   0   0
```

5. **Binary Variable x_{pv} :** This is a binary variable for indicating which package goes in which vehicle. Since the data is so large we have kept only the glimpse of it. We can reproduce the same using the model file and data file submitted.

```
ampl: display x;
x [*,*]
:     1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19 :=
1     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
2     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
3     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
4     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
5     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
6     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
7     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
8     0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
9     0   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
10    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
11    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
12    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
13    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
14    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
15    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
16    0   0   0   0   0   0   0   0   0   0   0   0   0   0   1   0   0   0   0
17    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
18    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
19    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
20    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
21    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
22    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
23    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
24    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
25    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
26    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
27    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
28    0   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
29    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
```

SENSITIVITY ANALYSIS

<u>Mandatory</u>

1. My son is sad that no planes are currently used. What is the maximum operating cost for a plane for which it is optimal to use at least one plane?

This simply demands us to find the reduced cost of operating the plane, so that we have at least one plane included in our operations of delivering all the packages.

To answer this question, we have gone with the trial-and-error method, incorporating various values of cost of operating the plane in our data file. Firstly, by executing the AMPL code for our parent model file, with the inputs in the data file as per the initial given data, we get the number of planes operating as zero. This can be inferred by referring the below decision variable highlighted in bold. When we display the decision variable var y {v in 1..V,i in 1..L, j in 1..L}, we get the values in the matrix from vehicle {96 to 108} as zero. Please note that, we have defined vehicle v = 96…108 as planes.

var x {p in 1..P, v in 1..V} binary;

var y {v in 1..V, i in 1..L, j in 1..L} binary;

var z {i in 1..L, j in 1..L}≥0, integer;

var T {p in 1..P} binary;

This question demands us to find the reduced cost of plane, so that we can operate at least one plane in our logistics. This is determined by reducing the value of cost of operation of plane.

```
ampl: display y;
y [*,1,*]
:     1   2   3   4   5   6   7   8   9   10
1     0   0   0   0   0   0   0   0   0    0
2     0   0   1   0   0   0   0   0   0    0
3     0   0   0   0   0   0   0   0   0    0
```

i.e., $50 per mile multiplied by the distance between each respective location, on trial-and-error basis and feeding them into the data file.

At this juncture, on reaching the value of $46 per mile starting from $50 per mile, we are getting one plane to be operated as can be inferred from the below screenshot of AMPL display of variable var y[v,i,j].

91	0	0	0	0	0	0	0	0	0	0
92	0	0	0	0	0	0	0	0	0	0
93	0	0	0	0	0	0	0	0	0	0
94	0	0	0	0	0	0	0	0	0	0
95	0	0	0	0	0	0	0	0	0	0
96	0	0	0	0	0	0	0	0	0	0
97	0	0	0	0	0	0	0	0	0	0
98	0	0	0	0	0	0	0	0	1	0
99	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0
101	0	0	0	0	0	0	0	0	0	0

2. Imagine that your boss, who is not as bright as an engineer/mathematician, is mad at you because of the large number of late packages which look bad on the company. To allow them to choose the best tradeoff between the competing "goals" of cost vs number of allowed late packages, create a graph that plots the optimal cost (y-axis) vs the maximum number of allowed late packages (x-axis). Hint: you will need to add a constraint to be to get the values for this graph. Hint2: If you are clever, you won't need to solve for "too many" values on the x-axis; regardless of you MUST justify in your report how you determined the maximum value on your x-axis. BTW, in my opinion (which is the LAW in this class) this is the best way to address bi-objective optimization problems with conflicting objectives. What we are doing here is computing the so-called set of Pareto-optimal solutions.

The case here asks us to limit the number of late packages and to find a best tradeoff between with the number of late packages and cost incurred to the company.

This will usually be considered to protect the brand value and the service that the company offers to its customers.

We all know from the problem statement and the model file that; the packages get delivered through the lease cost mode. Following the same, certain packages can also get delivered by incurring penalty. Here, we are limiting the number of late packages and are trying to find the cost to the company on performing the same.

Firstly, on running the model with the raw data provided, we find that the number of late packages that incurs penalty is 25.

This can be seen from the below screenshot by displaying T_p. The binary values when added, we are getting the value as 25.

```
ampl: display T;
T [*] :=
  1 0    19 0    37 0    55 0    73 0    91 0   109 0   127 0   145 1   163 0
  2 0    20 1    38 0    56 0    74 0    92 0   110 0   128 0   146 0   164 1
  3 0    21 0    39 0    57 0    75 0    93 0   111 0   129 0   147 1   165 0
  4 0    22 0    40 1    58 0    76 0    94 0   112 0   130 0   148 0   166 0
  5 0    23 0    41 0    59 0    77 0    95 0   113 0   131 0   149 1   167 0
  6 0    24 0    42 1    60 0    78 0    96 0   114 0   132 0   150 1   168 0
  7 0    25 0    43 0    61 0    79 0    97 0   115 0   133 0   151 0   169 0
  8 1    26 0    44 1    62 0    80 0    98 1   116 0   134 0   152 0   170 1
  9 0    27 1    45 1    63 0    81 1    99 0   117 1   135 0   153 0   171 0
 10 1    28 0    46 0    64 0    82 0   100 0   118 0   136 0   154 0   172 0
 11 0    29 0    47 0    65 0    83 0   101 0   119 0   137 1   155 0   173 0
 12 0    30 1    48 0    66 0    84 0   102 1   120 0   138 0   156 0   174 0
 13 0    31 0    49 0    67 0    85 0   103 0   121 1   139 0   157 0   175 0
 14 0    32 0    50 1    68 0    86 0   104 0   122 0   140 0   158 0   176 0
 15 1    33 0    51 0    69 0    87 0   105 0   123 0   141 0   159 0   177 0
 16 0    34 0    52 0    70 0    88 0   106 0   124 0   142 0   160 0   178 0
 17 0    35 0    53 1    71 0    89 0   107 0   125 0   143 0   161 0   179 0
 18 1    36 0    54 0    72 0    90 0   108 0   126 0   144 0   162 0   180 0
;
```

To limit the number of late packages, we should first introduce an additional constraint that limits the same. That is shown below (highlighted in bold)

Operatorconstraint{i in 1..L}:sum{j in 1..L,v in 1..V}y[v,i,j]+sum{j in 1..L}z[i,j]-sum{j in 1..L}z[j,i]<=a[i];

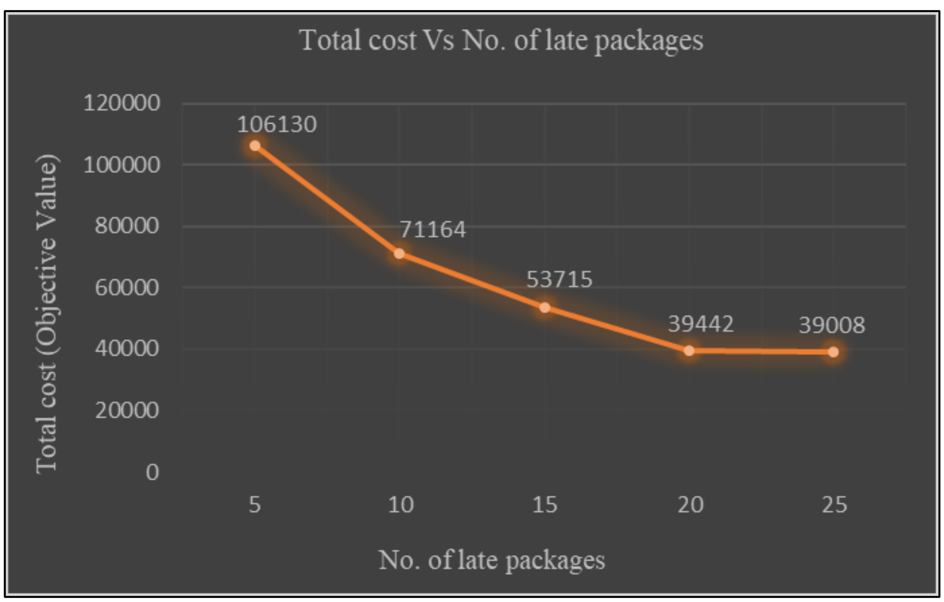
Weightconstraint{v in 1..V}:sum{p in 1..P}w[p]*x[p,v]<=h[v];

Packagepenalty:sum{p in 1..P}T[p]<=20;

In the above AMPL code defined, we are limiting the value of number of late packages as 20, and by solving the same, we get the below objective value shown.

```
C:\Users\Karthikeyan\Desktop
ampl: reset;reset;
ampl: option solver gurobi;
ampl: model"C:\Users\Karthikeyan\Desktop\AMPL\ampl_mswin64\ampl_mswin64\Updatedp16sensitivity2.txt";
ampl: data"C:\Users\Karthikeyan\Desktop\AMPL\ampl_mswin64\ampl_mswin64\Updatedp16data.txt";
ampl: solve;
Gurobi 9.5.2: optimal solution; objective 39008
1681189 simplex iterations
8744 branch-and-cut nodes
ampl: reset;reset;
ampl: option solver gurobi;
ampl: model"C:\Users\Karthikeyan\Desktop\AMPL\ampl_mswin64\ampl_mswin64\Updatedp16sensitivity2.txt";
ampl: data"C:\Users\Karthikeyan\Desktop\AMPL\ampl_mswin64\ampl_mswin64\Updatedp16data.txt";
ampl: solve;
Gurobi 9.5.2: optimal solution; objective 39442
203840 simplex iterations
808 branch-and-cut nodes
ampl:
```

In the similar way, on repeating the same procedure and by changing the values for T_p as 5, 10, 15, 20 & 25 we get the objective values as 106130, 71164, 53715, 39442 & 39008. The same is being plotted in graph as below. It is inferred that, as we limit the number of late packages the objective value is getting the increased, meaning that, the packages get delivered by following a quicker delivery mode wherever possible.



3. ALL of the computations related to this answer must be performed in the SAME computer (with no other task being run in parallel). This question is not a traditional sensitivity analysis question in that it doesn't provide managerial insight; instead, it shows how sensitive the solution time is to the nasty big-Ms.
   a. Solve the (BigM-less) model with the 'x' and 'n' variables: report the time it took to solve.

The Big M less model is coded in AMPL and solved optimally including the decision variable x_{pv} and n_{pv}. Here the model differs from the Big M models from the fact that, the decision variable x_{pv} and n_{pv} determines whether the package has arrived on time or late respectively. This is related through the constrains 'packages can be shipped with one vehicle, and all are shipped by including both these binary variables. Also, with the penalty constraint, the value x_{pv} can take the value 1, only if the calculated time is less than the deadline. If that is the case n_{pv} is zero. Thereby, not using the Big M values. On solving the Big M less model, the time take by AMPL to determine the solution is 98.96 seconds.

   b. For the BigM model, calculate the "theoretical" smallest correct value for each BigM for the model to be correct (recall that even constraints of the same type can have different BigMs). Report all of these BigM values.

For solving the Big M model, we need to calculate the value of M which is the maximum calculated time, that would be possible so that the penalty constraint defined in the model is satisfied. The calculated time in the LHS of the equation (penalty constraint) should be less than the summation of the deadline and the M value. With this the variable Tp will take values '1' or '0' to decide whether the package has to incur the penalty cost or not. By calculating the Big M values of each of the package, (calculating the distance the respective package should cover and the speed of the vehicle in which it is assigned), we have values ranging from 5.7 to 204. However, **by**

solving the AMPL assigning these values of ‘M’, we conclude that, the lease possible value for M is 87.

This is proved by solving the AMPL, for an M value of 86. Now, this can be inferred from the below AMPL output that the objective function is changing.

param L = 10;
param P = 180;
param V = 120;
param K = 18;
param M = 86;

```
Gurobi 9.5.2: optimal solution; objective 39063
2171228 simplex iterations
9660 branch-and-cut nodes
```

c. Solve the BigM model setting each BigM to its respective smallest value. Report the solution time. both in absolute value and relative to the time computed in (a).

Here the values of the smallest ‘M’ for each package is determined. The solution time for solving the model assigning these small M values agains packages are tabulated below. Here,we can see that from the data of absolute and relative values, the time reqiuired to solve for the max of the smallest value (1.e) M=204 is 240 seconds, which is higher relatively by 142 seconds.

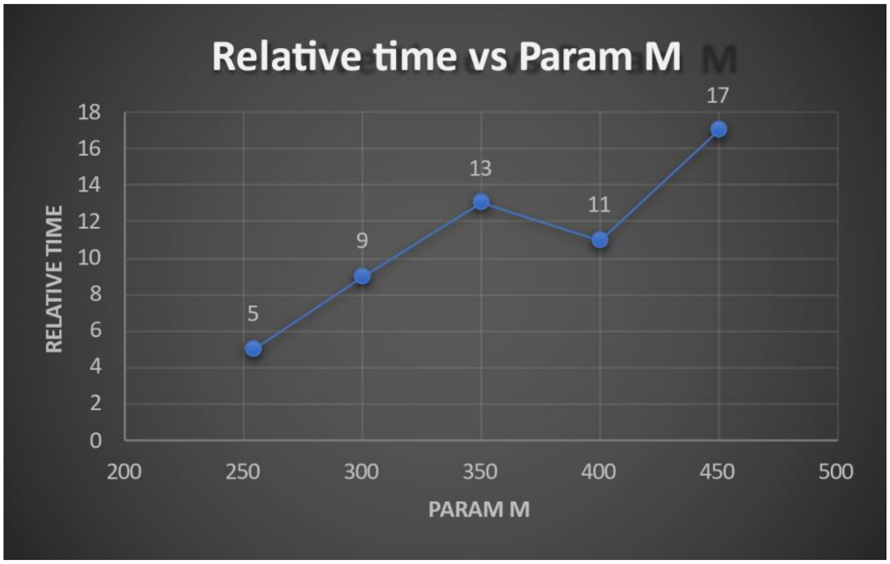
Also, for the smallest value of Big M (1.e) M=87, the solution time for solving is 105 seconds which is relatively 7 seconds higher than the time required to solve the big M less model.

d. Denote the value of the largest BigM as MegaM. Create a new BigM model, where all the BigM. values are set equal to a parameter Param M, whose initial value is MegaM. Report the solution time of the MegaM model in absolute value, relative to the time computed in (a), and relative to the time computed in (c).

The largest Big M is 204. The model is then solved by assigning the values of M to be greater than 204 from 250,300, 350 & 400. The values are then solved in model and the respective solution time is tabulated in absolute value and relative to the time calculate on running through bigM less model and small M value. It can be inferred that the time required for computing the model is increaseing for higher values of M.

e. Create a graph/plot of the "Solution Time Relative to time in (c)" vs. the value of ParamM (starting from MegaM and increasing thereafter). Plot enough points in the graph so that its shape is 'clear' In addition to including the graph in your report, also try to find the function who best fits your graph and report it. What are your takeaway conclusions from this part (e)?

The below graph is plotted for the solution time computed from assigning the Big M values starting from 204 and relative time with respect to the solution time derived by the incremental Big M values calculated from c.



Additional

1. The answer to the first what-if question above is a scalar (ie, the answer is a single number) and can be obtained solely by changing the data of the problem.

If the Capacity of the Loading and Unloading are increased first by 5% and second time by 20% of original capacity, then we have obtained the following change in the objective value.

- If the Loading and Unloading is increased by 5%, then overall cost of shipping has been reduced to $38434.
- If the Loading and Unloading is increased by 20% then overall cost of shipping has been reduced to $38285.

Un-Loading	Loading	Location (i)	Loading	Un-Loading
5% increase			**20% increase**	
13	17	1	19	14
13	14	2	16	14
16	17	3	19	18
15	15	4	17	17
16	16	5	18	18
17	15	6	17	19
13	15	7	17	14
13	17	8	19	14
14	14	9	16	16
17	16	10	18	19
Objective value - 38434			**Objective value - 38285**	

2. Create and solve ONE additional what-if question with a scalar answer that is obtained by modifying the model (and, if needed, also its data).

Here we are trying to limit the total number of operators reallocated between the various shipping locations. Suppose there arises a situation, where the operators cannot be reallocated, due to unavoidable circumstances. Then, this model defines the cost that is incurred to the company additionally in that scenario to plan for the best possible decisions and devise alternate options to optimize the operational cost.

Here, the model is modified by defining an additional constraint $z_{ij} \leq$ n.

When there is no limitation in reallocation of the operators (i.e.,) n=116 (max available operators), we get the objective value solution as $39,008 as mentioned below.

```
Gurobi 9.5.2: optimal solution; objective 39008
897389 simplex iterations
3878 branch-and-cut nodes
```

Here, we display the decision variable z_{ij} when there is no limitation in the operator relocation. From the below, graph AMPL output, we can find that there are totally seven operators being reallocated.

```
ampl: display z;
z [*,*]
:    1   2   3   4   5   6   7   8   9   10    :=
1    0   0   0   1   0   0   0   0   0   0
2    0   0   0   0   0   0   0   0   0   0
3    0   0   0   0   0   0   0   0   0   0
4    0   0   0   0   0   0   0   0   0   0
5    0   0   0   0   0   0   0   0   0   0
6    0   0   0   2   0   0   0   0   0   0
7    0   0   0   0   0   0   0   0   0   0
8    0   0   0   0   0   0   0   0   0   0
9    0   0   0   0   0   0   0   0   0   0
10   0   0   0   0   0   0   4   0   0   0
;
```

Now, we define the value of 'n' as 4, 8 & 10, literally limiting the number of operators to be reallocated. Below is the example of AMPL code defining the number of operators to be limited to 5. (Shown below)

Operator constraint {i in 1..L}: sum{j in 1..L,v in 1..V}y[v,i,j]+sum{j in 1..L}z[i,j]-sum{j in 1..L}z[j,i]<=a[i];
Operatorconstraint2: sum {i in 1..L, j in 1..L} z[i,j]<=5;

On executing the AMPL, we get the objective value function as $40068.

```
Gurobi 9.5.2: optimal solution; objective 40068
332436 simplex iterations
2238 branch-and-cut nodes
ampl: display z;
z [*,*]
:    1   2   3   4   5   6   7   8   9   10    :=
1    0   0   0   0   0   0   0   0   0   0
2    0   0   0   0   0   0   0   0   0   0
3    0   0   0   0   0   0   0   0   0   0
4    0   0   0   0   0   0   0   0   0   0
5    0   0   0   0   0   0   0   0   0   0
6    0   0   0   2   0   0   0   0   0   0
7    0   0   0   0   0   0   0   0   0   0
8    0   0   0   0   0   0   0   0   0   0
9    0   0   0   0   0   0   0   0   0   0
10   0   0   0   0   0   0   3   0   0   0
;
```

The various values of objective function against the maximum limit of reallocated operators are tabulated below from executing the AMPL codes.

Below is the tabulated value of various values of 'n' (i.e.) maximum number of operators who can be reallocated vs the objective value function.

Max operator that can be reallocated	**Objective value**
4	43157
5	40068
6	39512
7	39112

3. The answer to the second and third what-if question are graphs (of non-analytical You must create and solve ONE additional what-if question with a graph answer.

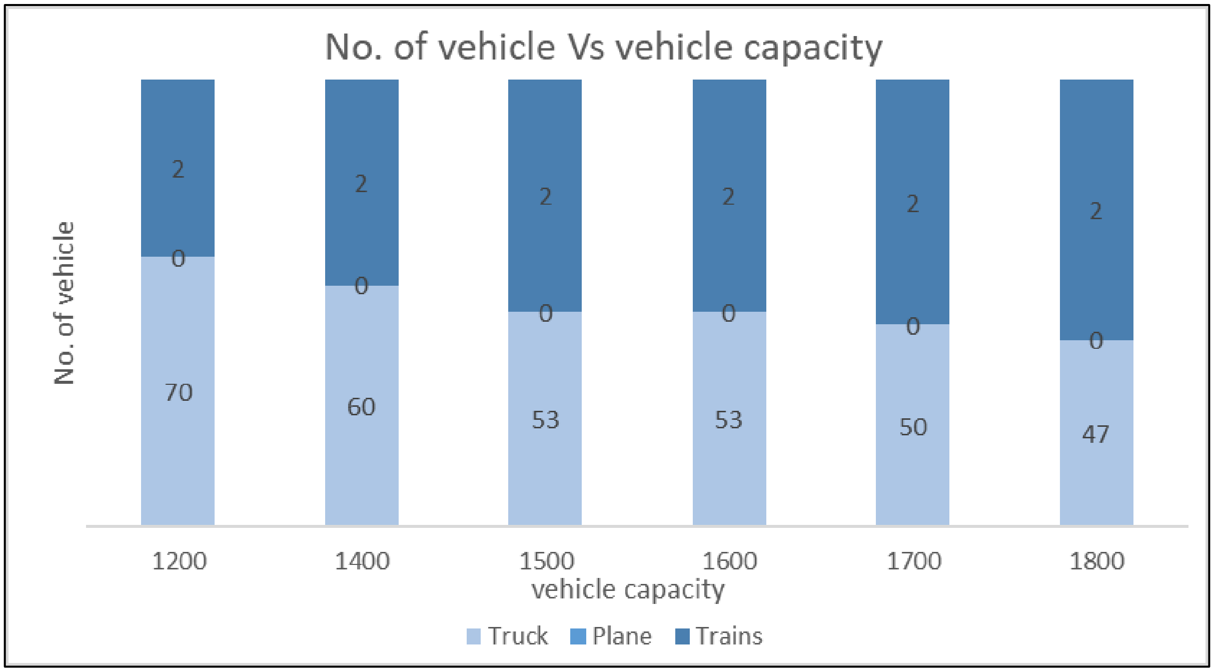
Here, we are trying to **find the total number of trucks, trains and planes** that are used among the available vehicle to deliver the packages and **what will be their relative value** if the **capacity** of these vehicles is **increased**. In the model, we are including the variable n, n_p & n_r in the weight capacity constraint to calculate the total number of trucks, planes and trains that are used for different scenarios.

var n>=0, integer;
var n_p>=0, integer;
var n_r>=0, integer;

In various supply chains operations, there will be situations to reduce the logistics cost, companies either try to increase the capacity of the current fleet or try optimizing the packaging efficiency to reduce the number of operating vehicles. Considering the same, we are exploring the scenario, where we are trying to increase the capacity of trucks, which is deployed in large numbers currently by increasing the capacity to 1200, 1400, 1500, 1600, 1700, 1800 kgs.

```
Weightconstraint:sum{p in 1..P,v in 1..96}w[p]*x[p,v]<=n*1800;
Plane:sum{p in 1..P,v in 97..108}w[p]*x[p,v]<=np*600;
Train:sum{p in 1..P, v in 109..120}w[p]*x[p,v]<=nr*6000;
```

The above AMPL code is an example of one of the trials done by increasing the capacity of truck to 1700kgs. Below is the AMPL screen shot of the display code. Please note that, with the increase in the capacity of the vehicles, the objective value greatly decreases because the number of vehicles operating between various locations decreases. However, there might be marginal increase in the operating cost of the vehicles with larger capacity, that should be considered before the tradeoff. This part of analysis is out of the scope of discussion.



- From the above graph, we can observe that with the gradually increase in the capacity of the vehicles, the no. of trucks utilized are decreasing, and the no. of trains utilized are increasing

CONCLUSION

In phase 1 of this project, we developed a model that solves the complex problem of middle mile logistics for shipping packages to various destinations. After the feedback on phase 1 report, slight modifications have been made to prepare this final consultation report. In this report, we started by outlining the objective of this project and then we give an executive summary of this report that highlights the important

aspects of this report before discussing the nature of the problem in the introduction and giving the model in its entirety. The report discusses the methodology which includes various parameters decision variables and constraints. The problem statement along with its solution is discussed in this report. To check the model behavior concerning changes in control parameter, a sensitivity analysis has been performed.

The total cost incurred (objective value) was found to $39008. From the sensitivity analysis, it was found that when the number of late packages were decreased, the total cost (objective value) was observed to increase.

APPENDIX

We aim to model effective planning and scheduling methods by optimizing the schedule of vehicles and operators for the OOPS shipping company.

The objective function is to minimize the Shipping cost:

Minimize Z $\sum_{p=1}^{P} \sum_{v=1}^{V} cl_p * n_{pv} + \sum_{v=1}^{V} \sum_{i=1}^{L} \sum_{j=1}^{L} co_{ijv} * y_{ijv} + \sum_{i=1}^{L} \sum_{j=1}^{L} cm_{ij} * z_{ij}$

CONSTRAINTS

1. Packages can only be shipped with one vehicle, and all new shipped
   $\sum_{v=1}^{V} x_{pv} = 1$ for p = 1 ,.., P
2. Vehicles can only travel on one root
   $\sum_{i=1}^{L} \sum_{j=1}^{L} y_{vij} \leq 1$ for v = 1 ,.., V
3. Packages must be shipped on a vehicle that is going from their origin to their destination
   $x_{pv} \leq y_{vs_pe_p}$ for p = 1 ,.., P for v = 1 ,.., V
4. Packages must reach their destination within the time limit if they are shipped without penalty
   $\sum_{v=1}^{V} \sum_{i=1}^{L} \sum_{j=1}^{L} t_{vij} * x_{pv} \leq f_p + M * T_p$ for p = 1 ,.., P
5. Each location can't exceed the limit on loading per day
   $\sum_{j=1}^{L} \sum_{v=1}^{V} y_{vij} \leq l_i$ for i = 1 ,.., L
6. Each location can't exceed the limit on unloading per day
   $\sum_{i=1}^{L} \sum_{v=1}^{V} y_{vij} \leq u_j$ for j = 1 ,.., L
7. Incompatible packages each need a separate vehicle
   $xI_{k1,v} + xI_{k2,v} \leq 1$ for k = 1 ,.., k, for v = 1 ,.., V
8. Cannot exceed the number of operators available in each location
   $\sum_{j=1}^{L} \sum_{v=1}^{V} y_{vij} + \sum_{j=1}^{L} z_{ij} - \sum_{j=1}^{L} z_{ji} \leq a_i$ for i = 1 ,.., L
9. Packages cannot exceed the weight of vehicles
   $\sum_{p=1}^{P} w_p * x_{pv} \leq h_v$ for v = 1 ,.., V
10. Non-negativity
    $z_{ij} \geq 0$ for i = 1 ,.., L, for j = 1 ,.., L
11. Integrality
    z_{ij} is integer for i = 1,.., L, for j = 1,.., L
12. Binary requirements
    x_{pv} is binary for p = 1 ,.., P, for v = 1 ,.., V
    y_{vij} is binary for v = 1 ,.., V, for i = 1 ,.., L, for j = 1
    T_p is binary for p = 1 ,.., P

REFERENCE

1) Isnafitri, M & Rosyidi, Cucuk & Aisyati, Azizah. (2021). A Truck Allocation Optimization Model in Open Pit Mining to Minimize Investment and Transportation Costs. IOP Conference Series: Materials Science and Engineering. 1096. 012024. 10.1088/1757-899X/1096/1/012024.
2) Ghiasvand, S., & Ejlaly, B. (2014). Integrated and periodic relief logistics planning for reaction phase in uncertainty condition and model solving by particles swarm optimization algorithm. European Online Journal Of Natural And Social Sciences, 3(1), pp. 92-101.
3) Karmaker, Chitra & Halder, Pobitra. (2017). Scheduling Project Crashing Time Using Linear Programming Approach: Case Study. International Journal of Research in Industrial Engineering (In press).
4) Greening, Lacy & Dahan, Mathieu & Erera, Alan. (2022). Lead-Time-Constrained Middle Mile Consolidation Network Design with Fixed Origins and Destinations.
5) Li Zhang, Zhongshan Liu, Lan Yu, Ke Fang, Baozhen Yao, Bin Yu, Routing optimization of shared autonomous electric vehicles under uncertain travel time and uncertain service time, Transportation Research Part E: Logistics and Transportation Review, Volume 157, 2022, 102548, ISSN 1366-5545,