# Exact and Heuristic Optimization Algorithms for Auto-rack Loading Problem

**Ashok Viswanathan[1], Bharath Narayanan[2], and Nagen Nagarur [3]**

[1] Best Buy Corp.,
7601 Penn Avenue,
South Richfield, MN 55423, USA

[2] Department of Industrial Engineering & Management,
Oklahoma University,
Stillwater, OK 74078, USA

[3] Department of Systems Science and Industrial Engineering,
State University of New York, Binghamton
Binghamton, NY 13902, USA

Corresponding author's Email: nnagarur@binghamton.edu

**Abstract:** The transportation of new vehicles over the rail network takes place in specially designed railcars / wagons called auto-racks. This paper introduces a new problem – Auto-Rack Loading Problem (ARLP), involving transportation of vehicles on auto-racks with the objective of maximizing the revenue from the loaded vehicles subject to constraints. Existing literature only addresses methods to model and optimize transportation of vehicles on trucks called the auto-carrier loading problem. Initially, a model is formulated as an integer programming problem which is proven to be NP-hard since a special case of the ARLP reduces to a 0-1 multiple knapsack problem. Consequently, a multi-phase heuristic procedure is developed to solve large instances of the problem in reasonable time. The performance of the integer programming model and the heuristic procedure are evaluated and compared with an upper bound for randomly generated instances involving auto-racks and vehicles designed for use in the Indian rail network.

*Keywords:* Vehicle Transportation, Heuristics, Finished Vehicle Delivery, Rail, Knapsack Problem

## 1. Introduction

An automotive supply chain comprises of two major components, namely inbound parts and finished vehicles (FV). For inbound parts logistics, parts are delivered from geographically spread suppliers to the assembly plant in a time-bound manner. The transportation of FV from the plant to the dealer constitutes the deliverable of an FV supply chain (Viswanathan et al., 2011). The delivery of new vehicles from the plant to the dealers is usually multi-modal, involving ocean shipping, rail, and trucks depending on the source and demand regions. Ocean shipping is used for inter-continental and long distance transportation. Rail is adopted as the mode of transportation when the origin plant region and the destination dealer region are connected by a rail network. Trucks transport vehicles directly to the dealers in close proximity to the plant, ports and rail terminals and between regional hubs that are not connected by a rail network. According to the Association of American Railroads (AAR), freight transportation by rail is on an average four times more fuel efficient than trucks thus reducing greenhouse gas emissions by 75% (Association of American Railroads, 2016). The transportation of new vehicles by rail takes place on a type of railcar / wagon called the auto-rack. Auto-racks are railcars with either a two and three level vehicle carrying structure also called bi-levels and tri-levels respectively as shown in Figure 1. The decks between the levels are movable depending on the size of the vehicle to be loaded in a level. Vehicles are loaded onto the auto-racks using special ramps and are attached to rails in the auto-rack using tie-down devices.


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

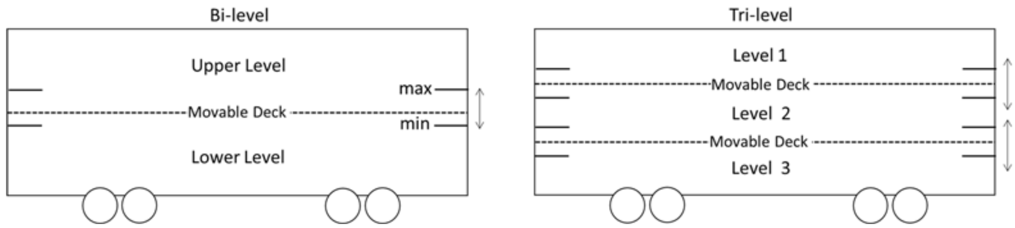


Figure 1. Schematic of a bi-level and tri-level auto-rack

Based on regional regulations, there is a limit to the number of auto-racks a train / rake can haul. From a service provider's perspective, maximizing the number of vehicles loaded in a train (load factor) increases the revenue / profit. A number of factors influence the load factor of vehicles in an auto-rack, example:

- Size and composition of the vehicles available for loading.
- Clearances needed to prevent abrasion between the vehicle and the inner surface of the auto-rack and between vehicles.
- Size of the auto-rack and the position of the movable deck.

The problem of maximizing the number of vehicles / revenue for transportation of vehicles by rail is the focus of this paper and is christened here as the Auto-Rack Loading Problem (ARLP). The ARLP can be defined as: Given a set of vehicle models with pre-determined length and height dimensions, and a train with a given set of auto-racks, each with known length and height, load the vehicles to the auto-racks to maximize the revenue from transporting the loaded vehicles, without violating the space constraints of the auto-racks.

The paper is organized as follows, Section 2 – Literature Review details the current state of research in vehicle loading and knapsack problems, Section 3 – Problem Description and Mathematical Model defines the problem and illustrates an Integer Programming (IP) model, Section 4 – Heuristic Solution for the ARLP discusses about a multi-phase heuristic procedure to solve the ARLP. Section 5 – Computational Experiments compares the computational results of the IP model and the heuristic procedure with the upper bound for a dataset involving auto-racks and vehicles in India, and Section 6 – Conclusion summarizes the conclusions and future work.

## 2. Literature Review

Historically, the automotive logistics industry has been at the forefront of innovation in the area of supplier-production interaction. In contrast, the production-distribution leg of the chain is a subject of relatively modest academic interest (Turner and Williams, 2004). A study on the design of an outbound automotive supply chain network based on lead times, location of distribution centers and the choice of the transportation modes was conducted by Eskigun et al. (2005). The authors employed a capacitated network design model (a linear programming approach) to minimize the cost with constraints on lead time, dwell time, capacity of distribution center, and transportation mode. The capacity of the distribution center was found to have a significant effect on the lead time, usage of trucks, dwell time, and eventually the cost.

### 2.1 Vehicle Loading

The problem is to optimize the load makeup – load a set of vehicles onto trucks or auto-racks involves multiple objectives, namely minimizing the dwell time at yards, minimizing the travel distance, and maximizing the load factor throughout the network. Another phenomenon occurring in vehicle shipment yards considered in literature is to perform the optimization on a dynamic basis. Published literature on the distribution of vehicles using trucks is in the form of the Auto-Carrier Loading Problem (ACLP), which is described in Section 1. Given demand for vehicles from dealers with respect to the models and delivery schedule, the daily schedule of a heterogeneous fleet of trucks is planned. In order to cater to the dynamic nature of the vehicle availability, the problem is solved either repeatedly on a reduced planning horizon that shifts forward each time the algorithm is applied (Cordeau et al., 2015) or using a multi-agent system triggered on an event basis (Yee, 2014). Cordeau et al. (2015) have used iterative local search employing a variety of move operators while Lin (2010) has used a formulation based on the tree loading network to solve the problem. Yee (2014) has used a multi-agent market based computational heuristic while Wang et al. (2019) formulated an integer programming model and a column generation based


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

heuristic to achieve the objectives set forth. Bonassa et al. (2019) have implemented a mixed integer programming model to solve a real-world dynamic multi-period auto-carrier problem over multiple days' demand.

Notably, the absence of published literature related to loading vehicles onto auto-racks or rail based transportation makes this study the first-of-its-kind.

## 2.2 Knapsack Problem

As will be discussed in Section 3.2, the ARLP is analogous to the multiple knapsack problem. The knapsack problem is a classic combinatorial optimization problem in which a set of items have a certain value and weight needs to be packed into a knapsack having a weight capacity, such that the total value of items packed into the knapsack is maximized. Knapsack problems in real life scenarios appear in situations like finding the least wasteful way to cut raw materials, financial investments and portfolio allocation. One of the initial application of algorithms to the knapsack problem was in the design and scoring of test papers where test takers could choose the questions they wanted to answer. Feuerman and Weiss (1973) proposed a system where students are given a test containing questions worth 125 total points. The students are then asked to answer all questions to the best of their ability and the subset totaling 100 points which maximizes each student's points is chosen through a knapsack algorithm to determine the maximum score by the student.

The knapsack problem in its most basic form is called the 0-1 knapsack problem. Given a list of items each with a value and weight, the items need to be loaded into a knapsack adhering to its weight capacity and the total quantity of each item loaded into the knapsack can be either 0 or 1. One common variant of the basic knapsack problem is that each item can be chosen multiple times. This problem called the bounded knapsack problem specifies an upper bound for the quantity of each item to be chosen. Another variant called the unbounded knapsack problem does not impose any upper bound on the quantity of each item that can be loaded into the knapsack. The multiple choice knapsack problem is one in which the items are divided in $k$ classes and exactly one item should be selected from each class. If each item has the same weight and value it is called the subset sum problem. The variant of the knapsack problem where there are multiple knapsacks is called the multiple knapsack problem. A special case of the multiple knapsack problem where the weights and the value of the items are same is called the multiple subset sum problem. Finally, if there is more than one capacity constraint, for example a volume capacity constraint and a weight capacity constraint, we get the multiply constrained knapsack problem. The multiply constrained problem has a 0-1, bounded and unbounded variants (Martello and Toth, 1990).

Pisinger (1999) developed an exact algorithm for solving large multiple 0-1 knapsack problems. The recursive branch-and-bound algorithm applies surrogate relaxation for deriving upper bounds while lower bounds are derived by solving a series of subset-sum problems after splitting surrogate solution into $m$ knapsacks. Khuri et al. (1994) developed a genetic algorithm for solving the 0-1 multiple knapsack problem which allows for the breeding and participation of infeasible strings in the population. They use a simple fitness function that uses a graded penalty term to penalize infeasibly bred strings. Chekuri and Khanna (2005) developed a polynomial time approximation scheme for the multiple knapsack problem. They show that slight generalizations of the multiple knapsack problem is APX-hard. Shah-Hosseini (2008) modeled it with an intelligent water drops algorithm, a population-based optimization algorithm that is modified to include a suitable local heuristic for the multiple knapsack problem. The proposed algorithm is tested for standard problem instances and it is also proved that the algorithm has the property of convergence in value. In addition, several algorithms have been developed to solve knapsack problems using dynamic programming approaches (Andonov et al., 2000) and hybrid approach consisting of branch and bound and dynamic programming. (Poirriex et al., 2009; Martello et al., 1999; Plateau and Elkihel, 1985; Martello and Toth, 1984).

# 3. Problem Description and Mathematical Model

The current problem involves loading vehicles onto freight trains made up of specially designed railcars / wagons called auto-racks. The length of a freight train and the number of auto-racks per train depends on the regulations existing in the region, example maximum length of a train is 630 m in Sweden, 750 m in Germany, 600 m in Slovak Republic, 3650 m in Canada, 5500 m in United States (Backaker and Krasemann, 2012; Islam et al., 2015; Dolinayova et al., 2015; Transport Canada, 2013; Joiner, 2010). The dimensions and the number of auto-racks per train also vary depending on the characteristics of the rail network (electrification, height of bridges, curve negotiability) in the region.

For a service provider, while the cost of operating a freight train between two points is largely fixed, the revenue is dependent on the vehicles loaded in the train. Small sized vehicles generate lower revenue per unit but provide higher load factor. Large sized vehicles generate a higher revenue per unit but provide lower load factor. Given a set of vehicles comprising of vehicles models of various sizes, identifying a subset of vehicles that can be feasibly loaded onto the auto-racks in a train while maximizing the revenue from the loaded vehicles is the objective of this paper.



DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

**Assumptions & Parameters:**

- Bi-level auto-racks with one movable deck considered. Tri-level auto-racks with two movable decks are a part of future research.
- The movable deck that can be raised / lowered in discrete length increments to increase / decrease the height of the two levels.
- Vehicle models are heterogeneous with known dimensions and revenue per unit.
- The width of all vehicle models are less than the width of the auto-rack.
- Auto-racks are homogenous with known dimensions.
- In order to prevent damages to the vehicles due to inadvertent abrasion, clearances are configured
  - Vehicle-to-Vehicle Clearance: Minimum required space between two vehicles loaded in the same level of auto-racks.
  - Vehicle-to-Door Clearance: Minimum space between door of auto-racks to the nearest vehicle.
  - Vehicle-to-Roof Clearance: Minimum space required between the roof of the vehicle to the roof of the level in which the vehicle is loaded.

In this section, an Integer Programming (IP) model is presented to derive solutions to ARLP with a goal of maximizing the revenue per train. A methodology to determine the upper bound for each dataset is also explained.

## 3.1 Integer Programming (IP) Model Formulation for ARLP

The model formulation uses the following notation.

**Parameters:**

| | |
|---|---|
| $i$ | Index for the number of a vehicle model, $i = 1, 2, \dots, n$, |
| $j$ | Index for the number of a auto-rack in a train, $j = 1,2, \dots, m$, |
| $N_i$ | Number of vehicles of vehicle model $i$, |
| $L_i$ | Length of vehicle model $i$, |
| $R_i$ | Revenue generated by transporting one unit of vehicle model $i$ in the auto-rack, |
| $H_i$ | Height of vehicle model $i$, |
| $LL_j$ | Available length of lower level of auto-rack $j$, |
| $UL_j$ | Available length of upper level of auto-rack $j$, |
| $LH_j$ | Available height of lower level of auto-rack $j$, |
| $UH_j$ | Available height of upper level of auto-rack $j$, |
| $CTC$ | Vehicle-to-vehicle clearance required while loading, |
| $CAL$ | Clearance required after loading a particular level, |
| $D$ | Unit increment of movable deck position from initial position, |
| $MDP$ | Maximum allowable movable deck position in number of increments from initial position, |
| $P$ | A large positive number. |

**Decision Variables:**

| | |
|---|---|
| $XU_{ij}$ | Number of vehicles of vehicle model $i$ loaded in the upper level of auto-rack $j$, |
| $XL_{ij}$ | Number of vehicles of vehicle model $i$ loaded in the lower level of auto-rack $j$, |
| $YU_{ij}$ | 1 if vehicle model $i$ has been loaded in the upper level of auto-rack $j$; 0 otherwise, |
| $YL_{ij}$ | 1 if vehicle model $i$ has been loaded in the lower level of auto-rack $j$; 0 otherwise, |
| $S_j$ | Number of increments of the movable deck from its initial position in auto-rack $j$, |



DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

**Model:**

$$Max \sum_{i=1}^{n}\sum_{j=1}^{m}(XU_{ij} + XL_{ij})R_i \tag{1}$$

$$\sum_{j=1}^{m}(XU_{ij} + XL_{ij}) \leq N_i \quad \forall i \tag{2}$$

$$LL_j - CTC\left(\sum_{i=1}^{n} XL_{ij} - 1\right) - \sum_{i=1}^{n} XL_{ij}L_i \geq CAL \quad \forall j, j \in 1,2 \ldots M \tag{3}$$

$$UL_j - CTC\left(\sum_{i=1}^{n} XU_{ij} - 1\right) - \sum_{i=1}^{n} XU_{ij}L_i \geq CAL \quad \forall j \tag{4}$$

$$YL_{ij}H_i \leq LH_j + DS_j \quad \forall i, j \tag{5}$$

$$YU_{ij}H_i \leq UH_j - DS_j \quad \forall i, j \tag{6}$$

$$XU_{ij} - PYU_{ij} \leq 0 \quad \forall i, j \tag{7}$$

$$XU_{ij} - YU_{ij} \geq 0 \quad \forall i, j \tag{8}$$

$$XL_{ij} - PYL_{ij} \leq 0 \quad \forall i, j \tag{9}$$

$$XL_{ij} - YL_{ij} \geq 0 \quad \forall i, j \tag{10}$$

$$0 \leq S_j \leq MDP \quad \forall j \tag{11}$$

$$XL_{ij} \epsilon \ \mathbb{Z}^+ \quad \forall i, j \tag{12}$$

$$XU_{ij} \epsilon \ \mathbb{Z}^+ \quad \forall i, j \tag{13}$$

$$S_j \epsilon \ \mathbb{Z} \quad \forall j \tag{14}$$

$$YU_{ij}, YL_{ij} \epsilon \ \{0,1\} \quad \forall j \tag{15}$$

The objective function (1) is to maximize the total revenue generated due to transportation of the loaded vehicles. The total revenue generated is by calculating the product of total number of vehicles loaded of a particular vehicle model and the revenue of the respective vehicle model and summing over for all the loaded vehicle models in all the auto-racks. Constraint (2) ensure that the upper bound on the number of vehicles to be loaded for vehicle model $i$ is equal to $N_i$. Constraint (3) ensures that the total length of loaded vehicles in the lower level of any auto-rack, including applicable vehicle-to-vehicle clearances and a minimum clearance after loading does not exceed the length of lower level of the respective auto-rack. It is important to know that if there are $Z$ vehicles loaded onto any level of any auto-rack, there are exactly $Z - 1$ vehicle to vehicle clearances that need to be accounted for. In constraint (3) $\sum_{i=1}^{n} XL_{ij}$ is the total number of vehicles loaded in auto-rack $j$. Hence, we would need to account for $\left(\sum_{i=1}^{n} XL_{ij} - 1\right)$ car-to-vehicle clearances and $CTC\left(\sum_{i=1}^{n} XL_{ij} - 1\right)$ is the total vehicle-to-vehicle clearance in length to be accounted for. An implicit assumption is that the available length of the auto-rack considers the appropriate car-to-door clearances. Constraint (4) captures the length loading constraints as in constraint (3), but for the upper levels of each auto-rack. Constraint (5) ensures that the height of the vehicles loaded onto the lower level of the auto-racks does not exceed the height of the respective lower levels. An implicit assumption is that the available height of a particular level takes into account the vehicle-to-roof clearance.



DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

Constraint (6) enforces the height constraint for the upper level of the auto-racks in a similar fashion to constraint (5). Constraints (7) and (8) ensure that if $XU_{ij}$ is greater than 0, then $YU_{ij}$ is 1 and 0 if $XU_{ij}$ is 0. This translates to the requirement that $YU_{ij}$ should take the value of 1 if any unit of vehicle model $i$ is loaded onto the upper level of auto-rack $j$ or 0 otherwise. Similarly, constraints (9) and (10) enforce that $YL_{ij}$ will take the value of 1 if $XL_{ij}$ is greater than 0 or 0 otherwise. Constraint (11) ensures that the number of increments of the movable deck from its initial position is utmost equal to the maximum allowable increment and that the movable deck cannot be pushed down from its initial position for all auto-racks. Constraints (12) and (13) imply that $XL_{ij}$ and $XU_{ij}$ are non-negative integers for all $i, j$. Constraint (14) implies that $S_j$ is an integer. Finally, constraint (15) enforces that $YU_{ij}$ and $YL_{ij}$ are binary for all $i, j$.

## 3.2 Proof for NP-hardness of ARLP

The ARLP is an NP-hard problem. This problem can be analogized to a multiple knapsack problem (Martello and Toth, 1990) as shown in Table 1:

Table 1. Analogy between ARLP and multiple knapsack problem

| Item | ARLP | Multiple Knapsack Problem |
|---|---|---|
| Input 1 | Levels (upper / lower) in auto-rack | Knapsacks |
| Input 2 | Vehicles | Items |
| Objective Function (Maximization) | Revenue of vehicles loaded in all auto-racks in a train | Profit from items loaded in all knapsacks |
| Constraint 1 | ∑ Length of vehicles loaded in a level ≤ Length of level | ∑ Weight of items loaded in a knapsack ≤ Capacity of knapsack |

Notably, the second constraint of the ARLP, namely "height of tallest vehicle in a level ≤ height of the level" is not a part of the multiple knapsack problem. This absence has been used in this paper to develop a relaxation of the ARLP to derive an upper bound.

Let us consider a special case of the ARLP with the following characteristics:

- There can be no adjustments of the movable deck levels i.e. $MDP = 0$. This would cause all $S_j$ to take the value 0.
- The height constraints are not restrictive as the input dataset consists of vehicles of height lesser than the available heights of either the lower or upper levels. This will make constraints (5) and (6) redundant and would remove the requirement to declare $YU_{ij}$ and $YL_{ij}$ which will inturn cause constraints (7), (8), (9) and (10) redundant.
- All clearances i.e $CTC$ and $CAL$ are 0.
- $N_i = 1$ for all vehicle models which will imply that $XL_{ij}$ and $XU_{ij}$ will become binary.

It can be observed that the above special case of the ARLP is reduced to a 0-1 Multiple Knapsack Problem where the number of knapsacks are $2m$. Since the 0-1 Multiple Knapsack problem is a well-known NP-Hard problem (Martello and Toth, 1990), it can be inferred that the special case of the ARLP is a NP-Hard problem. Since the special case of the ARLP is a relaxation of the ARLP, the latter would also be a NP-Hard problem. Hence, to obtain efficient solutions to the ARLP, especially if the problem dimensions are large, we have developed a heuristic approach to solve challenging instances of the problem in practical time.



DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

## 4. Heuristic Solution for the ARLP

The objective of the ARLP is to maximize the revenue from loading a set of vehicles onto a set of bi-level auto-racks while adhering to the length and height constraints. While the ARLP is shown to be a special case of the multiple knapsack problem in Section 3.2, the interplay of the two dimensions of length and height induces complexity that needs to be taken into consideration while developing a solution. Importantly, identifying the position of the movable deck so that both the upper and lower levels can be loaded with the highest revenue vehicles is critical.

Greedy search algorithms can be used to load vehicles onto the auto-racks. However, due to the tendency of greedy search algorithms to get stuck at local optima, an improvement procedure is required to further improve the solution quality. Considering that the ARLP is an NP-hard problem, the aforementioned characteristics are used to devise a multi-phase heuristic procedure to arrive at near-optimal solutions in reasonable time even for large instances of the problem. The schematic of the multi-phase heuristic is illustrated in Figure 2 and explained in detail in the following section.

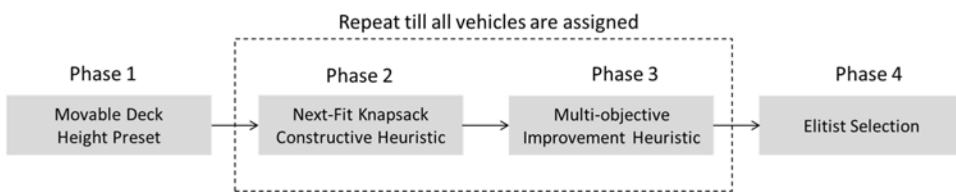


Figure 2. Heuristic Solution Phases

Phase 1 - Movable Deck Height Preset

Of the two dimensions of the ARLP, namely length and height, the length of auto-rack's levels is fixed while the height of a level is variable depending on the position of the movable deck. It can be observed that setting the position of the movable deck to maximize the revenue from loading vehicles in the two levels of the auto-rack will be critical to the performance of the heuristic procedure.

In this phase, every vehicle model will be assigned a default level (upper or lower) and a movable deck height that minimizes the height wasted. The assignment will be configured on the auto-rack if the vehicle model is the first vehicle to be loaded onto an auto-rack in Phase 2. The assignment that minimizes the height wasted is chosen as the default level and movable deck height. This step is relevant due to the fact that the movable deck moves in discrete increments, not continuous and thus has a significant impact on the performance of the heuristic as illustrated below:

$H_A$ → height of vehicle model A = 1590 units
$LH_1$ → height of lower level of the auto-rack '$1$' with the movable deck at 'min' position = 1474 units
$UH_1$ → height of upper level of the auto-rack '$1$' with the movable deck at 'min' position = 1693 units
$MDP$ → movable deck increment step = 50 units

Table 2. Numerical Illustration of Movable Deck Height Preset

| | Model A Default → Lower Level | Model A Default → Upper Level |
|---|---|---|
| Movable Deck Position for Minimal Wasted Height | 3 * $MDP$ = 150 units | 2 * $MDP$ = 100 units |
| Height wasted on Default Level | $LH_1$ + 3 * $MDP$ – $H_A$ = 34 units | $UH_1$ – 2 * $MDP$ – $H_A$ = 3 units |
| Tallest Allowable Vehicle Height on Non-Default Level | $LH_1$ – 3 * $MDP$ = 1543 units | $UH_1$ + 2 * $MDP$ = 1574 units |

It can be observed from the above Table 2 that assigning vehicle model A to default to the upper level is better (from a height perspective) since it allows taller vehicles to be loaded on the lower level.

Phase 2 - Next-fit knapsack constructive heuristic for initial solution

For every level of an auto-rack, an initial solution is generated using a constructive greedy heuristic, namely next-fit decreasing based on revenue. In addition to loading vehicles on both levels of the auto-rack, the position of the movable deck is set based on the first vehicle to be loaded. After generating the initial solution for the auto-rack, an improvement procedure (explained in Phase 3) is implemented to improve the revenue. The steps to perform next-fit decreasing are:

a) Sort all vehicles not yet assigned to an auto-rack by decreasing order of revenue.
b) Assign the first vehicle to a new auto-rack to the default level (upper or lower) of the vehicle model determined in Phase 1. Set the movable deck to the default deck position of the vehicle model also determined in Phase 1.
c) Perform next-fit decreasing till none of the unassigned vehicles can be assigned to the default level. The constraints for next-fit decreasing are length availability of the auto-rack and feasibility with respect to height of the level as set in Step b. The Vehicle-to-Vehicle, Vehicle-to-roof, and Vehicle-to-door clearances are preset parameters.
d) Resort all unassigned vehicles by height.
e) Repeat step c for the non-default level of the auto-rack.

At the end of Phase 2, both levels of the auto-rack have vehicles assigned to it based on next-fit decreasing.

Phase 3 - Improvement based on multi-objective decision making

Phase 3 involving improvement of the solution generated in Phase 2 is executed sequentially after completion of Phase 2 for every auto-rack. In this phase, a set of all feasible alternate solutions for the dominant vehicle model (the vehicle model with most units) in the auto-rack level is generated and the best alternate chosen. The vehicle model having the most units in the level is chosen as the dominant vehicle model since:

1) The greedy search procedure based on sorting unassigned vehicles by decreasing order of revenue in phase 2 will typically load similar vehicle models with higher revenue first in a level till there are no more unassigned units of the vehicle model
2) The vehicle model with more units in the level will have more possible (not necessary feasible) alternates than one with lesser units.

The steps to perform the improvement procedure are:

a) Devise a matrix comprising of two-way combinations of all vehicles models that can fit in a level of an auto-rack with the corresponding length of the auto-rack used. For example, for a given length *'l'* of an auto-rack, 3 units of vehicle model A and 3 units of vehicle model B are feasible using length $l_{ab}$ of the auto-rack, while 3 units of vehicle model A and 2 units of vehicle model C are feasible using length $l_{ac}$. where $l_{ab}$ & $l_{ac} <= l$.
b) From Phase 2, identify the dominant vehicle model in the level and list all two-way combinations with other vehicle models which are feasible with respect to the movable deck's position. For example, with vehicle model A being the dominant vehicle model and height of the current level $y_1$, only combinations involving vehicle models B, C, E are feasible. Vehicle models D and F are not feasible since the height of vehicle models D and F are greater than $y_1$. For each feasible combination $i$, record three fitness criteria, namely revenue ($R_i$), length of the auto-rack used ($L_i$), and total height of the vehicles ($H_i$) in the level. Even though the holistic objective of the heuristic procedure is to maximize revenue, the underlying premise of the multiple knapsack problem is to efficiently use the space (length and height) in the auto-rack (Kumar et al., 2008; Burke et al., 2005; Zitzler and Thiele, 1999). Since the improvement procedure is performed one auto-rack at a time, selecting a combination that only maximizes revenue for the current auto-rack might not yield the best results for upcoming auto-racks in the train due to space availability. Hence, during the selection process from among all feasible combinations, the fitness scores related to length and height occupancy are also taken into consideration (in addition to revenue) and the combination that is best with respect to all fitness criteria is selected.
c) Normalize each fitness criteria between 0 and 1. For example, the normalized value for revenue $R_i$ is $(R_i - R_{min})/(R_{max} - R_{min})$, where $R_{min}$ is the minimum revenue and $R_{max}$ is the maximum revenue among the feasible combinations for the level. Normalization is performed to equalize the range of all three fitness criteria.
d) In order to select the best combination among all feasible combinations, a weighted sum of the fitness criteria is calculated. The value for the weights that maximizes the revenue is determined from a sensitivity analysis.
e) Sort all feasible combinations in decreasing order of total fitness score.
f) The combination with the maximum total fitness score is the selected combination for this level.


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

The aforementioned constructive and improvement procedure is found to have a tendency to generate auto-racks with significant unused length and low revenue, rendering vehicles into low revenue auto-racks. In order to prevent this, all auto-racks with revenue below a pre-determined revenue threshold are disbanded (vehicles unassigned) and flagged till only flagged vehicles remain to be assigned. The flagged vehicles are also precluded from being the first vehicle to be loaded onto an auto-rack in Phase 2 to prevent the same solution from getting regenerated. It should be noted that a flagged vehicle can be a part of the next-fit decreasing procedure as long as they do not set the default level and default deck position of the auto-rack.

Phase 4 - Elitist Selection

After all vehicles have been assigned to an auto-rack in Phase 2 and Phase 3, the auto-racks are sorted in decreasing order of revenue. Based on the limitation on the number of auto-racks in the train / rake, the auto-racks with the highest revenue are chosen for the final solution.

## 4.1 Upper Bound for ARLP

Calculation of bounds helps in determining the target values for the optimization (IP model and heuristic procedure). For a maximization problem, the upper bound gives the optimal target and lower bound gives the worst case. This section explains the method to calculate the upper bound while the lower bound is not considered due to the fact that this is a maximization problem. The first upper bound UB1 comes from the branch and bound algorithm of Gurobi® after solving the IP model discussed in section 3.1 for 30 minutes of runtime for each test dataset. A 30-minute runtime is chosen because the ARLP is an operational problem and turnaround times for a loading solution is typically a couple of hours.

An alternate upper bound UB2 is derived using a relaxation of the original IP model. On relaxing the height constraints of the IP model (constraints 5, 6, 7, 8, 9 and 10) and enforce that no adjustments to the deck level be made i.e. $MDP = 0$, knapsack problem (with $2m$ knapsacks) ensues with the length constraints alone. However, the mix of vehicle models could contain models that when loaded on one level will result in the other level being unloadable (due to height restrictions). In addition, there might be instances where if a particular vehicle model is loaded onto one level, only certain vehicle model combinations will allow for feasible loading onto the other level. If there is an insufficient amount of these vehicle model combinations that can be loaded onto the other level, the excess quantity of the initial vehicle model, if loaded, will cause the other level to become unloadable. Consequently, assuming auto-racks containing such vehicle models to have both levels loaded with vehicles will result in an inefficient upper bound.

In order to alleviate this above issue, a pre-processing algorithm is developed that will identify vehicle models and their respective quantities that could potentially render the other level to be empty and double the length for those quantities. Since the upper bound is based on a relaxation that considers only the length constraints, doubling the vehicle length would in fact cause the loading efficiency of an auto-rack to be essentially halved, which is similar to one level rendered empty due to height restrictions. A flowchart of the pre-processing algorithm is shown in Figure 3.

The algorithm begins by calculating the tallest allowable vehicle on the non-default level of a vehicle model. If the same vehicle model can fit on the non-default level, then the default length of the vehicle model is used. If the vehicle model cannot fit on the non-default level, then the total count of vehicles which can fit on the non-default level is enumerated. If the total count of other vehicle models is more than the quantity of the vehicle model under consideration, then the default length of the vehicle model is used. Otherwise, the length is doubled for the excess quantity of the vehicle model under consideration. This procedure is repeated for all other vehicle models.

Once the pre-processing step is carried out on the dataset, the relaxed model is input into the optimization software Gurobi® and run for a time of 5 minutes.


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

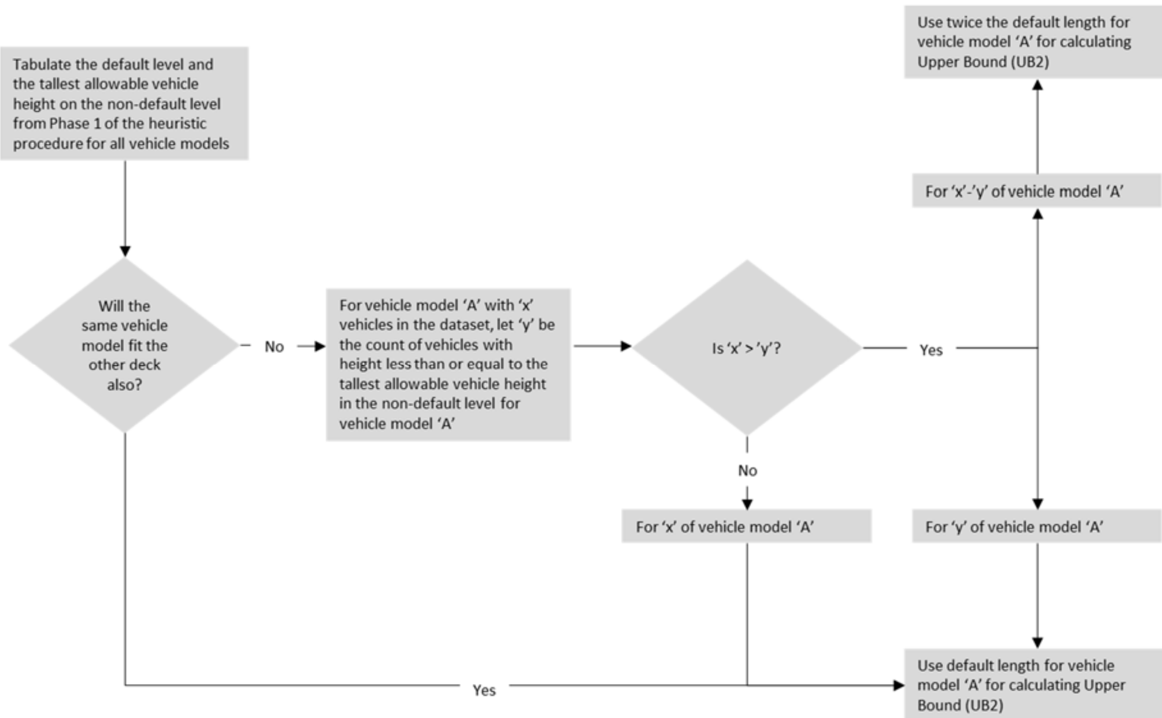


Figure 3. Pre-processing algorithm for Upper Bound UB2

## 5. Computational Experiments

In order to assess the efficiency of the IP model and the heuristic procedure, a series of computation tests are carried out on randomly generated problems. The IP model is executed in Gurobi® Solver while the heuristic procedure is coded in Microsoft Excel VBA, both run in a computer powered by Intel® i7 processor with 8GB RAM.

As mentioned in Section 2, this problem is the first of its kind to be addressed and hence standard datasets are not available. Consequently, datasets for testing are randomly generated (shown in Appendix B). For evaluating the IP model and the heuristic procedure developed as a part of this research, the auto-racks approved for use in the Indian rail network are chosen. The vehicles to be loaded comprise of the vehicles being sold in the Indian market.

As mentioned earlier, the objective of the IP model and the heuristic procedure is to maximize the revenue of vehicles loaded in a train / rake. While the actual revenue per vehicle for a vehicle model is based on contractual terms between the automotive manufacturer and the service provider, this research assumes the revenue of a vehicle model to be directly proportional to the dimensions (length, width, and height) of the vehicle.

### 5.1 Auto-rack Specifications

The auto-rack certified for operation on the Indian rail network is of type 'BCACBM' developed by Railway Design and Standards Organization (RDSO) and is a bi-level auto-rack with a movable deck. The maximum count of auto-racks in a train / rake is 27 (Government of India, 2013). While designs exist for two types of BCACBM auto-racks (A and B type), this research considers only the standard B auto-rack in the model. The detailed dimensions of the BCACBM auto-rack are documented by Touax (Touax, 2016).

The inside dimensions of the B-type BCACBM bi-level auto-rack are illustrated in Figure 4 and will be used to test the IP model and heuristic procedure. The movable deck can be raised from the "min" position in steps of 50 mm upto the "max" position. With the movable deck at the "min" position, the lower level has an available height of 1550 mm and the upper level has 1770 mm. At the "max" position, the lower level has an available height of 2050 mm and the upper level has 1270 mm.


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

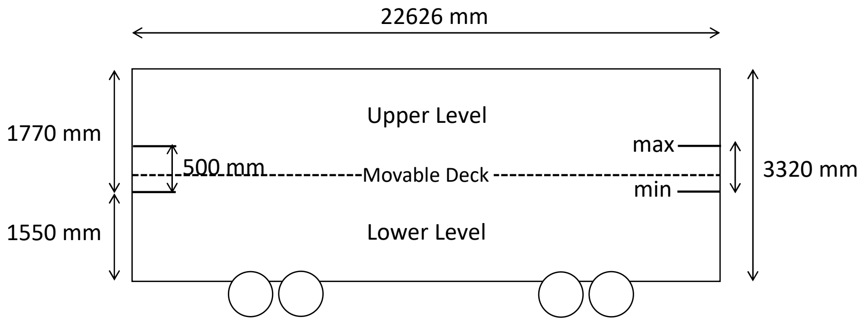


Figure 4. Dimensions of the BCACBM auto-rack

Neither the BCACBM specifications document nor the Government of India certification document specify the minimum clearance between surfaces to prevent damages due to abrasion. These include vehicle-to-vehicle, vehicle-to-door of auto-rack, vehicle-to-roof of auto-rack / movable deck. Consequently, this research uses the specifications set forth by Association of American Railroad (AAR) for these clearances (AAR, 2009).

Vehicle-to-vehicle = 4 inches or 101.6 mm
Vehicle-to-door = 6 inches or 152.4 mm
Vehicle-to-roof = 3 inches or 76.2 mm

It is assumed that the movable deck can be raised or lowered in discrete steps of 50 mm.

## 5.2 Vehicle Models

The vehicle models used for evaluating the performance of the IP model and the heuristic procedure are selected from the ones being sold in India and comprise of vehicles of varying sizes (large SUVs to small hatchbacks). The dimensions of the vehicles are obtained from the manufacturers' website. The list of vehicles, the dimensions, and the estimated revenue per vehicle are shown in Appendix A.

Notably, loading some large vehicle models with higher revenue on one level of the auto-rack will result in the other level being empty due to absence of commensurately sized small vehicles. For example, loading vehicle model "Model 1" (with height of 1930mm) on the lower level of an auto-rack will result in the movable deck being raised by 500mm (considering the vehicle-to-roof clearance and movable deck step increment) resulting in the upper deck having an available height of 1270mm. As can be observed from Appendix A, the shortest vehicle in the dataset is "Model 40" with a height of 1462mm. Hence, one of the critical decision parameters for the IP model and the heuristic procedure is to decide between loading one level of an auto-rack with higher revenue large vehicles or both levels with lower revenue small vehicles.

## 5.3 Datasets

The datasets used in this research are randomly generated for the vehicle models listed in Appendix A. Initially, the vehicles models are classified into 3 based on height as follows:

If Vehicle Height (VH) > 1700 mm, then Size = Big
Else If Vehicle Height (VH) > 1600 mm, then Size = Medium
Else, Size = Small

Next, datasets are generated for various combinations of the three size classifications (Big, Medium, and Small) as shown in Table 3.



DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

Table 3. Model – Vehicle Size Proportions

| Test Case | Big | Medium | Small |
|---|---|---|---|
| 1 | 33% | 33% | 33% |
| 2 | 25% | 25% | 50% |
| 3 | 50% | 25% | 25% |
| 4 | 25% | 50% | 25% |

For each test case, 10 datasets are randomly generated for a total vehicle count between 300 and 600 (shown in Appendix B) to be loaded on 27 auto-racks (as per the limit on Indian rail network).

## 5.4 Parameter Tuning

As in most heuristics, there are a number of parameters that may influence the behavior of the algorithm. For the heuristic procedure developed in this research, following are the parameters:

- Weightage for fitness criteria $R_i$,
- Weightage for fitness criteria $L_i$,
- Weightage for fitness criteria $H_i$.

In order to determine the value for the weights, a sensitivity analysis is performed using the datasets in Appendix B with combination of weights as in Table 4.

Table 4. Heuristic Parameter Weights

| Level | Weight for $R_i$ | Weight for $L_i$ | Weight for $H_i$ |
|---|---|---|---|
| 1 | 33% | 33% | 33% |
| 2 | 60% | 20% | 20% |
| 3 | 20% | 60% | 20% |
| 4 | 20% | 20% | 60% |

The results of sensitivity analysis on the test cases described in Table 3 is shown in Figure 5. An Analysis of Variance (ANOVA) is conducted on each test case to evaluate the statistical significance of the “levels” on the revenue. At $\alpha$ =5%, “level” is found to be statistically significant. A Tukey test for statistical difference of means showed that revenue for “level” 3 is statistically different and lower than “levels” 1, 2, and 4 for all test cases. The results for “levels” 1,2, and 4 are statistically similar for test cases 1, 3 and 4 while “level” 2’s revenue is higher for test case 2. From the aforementioned results, it can be inferred that for datasets containing a large number of small sized vehicles (like test case 2), “level” 2 is appropriate. For datasets dominated by medium or large sized vehicles, the choice of “level” is irrelevant and the weights can be arbitrarily chosen.


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

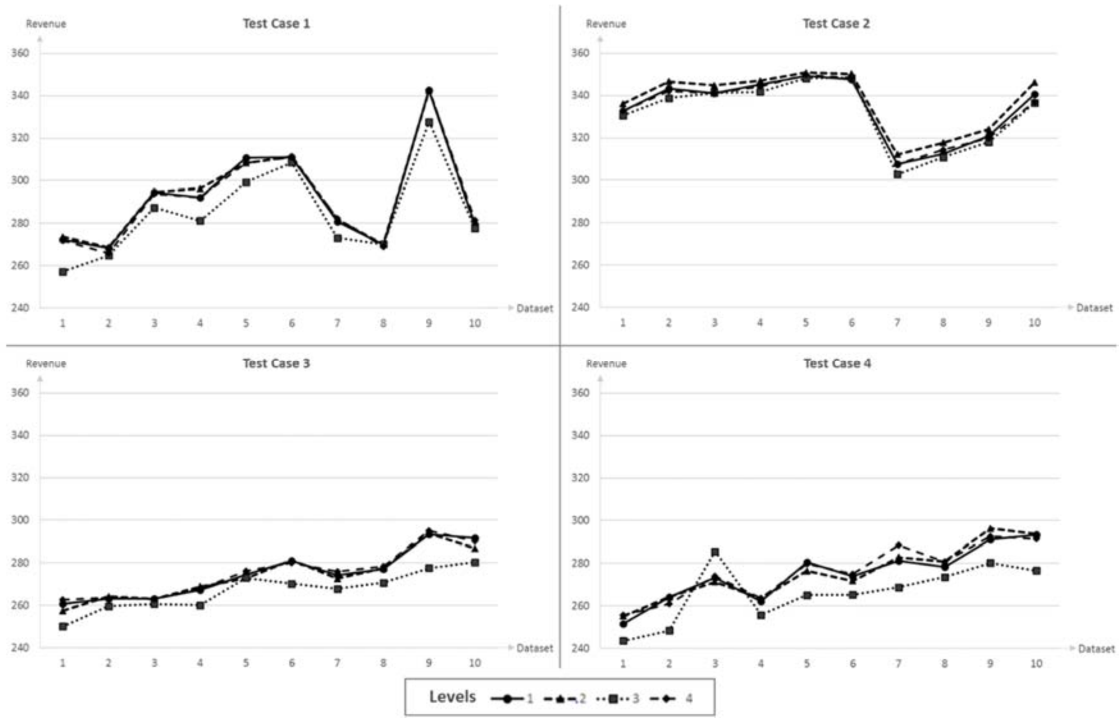


Figure 5. Parameter Tuning for Heuristic Procedure

## 5.5 Upper Bounds

As described in Section 4.1 Upper Bound UB1 is obtained from the output of Gurobi® after solving the IP model for 30 minutes. Upper Bound UB2 is derived from solving a relaxation of the IP model after applying the pre-processing algorithm to the dataset and solving the model in Gurobi® for 5 minutes. It can be observed from Table 5 that UB2 is a tighter upper bound in most cases and hence will be used for benchmarking the output of the IP model and the heuristic procedure.

Table 5. Comparison of Upper Bounds

| | Test Case 1 | | | Test Case 2 | | | Test Case 3 | | | Test Case 4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Dataset** | **UB1** | **UB2** | **Diff %** | **UB1** | **UB2** | **Diff %** | **UB1** | **UB2** | **Diff %** | **UB1** | **UB2** | **Diff %** |
| 1 | 351.4 | 287.2 | 22.3% | 363.6 | 350.9 | 3.6% | 307.0 | 278.3 | 10.3% | 386.7 | 279.0 | 38.6% |
| 2 | 347.6 | 288.8 | 20.4% | 376.5 | 359.1 | 4.9% | 322.0 | 281.3 | 14.5% | 392.1 | 289.9 | 35.3% |
| 3 | 341.0 | 310.3 | 9.9% | 386.4 | 356.7 | 8.3% | 308.1 | 278.8 | 10.5% | 326.2 | 301.2 | 8.3% |
| 4 | 348.8 | 313.5 | 11.3% | 364.5 | 355.2 | 2.6% | 352.2 | 290.2 | 21.4% | 331.3 | 291.7 | 13.6% |
| 5 | 346.0 | 332.2 | 4.1% | 366.1 | 366.6 | -0.1% | 333.2 | 295.8 | 12.6% | 333.7 | 308.6 | 8.1% |
| 6 | 370.0 | 327.4 | 13.0% | 366.6 | 363.7 | 0.8% | 325.8 | 301.2 | 8.2% | 339.1 | 297.9 | 13.8% |
| 7 | 344.9 | 296.0 | 16.5% | 339.3 | 316.6 | 7.2% | 342.5 | 292.0 | 17.3% | 340.7 | 312.0 | 9.2% |
| 8 | 357.4 | 288.4 | 23.9% | 361.0 | 328.4 | 9.9% | 386.0 | 297.7 | 29.7% | 344.2 | 307.5 | 11.9% |
| 9 | 356.3 | 353.6 | 0.8% | 376.0 | 335.3 | 12.1% | 334.1 | 313.9 | 6.4% | 335.7 | 328.0 | 2.4% |
| 10 | 331.2 | 300.3 | 10.3% | 363.3 | 358.2 | 1.4% | 325.1 | 313.6 | 3.6% | 349.8 | 327.6 | 6.8% |


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

## 5.6 Comparison of IP Model and Heuristic Procedure Results

For the test cases described in Appendix B, the IP model and the heuristic procedure are executed. The IP model is run for 30 minutes while the heuristic procedure takes between 5-6 minutes per run. Detailed results are tabulated in Table 6 to Table 9. The four tables represent the four test cases described in Table 3 with each test case containing 10 datasets. The results of the IP model and the heuristic procedure are shown from a revenue and vehicles loaded perspective. The output of the model used to determine the upper bound (discussed in Section 4.2) is used to benchmark the results of the IP model and the heuristic procedure.

Table 6. Results for Test Case 1 for Comparison between IP Model and Heuristic Procedure

| | | **Heuristic** | **IP Model** | | **Difference in Revenue %** | | |
|---|---|---|---|---|---|---|---|
| **Dataset** | **Vehicles in Dataset** | **Revenue (HR)** | **Revenue (IR)** | **Upper Bound (UB2)** | **HR - IR** | **UB2 - IP** | **UB2 - HR** |
| 1 | 282 | 273.7 | 279.1 | 287.2 | 1.9% | 2.9% | 4.9% |
| 2 | 304 | 264.8 | 265.0 | 288.8 | 0.1% | 9.0% | 9.1% |
| 3 | 368 | 294.8 | 298.9 | 310.3 | 1.4% | 3.8% | 5.3% |
| 4 | 395 | 296.3 | 297.4 | 313.5 | 0.4% | 5.4% | 5.8% |
| 5 | 456 | 310.8 | 316.2 | 332.2 | 1.7% | 5.1% | 6.9% |
| 6 | 481 | 311.4 | 313.8 | 327.4 | 0.8% | 4.3% | 5.1% |
| 7 | 332 | 282.3 | 284.5 | 296.0 | 0.8% | 4.0% | 4.9% |
| 8 | 302 | 270.2 | 273.0 | 288.4 | 1.0% | 5.6% | 6.7% |
| 9 | 602 | 342.8 | 346.9 | 353.6 | 1.2% | 1.9% | 3.2% |
| 10 | 389 | 281.6 | 287.0 | 300.3 | 1.9% | 4.6% | 6.6% |
| | | | | **Average** | 1.1% | 4.7% | 5.9% |

Table 7. Results for Test Case 2 for Comparison between IP Model and Heuristic Procedure

| | | **Heuristic** | **IP Model** | | **Difference in Revenue %** | | |
|---|---|---|---|---|---|---|---|
| **Dataset** | **Vehicles in Dataset** | **Revenue (HR)** | **Revenue (IR)** | **Upper Bound (UB2)** | **HR - IR** | **UB2 - IP** | **UB2 - HR** |
| 1 | 282 | 336.1 | 340.5 | 350.9 | 1.3% | 3.1% | 4.4% |
| 2 | 304 | 346.6 | 349.4 | 359.1 | 0.8% | 2.8% | 3.6% |
| 3 | 368 | 344.8 | 347.8 | 356.7 | 0.9% | 2.6% | 3.5% |
| 4 | 395 | 347.0 | 348.4 | 355.2 | 0.4% | 2.0% | 2.4% |
| 5 | 456 | 350.8 | 353.4 | 366.6 | 0.7% | 3.7% | 4.5% |
| 6 | 481 | 350.2 | 353.9 | 363.7 | 1.0% | 2.8% | 3.9% |
| 7 | 332 | 312.1 | 313.0 | 316.6 | 0.3% | 1.1% | 1.4% |
| 8 | 302 | 317.6 | 318.9 | 328.4 | 0.4% | 3.0% | 3.4% |
| 9 | 602 | 321.0 | 322.8 | 335.3 | 0.6% | 3.9% | 4.5% |
| 10 | 389 | 346.1 | 351.3 | 358.2 | 1.5% | 2.0% | 3.5% |
| | | | | **Average** | 0.8% | 2.7% | 3.5% |

Table 8. Results for Test Case 3 for Comparison between IP Model and Heuristic Procedure

| Dataset | Vehicles in Dataset | Heuristic Revenue (HR) | IP Model Revenue (IR) | Upper Bound (UB2) | Difference in Revenue % HR - IR | UB2 - IP | UB2 - HR |
|---|---|---|---|---|---|---|---|
| 1 | 282 | 262.7 | 267.2 | 278.3 | 1.7% | 4.2% | 5.9% |
| 2 | 304 | 264.3 | 268.1 | 281.3 | 1.4% | 4.9% | 6.4% |
| 3 | 368 | 263.3 | 266.5 | 278.8 | 1.2% | 4.6% | 5.9% |
| 4 | 395 | 268.9 | 274.8 | 290.2 | 2.1% | 5.6% | 7.9% |
| 5 | 456 | 276.2 | 280.0 | 295.8 | 1.3% | 5.7% | 7.1% |
| 6 | 481 | 281.4 | 282.2 | 301.2 | 0.3% | 6.7% | 7.0% |
| 7 | 332 | 276.1 | 283.3 | 292.0 | 2.5% | 3.1% | 5.8% |
| 8 | 302 | 278.5 | 280.2 | 297.7 | 0.6% | 6.2% | 6.9% |
| 9 | 602 | 295.1 | 301.0 | 313.9 | 2.0% | 4.3% | 6.4% |
| 10 | 389 | 291.7 | 298.1 | 313.6 | 2.1% | 5.2% | 7.5% |
| | | | | **Average** | 1.5% | 5.0% | 6.7% |

Table 9. Results for Test Case 4 for Comparison between IP Model and Heuristic Procedure

| Dataset | Vehicles in Dataset | Heuristic Revenue (HR) | IP Model Revenue (IR) | Upper Bound (UB2) | Difference in Revenue % HR - IR | UB2 - IP | UB2 - HR |
|---|---|---|---|---|---|---|---|
| 1 | 282 | 255.8 | 262.1 | 279.0 | 2.4% | 6.5% | 9.1% |
| 2 | 304 | 264.3 | 272.5 | 289.9 | 3.0% | 6.4% | 9.7% |
| 3 | 368 | 274.2 | 280.2 | 301.2 | 2.2% | 7.5% | 9.8% |
| 4 | 395 | 263.6 | 274.3 | 291.7 | 3.9% | 6.4% | 10.7% |
| 5 | 456 | 280.7 | 285.4 | 308.6 | 1.7% | 8.1% | 9.9% |
| 6 | 481 | 275.0 | 285.1 | 297.9 | 3.5% | 4.5% | 8.3% |
| 7 | 332 | 288.6 | 293.3 | 312.0 | 1.6% | 6.4% | 8.1% |
| 8 | 302 | 280.9 | 292.0 | 307.5 | 3.8% | 5.3% | 9.5% |
| 9 | 602 | 296.3 | 302.2 | 328.0 | 2.0% | 8.5% | 10.7% |
| 10 | 389 | 293.7 | 301.2 | 327.6 | 2.5% | 8.8% | 11.5% |
| | | | | **Average** | 2.6% | 6.8% | 9.7% |

Comparing the results of the IP model and the heuristic procedure on the datasets, it can be concluded that the heuristic procedure's results are lower by between 0.8% and 2.6% on an average from the IP model. Benchmarking the results against the upper bound UB2, it can be observed that the results of the IP model and the heuristic procedure are between 3.5% and 9.7% from the upper bound UB2. Notably, both the IP model and heuristic procedure perform best for the datasets in test case 2 and worst for datasets in test case 4. The inference from the aforementioned observation is that the IP model and the heuristic procedure work best for datasets dominated by small sized vehicles and worst for datasets dominated by large vehicles.


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

## 6. Summary and Conclusions

This paper introduces a new class of problems, named here Auto-Rack Loading Problem (ARLP), addressing loading of vehicles onto railcars called auto-racks, for the delivery of vehicles to a given destination. The problem is shown to be NP-hard. In addition to the formulation of an IP model, the NP-hardness of the ARLP necessitated the development of a heuristic procedure to solve large instances of the problem in reasonable time. Based on a reduction technique, an upper bound for the ARLP was determined for benchmarking the performance of the IP model and the heuristic procedure.

On executing randomly generated datasets for auto-racks designed for use in the Indian rail network, the results of the IP model and the heuristic procedure were found to be not less than 10% lower than the upper bound and within 3% of each other. Consequently, it can be inferred that the methods developed as part of this research are competitive and can be used for practical instances.

As future research direction, the scope could be expanded to consider tri-level auto-racks. Some practical use cases that can be studied are:

- Presence of a subset of vehicles that are mandatory to be loaded and incur a penalty if not loaded.
- Grouping of vehicles during loading due to certain common characteristics like model and dealer location.
- Another interesting extension would consist of taking advantage of stochastic information on vehicle demands to further improve planning.

## 7. Acknowledgements

The authors thank Vascor Logistics where this research was conducted. The authors confirm that the data supporting the findings of this study are available within the article.

## 8. References

Andonov, R., Poirriez, V., & Rajopadhye, S. (2000). Unbounded knapsack problem: Dynamic programming revisited. *European Journal of Operational Research*, *123*(2), 394-407.

Association of American Railroads. (2009). Association of american railroads. Manual of standards and recommended practices Section N Multi-level manual. *Transportation Technology Center, AAR, Cary, NC*, I24-I25.

Association of American Railroads. (2016) The environmental benefits of moving freight by rail. Retrieved from *https://www.aar.org/BackgroundPapers/Environmental Benefits of Moving Freight by Rail.pdf*

Backaker L., & Krasemann. J. T. (2012). Trip plan generation using optimization: A benchmark of freight routing and schedulling policies within the carload service segment. *Journal of Rail Transport Planning & Management*, *2*(1-2), 1-13.

Bonassa, A. C., Cunha, C. B., & Isler, C. A. (2019). An exact formulation for the multi-period auto-carrier loading and transportation problem in Brazil. *Computers & Industrial Engineering, 129,* 144-155.

Burke, E.K., Silva, J. D. L., & Soubeiga, E. (2005). Multi-objective hyper-heuristic approaches for space allocation and timetabling. In *Metaheuristics: Progress as Real Problem Solvers* (pp. 129-158). Springer, Boston, MA.

Chekuri, C., & Khanna, S. (2005). A polynomial time approximation scheme for the multiple knapsack problem. *SIAM Journal on Computing, 35*(3), 713-728.

Cordeau. J. F., Dell'Amico, M., Falavigna, S., & Iori, M. (2015). A rolling horizon algorithm for auto-carrier transportation. *Transportation Research Part B*, *76*, 68-80.

Dolinayova. A., Loch, M., & Kanis, J. (2015). Modelling the influence for wagon technical parameters on variable costs in rail freight transport. *Research in Transport Economics*, *54*, 33-40.

Eskigun, E., Uzsoy, R., Preckel, P. V., Beaujon, G., Krishnan, S., & Tew, J. (2005). Outbound supply chain network with mode selection, lead times and capacitated vehicle distribution centers. *European Journal of Operational Research*, *165*(1), 182–206.

Feuerman, M., & Weiss, H. (1973). A mathematical programming model for test construction and scoring. *Management Science*, *19*(8), 961-966.

Governnment of India. (2013). *Final maximum permissible speed certificate for operation of Broad Gauge Bogie Covered Autocar carrier wagon type BCACBM.* Retrieved from *www.rdso.indianrailways.gov.in/works/uploads/File/Final Speed ceritificate of BCACBM Wagon(1).pdf*

Islam, D. M. Z., Jackson, R., Robinson, M. (2015). European freight rolling stock fleet size in 2050 in light of Transport White Paper 2011. *Journal of Rail Transport Planning & Management*, *5,* 195-210.

Joiner, S. (2010). Is bigger better? Monster Trains vs Freight Trains. *Popular Mechanics*. Online Edition.

Khuri, S., Bäck, T., & Heitkötter, J. (1994). The zero/one multiple knapsack problem and genetic algorithms. In *Proceedings of the 1994 ACM symposium on Applied computing* (pp.188-193).

Kumar, R., Joshi, A. H., Banka, K. K., & Rockett, P.I. (2008). Evolution of hyperheuristics for the biobjective 0/1 knapsack problem by multiobjective genetic programming. In *Proceedings of the 10th annual conference on Genetic and evolutionary computation* (pp.1227-1234).

Lin. C-H. (2010). An exact solving approach to the auto-carrier problem. *Journal of Society for Transportation and Traffic Studies*, *1*, 93-107.

Martello, S., & Toth, P. (1984). A mixture of dynamic programming and branch-and-bound for the subset-sum problem. *Management Science*, *30*(6), 765-771.

Martello, S., & Toth, P. (1990). Knapsack Problems. 1st Edition. West Sussex, England: *John Wiley & Sons* (pp.221–240).

Martello, S., Pisinger, D., & Toth, P. (1999). Dynamic programming and strong bounds for the 0-1 knapsack problem. *Management Science*, *45*(3), 414-424.

Pisinger, D. (1999). An exact algorithm for large multiple knapsack problems. *European Journal of Operational Research*, *114*(3), 528-541.

Plateau, G., & Elkihel, M. (1985). A hybrid method for the 0-1 knapsack problem. *Methods of Operations Research*, *49*, 277-293.

Poirriez, V., Yanev, N., & Andonov, R. (2009). A hybrid algorithm for the unbounded knapsack problem. *Discrete Optimization*, *6*(1), 110-124.

Shah-Hosseini, H. (2008). Intelligent water drops algorithm: A new optimization method for solving the multiple knapsack problem. *International Journal of Intelligent Computing and Cybernetics*, *1*(2), 193-212.

Touax. (2016). 4-axle wagon for automotive transportation. Retrieved from *http://www.touaxtexmaco.com/content/bcacbm-rake*

Transport Canada. (2013). Overview of rail transportation. Retrieved from *http://www.tc.gc.ca/eng/policy/anre-menu-3020.htm*

Turner, K., & Williams, G. (2004). Modeling complexity in the automotive supply chain. *Journal of Manufacturing Technology Management*, *16*(4), 447–458.

Viswanathan, A., Wang, S., Joshi, A., Jones, L., & Srihari, K. (2011). Estimation of claims in an outbound automotive supply chain using capture-recapture methodology. *International Journal of Logistics Research and Applications*, *14*(1), 1-10.

Wang, Y., & Chen F. (2019). Auto-carrier Loading Problem with Inter-set Costs in Urban Outbound Logistics of Automobiles. In *Proceedings of 2019 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE)* (pp.164-168).

Yee. J. (2014). A dynamic optimization for automotive vehicle shipment and delivery. *Journal of the Korea Society for Simulation, 23*(4), 9-19.

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach, In *IEEE transactions on Evolutionary Computation*, *3*(4), 257-271.

## Appendix A: Vehicle Model Dimensions and Revenue

| Model | Length l (mm) | Width w (mm) | Height h (mm) | Revenue ($l*h/10^4$) | Standardized Revenue |
|---|---|---|---|---|---|
| Model 1 | 4456 | 1820 | 1930 | 8.60 | 1.70 |
| Model 2 | 4107 | 1745 | 1880 | 7.72 | 1.53 |
| Model 3 | 4170 | 1660 | 1880 | 7.84 | 1.55 |
| Model 4 | 4705 | 1840 | 1850 | 8.70 | 1.72 |
| Model 5 | 3995 | 1835 | 1839 | 7.35 | 1.45 |
| Model 6 | 3675 | 1475 | 1800 | 6.62 | 1.31 |
| Model 7 | 4585 | 1890 | 1785 | 8.18 | 1.62 |
| Model 8 | 4585 | 1760 | 1760 | 8.07 | 1.60 |
| Model 9 | 3999 | 1765 | 1708 | 6.83 | 1.35 |
| Model 10 | 3599 | 1495 | 1700 | 6.12 | 1.21 |
| Model 11 | 4315 | 1822 | 1695 | 7.31 | 1.45 |
| Model 12 | 3795 | 1680 | 1427 | 5.42 | 1.07 |
| Model 13 | 3164 | 1750 | 1652 | 5.23 | 1.03 |
| Model 14 | 4490 | 1730 | 1485 | 6.67 | 1.32 |
| Model 15 | 3679 | 1579 | 1478 | 5.44 | 1.07 |
| Model 16 | 3430 | 1490 | 1475 | 5.06 | 1.00 |
| Model 17 | 4370 | 1700 | 1475 | 6.45 | 1.27 |
| Model 18 | 3775 | 1680 | 1620 | 6.12 | 1.21 |
| Model 19 | 3850 | 1695 | 1530 | 5.89 | 1.16 |
| Model 20 | 3995 | 1695 | 1525 | 6.09 | 1.20 |
| Model 21 | 3955 | 1694 | 1524 | 6.03 | 1.19 |
| Model 22 | 4300 | 1756 | 1590 | 6.84 | 1.35 |
| Model 23 | 3995 | 1706 | 1570 | 6.27 | 1.24 |
| Model 24 | 3600 | 1600 | 1560 | 5.62 | 1.11 |
| Model 25 | 3895 | 1735 | 1555 | 6.06 | 1.20 |
| Model 26 | 3995 | 1695 | 1555 | 6.21 | 1.23 |
| Model 27 | 3585 | 1595 | 1550 | 5.56 | 1.10 |
| Model 28 | 3795 | 1695 | 1550 | 5.88 | 1.16 |
| Model 29 | 4413 | 1703 | 1550 | 6.84 | 1.35 |
| Model 30 | 3370 | 1410 | 1640 | 5.53 | 1.09 |
| Model 31 | 3765 | 1660 | 1520 | 5.72 | 1.13 |
| Model 32 | 3995 | 1660 | 1520 | 6.07 | 1.20 |
| Model 33 | 4265 | 1695 | 1510 | 6.44 | 1.27 |
| Model 34 | 3985 | 1734 | 1505 | 6.00 | 1.19 |
| Model 35 | 3990 | 1680 | 1505 | 6.00 | 1.19 |
| Model 36 | 3495 | 1550 | 1500 | 5.24 | 1.04 |
| Model 37 | 4440 | 1695 | 1495 | 6.64 | 1.31 |
| Model 38 | 4265 | 1695 | 1685 | 7.19 | 1.42 |
| Model 39 | 4270 | 1730 | 1685 | 7.19 | 1.42 |
| Model 40 | 3970 | 1901 | 1462 | 5.80 | 1.15 |

The standardized revenue is obtained by dividing the revenue of each model by the revenue of the model with the least revenue (Model 16).


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73

## Appendix B: Datasets

| Test Case | Dataset | Model | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | |
| 1 | 1 | 2 | 1 | 20 | 20 | 6 | 7 | 12 | 18 | 8 | 6 | 15 | 20 | 15 | 17 | 20 | 1 | 6 | 2 | 4 | 4 | 1 | 8 | 3 | 3 | 7 | 2 | 2 | 4 | 6 | 1 | 2 | 6 | 2 | 3 | 8 | 10 | 1 | 3 | 1 | 5 | 282 |
| | 2 | 5 | 3 | 8 | 6 | 20 | 18 | 7 | 14 | 20 | 18 | 15 | 20 | 11 | 4 | 15 | 19 | 8 | 3 | 2 | 8 | 7 | 1 | 8 | 2 | 1 | 2 | 3 | 2 | 6 | 7 | 2 | 8 | 1 | 3 | 3 | 6 | 8 | 1 | 7 | 2 | 304 |
| | 3 | 8 | 19 | 19 | 10 | 12 | 20 | 18 | 16 | 1 | 17 | 15 | 17 | 17 | 19 | 22 | 15 | 5 | 10 | 9 | 3 | 9 | 8 | 5 | 8 | 9 | 3 | 2 | 1 | 3 | 5 | 3 | 3 | 4 | 1 | 10 | 3 | 5 | 7 | 2 | 5 | 368 |
| | 4 | 17 | 20 | 10 | 8 | 17 | 16 | 16 | 16 | 12 | 22 | 21 | 8 | 18 | 16 | 23 | 24 | 9 | 7 | 2 | 5 | 7 | 3 | 2 | 7 | 6 | 5 | 6 | 4 | 6 | 3 | 5 | 9 | 7 | 10 | 1 | 9 | 3 | 4 | 7 | 4 | 395 |
| | 5 | 22 | 24 | 9 | 23 | 12 | 23 | 2 | 20 | 15 | 14 | 27 | 25 | 19 | 27 | 23 | 18 | 3 | 6 | 8 | 10 | 8 | 5 | 6 | 8 | 3 | 9 | 5 | 4 | 8 | 8 | 10 | 7 | 3 | 1 | 9 | 8 | 1 | 7 | 9 | 7 | 456 |
| | 6 | 24 | 21 | 6 | 24 | 13 | 18 | 17 | 13 | 24 | 31 | 26 | 2 | 23 | 22 | 32 | 24 | 10 | 10 | 5 | 10 | 2 | 6 | 6 | 8 | 2 | 3 | 6 | 6 | 2 | 5 | 10 | 12 | 11 | 11 | 12 | 7 | 1 | 5 | 8 | 3 | 481 |
| | 7 | 13 | 14 | 14 | 20 | 7 | 15 | 12 | 5 | 10 | 15 | 16 | 16 | 20 | 10 | 17 | 17 | 4 | 6 | 6 | 7 | 4 | 7 | 3 | 6 | 8 | 7 | 1 | 2 | 5 | 3 | 4 | 1 | 1 | 5 | 2 | 9 | 5 | 4 | 10 | 1 | 332 |
| | 8 | 5 | 2 | 12 | 21 | 13 | 9 | 21 | 10 | 7 | 8 | 13 | 11 | 12 | 12 | 25 | 20 | 4 | 2 | 2 | 3 | 6 | 2 | 9 | 6 | 1 | 7 | 3 | 3 | 1 | 8 | 3 | 2 | 6 | 7 | 6 | 7 | 4 | 2 | 3 | 4 | 302 |
| | 9 | 7 | 32 | 13 | 6 | 48 | 33 | 32 | 12 | 16 | 40 | 25 | 1 | 35 | 39 | 21 | 40 | 3 | 14 | 11 | 2 | 7 | 2 | 10 | 9 | 2 | 5 | 5 | 14 | 13 | 1 | 11 | 7 | 13 | 10 | 9 | 10 | 10 | 5 | 15 | 14 | 602 |
| | 10 | 22 | 20 | 16 | 9 | 21 | 14 | 24 | 1 | 3 | 27 | 23 | 17 | 10 | 2 | 20 | 30 | 3 | 5 | 1 | 10 | 1 | 2 | 4 | 8 | 9 | 1 | 1 | 8 | 8 | 5 | 8 | 5 | 9 | 1 | 6 | 6 | 9 | 7 | 7 | 6 | 389 |
| 2 | 1 | 2 | 3 | 19 | 20 | 10 | 17 | 14 | 24 | 6 | 18 | 25 | 23 | 2 | 19 | 16 | 12 | 5 | 3 | 11 | 6 | 16 | 7 | 7 | 1 | 1 | 18 | 18 | 15 | 8 | 1 | 14 | 3 | 19 | 5 | 20 | 12 | 12 | 12 | 10 | 6 | 460 |
| | 2 | 23 | 2 | 10 | 21 | 2 | 12 | 24 | 6 | 20 | 8 | 11 | 23 | 18 | 18 | 18 | 24 | 17 | 11 | 17 | 14 | 5 | 8 | 11 | 14 | 12 | 3 | 8 | 5 | 15 | 10 | 11 | 1 | 18 | 18 | 4 | 2 | 2 | 13 | 2 | 19 | 480 |
| | 3 | 10 | 11 | 9 | 8 | 20 | 3 | 16 | 23 | 25 | 6 | 29 | 16 | 17 | 7 | 24 | 26 | 18 | 2 | 11 | 14 | 16 | 9 | 6 | 10 | 15 | 4 | 3 | 19 | 9 | 3 | 4 | 4 | 14 | 20 | 19 | 5 | 8 | 19 | 1 | 17 | 500 |
| | 4 | 7 | 11 | 15 | 25 | 10 | 23 | 21 | 15 | 8 | 18 | 29 | 4 | 27 | 12 | 20 | 25 | 22 | 1 | 2 | 11 | 19 | 9 | 15 | 13 | 10 | 15 | 17 | 4 | 18 | 17 | 8 | 5 | 3 | 15 | 13 | 15 | 10 | 4 | 16 | 8 | 540 |
| | 5 | 22 | 11 | 22 | 23 | 6 | 25 | 2 | 12 | 22 | 7 | 16 | 24 | 20 | 30 | 19 | 29 | 13 | 7 | 17 | 8 | 8 | 8 | 19 | 9 | 2 | 11 | 8 | 20 | 11 | 18 | 16 | 10 | 3 | 8 | 20 | 8 | 5 | 21 | 24 | 16 | 580 |
| | 6 | 11 | 16 | 11 | 23 | 14 | 23 | 1 | 22 | 29 | 26 | 24 | 11 | 9 | 16 | 35 | 29 | 25 | 24 | 24 | 7 | 24 | 10 | 15 | 3 | 25 | 3 | 14 | 1 | 20 | 25 | 5 | 5 | 10 | 9 | 10 | 15 | 3 | 15 | 3 | 5 | 600 |
| | 7 | 14 | 14 | 10 | 15 | 13 | 4 | 3 | 3 | 4 | 6 | 18 | 18 | 18 | 11 | 8 | 1 | 7 | 11 | 11 | 6 | 4 | 10 | 7 | 10 | 4 | 9 | 6 | 10 | 3 | 4 | 4 | 4 | 5 | 12 | 3 | 4 | 2 | 5 | 10 | 9 | 320 |
| | 8 | 18 | 3 | 6 | 17 | 7 | 1 | 7 | 14 | 17 | 6 | 18 | 14 | 8 | 18 | 7 | 19 | 6 | 9 | 10 | 7 | 4 | 5 | 9 | 11 | 12 | 11 | 7 | 4 | 2 | 9 | 3 | 8 | 4 | 5 | 11 | 9 | 10 | 7 | 12 | 5 | 360 |
| | 9 | 9 | 9 | 13 | 10 | 5 | 19 | 9 | 12 | 14 | 5 | 18 | 9 | 16 | 16 | 16 | 20 | 2 | 15 | 13 | 9 | 1 | 14 | 3 | 1 | 4 | 15 | 14 | 8 | 2 | 11 | 9 | 3 | 14 | 13 | 1 | 14 | 10 | 1 | 15 | 8 | 400 |
| | 10 | 17 | 2 | 14 | 15 | 17 | 18 | 12 | 13 | 2 | 8 | 1 | 16 | 23 | 23 | 16 | 23 | 14 | 14 | 13 | 16 | 5 | 8 | 12 | 10 | 4 | 13 | 15 | 2 | 11 | 9 | 5 | 11 | 9 | 3 | 12 | 9 | 6 | 3 | 8 | 8 | 440 |
| 3 | 1 | 20 | 6 | 14 | 26 | 20 | 24 | 6 | 9 | 25 | 7 | 10 | 8 | 15 | 17 | 12 | 6 | 4 | 4 | 1 | 5 | 4 | 2 | 4 | 3 | 2 | 2 | 3 | 3 | 5 | 5 | 4 | 5 | 4 | 3 | 1 | 2 | 4 | 1 | 3 | 1 | 300 |
| | 2 | 26 | 17 | 3 | 4 | 30 | 22 | 8 | 27 | 23 | 17 | 4 | 17 | 18 | 9 | 8 | 7 | 2 | 2 | 2 | 3 | 4 | 5 | 4 | 4 | 5 | 1 | 1 | 4 | 3 | 4 | 2 | 5 | 3 | 3 | 4 | 4 | 2 | 3 | 5 | 5 | 320 |
| | 3 | 20 | 10 | 18 | 29 | 26 | 24 | 11 | 16 | 16 | 2 | 18 | 10 | 9 | 7 | 20 | 19 | 2 | 2 | 4 | 1 | 5 | 5 | 5 | 1 | 4 | 4 | 4 | 3 | 2 | 3 | 3 | 4 | 4 | 4 | 5 | 5 | 3 | 4 | 3 | 5 | 340 |
| | 4 | 28 | 9 | 9 | 32 | 7 | 25 | 33 | 30 | 7 | 12 | 9 | 8 | 14 | 20 | 16 | 11 | 4 | 2 | 1 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 3 | 5 | 5 | 4 | 5 | 2 | 5 | 2 | 2 | 4 | 2 | 2 | 4 | 4 | 360 |
| | 5 | 22 | 20 | 4 | 20 | 7 | 29 | 34 | 20 | 34 | 16 | 8 | 19 | 13 | 17 | 9 | 13 | 3 | 7 | 6 | 7 | 6 | 2 | 6 | 4 | 2 | 6 | 4 | 4 | 2 | 2 | 2 | 4 | 4 | 4 | 3 | 2 | 2 | 1 | 6 | 6 | 380 |
| | 6 | 29 | 29 | 10 | 33 | 14 | 22 | 9 | 21 | 33 | 9 | 7 | 23 | 13 | 23 | 15 | 10 | 3 | 3 | 5 | 5 | 6 | 4 | 3 | 1 | 5 | 3 | 3 | 7 | 4 | 1 | 4 | 6 | 2 | 3 | 7 | 6 | 7 | 4 | 3 | 5 | 400 |
| | 7 | 7 | 13 | 28 | 36 | 17 | 23 | 27 | 27 | 32 | 23 | 15 | 3 | 7 | 25 | 19 | 13 | 7 | 1 | 4 | 7 | 6 | 3 | 5 | 7 | 7 | 1 | 7 | 7 | 7 | 1 | 3 | 2 | 4 | 6 | 1 | 2 | 3 | 6 | 7 | 1 | 420 |
| | 8 | 19 | 29 | 37 | 15 | 10 | 24 | 40 | 11 | 35 | 18 | 23 | 14 | 10 | 14 | 15 | 16 | 6 | 4 | 5 | 6 | 1 | 6 | 5 | 4 | 6 | 6 | 1 | 3 | 7 | 6 | 6 | 7 | 6 | 1 | 5 | 5 | 4 | 3 | 4 | 3 | 440 |
| | 9 | 30 | 34 | 22 | 38 | 31 | 37 | 25 | 2 | 11 | 7 | 15 | 22 | 24 | 19 | 7 | 21 | 9 | 4 | 7 | 10 | 3 | 3 | 3 | 4 | 3 | 5 | 2 | 1 | 6 | 6 | 5 | 1 | 6 | 9 | 6 | 4 | 8 | 1 | 3 | 6 | 460 |
| | 10 | 29 | 35 | 24 | 6 | 45 | 38 | 36 | 10 | 17 | 20 | 11 | 24 | 24 | 14 | 15 | 12 | 7 | 2 | 3 | 10 | 9 | 8 | 1 | 9 | 2 | 7 | 8 | 3 | 3 | 3 | 5 | 6 | 9 | 2 | 2 | 3 | 4 | 5 | 6 | 3 | 480 |
| 4 | 1 | 4 | 10 | 18 | 11 | 1 | 3 | 2 | 11 | 15 | 35 | 17 | 20 | 36 | 2 | 3 | 37 | 3 | 2 | 4 | 1 | 3 | 2 | 4 | 3 | 5 | 3 | 4 | 4 | 5 | 4 | 4 | 3 | 3 | 2 | 4 | 2 | 1 | 3 | 1 | 5 | 300 |
| | 2 | 7 | 7 | 16 | 6 | 8 | 10 | 3 | 17 | 6 | 21 | 20 | 31 | 21 | 33 | 27 | 7 | 2 | 2 | 3 | 3 | 6 | 5 | 4 | 3 | 4 | 2 | 6 | 4 | 3 | 1 | 2 | 2 | 1 | 4 | 1 | 6 | 6 | 6 | 1 | 3 | 320 |
| | 3 | 18 | 6 | 16 | 12 | 4 | 6 | 20 | 5 | 3 | 40 | 16 | 24 | 18 | 37 | 12 | 33 | 6 | 5 | 1 | 1 | 2 | 2 | 6 | 3 | 6 | 2 | 4 | 6 | 4 | 6 | 4 | 3 | 5 | 5 | 4 | 3 | 5 | 1 | 3 | 3 | 360 |
| | 4 | 9 | 7 | 5 | 10 | 7 | 6 | 16 | 7 | 18 | 18 | 27 | 36 | 12 | 40 | 34 | 3 | 5 | 2 | 6 | 2 | 5 | 5 | 2 | 1 | 3 | 6 | 4 | 2 | 6 | 3 | 4 | 5 | 2 | 1 | 3 | 5 | 3 | 3 | 6 | 1 | 340 |
| | 5 | 13 | 10 | 8 | 13 | 13 | 15 | 15 | 2 | 6 | 23 | 19 | 39 | 30 | 26 | 39 | 14 | 5 | 5 | 6 | 2 | 2 | 2 | 5 | 2 | 6 | 6 | 2 | 4 | 4 | 2 | 2 | 5 | 4 | 1 | 5 | 6 | 5 | 4 | 4 | 6 | 380 |
| | 6 | 19 | 2 | 18 | 13 | 9 | 9 | 3 | 7 | 20 | 46 | 10 | 49 | 7 | 28 | 12 | 48 | 2 | 7 | 5 | 1 | 6 | 4 | 2 | 3 | 5 | 5 | 4 | 2 | 2 | 5 | 6 | 7 | 5 | 6 | 3 | 6 | 2 | 5 | 2 | 5 | 400 |
| | 7 | 3 | 1 | 18 | 9 | 22 | 4 | 23 | 23 | 2 | 24 | 26 | 38 | 28 | 44 | 46 | 4 | 2 | 3 | 5 | 2 | 5 | 4 | 2 | 6 | 7 | 4 | 5 | 2 | 4 | 4 | 5 | 3 | 6 | 7 | 3 | 7 | 7 | 2 | 6 | 4 | 420 |
| | 8 | 3 | 11 | 17 | 4 | 10 | 9 | 9 | 23 | 24 | 35 | 47 | 28 | 22 | 37 | 11 | 40 | 2 | 1 | 6 | 7 | 5 | 7 | 3 | 4 | 7 | 5 | 4 | 2 | 4 | 5 | 1 | 6 | 4 | 2 | 7 | 5 | 7 | 7 | 7 | 2 | 440 |
| | 9 | 16 | 16 | 23 | 20 | 11 | 9 | 12 | 1 | 12 | 25 | 46 | 33 | 33 | 23 | 42 | 38 | 7 | 2 | 6 | 6 | 8 | 3 | 1 | 6 | 3 | 8 | 8 | 7 | 8 | 6 | 6 | 1 | 3 | 3 | 4 | 3 | 2 | 4 | 7 | 8 | 480 |
| | 10 | 10 | 14 | 20 | 13 | 6 | 23 | 16 | 1 | 12 | 15 | 47 | 35 | 37 | 44 | 35 | 17 | 5 | 3 | 6 | 3 | 2 | 2 | 7 | 2 | 3 | 7 | 7 | 7 | 5 | 4 | 4 | 1 | 7 | 7 | 5 | 6 | 6 | 6 | 3 | 7 | 460 |


DOI: https://doi.org/10.37266/ISER.2022v10i1.pp55-73