Sādhanā (2018) 43:200
https://doi.org/10.1007/s12046-018-0958-z

A mathematical model and solution methods for rail freight transportation planning in an Indian food grain supply chain

K P ANOOP[1], VINAY V PANICKER[1,*], MANU NARAYANAN[1] and C T SUNIL KUMAR[2]

[1]Department of Mechanical Engineering, National Institute of Technology Calicut, Calicut, India
[2]Food Corporation of India (Zonal Office), Mumbai, India
e-mail: kpanoop05@gmail.com; vinay@nitc.ac.in; manunarayananvgr@gmail.com; ctsunil@gmail.com

MS received 20 February 2018; revised 20 June 2018; accepted 20 June 2018; published online 1 November 2018

Abstract. This paper addresses the rail transportation of food grains undertaken by Food Corporation of India (FCI) to meet the requirements of the food security programme called Public Distribution System (PDS). The research focuses on improving the allocation of railway rakes transporting food grains to a set of storage warehouses. A penalty factor based approach is adopted to represent the considerations in transportation planning and three penalty factors such as rake penalty factor, weekly penalty factor and capacity utilization penalty factor are introduced for the purpose. The single source - multiple destination problem is formulated and solved using exact method to minimize the sum of these three penalty factor values, termed total penalty. Further, a heuristic named optimum rake allocation algorithm is developed and tested using a set of 35 problem instances. The proposed heuristic is found to be highly efficient in terms of solution quality and computation time. A case study of FCI Kerala Region is also carried out to validate the formulated model and the proposed heuristic. The work provides valuable insights into the practical issues encountered in rail freight transportation planning and proposes an effective solution methodology to address them.

Keywords. Food grain supply chain; rail freight transportation; Food Corporation of India; penalty factors; optimum rake allocation algorithm.

1. Introduction

A food grain supply chain extends from farmer to the end consumer through different stages including miller, wholesaler and retailer. A typical characteristic of a food grain supply chain is the bulky nature of goods handled, which makes the distribution function more challenging. Food grain supply chains in India consist of extensive distribution networks since the producing and the consuming regions are located geographically apart. This work addresses a live problem on the food grain transportation and allocation as part of the Public Distribution System, a food security programme in India.

Public Distribution System is the largest distribution network of its kind in the world and Food Corporation of India (FCI) is the Central Government Agency which undertakes the procurement, storage, transportation and allocation of food grains for PDS [1–3]. FCI operates a large network of storage warehouses spread throughout the country and utilizes multiple modes of transportation for the bulk movement of food grains. Around 90% of this transportation is carried out by rail movement (in rail wagons) and this paper focuses on the long-haul rail transportation of food grains undertaken by FCI.

Recently, considerable research attention has been received for the transportation and distribution problem of FCI [4–9]. Still, there exists further scope for improvements in the current system. While all the above mentioned works approach the problem from a broader perspective (in some cases, the span of supply chain configuration extend to multiple stages), the present research seeks to go deeper into the practical challenges faced in inter-state rail transportation of food grains at the regional level.

The problem under consideration corresponds to a multi-period problem and the planning horizon considered is four weeks (each month is assumed to be consisted of four weeks). The selection of the source warehouse for food grain transportation does not fall under the regional level decision making and as a result, the problem is modeled as a single source - multiple destination problem. The unaddressed challenges faced by FCI with regard to rail transportation of food grains are listed as follows:

- Detention of rail wagons carrying food grains at warehouses due to delay in the unloading process. This incurs a penalty cost called Demurrage Cost (DC) which is to be paid to the carrier organization, Indian Railway.

*For correspondence

- Maintaining a uniform capacity utilization of 80% at all warehouses.
- Providing prompt supply of food grains to warehouses to match the time of its outflow to the succeeding stage in the supply chain.

This work focuses on the above mentioned challenges and adopts a penalty factor based approach to address them. Three penalty factors introduced in this research are given below.

1. Rake penalty factor – A factor which quantifies the relative risk in allocating a full rake (full train load) to a warehouse over a half rake (half train load) in terms of the incurrence of demurrage cost. Since FCI plans food grain transportation in terms of half rakes and full rakes to avail quantity discount on bulk movement, this distinction is deemed important. The risk of allocating a full rake is calculated to be 2.52 times the risk of allocating a half rake. The proportion is calculated taking into consideration the following two aspects:

   i. extra cost incurred as demurrage cost when a full rake is placed instead of a half rake,
   ii. relative difference in the free time available for unloading the rake in both cases.

   The calculation of rake penalty factor is further detailed in appendix A

2. Capacity utilization penalty factor – The capacity utilization across warehouses is made uniform by incorporating this factor. It is the ratio of the storage capacity of the warehouse to the existing stock level. To minimize the capacity utilization penalty factor, the warehouses with less capacity utilization will be given preference for rake allocation.
3. Weekly penalty factor – Weekly penalty factor quantifies the priority of a particular week in a month over others for rake allocation. It is defined as the ratio of the total outflow (of food grains) from a warehouse in a month to the outflow (from the same warehouse) during a particular week in that month. Rake will be allocated to the week with relatively more outflow to minimize the weekly penalty factor. It is calculated using a three-month moving average forecasting method.

The scales of penalty factors are set based on their priorities, in accordance with the expert opinion of transportation managers at FCI. Rake penalty factor is given the highest priority and it takes a value of 20 for half rake allocation and 50 for full rake allocation. Second priority is given to the capacity utilization penalty factor that takes values starting from 10, where the value 10 represents 100% capacity utilization. Usually the scale ranges from 10 to 20 since all warehouses are half filled most of the time. Weekly penalty factor is fitted on a scale between 0 and 10 and thus receives the least priority.

The objective of the formulated mathematical model is to minimize the total penalty value, which is the sum of the three penalty factor values. An exact solution approach and a heuristic named Optimum Rake Allocation Algorithm (ORAA) are also presented. The proposed heuristic ORAA offers a time-efficient solution approach especially when the problem complexity increases. The formulated model and the proposed heuristic form the basis of the Decision Support System (DSS) being developed for assisting FCI managers in planning food grain transportation effectively.

To the best of authors' knowledge, this is the first work which addresses food grain transportation-allocation problem using a penalty factor based approach. Also, no work has been found in literature that considers the provision called two-point combination offered by Indian Railway. The provision allows to combine the demand of two destinations and place the order for a full rake from the source warehouse so that the cost discount for full rake ordering can be availed. However, there are restrictions on the destinations which can be combined under the provision and on splitting the wagons between the combined destinations. The specific characteristics of the problem demanded an unconventional approach to tackle it and this adds to the novelty of the work presented here.

The rest of the paper is organized into seven sections. Section 2 provides a review of relevant literature in the domain of rail freight transportation and food grain transportation. The formulated mathematical model is presented in section 3. Section 4 explains the proposed solution methodologies. The details and findings of the extensive computational study are presented in section 5. To support the problem examined, a case study is presented in section 6. Finally, the work is summarized and concluded in section 7 along with the scope for future research.

2. Review of relevant literature

Rail freight transportation is a well-studied area that opens up newer opportunities for research regularly, as a result of the emergence of new technologies and challenges. Crainic and Laporte [10] provide a review of the operations research models and methods used in freight transportation planning. The paper points out the planning issues in freight transportation from the strategic, tactical and operational perspectives.

It is interesting to note that even though a lot of research has been carried out in the area of rail freight transportation, there is a visible divide that exists between the solutions proposed in existing literature and their practical implementation [11, 12]. The reasons are attributed to the high complexity of the overall task and other practical issues in implementing optimum solutions. Disruptions and risks associated with rail freight operations has also attracted research attention. It is alarming to find from literature that, for each day of movement of inventory in the supply chain, it idles as pipeline inventory for another 20 days [13]. This emphasizes the need for maintaining a continuous flow of inventory throughout the supply chain using efficient transportation practices.

The assumption that the product is sourced from a single origin is made due to various reasons. The model proposed by Jula *et al* [14] optimizes the import of containerized goods from Asia to USA considering the product to be sourced from a single Asian origin. As the source and destination stations are located far apart, the distance between the sources are considered insignificant. Single source multiple destination routing problems can also be found in the domain of computer networks [15]. Messages originating from a single source node are routed to multiple destination nodes to minimize the cost in the network. Further, Banerjee *et al* [16] also address a problem with single supplier and multiple buyers.

Gelareh *et al* [17] and Baykasoğlu and Subulan [18] address freight transportation for a multi-period scenario. The former considers multi-period hub location problem while the latter examines a multi-objective, multi-period sustainable load planning problem. A multi-period scenario adds additional dimensions to the problem, especially with respect to the translation of effects of delays and deviations in one period to the succeeding periods. The present work addresses a multi-period scenario where the optimum period for allocation of food grain rakes are to be determined.

Recent articles related to food grain transportation reveal the growing research interest in the area. Asgari *et al* [19] present a case study of wheat production in Iran. A linear integer programming model is formulated to determine the optimal amounts of wheat to be transported from each producing province to consuming province. A genetic algorithm based solution methodology is also proposed. Hyland *et al* [20] propose conceptual and mathematical models for domestic grain supply chain that incorporates trucking, elevator storage and rail transportation.

The following works reported from Indian scenario address the various transportation issues in PDS. Being the largest distribution network of its kind in the world, it throws open greater opportunities for research in transportation and logistics.

Maiyar *et al* [4] propose a bi-level model that represents the initial stage of the four stage food grain transportation-distribution process taken up by FCI. A linear model is formulated for the first level of the transportation-distribution process which considers only a single mode of transportation (by road). The second level of the transportation-distribution process utilizes multimodal rail-road transportation and is represented by a mixed integer non-linear programming model. The article also presents two variants of particle swarm optimization algorithm as solution methodologies. The limitation of the work lies in the fact that it considers only the first stage of the transportation-distribution process which is relatively less complex than the remaining stages.

Tanksale and Jha [6] address the problem concerning storage and inter-state transportation of food grains taken up by FCI. The work aims to determine the optimal quantity of food grains that is to be stored (at) and transported to various warehouses within the FCI network. The paper proposes a mixed integer linear programming model to minimize the transportation and inventory costs associated with the supply chain. A heuristic based solution methodology is also presented. Nevertheless, it can be found that the model does not consider any particular mode of transportation and hence the related characteristics are omitted.

Food grain transportation-allocation problem of PDS is investigated by Mogale *et al* [5]. The paper considers the first two stages of the four stage transportation process. The researchers propose a mixed integer non-linear programming model to minimize the total cost which includes transportation, inventory and operational costs. A chemical reaction optimization meta-heuristic is also presented for solving the model. Further, Mogale *et al* [9] extend this work by considering three stages in the transportation process. Constraints related to vehicle capacity are newly added to the problem.

A similar contextual problem for bulk transportation of wheat in PDS is addressed by Mogale *et al* [7]. A mixed integer non-linear programming model is formulated to minimize the transportation, storage and operational costs. A hybrid chemical reaction optimization algorithm combined with tabu search is proposed for solving the model. Mogale *et al* [8] also consider the transportation and distribution problem of FCI by including additional factors like inventory holding cost and vehicle preference constraints. However, these works consider the transportation and distribution problem of FCI from a broader perspective leaving out ample scope for a deeper analysis of the practical difficulties faced in the inter-state rail transportation of food grains, especially at the regional level. The present work proceeds in this direction.

3. Problem formulation

This section presents a multi-period Integer Non-Linear Programming (INLP) model for the food grain transportation-allocation problem under consideration. The model develops a rake allocation plan which minimizes the sum of the values of three penalty factors such as rake penalty factor, weekly penalty factor and capacity utilization penalty factor. Various constraints are also included in the model considering the context of the problem.

The allocation of wagons to warehouses is done either as a full rake or a half rake (under the two-point combination provision) and hence the parameters such as demand, storage capacity and available storage space are also represented in terms of multiples of half rake (the carrying capacity of a half rake is considered as one unit). In the rake allocation plan, the value of '1' represents a half rake while '2' represents a full rake. Unbalanced scenarios where the total number of food grain rakes allotted to the set of warehouses exceeds its total demand are also considered in the model.

The set of rail-fed warehouses and goods sheds (transshipment terminals offered by Indian Railway for serving warehouses without rail connectivity) is represented by W = {W1, W2…, G1, G2,…} where W1, W2,… represent the rail-fed warehouses and G1,G2,… denote the railway goods sheds. The set of weeks is represented by P = {P1, P2, P3, P4}.

3.1 *Assumptions*

The various assumptions considered in the model are as follows:

- A month consists of four weeks and hence the planning horizon is taken as four weeks.
- The available storage space at a warehouse is known in advance and it changes only when an allocation is made to that warehouse during the planning horizon.
- All warehouses have the infrastructure capabilities to accommodate a full rake placement of food grains.
- Only one allocation is possible at a warehouse in a particular week irrespective of the allocation quantity.
- A full rake ordered under two-point combination provision can only be split equally.

3.2 *Mathematical model*

The INLP model formulated for the problem under consideration is presented in this sub-section.

Indices

i, j Warehouses in set W

p Weeks in set P

Input parameters

tr Total number of incoming half-rakes

s_i Storage capacity (in half-rakes) of warehouse $i \in W$

d_i Monthly demand (in half-rakes) of warehouse $i \in W$

st_i Storage space available (in half-rakes) at warehouse $i \in W$

$$c_{ij} = \begin{cases} 1 & \textit{if the demand of warehouses } i \textit{ and } j \textit{ can be combined} \\ 0 & \textit{otherwise} \end{cases}$$

Binary valued parameter which determines if two warehouses $i,j \in W$ can be combined or not.

pr_{ip} Weekly penalty value corresponding to week $p \in P$ at warehouse $i \in W$

$$rf_{ip} = \begin{cases} 20 & \textit{if a half rake is placed at warehouse } i \in W \textit{ in week } p \in P \\ 50 & \textit{if a full rake is placed at warehouse } i \in W \textit{ in week } p \in P \end{cases}$$

Rake penalty factor at warehouse $i \in W$ in week $p \in P$

Decision variables

w_{ip} Number of half-rakes allotted to warehouse $i \in W$ in week $p \in P$ *(Non-negative integer)*

$$m_{ijp} = \begin{cases} 1 & \textit{if the demands of warehouses } i \textit{ and } j \textit{ are combined in week } p \\ 0 & \textit{otherwise} \end{cases}$$

Binary variable which determines if the demands of warehouses i and j are combined in week p or not

Objective function

$$\text{Minimize Total Penalty} = \sum_{i \in W} \sum_{p \in P} \left(rf_{ip} + \min\left(1, w_{ip}\right) \times pr_{ip} \right) + \sum_{i \in W} \left(10 \times \frac{s_i}{\left(s_i - st_i + \sum_{p \in P} w_{ip} \right)} \right)$$

The first term of the objective function corresponds to the sum of rake and weekly penalty factors. The minimum value among '1' and w_{ip} is multiplied with the respective weekly penalty factor value. The variable w_{ip} may assume a value of 0, 1 or 2 depending on the allocation made to a warehouse in a particular week. When no allocation is made ($w_{ip} = 0$), the value of weekly penalty factor is also taken as zero since it is multiplied with the minimum value among 1 and 0. Further, when w_{ip} takes a value of one or two, the original value of weekly penalty factor is maintained by multiplying it with the minimum value among '1' and w_{ip}. Taking the minimum value among '1' and w_{ip} ensures that the weekly penalty factor does not get doubled in case the value of w_{ip} is two. The second term in the objective function represents capacity utilization penalty factor. Overall, the objective is to minimize the sum of these three penalty factor values which is termed as total penalty.

Subject to the following constraints:

$$\sum_{p \in P} w_{ip} \leq st_i \quad \forall i \tag{1}$$

Constraint (1) limits the total monthly allocation to a warehouse (over four weeks) within the storage space available at that warehouse.

$$\sum_{p \in P} w_{ip} \geq d_i \quad \forall i \tag{2}$$

Constraint (2) ensures that the monthly allocation to each warehouse is at least equal to meet the demand of that warehouse. Allocation of excess food grains is allowed considering the problem context.

$$\sum_{i \in W} \sum_{p \in P} w_{ip} = tr \tag{3}$$

Constraint (3) makes sure that all the incoming food grain rakes are allotted to warehouses.

$$m_{ijp} = \begin{cases} 0 \text{ or } 1 & c_{ij} = 1 \\ 0 & c_{ij} = 0 \end{cases} \quad \forall i, \forall j, \forall p \tag{4}$$

Constraints (4) to (8) correspond to the two-point combination provision. The restriction on combining the demand of warehouses under two-point combination is represented by constraint (4).

$$w_{ip} = w_{jp} \begin{cases} = 1 & m_{ijp} = 1 \\ \neq 1 & m_{ijp} = 0 \end{cases} \quad \forall i, \forall j, \forall p \tag{5}$$

The allocation of half rakes to warehouses combined under two-point combination provision is ensured by constraint (5). It also avoids half rake allocation to warehouses which are not combined.

$$m_{ijp} = m_{jip} \quad \forall i, \forall j, \forall p \tag{6}$$

Constraint (6) maintains consistency between the two-point destinations.

$$\sum_{j \in W} m_{ijp} \leq 1 \quad \forall i, \forall p \tag{7}$$

Constraint (7) avoids a warehouse from getting combined with more than one warehouse in a particular week.

$$rf_{ip} = \begin{cases} 0 & w_{ip} = 0 \\ 20 & w_{ip} = 1 \\ 50 & w_{ip} = 2 \end{cases} \quad \forall i, \forall p \tag{8}$$

Constraint (8) assigns value for the rake penalty factor in line with the allocation made to a warehouse in a particular week.

4. Solution methodology

In this section, the exact solution approach and a heuristic based solution methodology are presented.

4.1 *Exact solution approach*

The formulated mathematical model is solved using commercially available optimization software named 'A Mathematical Programming Language' (AMPL). The solver chosen is 'gecode', an open source non-linear solver which supports logical constraints.

The suitability of a solution approach for practical application can be ascertained only after an extensive study of its performance for different problem sizes in terms of solution quality and computation time. As a step in this direction, problem instances corresponding to different problem sizes are solved and analyzed. The problem size is varied by changing the number of warehouses in the set from four to ten and the computation time required for obtaining an optimum solution in each case is also noted. The details of this study are presented in section 5. The study revealed the inability of the proposed exact solution approach in providing optimum solutions within a reasonable time when the number of warehouses in the set goes beyond eight. Since the proposed model is to be applied in scenarios where the number of warehouses in the set may go well beyond eight, the need of an alternative solution methodology, especially for bigger problem sizes is identified. In this context, a heuristic based solution methodology is proposed.

4.2 *Heuristic based solution approach*

In order to improve the solvability limit of the problem instances, especially for bigger problem sizes where exact method becomes computationally inefficient, a heuristic based solution methodology is proposed.

4.2a *Optimum Rake Allocation Algorithm (ORAA)*: The Optimum Rake Allocation Algorithm (ORAA) proposed in the present research is a heuristic developed to allocate the incoming food grain rakes to a set of warehouses by minimizing the total penalty value. ORAA generates a feasible solution in each iteration, which is compared with the previous best solution (in terms of the total penalty value) to decide whether to retain it or not. As the iteration progresses, the best solution generated till that instant is retained and the inferior solutions are discarded. ORAA can solve the model in both balanced and unbalanced scenarios depending on the problem context. It is coded using MATLAB (version R2015a) and is verified and validated using a set of problem instances. The performance of ORAA in terms of solution quality and computation time is compared with that of solutions obtained using the exact method. Figure 1 presents the flowchart of the proposed heuristic.

4.2b *Working of ORAA:* The proposed optimum rake allocation algorithm progresses through five stages as follows:

Stage 1 – Initialization and setting the number of iterations

ORAA starts by initializing various parameters and setting the rake allocation matrix (*ra*) to zero. The rake allocation matrix is an *m* x *n* matrix representing the rake

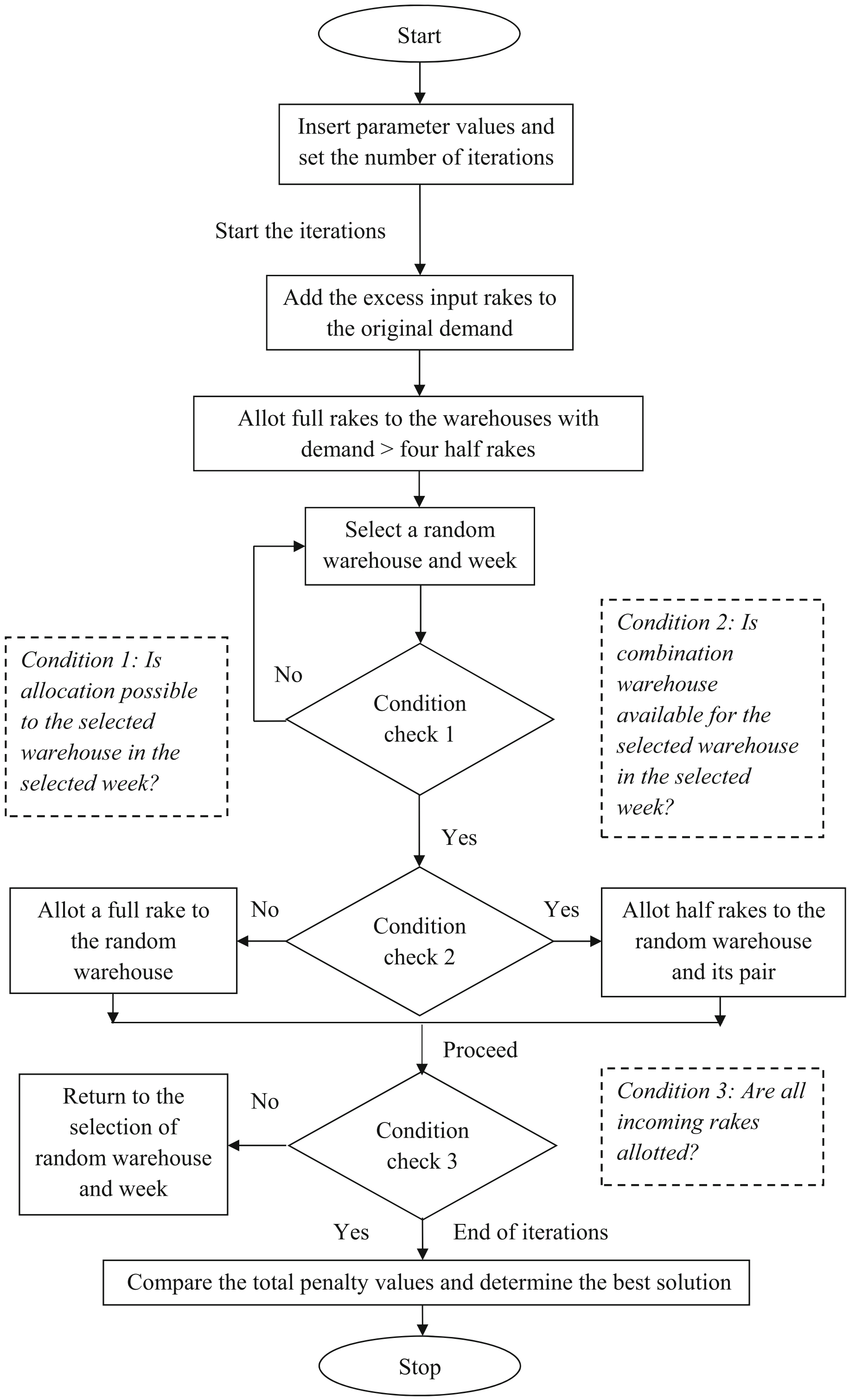

Figure 1. Flow chart of the proposed heuristic ORAA.

allocation to m warehouses in n weeks. The various parameters considered in the heuristic include the following:

(i) total number of rakes allotted to the set of warehouses for a particular month, tr

(ii) monthly demand of warehouse i, d_i

(iii) storage capacity of warehouse i, sc_i

(iv) the initial stock level at warehouse i, isl_i

(v) rake penalty factor corresponding to rake allocation at warehouse i in a particular

(vi) week p, rp_{ip}
(vii) capacity utilization penalty factor corresponding to warehouse i, cp_i
(viii) weekly penalty factor at warehouse i in a particular week p, wp_{ip}
(ix) maximum monthly allocation at warehouse i, mma_i, which is eight (a full rake in all four weeks which equals eight half rakes) in the problem under consideration due to the assumption that there can only be a single allocation at a warehouse in a particular week.
(x) number of iterations, $iter$

Stage 2 – Addition of excess rakes to original demand

Once the input parameters are set, the total demand of the set of warehouses and the total number of rakes allotted to the set of warehouses for the month are compared to determine whether the problem is balanced or not. In case the total number of rakes allotted is greater than the total demand (unbalanced scenario), the excess rakes are added to the original demand of various warehouses subject to space availability. This addition is done either randomly or by choosing the warehouse with the maximum capacity utilization penalty factor in an equally probable manner. The pseudocode for this stage is presented below.

- for $b = 1 \ldots iter$

  do

  - set ra = zeros (m,n)
  - total quantity, tq = sum (d_i)
  - excess, $ex = tr - tq$
  - deciding variable, $dv1$ = random number
  - available space, $ap_i = sc_i - isl_i$
  - while ($ex > 0$ & $dv1 < 0.5$)

    do

    - find the warehouse j with the maximum capacity utilization penalty factor
    - maximum possible addition, $mpa_j = ((\text{minimum}\ (mma_j, ap_j)) - d_j)$
    - maximum addition required, $mdr_j = \text{minimum}\ (mpa_j, ex)$
    - addition, a = random integer (1 to mdr_j)
    - $d_j = d_j + a$
    - $ex = ex - a$

    end while
  - while ($ex > 0$ & $dv1 > 0.5$)

    do

    - select a random warehouse j with $mpa_i > 0$ for addition of demand
    - maximum possible addition, $mpa_j = ((\text{minimum}\ (mma_j, ap_j)) - d_j)$
    - maximum addition required, $mdr_j = \text{minimum}\ (mpa_j, ex)$
    - addition, a = random integer (1 to mdr_j)
    - $d_j = d_j + a$
    - $ex = ex - a$

    end while

Stage 3 – Allocation of full rakes to warehouses with higher demand

A monthly demand greater than four half rakes at a warehouse is considered high since at least one full rake is required to meet the demand. This is due to the assumption that only a single allocation is possible at a warehouse in a particular week. Full rake allocation to the high demand warehouses are made in the third stage of ORAA. The high demand warehouses may be allocated the entire full rakes it require or a part of it. This distinction is made randomly. However, at least one full rake is allocated to the high demand warehouses. Remaining full rakes, if required will be allocated in stage 4. The pseudocode for this stage is presented below.

```
o  sum = sum(ra)
o  for all warehouses j with d_i > 4
   do
      for x = 1...(d_j – 4)
      do
         •  week, w = random integer (1 to n)
         •  if ra_jw = 0
               o  ra_jw = ra_jw + 2
               o  sum = sum + 2
               o  d_j = d_j - 2
            end if
      end for
   end for
```

Stage 4 – Half and full rake allocation to warehouses

In the working logic for this stage, a while loop is initiated for allocating half rakes and full rakes required to meet the remaining demand of the set of warehouses. The loop terminates only when the total demand of all the warehouses are met. Warehouses and weeks are selected at random. The rakes are allocated if the monthly demand of the selected warehouse is not met and if there is no allocation made to the selected warehouse in the selected week. The allocation process begins with half rake allocation and the full rake allocation is initiated only after certain number of iterations (k_{shift}) of the while loop. This is to ensure a higher preference for half rake allocations since it incurs lower rake penalty. Once the number of iterations goes beyond k_{shift}, the selection between half rake and full rake allocation is done in an equally probable manner based on the value of a random number generated. Combination warehouses are found out before initiating a half rake allocation and if no combination is available for a particular warehouse, the heuristic chooses a full rake allocation.

The chance of ORAA getting stuck at various points in the search space is averted by resetting the rake allocation matrix and other parameters to initial state when the iteration inside the while loop goes beyond a particular value (k_{reset}). In such cases, the execution restarts from stage 3. In addition, a provision is made for terminating the while loop for rake allocation if the number of iterations within the loop goes beyond a high value ($k_{terminate}$) and execution restarts from stage 2. The values of parameters k_{shift}, k_{reset} and $k_{terminate}$ are set based on extensive computational experiments. The pseudocode for this stage is provided below.

```
○ while sum < tq
  do
    ▪ iteration count, k =1
    ▪ warehouse, j = random integer (1 to m)
    ▪ w = random integer (1 to n)
    ▪ if ra_jw = 0 & d_j > 0
        • deciding variable, dv2 = random number
        • if combination warehouse, cw exist for j
            ○ if ra_cw,w = 0 & d_cw > 0 & dv2 < 0.5
                ▪ ra_jw = ra_jw + 1
                ▪ ra_cw,w = ra_cw,w + 1
                ▪ sum = sum + 2
                ▪ d_j = d_j – 1
                ▪ d_cw = d_cw – 1
              end if
            ○ if dv2 > 0.5 & k > k_shift
                ▪ ra_jw = ra_jw + 2
                ▪ sum = sum + 2
                ▪ d_j = d_j – 2
              end if
        • else
            ○ ra_jw = ra_jw + 2
            ○ sum = sum + 2
            ○ d_j = d_j – 2
      end if
    ▪ if k > k_reset
        • reset ra = zeros (m,n)
        • reset sum, d_i
        • restart from stage 3
      end if
    ▪ if k > k_terminate
        • break
        • restart from stage 2
      end if
    ▪ k = k + 1
○ end while
```

Stage 5 – Calculation of total penalty value and determination of the best solution

The total penalty value corresponding to the rake allocation plan in a particular iteration is calculated and compared with that obtained in the previous iteration. The rake allocation plan obtained in the first iteration (b = 1) is set as the final rake allocation plan (*fra*) which gets replaced with better solutions in subsequent iterations. The final rake allocation plan is replaced with the latest rake allocation plan if the total penalty value obtained in the latest iteration is less than that obtained in the previous iteration. The final rake allocation plan is retained in other cases. The final rake allocation plan and the corresponding final total penalty value (*ftp*) are displayed at the end. The pseudocode for this stage is as follows:

```
○ total penalty, tp = sum (rp_ip) + sum (wp_ip) + sum (cp)
○ if tp (iter) < tp (iter-1)
    ▪ final rake allocation, fra = ra (iter)
    ▪ final total penalty, ftp = tp (iter)
○ else
    ▪ retain fra and ftp
  end if

end for

display fra and ftp
```

For better understanding of the proposed heuristic, a numerical illustration of ORAA is provided in appendix B. The details of the computational study carried out to assess the performance of the exact solution approach and the proposed ORAA are provided in the following section.

5. Computational study

The objective of the computational study carried out as a part of this work is two folded as follows:

1. To verify and validate the proposed model and solution methodologies using problem instances that resemble practical scenario.
2. To compare the performance of the proposed solution methodologies with respect to the solution quality and computation time for different problem sizes.

5.1 *Design of computational experiments*

A set of 35 problem instances are used in the computational study to fulfill the aforementioned objective. The problem instances are designed such that the problem size (number of warehouses in the set) varies from four to ten and hypothetically generated data resembling real data is used to set various parameter values. The problem instances are designed to include both balanced (the total number of rakes allotted to the set of warehouses equals its total demand) and unbalanced (total number of rakes allotted to the set of warehouses exceeds its total demand) scenarios. In the set of problem instances, the first instance corresponding to each problem size (seven different problem sizes are considered) represents balanced scenario. The rest of the instances are generated by increasing the total number of half rakes allotted to the set of warehouses, keeping all other parameters constant. An increase of two half rakes is made per instance till the number of half rakes in excess of demand becomes not less than 30% of the original demand (the average figure from FCI indicates that the total number of rakes that gets allotted to the warehouses may go up to 30% in excess of the original demand). Hence, seven problem instances represent balanced scenarios while the remaining 28 instances are unbalanced.

The computational study is performed on a machine with Intel Core i5 3.20 GHz CPU with 8 GB RAM using the set of 35 problem instances. Exact solutions are generated by using the commercially available optimization software called 'A Mathematical Programming Language (AMPL)' with the non-linear solver named 'gecode' which supports logical constraints. The proposed ORAA is coded and solved using MATLAB (version R2015a). The same problem instances are solved using both the solution methodologies thereby enabling a direct comparison between their performances.

The parameter values corresponding to a sample problem instance are provided in table 1.

The combination matrix used in the above mentioned sample problem instance is shown in table 2. A value of '1' represents a possible combination between the destinations while the value '0' indicates a negation.

Table 1. Parameter values for the sample problem instance.

Warehouse	Parameter values in half rakes: Storage capacity	Monthly demand	Initial stock level
A	6	1	2
B	10	2	4
C	14	5	6
D	20	0	8
E	22	4	10
F	18	2	9
G	16	3	7
H	22	2	12
I	18	3	11

Table 2. Combination matrix for the sample problem instance.

Warehouse / Goods shed	Warehouse/Goods shed: A	B	C	D	E	F	G	H	I
A	–	1	1	0	0	0	0	0	0
B	1	–	0	0	0	1	0	0	0
C	1	0	–	1	0	0	0	0	0
D	0	0	1	–	1	0	0	0	0
E	0	0	0	1	–	1	0	0	0
F	0	1	0	0	1	–	1	0	0
G	0	0	0	0	0	1	–	1	1
H	0	0	0	0	0	0	1	–	1
I	0	0	0	0	0	0	1	1	–

Table 3. Weekly Penalty Matrix for the sample problem instance.

Warehouse	Weekly penalty factor: Week 1	Week 2	Week 3	Week 4
A	3	1	5	2
B	4	1	6	1
C	1	2	10	1
D	2	2	5	1
E	5	4	1	1
F	2	1	3	1
G	2	3	4	2
H	1	4	2	3
I	3	9	1	1

The weekly penalty matrix for solving the sample problem instance is included in table 3.

5.2 *Results and discussion*

The results and findings from the computational study are presented in this sub-section. The rake allocation plan for the sample problem instance solved using the exact methodology is shown in table 4. Solution for the same problem instance obtained by ORAA is provided in table 5. The same alphabets used as superscripts for half rake allocations (for a pair of warehouse) in the tables denote those warehouses whose orders are combined under the two-point combination provision. Total penalty value corresponding to rake allocation plans is also provided in each table.

On comparing the solutions presented in table 4 and table 5, it can be found that the total number of full rake and half rake allocations are the same in both plans. Also, the total number of rakes allotted to each warehouse is equal. Hence, the value of rake penalty factor and capacity utilization penalty factor will be the same in both the plans. However, there are variations in allocation weeks. For

Table 4. Rake allocation plan for the sample problem instance obtained using exact method.

Warehouse	Week 1	Week 2	Week 3	Week 4
A	0	1[a]	0	1[b]
B	0	1[a]	0	1[b]
C	2	1[c]	0	2
D	1[d]	1[c]	0	1[e]
E	1[d]	1[f]	1[g]	1[e]
F	0	1[f]	1[g]	0
G	1[h]	1[j]	0	1[m]
H	0	1[j]	1[k]	0
I	1[h]	0	1[k]	1[m]
Total penalty = 722				

The alphabet represents the pair of warehouses whose orders are combined under two-point combination provision.

Table 5. Rake allocation plan for the sample problem instance obtained using the proposed heuristic ORAA.

Warehouse	Week 1	Week 2	Week 3	Week 4
A	0	1[a]	0	1[b]
B	0	1[a]	0	1[b]
C	2	2	0	1[c]
D	0	1[d]	1[e]	1[c]
E	1[g]	1[d]	1[e]	1[f]
F	1[g]	0	0	1[f]
G	0	1[h]	1[j]	1[k]
H	1[m]	1[h]	0	0
I	1[m]	0	1[j]	1[k]
Total penalty = 725				

The alphabet represents the pair of warehouses whose orders are combined under two-point combination provision.

example, in the exact solution, half rake allocations are made to warehouse D in the first, second and fourth weeks. In the solution obtained using ORAA, the allocations are made in second, third and fourth weeks. As a result, weekly penalty factor value will be different in both the solutions and this is reflected in the total penalty value. The total penalty value is obtained as 722 for the exact solution and 725 for the proposed heuristic ORAA.

5.2a *Generation of balanced rake allocation plans:* The solutions provided in tables 4 and 5 display the effectiveness of the model and solution methodologies in generating balanced rake allocation plans. Since rake penalty for half rake allocation is the minimum, the demands of warehouses are met by half rake allocations to the greatest possible extent. Full rake allocations are used only in case half rake allocations cannot meet the demand.

It can be found that the sample problem instance corresponds to an unbalanced scenario where the inflow of food grains exceeds the total demand of warehouses. The total demand of warehouses is 22 half rakes and the total allotment corresponds to 26 half rakes. Hence, a decision is to be made regarding the accommodation of these excess rakes taking into consideration the capacity utilization at various warehouses. Capacity utilization factor assumes significance in this case. In both the plans, one excess half rake is allocated to warehouse 'A' and the remaining three to warehouse 'D'. It can be observed that these warehouses have comparatively low capacity utilization (50% and 40%, respectively).

The third criterion for planning rake allocation is the week in which the allocation is to be effected. Rake allocation to each warehouse is made in those weeks which minimize the weekly penalty factor value. That is, allocation is planned in those weeks where the outflow is comparatively higher. The solutions for the sample problem instance also reflect the ability of the model and solution methodologies to accommodate this consideration.

In short, the ability of the proposed model and the solution methodologies to generate solutions which are balanced with respect to all the three considerations of FCI is reflected in the sample rake allocation plans. The proposed approach for transportation planning helps FCI to overcome the challenges faced by it currently. Accommodating the risk of incurring demurrage cost in rake allocation planning is a step aimed at reducing the detention of wagons at warehouses and thus, the demurrage cost. Weekly penalty factor tries to ensure timely allocation of food grains to warehouses so that it matches the outflow to the succeeding stage. Further, capacity utilization penalty factor strives to achieve uniform capacity utilization across warehouses.

5.2b *Comparative performance analysis of solution methodologies for different problem sizes:* Table 6 presents the results from the set of problem instances using different problem sizes (different number of warehouses in the set).

The problem instances included in table 6 are represented by the problem size considered in that instance (the number that follows the letter 'W') and the total number of half rakes allotted to the set of warehouses for that month (the number that follows the letters 'TR'). For example, W6_TR22 represents the problem instance corresponding to problem size six (a set of 6 warehouses) with an allotment of 22 half rakes. The first problem instance corresponding to each problem size represents the balanced scenario while the subsequent problem instances represent unbalanced scenarios.

The results provided in table 6 are helpful in comparing the performance of the two solution methodologies for different problem sizes. It can be found that ORAA is able to provide solutions with total penalty value which fall within two percent deviation from those obtained by the exact method. The maximum percentage deviation observed in table 6 is 1.92 for the problem instance W6_TR22 and the minimum percentage deviation is 0 that is obtained in 14 instances. In the case of two problem instances with 10 warehouses (W10_TR28 and

Table 6. Results from the set of problem instances with varying problem size.

		Total penalty value			Computation time (in seconds)	
Sl. no.	Problem instance	Exact method	The proposed heuristic ORAA	Percentage deviation	Exact method	The proposed heuristic ORAA
1	W4_TR8	254	254	0	0.057	0.733
2	W4_TR10	287	287	0	0.146	1.547
3	W4_TR12	324	324	0	0.316	1.383
4	W4_TR14	370	370	0	0.543	1.392
5	W4_TR16	417	417	0	0.666	1.291
6	W5_TR12	372	372	0	0.712	3.947
7	W5_TR14	401	401	0	1.855	2.408
8	W5_TR16	439	439	0	4.318	1.957
9	W5_TR18	483	483	0	8.370	2.972
10	W5_TR20	530	540	1.89	12.111	2.784
11	W6_TR14	413	413	0	25.282	1.824
12	W6_TR16	445	445	0	79.714	1.815
13	W6_TR18	482	482	0	208.919	3.968
14	W6_TR20	525	526	0.19	460.629	2.214
15	W6_TR22	572	583	1.92	804.139	2.802
16	W7_TR18	545	545	0	87.471	7.471
17	W7_TR20	564	566	0.35	306.801	25.880
18	W7_TR22	601	602	0.17	1008.954	6.005
19	W7_TR24	643	647	0.62	2817.541	38.434
20	W7_TR26	688	700	1.74	6350.912	51.950
21	W8_TR20	587	587	0	494.556	32.168
22	W8_TR22	619	624	0.81	2242.977	13.457
23	W8_TR24	659	660	0.15	9335.162	6.803
24	W8_TR26	702	704	0.28	32182.743	8.830
25	W8_TR28	745	754	1.21	87455.207	59.387
26	W9_TR22	656	658	0.30	7650.751	26.974
27	W9_TR24	685	691	0.88	39476.908	6.152
28	W9_TR26	722	725	0.42	173373.252	30.498
29	W9_TR28	762	768	0.79	>244959.457*	50.234
30	W9_TR30	811	820	1.11	>244187.802*	45.096
31	W10_TR24	723	729	0.83	60498.009	6.473
32	W10_TR26	753	757	0.53	>229900.689*	7.781
33	W10_TR28	793	789	−0.50	>226893.705*	62.758
34	W10_TR30	836	837	0.12	>185577.096*	64.104
35	W10_TR32	884	880	−0.45	>184026.262*	46.085

Problem instance representation: W problem size_TR number of half rakes allotted), * Interrupted solutions.

W10_TR32), the solutions provided by ORAA have total penalty values lesser than the solutions obtained by interrupting the exact solver after 48 hours of execution.

The general trend observed in the results provided in table 6 is that the deviation in total penalty value increases with problem complexity which increases with problem size. Nevertheless, it is to be noted that problem complexity is not solely dependent on problem size. The total number of rakes allotted to the set of warehouses is also found to influence the expanse of the search space that in turn affects the complexity of the problem. This is obvious from the results in table 6 since the only change in data among the set of problem instances for the same problem size correspond to the total number of rakes allotted.

The advantage of using a heuristic based solution methodology is its ability to provide solutions with acceptable quality within a reasonable computation time. The time efficiency of the proposed heuristic named ORAA is ascertained by comparing the computation time required for the two solution methodologies. As observed in the case of deviation in total penalty value, the computation time is also found to vary with problem size and the total number of rakes allotted to the set of warehouses. This is evident from the results given in table 6.

Figure 2 represents the variation of average computation time with problem size in case of exact solution methodology.

The results of problem instances corresponding to balanced scenarios are omitted since it does not represent all

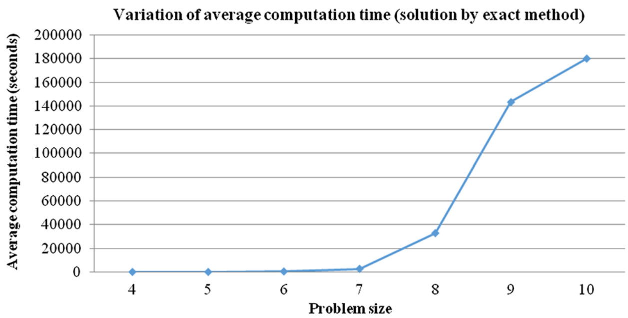

Figure 2. Variation of average computation time with problem size using exact solution approach.

the characteristics of the problem due to an insignificant capacity utilization penalty factor. The curve is found to have a steep rise when the problem size exceeds seven and this trend continues for the subsequent problem sizes. It is therefore understood that the exact method is unable to provide optimum solutions for certain problem instances with size nine and ten even after 48 hours of execution (The curve is plotted by taking the computation time for these instances as 50 hours, i.e., 180000 seconds). This finding indicates the inadequacy of the exact solution methodology in providing solutions within practically acceptable time limits when problem size goes beyond eight.

Figure 3 shows the variation of the average computation time with problem size for the solutions obtained by ORAA.

Like the curve shown in figure 2, the results corresponding to balanced scenarios are omitted for plotting the curve in this case also. There is variation in computation time with increase in problem size, but the magnitude is much smaller in this case with the maximum average value falling below one minute. The curve is found to be less steep compared to the exact case and it also shows a reduction in computation time when the problem size increases from seven to eight. This indicates that a constant trend of increase of computation time with problem size is absent in this case. These observations clearly suggest that the heuristic named ORAA can be adopted as a solution methodology for the problem, especially when the problem size equals or goes beyond eight.

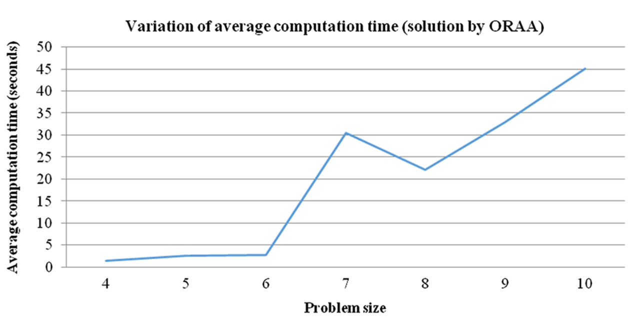

Figure 3. Variation of average computation time with problem size using the proposed heuristic ORAA.

6. Case study

To validate the formulated model and to evaluate the performance of the proposed heuristic ORAA, a real case involving eight warehouses of FCI Kerala Region is presented in this section.

6.1 *FCI Kerala Region*

The State of Kerala is in the southern part of India and the FCI network in Kerala constitutes the Kerala Region of FCI. Kerala is a deficit state in terms of food grain availability since production within the state is insufficient to meet the demand of its entire population. Hence, Kerala depends heavily on food grains transported inward from other surplus states in the country. The total quantity of food grains that get transported to FCI warehouses in Kerala account to 1.453 million metric tons annually. More than 90% of this transportation is taken up by Indian Railway and the transportation planning is done in line with the operating regulations put forward by Indian Railway.

The average demurrage cost per metric ton paid by FCI Kerala region during the financial year 2016-17 amounts to INR 27.22 while the national average stands at INR 6.80, as per the Official data. This clearly indicates the need for improving the transportation-allocation function of FCI Kerala Region with utmost urgency. FCI Kerala Region consists of 23 owned warehouses and a hired warehouse. The allocation of food grain rakes to a set of eight destinations (four warehouses and four goods sheds) located in the northern part of Kerala is considered in this case. These warehouses and goods sheds fall under Palakkad railway division and the list is as follows:

- Goods shed at Neeleswaram (for serving FCI warehouse at Neeleswaram)
- FCI warehouse at Payyannur
- Goods shed at Etakot (for serving FCI warehouse at Muzhapilangad)
- FCI warehouse at Thikkodi
- FCI warehouse at West Hill
- Goods shed at Thirunavaya (for serving FCI warehouse at Kuttipuram)
- Goods shed at Angadipuram (for serving FCI warehouse at Angadipuram)
- FCI warehouse at Palakkad

Figure 4 shows the geographical locations of FCI warehouses and railway goods sheds as obtained from the Google Maps.

The matrix representing the possible combination of destinations under Palakkad railway division is provided in table 7. A value of '1' represents a possible combination between the destinations while the value '0' indicates a negation.

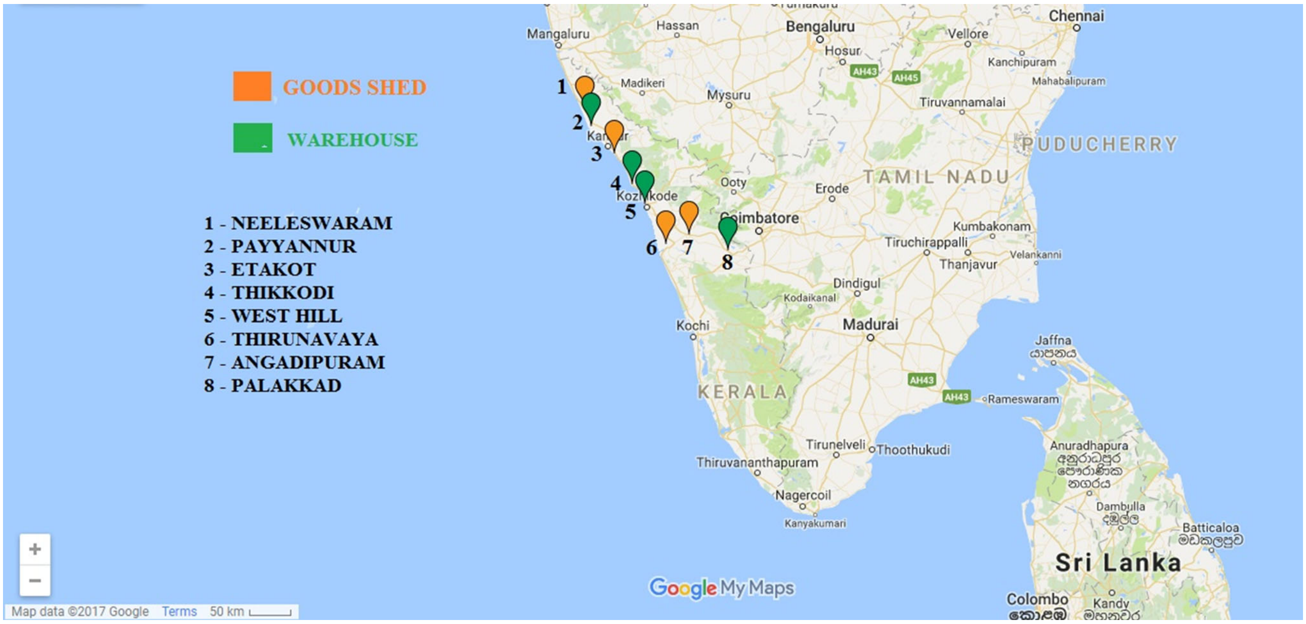

Figure 4. Geographical locations of FCI warehouses/goods sheds considered in the case study.

Table 7. Combination matrix used in the case study.

	Warehouse/Goods shed							
Warehouse/Goods shed	WHG	TKFS	ETK	NLE	PAYS	AAM	PGTS	TUA
WHG	–	1	0	0	0	1	0	1
TKFS	1	–	0	0	1	0	0	0
ETK	0	0	–	1	0	0	0	0
NLE	0	0	1	–	1	0	0	0
PAYS	0	1	0	1	–	0	0	0
AAM	1	0	0	0	0	–	1	1
PGTS	0	0	0	0	0	1	–	1
TUA	1	0	0	0	0	1	1	–

Value ‘1’ represents a possible combination and ‘0’ represents a negation.

The representation of warehouses and goods sheds in Table are as follows: WHG – West Hill, TKFS – Thikkodi, ETK – Etakot, NLE – Neeleswaram, PAYS – Payyannur, AAM – Angadipuram, PGTS – Palakkad and TUA – Thirunavaya.

6.2 *Computational experiments using historical data*

The problem instances generated using historical data from FCI are solved to verify and validate the proposed model and solution methodologies. Six problem instances are generated from the historical data of FCI corresponding to six months from January to June 2016. The monthly demand, storage capacity and the initial stock level at various warehouses, as well as the data corresponding to the outflow of food grains are taken from the historical data of FCI Kerala Region. Outflow of food grains from different warehouses is used to calculate weekly penalty factor. The combination matrix is generated based on the Rates Master Circular dated 05.06.2015 and the related Corrigendum No.03 dated 23.09.2015 issued by Railway Board, Ministry of Railways, Government of India.

The parameter values corresponding to the month of March 2016 are provided in table 8.

The total monthly demand of the set of warehouses for the month of March 2016 is 22 half rakes. The weekly penalty factor values for March 2016 are provided in table 9. The values are generated using data corresponding to the outflow of food grains during the three month period from December 2015 to February 2016.

The rake allocation plans obtained by solving the problem instance corresponding to March 2016 using the exact

Table 8. Parameter values corresponding to March 2016.

	Parameter values in half rakes		
Warehouse	Storage capacity	Monthly demand	Initial stock level
West Hill	13	4	2
Thikkodi	25	4	14
Muzhapilangad	6	2	2
Neeleswaram	5	2	3
Payyannur	16	1	15
Angadipuram	5	0	4
Palakkad	40	7	33
Kuttipuram	6	2	1

Table 9. Weekly penalty factor values corresponding to March 2016.

	Weekly penalty factor			
Warehouse	Week 1	Week 2	Week 3	Week 4
West Hill	2	4	10	1
Thikkodi	4	2	2	1
Muzhapilangad	2	2	2	1
Neeleswaram	2	2	2	1
Payyannur	4	4	2	1
Angadipuram	1	2	3	1
Palakkad	4	3	2	1
Kuttipuram	5	5	1	1

Table 10. Rake allocation plan for March 2016 obtained using exact method.

Warehouse	Week 1	Week 2	Week 3	Week 4
West Hill	1^a	1^b	1^c	1^f
Thikkodi	1^a	1^b	1^c	1^g
Muzhapilangad	0	0	1^d	1^h
Neeleswaram	0	0	1^d	1^h
Payyannur	0	0	0	1^g
Angadipuram	0	0	0	0
Palakkad	2	2	1^e	2
Kuttipuram	0	0	1^e	1^f
Total penalty = 626				

Pairs of superscripted alphabets denote combined warehouses.

method and ORAA are provided in tables 10 and 11, respectively.

A comparison between the rake allocation plans represented in Table 10 and Table 11 reveals that one pair of half rake allocations made to Muzhapilangad and Neeleswaram are in different weeks. In Table 10, the allocations are made in week 3, while in Table 11 the same is done in week 2. This is because the weekly penalty factor values for week 2 and week 3 are the same (2 each) for Muzhapilangad and Neeleswaram as shown in Table 3. No other penalty values are varied by the change of the allocation week and hence the total penalty value remains the same in both the cases. Multiple optima exist in this case.

Table 11. Rake allocation plan for March 2016 obtained using the proposed heuristic ORAA.

Warehouse	Week 1	Week 2	Week 3	Week 4
West Hill	1^a	1^b	1^d	1^f
Thikkodi	1^a	1^b	1^d	1^g
Muzhapilangad	0	1^c	0	1^h
Neeleswaram	0	1^c	0	1^h
Payyannur	0	0	0	1^g
Angadipuram	0	0	0	0
Palakkad	2	2	1^e	2
Kuttipuram	0	0	1^e	1^f
Total penalty = 626				

Pairs of superscripted alphabets denote combined warehouses.

The total penalty value and computation time corresponding to various problem instances solved by the exact method and ORAA are provided in table 12. The percentage deviation of the total penalty values obtained by ORAA from those obtained by the exact method is also included for comparing the performance of the two solution methodologies.

In table 12, each problem instance is represented by the first three letters of the corresponding month, four digits of the respective year and the total number of half rakes allotted to the set of warehouses for that month (the number that follows the letters 'TR'). For example, Jan 2016_TR16 represents the problem instance corresponding to the month of January in the year 2016 with an allotment of 16 half rakes.

The maximum percentage deviation of the total penalty value observed in Table 12 is 0.23 (Problem instance: Jun 2016_TR16) and the minimum percentage deviation is zero which occurs in five out of six instances. This means that ORAA has provided optimal solutions in five out of six problem instances. A comparison of the computation time between the two solution methodologies shown in Table 12 reveals that the maximum computation time for ORAA is 17.388 seconds (Problem instance: Mar 2016_TR22) while that for the exact method is 101.956 seconds (Problem instance: Jun 2016_TR16) in the worst-case scenario.

Overall, the results for the six problem instances reveal the ability of the proposed model and solution methodology in generating balanced rake allocation plans in terms of the three penalty factors. Comparison between the rake allocation plans generated by ORAA and the historical plans of FCI is included in the following sub-section.

Table 12. Results of problem instances solved as part of case study.

		Total Penalty Value			Computation time (in seconds)	
Sl. no.	Problem instance	Exact method	The proposed heuristic ORAA	Percentage deviation	Exact method	The proposed heuristic ORAA
1	Jan 2016_TR16	521	521	0	0.483	2.526
2	Feb 2016_TR12	383	383	0	4.154	2.234
3	Mar 2016_TR22	626	626	0	6.516	17.388
4	Apr 2016_TR18	476	476	0	28.685	5.500
5	May 2016_TR14	391	391	0	30.428	4.543
6	Jun 2016_TR16	433	434	0.23	101.956	4.870

Problem instance representation: month year_TR number of half rakes allotted.

Table 13. Comparison of proposed rake allocation plans with historical plans of FCI.

		Total penalty value		
Sl. no.	Problem instance	Proposed plan using ORAA	Historical plan of FCI	Percentage improvement
1	Jan 2016_TR16	521	522	0.19
2	Feb 2016_TR12	383	385	0.52
3	Mar 2016_TR22	626	631	0.79
4	Apr 2016_TR18	476	478	0.42
5	May 2016_TR14	391	415	5.78
6	Jun 2016_TR16	434	470	7.66

6.3 *Comparison of proposed rake allocation plans with historical plans of FCI*

A comparison is made between the rake allocation plans generated by ORAA and the historical plans of FCI Kerala Region for the six problem instances under consideration. Table 13 provides the total penalty value corresponding to the rake allocation plans obtained from ORAA and the historical plans of FCI.

It can be observed from Table 13 that the total penalty value of rake allocation plans proposed by ORAA is less than historical plans of FCI in all the six problem instances. The improvement varies from 0.19% to 7.66% and this indicates the potential of the proposed model and solution methodology in generating better rake allocation plans as compared to the existing planning method. The improvement in total penalty value may depend on individual problem instances due to variation in problem complexity which depends on the number of warehouses and the demand values.

The effect of number of warehouses and demand values on the complexity of the problem is evident from the results of computational study presented in section 5. Hence, the proposed method can have huge impact for bigger problem sizes and when the complexity is high. Even in a limited comparative study of six problem instances shown in Table 13, the highest improvement is found in the sixth problem instance for which the computation time for the exact method is also high (Table 12), thus indicating higher complexity. Even a small improvement in rake allocation plan can result in substantial financial gains for the organization as a result of the huge size of the system. This adds to the relevance of efforts to improve the transportation-allocation function of FCI.

7. Conclusions and managerial implications

The paper addresses a live problem on food grain transportation and allocation undertaken by the Food Corporation of India as part of the Public Distribution System in India. A mathematical model is formulated to optimize the allocation of food grain rakes to a set of FCI warehouses and a heuristic named ORAA is proposed for solving it. A case involving the transportation-allocation function of FCI Kerala Region is also studied and reported.

The formulated model minimizes the total penalty value which is the sum of three penalty factors such as rake penalty factor, weekly penalty factor and capacity utilization penalty factor. Rake penalty factor tries to minimize the risk of incurrence of demurrage cost, capacity utilization penalty factor tries to maintain uniform capacity utilization across the set of warehouses and weekly penalty factor tries to match the period of allocation and outflow of food grains at warehouses. The two-point combination provision offered by Indian Railway and its associated constraints are also considered.

The proposed heuristic ORAA aims to bridge the gap between the large computation time for an exact solution and the practically acceptable time limit from a decision maker's perspective. ORAA is tested using problem instances corresponding to different problem sizes and complexities. The proposed heuristic ORAA is found to provide good quality solutions within a short computation time. The obtained solutions fall within two percent deviation from the optimum and the maximum computation time observed is 66.502 seconds. This clearly indicates that the proposed heuristic ORAA is fit for use in real case applications where the problem complexity can vary significantly. The case study of FCI Kerala Region also suggests the effectiveness of the proposed model and solution methodology in generating better rake allocation plans compared to the existing method. A comparison between the historical plans and proposed plans shows that the latter is superior to the former in terms of total penalty value.

The formulated model and the proposed heuristic named ORAA are intended to serve as the basis for the Decision Support System (DSS) being developed for food grain transportation planning in FCI. The DSS aims to provide a scientific approach for transportation planning in the organization, which currently relies on conventional methods. The DSS is supposed to generate rake allocation plans which are more balanced in terms of the various considerations of FCI within practically acceptable time limits.

The transportation-allocation problem of FCI presented in this paper represents the typical considerations and challenges of firms which undertake long-haul rail transportation on a massive scale. A deeper analysis of these issues may prompt researchers to go beyond the commonly perceived modelling methodologies and performance measures to tackle them. The present work heads in this direction paving a path for more research that attend the real world problems closely. The proposed model can be modified to meet the requirements of similar firms undertaking long-haul rail transportation of goods by revising the objective function and constraints. The problem can also be approached as a multi-objective problem by considering the minimization of each penalty factor as distinct objectives. The authors are currently working in this direction.

Acknowledgements

This work has been substantially supported by the Ministry of Human Resource Development (MHRD), Government of India. The authors acknowledge the valuable support and guidance provided by Mr RD Nazeem (IAS), Executive Director (FCI South Zone), Mr Syju J S, former General Manager (FCI Kerala Region) and the employees at FCI Regional office (Kerala) and Calicut District office. The authors are also thankful to the referees and the Corresponding editor for their constructive suggestions and comments which have immensely helped to bring this article to the present form.

Appendix A

Setting rake penalty factor

The problem considers rail wagons of type BCNHL offered by Indian Railway for food grain distribution. For this type, a half rake consists of 29 wagons and a full rake consists of 58 wagons.

Free time available for unloading a half rake = 7 hours (stipulated by Indian Railway)

Free time available for unloading a full rake = 11 hours

Average free time per wagon for unloading a half rake = 7/29 = 0.24 hours

Average free time per wagon for unloading a full rake = 11/58 = 0.19 hours

Difference in average free time per wagon between a half rake and a full rake = 0.05 hours

Excess free time per wagon available for a half rake with respect to a full rake = (0.05/0.19) x 100 = 26.31%

That is, when the free time available for a full rake is 'x' hours, a half rake gets '1.26 x' hours.

Since the demurrage cost is calculated on the basis of number of wagons in the rake, a full rake allocation incurs double the cost as that of half rake allocation for the same amount of detention time. It can be seen that the risk of incurrence of demurrage cost is affected by the number of wagons in the rake and the free time available for unloading it.

The risk of incurrence of demurrage cost for a full rake = 2 x 1.26 = 2.52 times that of a half rake, combining the relative time and cost considerations.

Rake penalty factor is chosen as 20 and 50 for half rake and full rake allocations respectively, to keep up with the proportion of risk of incurrence of demurrage cost, which is 1: 2.52 (The value is rounded off for convenience). Setting values of 20 and 50 also ensure the highest priority for rake penalty factor among the three penalty factors since the other penalty factors are set below the lowest rake penalty factor of 20.

Appendix B

Numerical illustration of ORAA

Stage 1: Initialization and setting the number of iterations

$m = 8$; $n = 4$

ra = [0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0]

$tr = 24$; d = [4; 4; 2; 2; 1; 0; 7; 2]; sc = [13; 25; 6; 5; 16; 5; 40; 6]

isl = [2; 14; 2; 3; 15; 4; 33; 1]; rp_{ip} = [20 for half rake; 50 for full rake]

wp = [2 4 10 1; 4 2 2 1; 2 2 2 1; 2 2 2 1; 4 4 2 1; 1 2 3 1; 4 3 2 1; 5 5 1 1]

mma = [8, 8, 8, 8, 8, 8, 8, 8]; $iter = 5000$

Stage 2: Addition of excess rakes to original demand

(Let the current iteration be 101)

$b = 101$

ra = [0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0]

$tq = 22$

$ex = 24 - 22 = 2$

$dv1 = 0.3$

ap = [13; 25; 6; 5; 16; 5; 40; 6] - [2; 14; 2; 3; 15; 4; 33; 1] = [11; 11; 4; 2; 1; 1; 7; 5]

($ex > 0$ & $dv1 < 0.5$) >> Proceed

capacity utilization penalty factor (rounded) = round down (10 x ($sc_i / (isl_i + d_i)$))

= [21; 13; 15; 10; 10; 12; 10; 20]

$j = 1$;

mpa_1 = (minimum (8, 11) – 4) = 4

mdr_1 = minimum (4, 2) = 2

$a = 2$

$d_1 = 4 + 2 = 6$

$ex = 2 - 2 = 0$

($ex = 0$ & $dv1 < 0.5$) >> Jump to stage three

Hence, d = [6; 4; 2; 2; 1; 0; 7; 2]

Stage 3: Allocation of full rakes to warehouses with higher demand

sum = 0

j = [1,7]

j = 1

x = 1

w = 3

ra_{13} = 0 >> Proceed

ra_{13} = 0 + 2 = 2

sum = 0 + 2 = 2

d_1 = 6 – 2 = 4

x = 2

w = 4

ra_{14} = 0 >> Proceed

ra_{14} = 0 + 2 = 2

sum = 2 + 2 = 4

d_1 = 4 – 2 = 2

j = 7

x = 1

w = 1

ra_{71} = 0 >> Proceed

ra_{71} = 0 + 2 = 2

sum = 4 + 2 = 6

d_1 = 7 – 2 = 5

x = 2

w = 2

ra_{72} = 0 >> Proceed

ra_{72} = 0 + 2 = 2

sum = 6 + 2 = 8

d_1 = 5 – 2 = 3

x = 3

w = 1

$ra_{71} \neq 0$ >> Exit the for loops and jump to stage four

Hence, *ra* = [0 0 2 2; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0; 2 2 0 0; 0 0 0 0]

Stage 4: Half and full rake allocation to warehouses

k_{shift} = 135; k_{reset} = 575; $k_{terminate}$= 7450

(*sum* (= 8) < *tq* (= 24)) > > Proceed

k = 1

j = 3

w = 1

(ra_{31} = 0 & d_3 (= 2) > 0) > > Proceed

dv2 = 0.2

cw = [4] > > Proceed

(ra_{41} = 0 & d_4 (= 2) > 0 & *dv2* < 0.5) > > Proceed

ra_{31} = 0 + 1 = 1

ra_{41} = 0 + 1 = 1

sum = 8 + 2 = 10

d_3 = 2 – 1 = 1

d_4 = 2 – 1 = 1

(*dv2* = 0.2 & *k* = 1) > > Jump to the penultimate step inside the while loop in stage four

k = 1 + 1 = 2

Hence, *ra* = [0 0 2 2; 0 0 0 0; 1 0 0 0; 1 0 0 0; 0 0 0 0; 0 0 0 0; 2 2 0 0; 0 0 0 0]

(After certain iterations inside the while loop in stage four)

Let *ra* = [1 1 2 2; 1 1 0 2; 1 0 0 1; 1 0 0 1; 0 1 0 0; 0 0 0 0; 2 2 0 1; 0 1 0 1]

k = 151

j = 7

w = 3

(ra_{73} = 0 & d_7 (= 2) > 0) > > Proceed

dv2 = 0.7

cw = [6, 8] > > Proceed

(ra_{63}, ra_{83} = 0 & d_6, d_8 = 0 & *dv2* > 0.5) > > Jump to the next if statement

(*dv2* > 0.5 & *k* > 135) > > Proceed

ra_{73} = 0 + 2 = 2

sum = 22 + 2 = 24

d_7 = 2 – 2 = 0

(*k* < 575) > > Jump to the penultimate step inside the while loop in stage four

k = 151 + 1 = 152

Exit while loop and jump to stage five

Hence, *ra* = [1 1 2 2; 1 1 0 2; 1 0 0 1; 1 0 0 1; 0 1 0 0; 0 0 0 0; 2 2 2 1; 0 1 0 1]

Stage 5: Calculation of total penalty value and finding the best solution

sum (rp_{ip}) = (Number or 1 s in the final rake allocation plan x 20 + Number of 2 s in the final rake allocation plan x 50) = (12 x 20 + 6 x 50) = 540

sum (wp_{ip}) = Sum of the corresponding values in the *wp* matrix for which there is an allocation made in the rake allocation plan = 2 + 4 + 10 + 1 + 4 + 2 + 1 + 2 + 1 + 2 + 1 + 4 + 4 + 3 + 2 + 1 + 5 + 1 = 50

sum (cp_i) = sum (round down (10 x (sc_i / (isl_i + d_i))))- = round down each (10 x ((13/8) + (25/18) + (6/4) + (5/5) + (16/16) + (5/4) + (40/40) + (6/3))) = 16 + 13 + 15 + 10 + 10 + 12 + 10 + 20 = 106

tp (101) = 540 + 50 + 106 = 696

(Let the *tp* value corresponding to the rake allocation plan in the previous iteration, *tp* (100) = 700)

tp (101) < 700 > > Proceed

fra = [1 1 2 2; 1 1 0 2; 1 0 0 1; 1 0 0 1; 0 1 0 0; 0 0 0 0; 2 2 2 1; 0 1 0 1]

ftp = 696
(continue for loop till *b* = 5000)
Exit the loop and display *fra* and *ftp* in the end.

References

[1] Ambekar S, Kapoor R and Mehta P 2015 Structural mapping of public distribution system using multi-agent systems. *Bus. Process Manag. J.* 21: 1066–1090

[2] NITI Ayog 2016 Evaluation study on role of public distribution system in shaping household and nutritional security India. http://niti.gov.in/writereaddata/files/document_publication/Final%20PDS%20Report-new.pdf. Accessed 13 December 2017

[3] Planning Commission 2002 Public distribution system and food security. *Final document on midterm appraisal - ninth five year plan (1997–2002)* http://planningcommission.nic.in/plans/mta/mta-9702/mta-ch8.pdf. Accessed 13 December 2017

[4] Maiyar L M, Thakkar J J, Awasthi A and Tiwari M K 2015 Development of an effective cost minimization model for food grain shipments. *IFAC-PapersOnLine* 48: 881–886

[5] Mogale D G, Kumar S K and Tiwari M K 2016 Two stage Indian food grain supply chain network transportation-allocation model. *IFAC-PapersOnLine* 49: 1767–1772

[6] Tanksale A and Jha J K 2016 An effective heuristic for multi-period multi-foodgrain inventory transportation problem in India. *Infor* 54: 355–379

[7] Mogale D G, Kumar S K, Márquez F P G and Tiwari M K 2017a Bulk wheat transportation and storage problem of public distribution system. *Comput. Ind. Eng.* 104: 80–97

[8] Mogale D G, Dolgui A, Kandhway R, Kumar S K and Tiwari M K 2017b A multi-period inventory transportation model for tactical planning of food grain supply chain. *Comput. Ind. Eng.* 110: 379–394

[9] Mogale D G, Kumar S K and Tiwari M K 2018 An MINLP model to support the movement and storage decisions of the Indian food grain supply chain. *Control Eng. Pract.* 70: 98–113

[10] Crainic T G and Laporte G 1997 Planning models for freight transportation. *Eur. J. Oper. Res.* 97: 409–438

[11] Ferreira L 1997 Planning Australian freight rail operations: An overview. *Transport. Res. A-Pol.* 31: 335–348

[12] Gorman M F, Clarke J P, Gharehgozli A H, Hewitt M, de Koster R and Roy D 2014 State of the Practice: A Review of the Application of OR/MS in Freight Transportation. *Interfaces* 44: 535–554

[13] Morash E A and Clinton S R 1997 The role of transportation capabilities in international supply chain management. *Transport. J.* 36: 5–17

[14] Jula P and Leachman R C 2011 A supply-chain optimization model of the allocation of containerized imports from Asia to the United States. *Transport. Res. E-Log.* 47: 609–622

[15] Bharath-Kumar K and Jaffe J M 1983 Routing to multiple destinations in computer networks. *IEEE T. Comm.* 31: 343–351

[16] Banerjee A, Burton J and Banerjee S 2003 A simulation study of lateral shipments in single supplier, multiple buyers supply chain networks. *Int. J. Prod. Econ.* 81–82: 103–114

[17] Gelareh S, Neamatian Monemi R and Nickel S 2015 Multi-period hub location problems in transportation. *Transport. Res. E-Log.* 75: 67–94

[18] Baykasoğlu A and Subulan K 2016 A multi-objective sustainable load planning model for intermodal transportation networks with a real-life application. *Transport. Res. E-Log.* 95: 207–247

[19] Asgari N, Farahani R Z, Rashidi-Bajgan H and Sajadieh M S 2013 Developing model-based software to optimise wheat storage and transportation: A real-world application. *Appl. Soft Comput.* 13: 1074–1084

[20] Hyland M F, Mahmassani H S and Bou Mjahed L 2016 Analytical models of rail transportation service in the grain supply chain: Deconstructing the operational and economic advantages of shuttle train service. *Transport. Res. E-Log.* 93: 294–315