

Paper 3857-2019

Railways Scheduling and Rolling Stock Optimization using SAS®

Ayush Tiwari, Lokendra Devangan, Malleswara Sastry, Core Compete; Faiz Hamid, IIT Kanpur

ABSTRACT

Passenger comfort, optimal utilization of rolling stock and proper allocation of operational resources are the essential aspects for efficient management for any rail infrastructure. In this paper we address a scheduling problem for railways with the aim to minimize travel time of all the trains of a route by reducing the number of stoppages and optimal allocation of passenger quota/seats between every pair of stations in each train using modern SAS® technologies and SAS®/OR® software. An attempt is also made to investigate whether the given demand can be met by lesser number of trains. Mixed integer linear programming model is formulated to meet the two simultaneous decisions in scheduling objective- to minimize the number of running trains as well as their stoppages. The computational results with real life study on Indian Rail Network show a significant reduction in the number of stoppages and trains while satisfying the given demand.

Keywords: berth allocation, rolling stock, coach configuration, Proc OPTMODEL

INTRODUCTION

Indian Railways (IR) is one of the world's largest rail network. It is a departmental undertaking of the Government of India, which owns and operates most of India's rail transport. As of March 2017, IR network spans 121,407 km (75,439 mi) of track length, while the route length is 67,368 km (41,861 mi), spread across around 8500 stations. It operates more than 22,300 trains daily including passenger and freight trains. Over 23 million passengers and over 3 million tonnes of freight travel by trains daily in India. The passenger traffic stood at 8,219 million in FY17. India's railway network is recognised as one of the largest railway systems in the world under single management. The increase in the use of railway transport has led to more dense timetables in which small delay in operation of one train easily propagates and affects other traffic. Hence there arises the need to ensure that the train travel time is minimised, and utilisation of rolling stock is maximised while meeting the passenger demand. This motivates us to study timetabling/scheduling of trains and tuning the stoppages such that travel time is minimised and customer satisfaction is improved.

Timetable management, such as train scheduling, rescheduling, timetable compression, are common techniques to improve the timetables with an objective to increase track/ line capacity and allow for additional trains along a given corridor. With ever growing demand of railways as a means of transport, both for passengers and freight operations, there remains a huge gap between the currently applied and the optimal model in Indian Railways. Further, different trains have different stoppages and different speeds though running on the same route; owing to their status, passenger and structural requirements. Most often, trains in Indian Railways run late compared to their original schedule. There are several reasons behind that, an important one being non-optimal stoppages assigned to different trains and speed differential of trains.

Our study aims to solve the problem two folds i.e. to find the minimum number of trains that are required to completely satisfy the demand between every pair of stations on the given route, along with the optimal/ minimum number of stoppages for all the trains running on a given route/ segment. This leads to optimal utilization of rolling stock.

RAILWAYS NETWORK OPTIMIZATION PROBLEM

Since Indian Railways is a public corporation, it is necessary to simultaneously optimize passenger as well as operation centric concerns. Operations centric approaches emphasize on curtailing adverse effects related to railway corporations, such as interruptions, number of cancelled and running trains, etc whereas passenger centric approaches focus on lessening adverse effects related to passengers, such as total travel time or number of connections. There is a novel attempt in this research to achieve the dual objective of finding the minimal stoppages for each train running on a given route while satisfying the passenger demand between every pair of stations along with minimising the number of trains using the capabilities of SAS®. The formulated model is implemented using the MILP solver of the Proc OPTMODEL in SAS® Enterprise Guide® and the results visualization is performed using SAS® Visual Analytics®.

MODEL FORMULATION AND ANALYSIS

In this section, we formulate the model for railways scheduling and rolling stock management. The passenger demand in the model is assumed to be invariable with respect to the different time slots of operation. Hence the demand data in this study is taken as fixed between the pair of stations. The problem is to ascertain the number of berths to be allocated between every pair of stations in each passenger train of any route of Indian Railways such that total demand of passengers of that route is met and stoppages are minimized. The identified problem is formulated as a mixed integer linear programming problem and solved for Howrah to Delhi via main line (HWH-DHN-GAYA-MGS-ALD-CNB-NDLS) of Indian Railways. However, the proposed model is applicable to any other route as well.

INPUT PARAMETERS

M: set of trains, $M = \{1, 2, 3, \dots, m\}$

N: set of stations, $N = \{1, 2, 3, 4, \dots, n\}$

L: set of classes, $L = \{1, 2, 3, 4, \dots, l\}$

$C_{r,k}$: capacity of class r in train k

$d_{i,j,r}$: passenger demand between stations i and j in class r

DECISION VARIABLES

$x_{i,j,r,k}$: number of berths to be allocated between stations i and j in class r of train k

$u_{i,k}$: binary variable = 1 if train k stops at station i , 0 otherwise

y_k : binary variable = 1 if train k will run, 0 otherwise

MATHEMATICAL FORMULATION: TRAINS AND STOPPAGES MINIMIZATION MODEL

In this section, a model is proposed with two components of different weightages in its objective which allows minimizing the number of trains as its first priority decision and then minimizing the total number of stoppages for those running trains as second priority decision, for a given demand. It is important to note that, since the demand is completely satisfied, this model leads to better utilization of capacity. The model is formulated as follows:

$$\text{Minimize} \quad \alpha_1 \times \sum_{k=1}^m y_k + \beta_1 \times \sum_{i=1}^n \sum_{k=1}^m u_{i,k} \quad (1)$$

subject to,

$$\sum_{p=1}^i \sum_{q=i+1}^n x_{p,q,r,k} \leq C_{r,k} \quad i \in \{1, 2, 3, \dots, n-1\}, \forall r \in L, \forall k \in M \quad (2)$$

$$\sum_{k=1}^m x_{i,j,r,k} \geq d_{i,j,r} \quad \forall (i,j) \in N \times N \text{ and } i < j, \forall r \in L \quad (3)$$

$$\sum_{j=i+1}^n \sum_{r=1}^l x_{i,j,r,k} \leq D_1 \times u_{i,k} \quad \forall i \in N, \forall k \in M \quad (4)$$

$$\sum_{j=1}^{i-1} \sum_{r=1}^l x_{i,j,r,k} \leq D_2 \times u_{i,k} \quad \forall i \in N, \forall k \in M \quad (5)$$

$$u_{i,k} \leq y_k \quad \forall i \in N, \forall k \in M \quad (6)$$

$$x_{i,j,r,k} \geq 0, \quad u_{i,k}, y_k \in \{0,1\}$$

where D_1 and D_2 are large constants which may take the following values,

$$D_1 = \text{Min}\{\sum_{r=1}^l C_{r,k}, \sum_{j=i+1}^n \sum_{r=1}^l d_{i,j,r}\} \quad (7)$$

$$D_2 = \text{Min}\{\sum_{r=1}^l C_{r,k}, \sum_{j=1}^{i-1} \sum_{r=1}^l d_{i,j,r}\} \quad (8)$$

{Model M_1 }

The objective (1) here is to minimise the total number of trains and the total number of stoppages. Minimising the number of trains is the first priority objective followed by minimising the number of stoppages of all trains. Hence, variables y_k are assigned higher penalty ($\beta_1 \ll \alpha_1$) than $u_{i,k}$ variables. Constraints (2) denote the leg capacity constraints which implies that for each leg of a class of a train, the total number of passengers travelling in any class in any train between any two pair of stations should not exceed the capacity of that class. Demand constraints (3) necessitate that the total number of seat allocations in a particular class over all trains between a given pair of stations should be greater than or equal to the passenger demand between this pair of stations for that class. Outgoing linking constraints (4) link the stoppage variables with the allocation variables such that if any train is not halting at a station, all the allocations from that station for all the classes of that train will be zero. However, if the train stops at a station then the sum of the total allocation from that station must be less than or equal to the minimum of the sum of outgoing demand or capacity of the train. Similarly, incoming linking constraint (5) restricts incoming allocation at station i . Constraints (6) are train constraints which ensure that any train would have stoppage at any station, only if it is running.

COMPUTATIONAL STUDY

The mixed integer linear programming model for the stoppage and train minimization problem formulated in the previous section has been implemented using SAS® in Enterprise Guide (EG). Proc OPTMODEL optimization solver was used to solve the linear and integer programs. This section covers the experimental experience of the proposed

model with the real-life data. The railway segment covered here is the Howrah-Delhi route. Analysis of the computational results for the model is described. The results obtained were processed to identify the allocation of seats in each class between a pair of stations, the number of running trains and their stoppages.

DATASET DESCRIPTION

Passenger demand data is the basis for the whole scheduling process. The model formulated in the previous section is tested on the real-life demand dataset for trains on the Howrah-Delhi route. This route is one of the busiest routes of the Indian rail network hence it was selected for our study. Any decrease in the number of stoppages or trains will be boon for Indian Railways. However, the model can be generalised for any other route as well. The data was obtained from CRIS (Centre for Railway Information Systems), which contained information on the total number of bookings, cancellations, actual passengers travelled, number of available berths between every pair of stations in each class of each train in aggregated form for 11 months i.e. April 2017 to February 2018 for 32 UP and DOWN trains enroute.

DATA PRE-PROCESSING

For the study, 16 UP and their counterpart 16 DOWN trains were selected for the study from the data obtained. Different trains had different stoppages ranging from 7 intermediate stations for some Rajdhani/Durontos to even 112 stoppages for trains like Udhayan Abha Toofan Express (13007). All the trains should have the same candidate stoppages for analysis and this number has to be small to reduce the problem complexity. Hence, there was an immediate need of converting the raw data to useful format. Following steps were applied to prepare the input data to the optimization model.

DATA PREPARATION

Merging of multiple routes to one: Howrah-Delhi route is a combination of main line and other chord lines. So, for simplified study in this research, routes are artificially merged to one to reduce the intermediate entry and exit of trains from the route.

Aggregated demand data generation: The demand between every pair of stations is estimated using the station-to-station booking data for all the classes. This includes passengers who have actually travelled as well as wait-listed passengers who could not get confirmed tickets till the end. Further, the demand between two minor stations is aggregated with the demand between the major stations. This is done using the following steps:

Step 1. Retain the demand from major to major stations.

Step 2. Aggregate the demand from a major station i to a minor station j with the demand from i to first major station after j .

Step 3. Aggregate the demand from a minor station i to a major station j with the demand from i to first major station before i to station j .

Step 4. Aggregate the demand between minor stations i and j with the demand from the first major station before i to the first major station after j . This is done to reduce the number of stations and consequently the problem complexity. This step brings out 10 major stations on the route for all the trains.

Capacity calculation for each train: To get the train details i.e. the number of seats available in each class and the number of coaches of all the classes present in the trains, Indian railways website is referred and accordingly the capacities are calculated.

RESULTS WITH INDIAN RAILWAYS DATA

The proposed mixed integer programming model has decision variables and constraints of the following order as listed in Column 1 and for the dataset obtained from Indian Railways, the size of the model is given in Column 2 of **Table 1: Theoretical and actual count of variables and constraints** Table 1.

The model is solved for the real dataset having 16 trains, 10 stations, 6 classes and the given passenger demand.

Table 1: Theoretical and actual count of variables and constraints

Number of	Theoretical	Actual
Continuous Variables	$\frac{n}{2}c.l.m$	4320
Binary Variables	$n.m$	160
Leg Capacity Constraints	$(n - 1).l.m$	864
Demand Constraints	$\frac{n}{2}c.l$	270
Incoming Linking Constraints	$\frac{n}{2}c.l.m$	4320
Outgoing Linking Constraints	$\frac{n}{2}c.l.m$	4320
Additional Linking Constraints	$2.n.m$	320
Train Constraints	-	-
Coach Restriction Constraints	-	-
Total Constraints	-	10094

where n , m and l are the number of stations, trains and classes respectively.

SOLVING OPTIMIZATION USING PROC OPTMODEL

SAS programming was used for creating data for optimization model. The OPTMODEL procedure was used to solve the optimization problem. Proc OPTMODEL code is below:

```
/*Optimization model*/
```

```

/*%Macro OR_process();*/
proc optmodel;

/*Declare dimensions*/
set <num, num, num, num>x_dim;
set <num>y_dim;
set <num, num>u_dim;
set <num, num>v_dim;
set <num, num>capacity_dim;
set <num, num, num>demand_dim;
set <num>source_st_dim;
set <num>destin_st_dim;

/* Decalare parameters*/
num capacity{capacity_dim};
num demand{demand_dim};
num train_capacity{y_dim};
num out_demand{source_st_dim};
num in_demand{destin_st_dim};

/*Declare decision variables*/
var y{y_dim} binary;
var u{u_dim} binary;
var v{v_dim} binary;
var x{x_dim};

/*Read the data into the dimension variable*/
read data sasgf.demand into demand_dim= [sourceid destinationid classid]
demand=demand ;
read data sasgf.capacity into capacity_dim= [trainid classid]
capacity=capacity;
read data sasgf.addedcaps into y_dim= [trainid]
train_capacity=train_capacity;
read data sasgf.sourceid into u_dim= [ sourceid trainid];
read data sasgf.Destinationid into v_dim= [ Destinationid trainid];
read data sasgf.berths into x_dim= [Sourceid Destinationid Classid Trainid
];
read data sasgf.addedoutdem into source_st_dim= [sourceid]
out_demand=out_demand;
read data sasgf.addedindem into destin_st_dim= [destinationid]
in_demand=in_demand;

/*Optimization Objective Function*/
Min Total_transit= sum{<trainid> in y_dim}
(100*y[trainid]) + sum{<sourceid, trainid > in u_dim}(10*u[sourceid,
trainid ])
+ sum{<destinationid, trainid > in v_dim}(10*v[destinationid, trainid ]);

/*Constraints*/
/*Note: For sum{< put brackets () for those over which sum is NOT to be
taken>}
For better understanding, here we are doing as following:
For combination of each cluster_id and analysis_id, total skuids should be
less than 3
ie we are summing binary variable skuid, so summing variable should be outside
of () brackets
and 'for each variables' should be in brackets
*/

```

```

/*Constraint 1*/
con capacity_constraint{<trainid, classid> in capacity_dim}:
sum{<sourceid, destinationid, (classid), (trainid) > in x_dim}
  x[sourceid, destinationid, classid, trainid]<=capacity[trainid, classid];

/*Constraint 2*/
con demand_constraint{<sourceid, destinationid, classid> in demand_dim}:
sum{<(sourceid), (destinationid), (classid), trainid > in x_dim}
  x[sourceid, destinationid, classid, trainid]>=demand[sourceid,
destinationid, classid];

/*Constraint 3.1a*/
/*con linking_t1a_cons{<Sourceid, trainid > in u_dim}:*/
/*sum{<(Sourceid), Destinationid, classid, (trainid) > in x_dim}*/
/*
      x[Sourceid, Destinationid, classid,
trainid]<=train_capacity[trainid]*u[sourceid, trainid];*/

/*Constraint 3.1b*/
/*con linking_t1b_cons{<Sourceid, trainid > in u_dim}:*/
/*sum{<(Sourceid), Destinationid, classid, (trainid) > in x_dim}*/
/*
      x[sourceid, destinationid, classid,
trainid]<=out_demand[sourceid]*u[sourceid, trainid];*/

/*Constraint 4.1a*/
/*con linking_t2a_cons{<destinationid, trainid > in v_dim}:*/
/*sum{<sourceid, (destinationid), classid, (trainid) > in x_dim}*/
/*x[sourceid, destinationid, classid,
trainid]<=train_capacity[trainid]*v[Destinationid, trainid];*/

/*Constraint 4.1b*/
/*con linking_t2b_cons{<destinationid,trainid > in v_dim}:*/
/*sum{<sourceid, (destinationid), classid, (trainid) > in x_dim}*/
/*x[sourceid, destinationid, classid,
trainid]<=in_demand[destinationid]*v[Destinationid, trainid];*/

/* Constraint 5.1*/
con train_constraint1{<sourceid, trainid> in u_dim}:
u[sourceid,trainid]<=y[trainid];

/* Constraint 5.2*/
con train_constraint2{<destinationid, trainid > in v_dim}:
v[destinationid, trainid]<=y[trainid];

/*Constraint 6*/
con berth_constraint{<sourceid, destinationid, classid, trainid > in
x_dim}:
x[sourceid, destinationid, classid, trainid]>=0;

solve;
/*solve with milp / relobjgap=1e-1; */

/*Output Solutions*/

create data sasgf.solution_x from [sourceid destinationid classid trainid]
={<sourceid, destinationid, classid, trainid> in x_dim:
x[sourceid, destinationid, classid, trainid].sol}
x=x;
quit;
/*%mend OR_process;*/

```

Following are the brief snapshots of the generated log file, output data and results obtained:

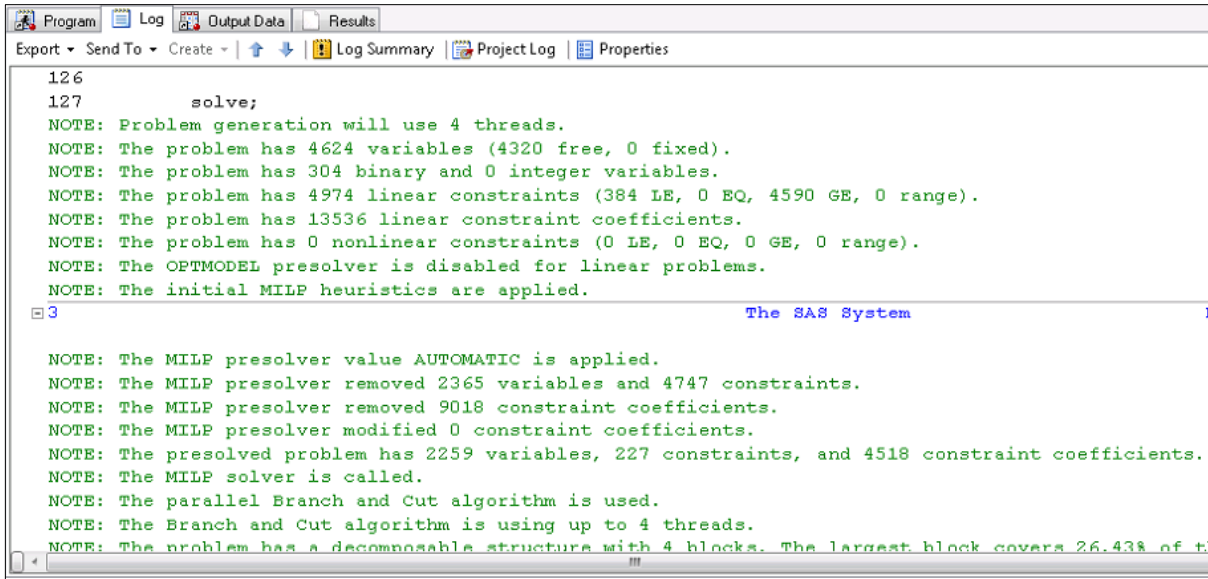


Figure 1: Snapshot of Log Generated in SAS® Enterprise Guide®

The screenshot shows the Results window for the OPTMODEL Procedure. The table below summarizes the problem details:

The OPTMODEL Procedure	
Problem Summary	
Objective Sense	Minimization
Objective Function	Total_transit
Objective Type	Linear
Number of Variables	4624
Bounded Above	0
Bounded Below	0
Bounded Below and Above	304
Free	4320
Fixed	0
Binary	304
Integer	0
Number of Constraints	4974
Linear LE (<=)	384
Linear EQ (=)	0
Linear GE (>=)	4590
Linear Range	0

Figure 2: Snapshot of Results Summary in SAS® Enterprise Guide®

	sourceid	destinationid	classid	trainid	x
1	1	2	3	1	33280
2	1	2	5	1	4394
3	1	2	6	1	6108
4	1	3	6	1	5535
5	1	4	5	1	3342
6	1	4	6	1	2442
7	1	5	6	1	1022
8	1	6	5	1	2421
9	1	6	6	1	1382
10	1	7	5	1	4138
11	1	7	6	1	991
12	1	8	6	1	204
13	1	9	6	1	213
14	1	10	5	1	5921
15	1	10	6	1	5504
16	2	3	5	1	10
17	2	3	6	1	1616
18	2	4	5	1	832
19	2	4	6	1	403
20	2	5	6	1	104
21	2	6	5	1	1165
22	2	6	6	1	479
23	2	7	5	1	1582
24	2	7	6	1	389
25	2	8	6	1	114
26	2	9	6	1	125
27	2	10	5	1	11749
28	2	10	6	1	1943
29	3	4	6	1	122
30	3	5	6	1	46

Figure 3: Snapshot of Produced Output in SAS® Enterprise Guide®

DASHBOARD ON SAS® VISUAL ANALYTICS

SAS® Visual Analytics® (VA) is used to display and visualize the results obtained by the optimization. Reports have the capability to compare the multiple scenarios and much better visualization as compared to raw results.

CONCLUSIONS

In this paper a mixed integer programming model for passenger trains scheduling and optimization of rolling stock is formulated and solved using the capabilities of SAS®. The basic objective framework of the model is to minimize the total number of stoppages of all trains for a given demand and trains. The model is further improved to minimise both the number of trains as well as the total stoppages of all the trains while satisfying the given passenger demand between all the pair of stations.

To study the computational experience, the model has been implemented using SAS® optimization solver to solve the linear and integer programs. The model is tested in real life data of Howrah-Delhi route obtained from CRIS, Indian Railways. This route has 10 major stations and currently 16 trains having six classes run on this route to meet the passenger demand.

The results of the model suggest 46 stoppages against the 160 current stoppages (reduction of 71.25%) while reducing the number of trains to 13 (18.75% reduction) as compared to the original. The model recommends several end-to-end non-stop trains and therefore minimizes the total travel time of all the trains. As per the current practice of Indian Railways, each stoppage on an average consumes about 12 minutes for retardation and acceleration plus the stoppage time.

The model shows better results when compared with the current practice followed by Indian Railways. The utility of the results of this research is two folds. It aids in railway trac planning/scheduling and improves customer satisfaction as well by reduced travel

time. The results will probably also help in reducing the track congestion since the number of trains are minimized.

Solving optimization problems with the capabilities of SAS®, especially with the OPTMODEL procedure which includes the powerful OPTMODEL modelling language and state-of-the-art solvers for several classes of mathematical programming problems, the process becomes quite easy to operate and generate results. The challenges of implementing complicated mathematical formulations can easily be overcome by simplistic procedural programming in OPTMODEL.

REFERENCES

SAS Institute Inc. 2011. SAS/OR 9.3 Users Guide: Mathematical Programming. Cary, NC: SAS Institute Inc.

Raja Gopalakrishnan and Narayan Rangaraj. Capacity management on long-distance passenger trains of Indian railways. Interfaces, 40(4):291302, 2010.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Ayush Tiwari, Core Compete
E-mail: ayush.tiwari@corecompete.com
www.corecompete.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.