



# Timetable Optimization For Single Line Railway

Malik Muneeb Abid <sup>1\*</sup>, Muhammad Babar Khan<sup>2</sup>, Iqbal Muhammad<sup>3</sup>

<sup>1</sup> School of Transportation and Logistics, South West JiaoTong University, Chengdu, China, 610031

<sup>2</sup> National Institute of Transportation, School of Civil & Environmental Engineering (SCEE), National University of Sciences & Technology (NUST), Islamabad-44000, Pakistan.

<sup>3</sup> School of Information Sciences and Technology, South West JiaoTong University, Chengdu, China.

---

## Abstract

This paper considers the train scheduling problem of single line railway. Formulation of this problem is based on job shop scheduling structure with objective to minimize total travel time. Real life constraints of this track segment are applied to this problem. Branch and bound technique with priority rules is used to solve the formulated problem in reasonable time. An exact lower bound rule is used to estimate the least train delay for resolving the remaining crossing conflicts in a partial schedule and Cutset/ dominance rule is used to reduce search space by eliminating less promising nodes. A number of examples are presented to illustrate the proposed model and solution algorithm. A real world implementation is presented by using the data for single line track segment from Rawalpindi to Lalamusa, Pakistan.

*Keywords:* Train Timetable, Single line Railways, Optimization, Branch and bound

---

## 1. INTRODUCTION

Train scheduling problem is interplay between different resources and shared rail network, which makes it a complex optimization problem involving millions of decision variables and constraints. A good solution approach to solve this problem must consider all resources integrated. Train scheduling based on manual calculations is time consuming and based on thumb rules. In order to solve this problem, we applied latest developed computer based techniques here and found profit in the form of time savings and improved schedules.

This study aims to provide an aiding tool which helps traffic managers/dispatchers in traffic management. This tool must be sophisticated and capable to estimate the effects of each and every dispatching measure taken by the controller. So, it can help rail traffic control sections to decide well in a short decision time by viewing the impact of their decision on the whole network. This may enable traffic controllers to modify of the actual timetable in order to adjust the sudden traffic disturbances.

Formulation of this problem is based on job shop scheduling structure with objective to minimize total travel time. Real life constraints of the track segments are applied to this problem. To be more specific, it is optimization problem with a set of trains running over a rail network composed of set of single line segments. It is assumed that each train has pre specified departure time and travelling route. Moreover, it is also assumed that free running times at segments are constant. Travelling of trains is assumed as tasks to assign to machines (tracks and stations). To ensure safety minimum required headway is maintained on arrival of trains at same station and on departure to same track. Formulated problem is too complex to solve this problem with Branch and Bound standard combinatorial search. Cutset/ dominance rule is used to reduce search space by eliminating less promising nodes.

---

\* Corresponding author: Malik Muneeb Abid ([malik.muneeb.abid@hotmail.com](mailto:malik.muneeb.abid@hotmail.com))

This is an extended version of the paper presented at the ICTE 2013, Chengdu, China.

The organization of this paper is as follows. In section 2 literature review about the topic is presented. Section 3 introduces the methodology adopted in this paper for modeling and solution of train timetable optimization problem. Section 4 represents the example and real world case scenario for the application of proposed algorithm and finally section 5 concludes this paper and recommend about future directions.

## **2. LITERATURE REVIEW**

From the very beginning the train timetabling has been an active area of research. D'Ariano (2008) classifies the modeling approaches in the field of railway as off-line timetabling is the process to construct schedule of operations before sometime of actual execution, while in real-time management is modifying the existing one according to new real time scenario. Figure 1 shows the classifications of the railway traffic management models used to efficiently design timetables. Tomii et al. (2005) proposed the algorithm based on PERT (Program Evaluation and Review Technique) to reduce the dissatisfaction of passengers. Kanai et al. (2011) counted the number of passengers by combing the PERT formulation and passenger flow simulation. Kliewer and Suhl (2011) minimized the weighted sum of traveling and passenger waiting times by simulating the several dispatching strategies. Dollevoet et al. (2013) proposed three heuristics to minimize the travel time of passengers. Bi objective problem considered by Corman et al. (2012) minimizes the train delays and missed passenger connections.

There are a lot of studies which have been devoted to timetable rescheduling; we refer readers to see Tornquist (2006), Jespersen-Groth et al. (2009) and Cacchiani et al. (2014). D'Ariano et al. (2007, 2008a, and 2009) introduced a real-time traffic management system ROMA (Railway traffic Optimization by Means of Alternative graphs) and Corman et al. (2010a) extended this system by applying met heuristics algorithms. D'Ariano et al. (2008b) introduced the concept of flexible departure and arrival times for absorb minor delays and Corman et al. (2011) used priorities for trains for rescheduling. Corman and D'Ariano (2012) evaluated and rescheduled the several disturbance scenarios by implementing cancelling and rerouting strategies in ROMA. Törnquist (2007) and Törnquist Krasemann (2012) adopted heuristics to prevent the delay propagation among the trains.

### **2.1 Branch and Bound Technique**

Last three decades have a wide range of studies devoted to train scheduling using branch and bound technique. Table 1 shows the review of different modeling techniques for train timetable optimization problem with solution algorithms. Szpigel (1973) first modeled the train scheduling problem as mixed integer program and applied the branch and bound technique proposed by Greenberg's (1968) to minimize the total transit times subjected to crossing and overtaking constraints. Jovanovic and Harker (1991) used branch and bound for non linear integer model to generate feasible meet-pass plans for trains. Higgins et al. (1996) proposed nonlinear model to minimize total train delays and presented branch and bound solution strategy with look-ahead rule for lower bound generation. Look-ahead rule estimated the least delays for remaining conflicts in partial schedule. However, the authors remarked that the lower bound rule overestimates the actual remaining delay. Kroon and Peeters (2003) considered variable trip times to develop the periodic event scheduling model for cyclic train timetable. Zhou and Zhong (2005) considered acceleration and deceleration time losses to formulate multi-mode resource-constrained project scheduling model. Zhou and Zhong (2007) used branch and bound technique to solve the total travel time minimization problem and used i) Lagrangian lower bound estimate, ii) exact lower bound based on remaining conflicts in the partial schedule and, iii) Upper bound using beam search heuristic. D'Ariano et al. (2007) modeled real time traffic control as huge job shop scheduling problem with no store constraints. Shafia et al. (2012) formulated to find the robust time table on single line track. They used branch and bound with beam search heuristic to solve the problem. Comparison of results of branch and bound technique with Lingo shows optimality of solution. Abid and Khan (2013) proposed branch and bound technique to minimize the total travel time of trains for single line railways and Abid and Khan (2015) extended this model to find the optimum position of sidings on track. However, the complexity of problem demands more studies and techniques (heuristics) to solve this problem within reasonable time.

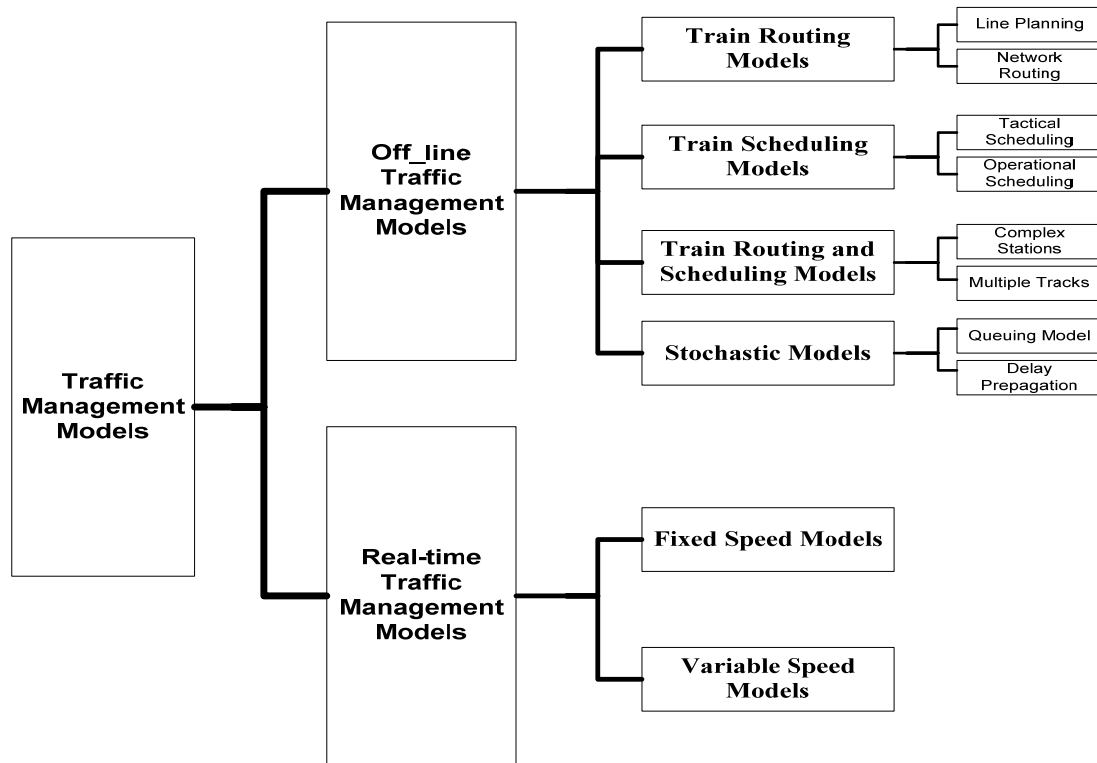


Figure 1: Classifications of Rail Traffic Management Models.

Table 1: Synthesis of Past Research

Publication	Modeling Technique	Objectives	Solution Algorith
Tornquist and Persson(2007)	Mixed integer programming	To minimize total delay and cost.	H
D'Ariano et al. (2007)	Mixed integer programming	To minimize secondary delays.	B & B, H
D'Ariano et al. (2008)	Mixed integer programming	To minimize the maximum and average consecutive delays in lexicographic order.	B & B, LS and PR.
D'Ariano and Pranzo(2009)	Mixed integer programming, Computer simulation model	To minimize secondary delays for all trains over multiple timetable hours.	B & B, H and PR.
Corman et al. (2009)	Mixed integer programming	To minimize average delay and average cost of train.	B & B, H
Corman et al. (2010a)	Mixed integer programming	To minimize the maximum and average consecutive delays in lexicographic order.	B & B, H
Corman et al. (2010b)	Computer simulation model	To minimize total passengers' delay.	AG.
Corman et al. (2011)	Mixed integer programming	To minimize train delay combined with the objectives of train companies.	B & B, H

Schachtebeck and Schöbel (2010)	Mixed integer programming	To minimize the sum of all delays of all passengers at their final destinations.	H, D
Meng and Zhou (2011)	Mixed integer programming, Stochastic programming	To minimize the expected additional delay under different forecasted operational conditions.	B & B, H
Shafia et al. (2012)	Mixed integer programming	To maximize delay absorption	B & B
Sato et al. (2013)	Mixed integer programming	To minimize passenger inconvenience	B & C
Andersson et al. (2015)	Mixed integer programming	To increase the robustness	H

Where: Branch-and-bound (B&B). Branch and cut (B&C) Alternative graphs (AG). Heuristics (H). Dynamic programming (D). Local search (LS). Practical rules (PR).

### 3. METHODOLOGY

This work considers single line train scheduling problem, in which trains are considered as tasks which are assigned to tracks (considered as machines). In this way jobshop scheduling problem is formulated with set of single line track segments and a set of trains having predefined traveling directions and fixed running times. B&B technique with cut set dominance rule and lower bound elimination rules are used to reduce the search space. Figure 2 and Table 2 show the variables used in modeling.

#### 3.1 Assumptions and Variables

In order to model the train scheduling problem we have assumptions regarding the railway network. First, it is assumed that track is composed of segments which are separated by sidings. Sidings are places where trains can cross, overtake and change their directions. Second, all trains have pre specified route, direction and constant free running times for tracks. Third, jobshop scheduling is considered here with trains as tasks to be assigned to tracks considering as machines. Fourth, for safety minimum headways are maintained between the trains. Fifth, stations have capacity for more than one trains, however, it is assumed that only one train capacity for each station. Sixth, a train can wait for other train at station for maximum 30 minutes which is the maximum limit of wait.

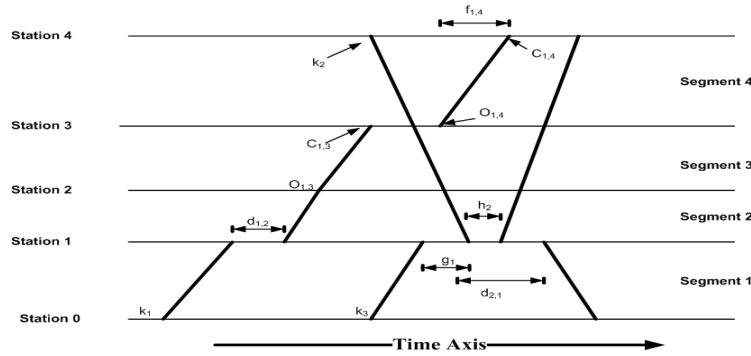


Figure 2: Illustration of variables

Table 2: Table of Notations

	Definition	Symbol
Train index		t
Segment index		s
Segment sequence number in train route		j
Station index		i
Set of trains		T

Set of segments $ S =m$	S
Set of stations $ J =m+1$	J
Direction indicator for train t, $p(t) = 0$ for an inbound train and $p(t) = 1$ for an outbound train	$p(t)$
Segment index of the $j^{\text{th}}$ traveling segment in a route for train i, $\sigma(t,j) = j$ for outbound trains, $\sigma(t,j) = m+1-j$ for inbound trains	$\sigma(t, j)$
Downstream station number of the $j^{\text{th}}$ traveling segment in a given route for train t, $b(t, j) = j$ for outbound trains, $b(t, j) = m - j$ for inbound trains	$b(t, j)$
Planned departure time for train t at its first station	$k_t$
Free running time for train t at segment s	$f_{(t,s)}$
Minimum required station dwell time before train t entering segment s	$d_{(t,s)}$
Maximum allowed station dwell time before train i entering segment j	$\bar{d}_{(t,s)}$
Minimum headway between arrival and departure times of two consecutive trains at segment j	$h_s$
Minimum headway between arrival times of two consecutive trains at station u	$g_i$
Entering time for train t at segment s, i.e., start time for job t on machine s	$o_{t,s}$
Leaving time for train t at segment s, i.e., end time for job t on machine s	$c_{t,s}$
Binary Variable, 1 if train t is scheduled before train t' on segment s, 0 otherwise	$A_{t',s}$
Sufficiently large constant	M

### 3.2 Objective function

$$\text{Min}Z = \sum_{t=1}^n c_{t,\sigma(t,m)} \quad (1)$$

Objective function of model is to minimize the total of completion time of each activity which will give result in the minimization of total travel time.

### 3.3 Constraints

#### 3.3.1 Departure Time Constraint

$$o_{t,\sigma(t,1)} \geq k_t \quad \forall t \in T \quad (2)$$

This constraint ensures that planned departure time is not more than the actual departure time.

#### 3.3.2 Free Running Time Constraint

$$c_{t,\sigma(t,j)} = f_{t,\sigma(t,j)} + o_{t,\sigma(t,j)} \quad \forall t \in T, j = 1, 2, \dots, m \quad (3)$$

This equation states that leaving time of a section must be equal to the entering time plus free running time.

#### 3.3.3 Minimum Dwell Time Constraint

$$o_{t,\sigma(t,j)} \geq c_{t,\sigma(t,j-1)} + \underline{d}_{t,\sigma(t,j)} \quad \forall t \in T, j = 2, \dots, m \quad (4)$$

This constraint is ensuring that scheduled stop is more than minimum dwell time which is practically required to load and unload passengers and freight trains.

#### 3.3.4 Headway Constraint at Track Segment

$$o_{t,s} \geq c_{t',s} + h_s \quad \text{Or} \quad o_{t',s} \geq c_{t,s} + h_s \quad \forall t', t \in T, t \neq t', s \in S \quad (5)$$

Minimum headway is required for safe operation of trains running in opposite or same direction on the same track. This constraint ensures safe operations at the rail corridor.

#### 3.3.5 Headway Constraint on Arrival Time at Stations

$$c_{t,\sigma(t,j)} \geq c_{t',\sigma(t',j)} + g_i \quad \text{Or} \quad c_{t',\sigma(t',j)} \geq c_{t,\sigma(t,j)} + g_i, \quad \forall t', t \in T, t \neq t', b(t, j) = b(t', j) = s \quad (6)$$

This constraint imposes that minimum headway on two consecutive trains approaching at same station.

### 3.3.6 Maximum Dwell Time Constraint

$$o_{t,\sigma(t,j)} \leq c_{t,\sigma(t,j-k)} + \bar{d}_{t,\sigma(t,j)} \quad \forall t \in T, j = 1, 2, \dots, m \quad (7)$$

This constraint gives the upper bound of time which a train can wait for other train.

### 3.3.7 Either-or Relationship

$$\begin{aligned} o_{t,s} &\geq c_{t',s} + h_s - M \times A_{t,t',s} \\ o_{t',s} &\geq c_{t,s} + h_s - M \times (1 - A_{t,t',s}) \end{aligned} \quad (8)$$

Either – or relations are decision variables which will decide which train will traverse segment first.

## 3.4 Solution Methodology

B&B solution procedure with depth and breadth first search technique is described in this section to resolve conflicts. Each node in the B&B tree represents a partial solution (i.e. partially resolved schedule) and the depth (in terms of number of levels) in the tree determines the number of conflicts resolved in this partial solution. For example, a node at the ninth level of the tree will be a partially resolved schedule where the first nine conflicts are resolved. Each node will have two branches as either of the two trains in the conflict can be delayed. A train is delayed at the nearest feasible siding.

### 3.4.1 Search Techniques

Search techniques are used to find the specified item with in a set of items. These techniques are useful to find the solution early. There are so many algorithms applied for searching some of those used in this thesis for B&B searching are as follows:

- *Depth First Search:* It is a search technique used for traverse the search tree this technique start search from root and terminates at leaf and then go next unvisited node. Rule is if a node is not pruned than next search node will be its children node.
- *Breadth First Search:* It is a search technique used for traverse the search tree this technique start search from root and visit all nodes at same level first than go to next level nodes. Rule of this search is exploring all nodes of given level before proceeding to next level.

### 3.4.2 Priority Rules

It is practically impossible to attain optimal solution of large scale network of NP-hard train scheduling problem (Cai et al (1998), Caprara et al (2002)). To find the feasible solutions of these problems within a reasonable time limit heuristic techniques are generally applied. Priority rules are also incorporated which decides, according to predefined objectives, the next train to be scheduled from a pair of conflicting trains.

These priority rules order the running trains by using the decision criteria adopted by traffic controllers at each junction. We are using most commonly used rules which are elaborated in more detail in the example next section.

- Random Priority Rule (RPR)
- Best Cost Priority Rule (BCPR)
- Early Start Priority Rule (ESPR)
- Early Finish Priority Rule (EFPR)
- Minimum Processing Time Priority Rule (MPTPR).

## 3.5 Lower bound Calculation

In this work simple lower bound rule proposed by Higgins (1995b) and later on modified and used by Zhou and Zhong (2007) is used. This rule simply estimates the additional delay required for resolving the remaining crossing conflicts in a partial schedule, it ignores the existing overtaking and new generated conflicts in schedule. Figure 3; differentiate between crossing and overtaking conflicts. When two trains are in opposite direction than it will be crossing conflict while in the case of overtaking they are running in the same direction.

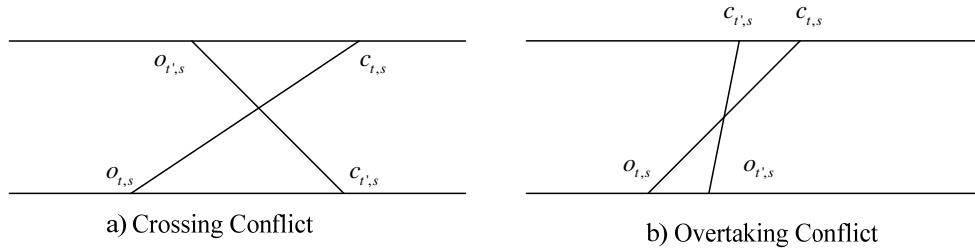


Figure 3: Illustration of Crossing and Overtaking Conflicts.

First minimum additional delay to resolve a single crossing conflict is estimated and then it can be used to calculate the total additional delay for all conflicts in a partial schedule.

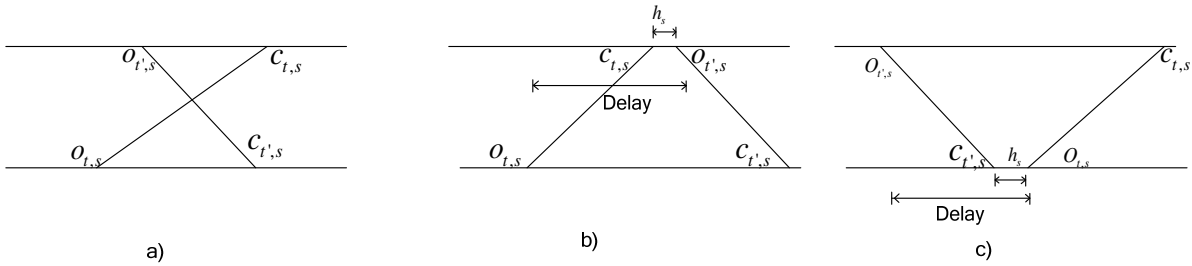


Figure 4: Calculation of Additional Delay for Resolving a Conflict.

Figure 4, shows a conflict between two trains at segment  $s$ . If train  $t'$  is delayed first and train  $t$  is allowed to traverse track first than resulting delay to  $t'$  will be  $h_s + c_{t,s} - o_{t',s}$ . Otherwise, train  $t$  will be delayed for  $h_s + c_{t',s} - o_{t,s}$ . From the above explanation it can be concluded that minimum additional delay for resolving a single conflict at segment  $s$  will be

$$\min(h_s + c_{t,s} - o_{t',s}, h_s + c_{t',s} - o_{t,s}) = h_s + \min(c_{t,s} - o_{t',s}, c_{t',s} - o_{t,s}) \geq h_s \quad (9)$$

When resolving a conflict at intermediate siding minimum headways of both upstream and downstream segments should maintained. Figure 5, conflict resolution of two trains running in opposite directions at intermediate siding. From both cases it can be concluded that minimum additional delay for conflict resolution at intermediate siding will be  $g + h$ .

It is clear from Figure 5 that additional delay has two components. First, headway between the arrival times of two trains at a station  $J$ . Second, headway between the departure times of trains.  $h$  time units are the accurate estimation of second part of lower bound because the only constraint applied over the departure time of second train is minimum headway. On the contrary, arrival time of trains is dependant of path trajectories of trains which may lead a train to arrive a station much earlier than other. Hence, in first part  $g$  units delay underestimates the additional delay. Alternatively, it would be suitable to change meeting station if actual delay of first part is greater than the free running time of consecutive segment.

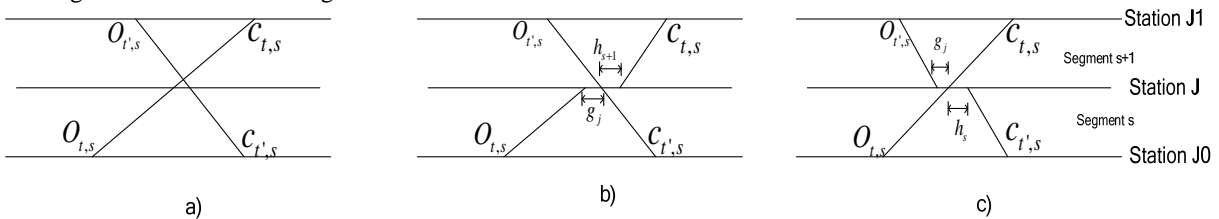


Figure 5: Calculation of Additional Delay for Resolving a Conflict at Intermediate Siding.

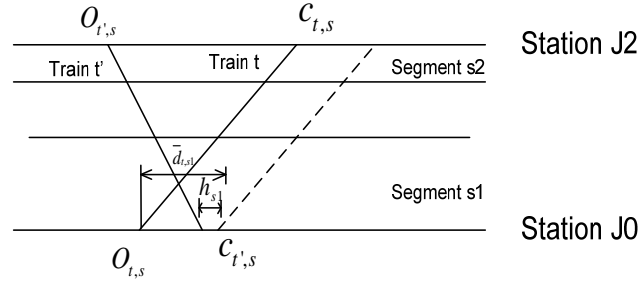


Figure 6: Estimation of Feasible Region for Conflict Resolution.

Based on the dwell time window of both trains conflicting, possible region of their conflict can be estimated. As shown in Figure 6, if  $c_{t',J1} \geq o_{t',J1} + \bar{d}_{t',J1}$  and  $c_{t,J2} \geq o_{t',J2} + \bar{d}_{t',J2}$  then  $c_{t',J1} \geq o_{t',J1}$  and  $c_{t,J2} \geq o_{t',J2}$  indicates that conflict is to be resolved at some where intermediate siding.

Summarizing, the above discussion the conflict based lower bound at any node  $n$  is

$$LB(n) = \sum c_{t,\sigma(t,s)} + \sum_{t,t'} \Theta_{t,t'} \quad (10)$$

$\Theta_{t,t'}$  can be determined from following expressions:

---

**Algorithm 1: Lower Bound Estimation**


---

if  $c_{t',J1} \geq o_{t',J1} + \bar{d}_{t',J1}$  and  $c_{t,J2} \geq o_{t',J2} + \bar{d}_{t',J2}$

$$\Theta_{t,t'} = g + h$$

Else if  $c_{t',J1} \leq o_{t',J1}$  and  $c_{t,J2} \leq o_{t',J2}$

$$\Theta_{t,t'} = h$$

End.

---

**3.6 Node Elimination Rule**

Node elimination rules are incorporated to reduce the search space by pruning those nodes which are not providing the promising results. Function value is calculated at each node and it is eliminated if it does not provide solution leading to feasible solution.

**3.6.1 Cut set /Dominance Rule**

This rule calculates the function value of each child and active nodes and compares it with current best solution. If cost of this node is equal or greater than the current best solution than it will be eliminated.

It compares two nodes only if these conditions are satisfied:

- i. Finish set of trains at both nodes are same.
- ii. Departure time of considered train is same at both nodes,  $k_i(n_1) = k_i(n_2)$ .
- iii.  $z(n_1) < z(n_2)$ ,  $z$  is objective function value at nodes.

Then  $n_2$  is dominated by  $n_1$ .

Figure 7 gives example of cut set rule. Train 3 is the last scheduled high speed train with  $k_i(n_1) = k_i(n_2)$ . Both nodes have same set of scheduled trains which include high speed trains 1, 2, 2 and medium speed train 8. Both active nodes  $n_2$  and  $n_1$  has to schedule train 9, which yields to train 3 at station 3. Furthermore, it is easy to verify that  $z(n_1) < z(n_2)$  for two schedules. This concludes that node  $n_1$  dominates  $n_2$ .

**3.6.2 Lower bound Rule**

Lower bound value of each newly generated node is calculated. If value of lower bound of a node does not fall within Optimality gap of upper bound function than it is pruned.



$$LB(n) \geq UB(\text{optimality gap}\%) \quad (11)$$

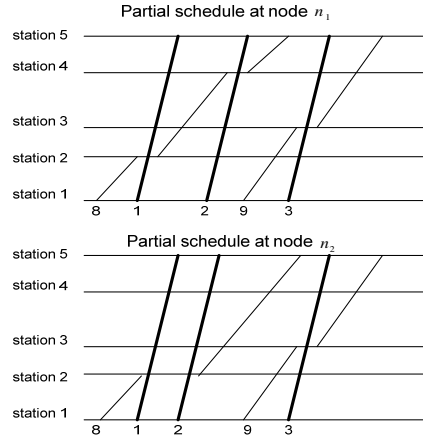


Figure 7: Example of Cut Set Rule

### 3.7 Solution Algorithm

#### Algorithm 2: Optimization of Schedule

**INPUT:** Set of trains (T), Set of segments (S), Set of stations (J), Train route and direction ( $\sigma(t, j)$ ),  $p(t)$ , Planned departure time ( $k_t$ ), Free running time for train ( $f_{(t,s)}$ ),

**OUTPUT:** Optimized Train Timetable (minimized total travel time)

1. Set  $UB = \infty$

for  $t \in T, j \in J, s \in S$

$$o_{t,\sigma(t,1)} \geq k_t$$

$$c_{t,\sigma(t,j)} = f_{t,\sigma(t,j)} + o_{t,\sigma(t,j)}$$

$$o_{t,\sigma(t,j)} \geq c_{t,\sigma(t,j-1)} + \underline{d}_{t,\sigma(t,j)}$$

$$o_{t,s} \geq c_{t',s} + h_s \text{ or } o_{t',s} \geq c_{t,s} + h_s$$

$$c_{t,\sigma(t,j)} \geq c_{t',\sigma(t',j)} + g_i \text{ or } c_{t',\sigma(t',j)} \geq c_{t,\sigma(t,j)} + g_i$$

$$o_{t,\sigma(t,j)} \leq c_{t,\sigma(t,j-k)} + \bar{d}_{t,\sigma(t,j)}$$

$$o_{t,s} \geq c_{t',s} + h_s - M \times A_{t',s} \text{ or } o_{t',s} \geq c_{t,s} + h_s - M \times (1 - A_{t',s})$$

2. Generate node

for  $t, t' \in T, j \in J, s \in S$

3. Lowerbound (LB)

$$LB(n) = \sum c_{t,\sigma(t,s)} + \sum_{t,t'} \Theta_{t,t'}$$

4. Optimization

4.1 Search method (DFS, BFS)

4.2 Priority Rule (MCPR, BCPR, etc.)

4.3 Node elimination rule

## 4 EXPERIMENTAL SETUP

### 4.1 Example

Schedule 3 trains using B&B technique, with running speed 60 km/h over a single line track with five segments and six sidings/terminals. One train is inbound and two are outbound. Track lengths are 10, 10, 10, 15 and 10 km. Departure times of each train at terminal stations is given in Table 3.

Table 3: Problem Input Data

Train Number	Departure time	Direction
0	0:05	Outbound
1	0:17	Inbound
2	0:35	Outbound

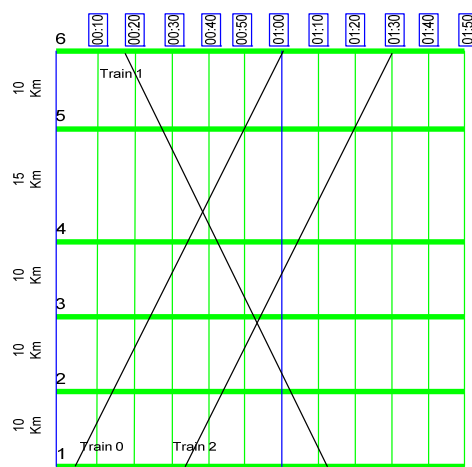


Figure 8: Problem Representation

#### 4.1.1 Depth First Search

DFS technique will take the last generated node as active node. Here, we have started with Node No.0 with an empty schedule and proceed further by scheduling tasks on the available tracks until we find a conflict among the trains. Then the available options will be new nodes of search tree. Train 0 and Train 1 are conflicting at section 3 which will give two child nodes, one node when Train 1 is allowed to traverse the track segment first and other when Train 1 is waiting for Train 0. Train 1 is trying to use the track 27 to 42 minutes and Train 0 is 35 to 50 minutes. According to time Node No.1 will be option when Train 1 traverses the track 27 to 42 minutes and other one is Node No.2.

Node No.2 will be evaluated first because it is the last one node generated in search space. At level 2 Node No.2 is generating two more nodes 3 and 4 with different available options. Now Node No.4 will be evaluated giving 5 and 6 root nodes of this search tree. First Node No.6 is evaluated giving optimum value 49 and then Node No.5 gave 35. No more child node available now here than it will move one step back at level 2 and evaluate Node No.3 which gives two root nodes 7 and 8. Node No.8 will be first one to get evaluated but it will be pruned as it is not giving good results as compared to previous results than Node No.7 evaluation gives 31. Next, algorithm will move back to unexplored nodes as no more nodes are available after this root node in this path. At level 1 Node No.1 is evaluated it is giving two options Node No.9 and Node No.10. DFS will evaluate Node No.10 first with optimum value 28 and then finally it will go to the last available option Node No.9 which will give optimum value 14, which is less than all previous values. Hence, Figure 9 (a) shows the result of this search tree is 11 nodes generated with depth 4 and optimum value 14 minutes delay.

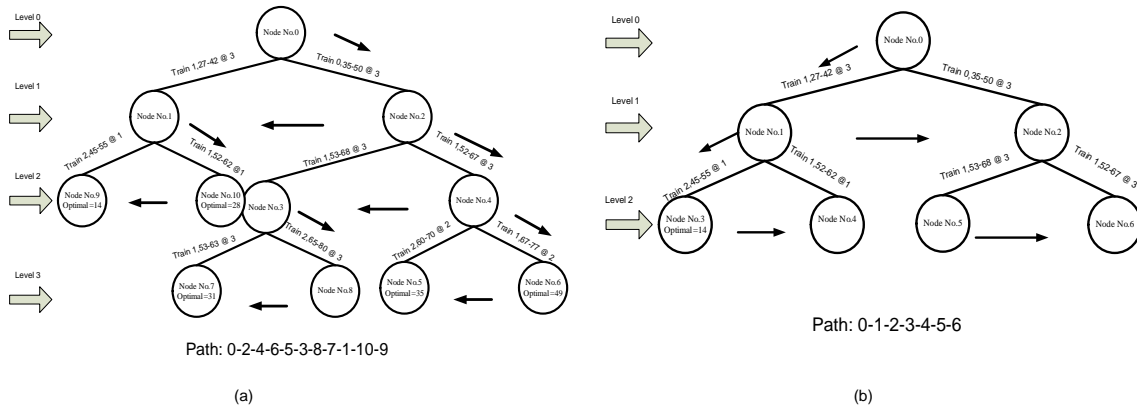


Figure 9: (a) Search Tree of Depth First Search Technique. (b) Search Tree of Breadth First Search Technique

#### 4.1.2 Breadth First Search

BFS technique traverse the search tree in a such manner that node generated first will be evaluated first and all nodes at the same level are evaluated before going to next level (Figure 9 (b)). When considering the above problem given with BFS (as compared to DFS), Node No.1 will be explored first here. Node No.1 gives birth to 2 child nodes Node No.3 and 4. Next, Node No.2 is explored with two options available Node No.5 and 6. All the nodes at level 1 are evaluated now algorithm will go to next level where Node No.3 comes first, which gives optimal value 14. Then search algorithm will evaluate Node No.4, 5 and 6 but these all available options will not update solution. It concludes that this search algorithm has evaluated 7 nodes up to depth level 3 and generated optimum solution 14 minutes delay to trains.

#### 4.1.3 Priority Rules

Random priority rule select active node from the child nodes randomly and evaluate only one node at each level and all others remain unexplored. Result of random priority rule is given in Figure 10(a). All other rules are generating same results because here free running time of each track segment is fixed and only one type of train is consider in this example so early arrival and early departure and processing times will be same. Here this example only describes the working of different techniques in the next chapter in the application of this model; two different types of trains are considered which will evaluate the effect of different priority rules. Table 4 and Figure 10(b) shows the results of DFS, BFS, early arrival, early finish, early departure and minimum processing times.

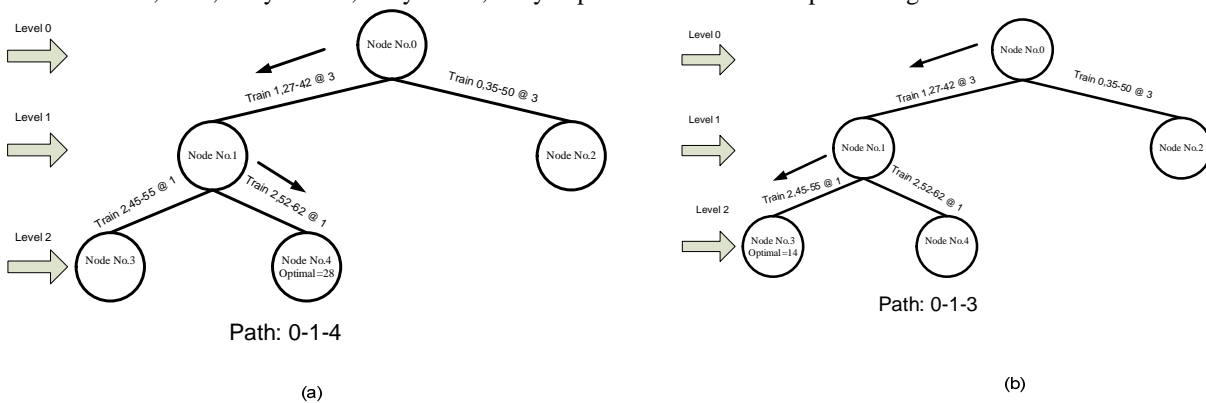


Figure 10: (a) Search Tree of Random Priority Rule. (b) Search Tree of BCPR, ESPR, EFPR, MPTPR.

### 4.2 Comparison of Approaches

The proposed model is tested on a train schedule of 5 sections track segments with 3 to 15 trains running with constant segment running times. Figure 11(a) shows the summary of results in terms of number of nodes generated by each search technique and Figure 11(b) shows the solution quality of each technique in terms of conflict delays for each problem set. Data set assumed here contains constant running times so best cost, late arrival, late departure and minimum processing time priority rules are generating same results. Priority rules has generated less than 1% nodes as compared to exact solution techniques with 43.4% and 2.71% optimality gap in random and other priority rules. It can be concluded that priority rules are generating results almost same as Branch and Bound solution techniques within much less time except random priority rule. Results in Figure 12 and 13 are generated after considering all nodes because the optimality gap used in Lower bound rule is 100%, which do not prune any node. Dominance or cutset rule performs well with breadth first technique.

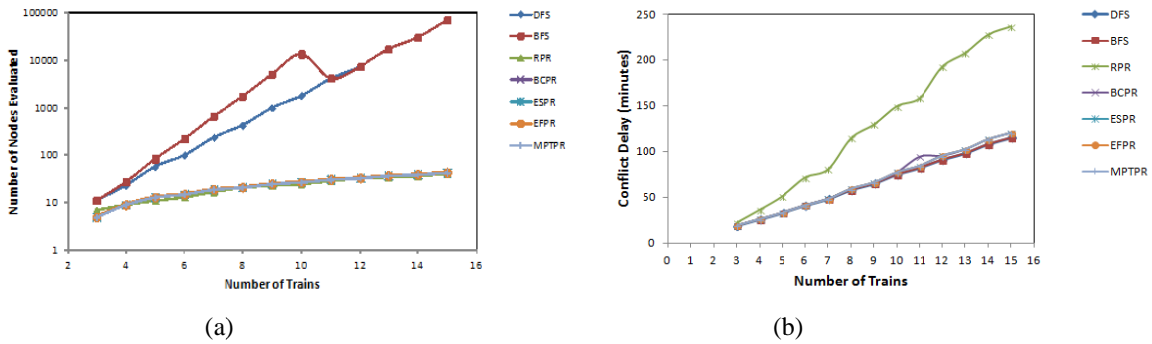


Figure 11: (a) Comparison of node evaluated by different approaches, (b) Solution quality of different approaches in terms of conflict delays.

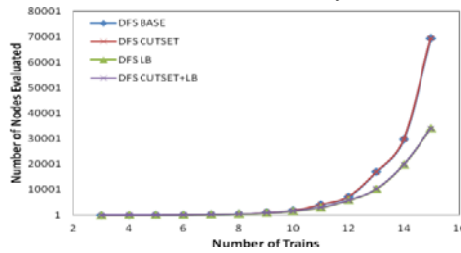


Figure 12: Number of Nodes Evaluated by Depth First Search Technique.

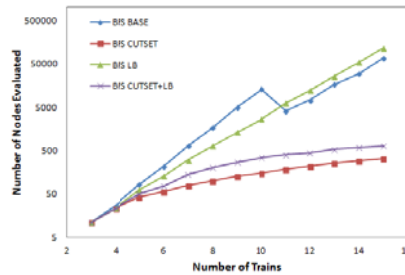


Figure 13: Number of Nodes Evaluated by Breadth First Search Technique.

### 4.3 Real World case

The track chosen to apply the model and get optimized results is from Rawalpindi to Lalamusa. Track chosen is mainly single line track with length 156 Km. On the busiest day of week about 30 trains traverse this track. There are four different types of trains scheduled over this track, namely; Mail and Express, Intercity, Passenger and Mixed trains. To ensure safety minimum 3 minute headway between the departure time of two trains to a track and 2 minutes between arrival times of two trains at a station is maintained (Rizvi, 2010). Description of each train and their input values in the model to get optimum results are given in Table 4.

Table 4: Description of Trains using Track segment

Inbound train						Outbound Train					
Train no	Train No input	Type	Origin	Destination	Start time	Train no	Train No input	Type	Origin	Destination	Start time
105	3	A-C/P,A-C/L & EC	LLM	RWP	2:13	106	1	A-C/P,A-C/L & EC	RWP	LLM	0:30
327	4	MIXED	LLM	RWP	2:25	2	2	A-C,A-C/B & EC	RWP	LLM	2:00
131	5	EC	LLM	RWP	3:20	46	6	EC	RWP	LLM	6:15
11	9	EC	LLM	RWP	8:10	110	7	A-C/P,A-C/L & EC	RWP	LLM	7:00
107	10	A-C/P,A-C/L & EC	LLM	RWP	8:38	104	8	A-C/P,A-C/L & EC	RWP	LLM	7:30
101	12	A-C/P,A-C/L & EC	LLM	RWP	9:47	8	11	A-C,A-C/B & EC	RWP	LLM	8:45
13	13	A-C/L & EC	LLM	RWP	10:40	40	14	A-C,A-C/B & EC	RWP	LLM	10:45
23	15	1ST & EC	LLM	RWP	11:25	24	16	1ST & EC	RWP	LLM	11:45
39	17	A-C,A-C/B & EC	LLM	RWP	11:57	14	19	A-C/L,EC	RWP	LLM	13:10
45	18	EC	LLM	RWP	12:28	132	20	EC	RWP	LLM	15:30
7	23	A-C,A-C/B & EC	LLM	RWP	17:22	102	21	A-C/P,A-C/L & EC	RWP	LLM	16:30
103	25	A-C/P,A-C/L & EC	LLM	RWP	18:37	12	22	EC	RWP	LLM	17:15
109	26	A-C/P,A-C/L & EC	LLM	RWP	19:40	108	24	A-C/P,A-C/L & EC	RWP	LLM	18:00
1	0	A-C,A-C/B & EC	LLM	RWP	0:00	328	27	MIXED 2ND	RWP	LLM	19:45

A-C=Air conditioned, B=Business, EC=Economy, 1<sup>ST</sup>=First Class, L=Standard, P=Parlour.

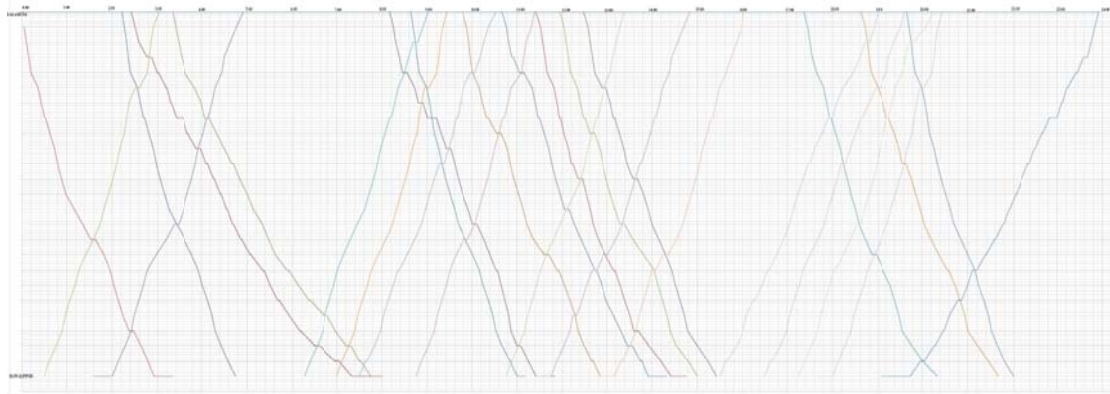


Figure 12: Actual Train Schedule of Track Segment from Rawalpindi to Lalamusa.

#### 4.2.1 Results of Real World Case

The branch and bound technique with depth first search and dispatching priority rules (i.e., based on arrival, departure and processing times) is implemented in Visual C++. The objective function was to minimize total travel time by optimizing the conflicts of trains running over track segment taken for case study. There is total running time 85 hours 52 minutes with 304 minutes conflict delay and 238 minutes scheduled stops. The schedule was optimized using B&B technique with priority rules. The optimal schedule generated by BCPR is displayed in Figure 13. This has total 72 hours and 36 minutes and 291 minutes conflict delay. All other priority rules optimize this schedule with 72 hours and 40 minutes with 295 minutes conflict delay, as shown in Figure 13. Efficiency of optimal schedules can be found by visual inspection of both schedules.

There are unnecessary delays to train 327 and 131, conflicts of these trains can be reduced considerably without causing extra delay to other trains. Figure 14 and Figure 15 are providing us with comparison of total travel time of inbound and outbound trains separately.

It can be seen from comparison of actual and optimized schedules that conflict delay to inbound Train 11 and outbound Train 104 are much more than other trains in actual schedule. In actual schedule, at the start of Train 104 path conflicts are with Train 327 and Train 131. While in optimized schedule because conflict resolution of inbound Train 327 and Train 131 with outbound Train 106 and Train 2 path are adjusted in such a way that the path trajectories

of these trains(Train 327 and Train 131) end before the Departure of Train104. In case of Train 11, in actual schedule it is conflicting with four outbound trains (Train 46, Train 110, Train 104 and Train 8). In all conflicts it is delayed and it is also overtaken by Train107(Fast train), which cause more delay to this path. In optimized schedule, in conflict with Train 46 is given priority over this and in all other crossing conflicts it is given priority over other three outbound trains. This arrangement also omitted overtaking conflict with Train 107.

Conflicts between inbound Train 327, Train 131, Train 105 and outbound Train 106 and Train 2 are adjusted such that in optimized schedule inbound trains are given priority over outbound trains, which reduces next conflicts of these trains with Train 46, Train 110 and Train 104.

In actual schedule outboud Train 40 is given priority over inbound Train 11, Train 107, Train 101, Train 13, Train 23, Train 39 and Train 45. This delay of 7 inbound trains is generating more conflicts with next outbound trains. In optimized schedule these inbound trains are given priority over Train 40, which results in omission of conflict of Train 13 and Train14, Train 45 and Train132.

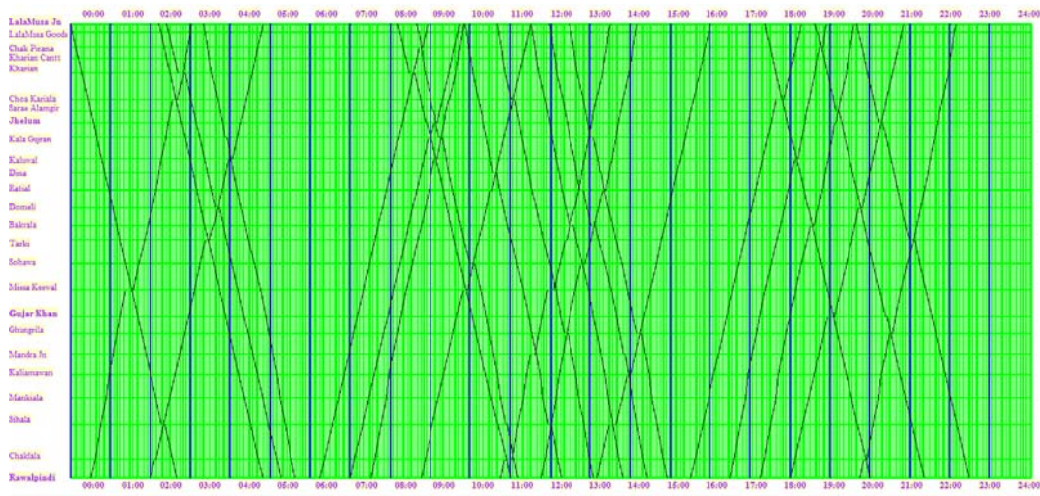


Figure 13: Optimize schedule

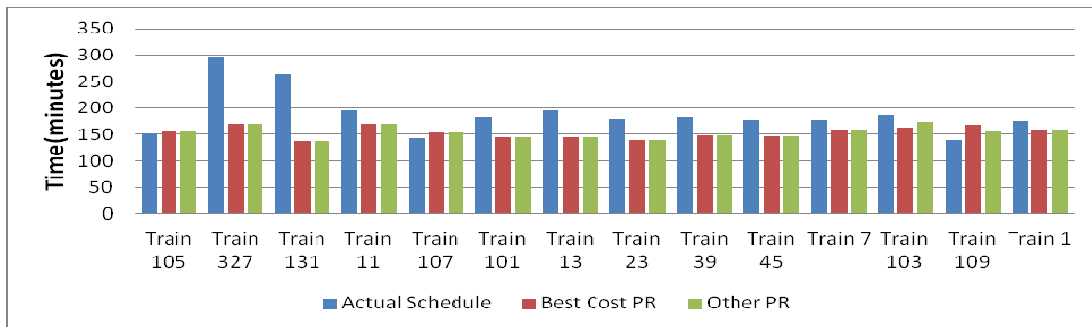


Figure 14: Comparison of Total Travel Time of Inbound Trains for Actual and Optimized Schedule.

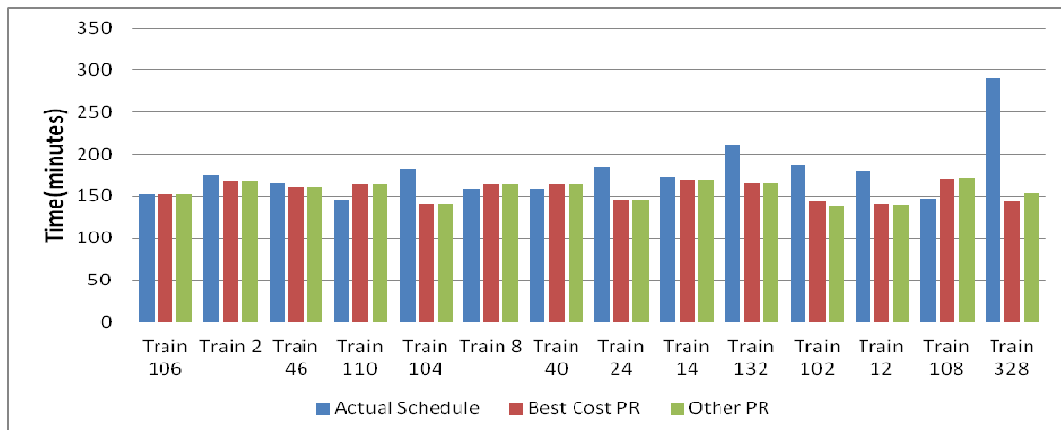


Figure 15: Comparison of Total Travel Time of Outbound Trains for Actual and Optimized Schedule.

## 5 CONCLUSIONS AND RECOMMENDATIONS

This paper presents the modeling and optimization of single line track railways timetables. A mathematical model is developed here by considering the real world constraints at the railway segments. Branch and bound algorithm is devised with lower bound and cut set dominance rules to solve the proposed model. Mathematical model developed in is applied over a real rail corridor of track segment from Rawalpindi to Lalamusa. Actual schedule of this track is optimized here using priority rules, while considering both slow and fast trains traversing this track segment daily. Results have shown that these rules generate optimized schedule with less running time as compared to actual one, which was the objective function of this modeling.

This work assumed constant free running time to make this more realistic its extension would be inclusion of variable running time into this model. By using this approach in modeling, this may lead to more complex problem formulation hence there will be necessity of an efficient heuristic technique which will solve it with in less time.

### Acknowledgements

The authors would like to thank Mr. Umar Riaz, Divisional Planning Officer and Mr. Ehtsham Khan, Divisional Signal Engineer of Rawalpindi Division for their cooperation and guidance during this research. The study has benefited from valuable assistance of Mr. Zohaib Amjad in coding the algorithms.

### REFERENCES

- Abid, M., M., Khan, M., B., 2015. Sensitivity analysis of train schedule of a railway track network using an optimization modeling technique. *Eur. Transp. Res. Rev.* (2015) 7:3.
- Abid, M., M., Khan, M., B., 2013. Optimization of railways schedule using branch and bound Technique with priority rules. *ICTE 2013, Chengdu*, 18-20 October, 3171-3181.
- Andersson, E., V., Peterson, A., Krasemann, J., T., 2015. Improved Railway Timetable Robustness for Reduced Traffic Delays – a MILP approach. *6th International Conference on Railway Operations Modelling and Analysis - RailTokyo2015*.
- Cai, X., Goh, C. J., & Mees, A. I., 1998. "Greedy heuristics for rapid scheduling of trains on a single track." *IIE Transaction*, 30, 481–493.
- Caprara, A., Fischetti, M., & Toth, P., 2002. "Modeling and Solving the Train Timetabling Problem." *Operations Research*, 50, 851–861.
- Corman, F., D'Ariano, A., Pacciarelli, D., Pranzo, M., 2009. Evaluation of green wave policy in real-time railway traffic management. *Transportation Research Part C* 17 (6), 607–616.
- Corman, F., D'Ariano, A., Pacciarelli, D., Pranzo, M., 2010a. A tabu search algorithm for rerouting trains during rail operations. *Transportation Research Part B* 44 (1), 175–192.
- Corman, F., D'Ariano, A., Hansen, I.A., 2010b. Disruption handling in large railway networks. In: *Proceeding of Computers in Railways, Beijing, China*. pp. 629–640.
- Corman, F., D'Ariano, A., Hansen, I.A., Pacciarelli, D., 2011. Optimal multi-class rescheduling of railway traffic. *Journal of Rail Transport Planning & Management* 1, 14–24. <http://dx.doi.org/10.1016/j.jrtpm.2011.06.001>.



- Corman, F., D'Ariano, A., Pacciarelli, D., Pranzo, M., 2012. Bi-objective conflict detection and resolution in railway traffic management. *Transportation Research Part C: Emerging Technologies* 20, 79–94. <http://dx.doi.org/10.1016/j.trc.2010.09.009>.
- Corman, F., D'Ariano, A., 2012. Assessment of advanced dispatching measures for recovering disrupted railway traffic situations. *Transportation Research Record: Journal of the Transportation Research Board* 2289, 1–9. <http://dx.doi.org/10.3141/2289-01>.
- D'Ariano, A., Pacciarelli, D., Pranzo, M., 2007. A branch and bound algorithm for scheduling trains in a railway network. *European Journal of Operational Research* 183, 643–657. <http://dx.doi.org/10.1016/j.ejor.2006.10.034>.
- D'Ariano, A., Corman, F., Pacciarelli, D., Pranzo, M., 2008a. Reordering and local rerouting strategies to manage train traffic in real time. *Transportation Science* 42, 405–419. <http://dx.doi.org/10.1287/trsc.1080.0247>.
- D'Ariano, A., Pacciarelli, D., Pranzo, M., 2008b. Assessment of flexible timetables in real-time traffic management of a railway bottleneck. *Transportation Research Part C: Emerging Technologies* 16, 232–245. <http://dx.doi.org/10.1016/j.trc.2007.07.006>.
- D'Ariano, A., Pranzo, M., 2009. An advanced real-time train dispatching system for minimizing the propagation of delays in a dispatching area under severe disturbances. *Networks and Spatial Economics* 9, 63–84. <http://dx.doi.org/10.1007/s11067-008-9088-1>.
- Dollevoet, T., Huisman, D., 2013. Fast heuristics for delay management with passenger rerouting, *Public Transport*. Doi:<http://dx.doi.org/10.1007/s12469-013-0076-6>.
- Greenberg, H.H., 1968. A branch-and-bound solution to the general scheduling problem. *Operations Research* 16 (2), 352–361.
- Higgins, A., Kozan, E., Ferreira, L., 1996. Optimal scheduling of trains on a single line track. *Transportation Research Part B* 30 (2), 147–161.
- Jespersen-Groth, J., Potthoff, D., Clausen, J., Huisman, D., Kroon, L., Maróti, G., Nielsen, M.N., 2009. Disruption management in passenger railway transportation. In: Ahuja, R.K., Möhring, R.H., Zoroliagis, C.D. (Eds.), *Robust and Online Large-Scale Optimization, Lecture Notes in Computer Science*, vol. 5868. Springer, Berlin, pp. 399–421. [http://dx.doi.org/10.1007/978-3-642-05465-5\\_18](http://dx.doi.org/10.1007/978-3-642-05465-5_18).
- Jovanovic, D., Harker, P.T., 1991. Tactical scheduling of rail operations: the SCAN I system. *Transportation Science* 25 (1), 46–64.
- Kanai, S., Shiina, K., Harada, S., Tomii, N., 2011. An optimal delay management algorithm from passengers' viewpoints considering the whole railway network. *Journal of Rail Transport Planning & Management* 1, 25–37. <http://dx.doi.org/10.1016/j.jrtpm.2011.09.003>.
- Kliewer, N., Suhl, L., 2011. A note on the online nature of the railway delay management problem. *Networks* 57, 28–37. <http://dx.doi.org/10.1002/net.20381>.
- Kroon, L.G., Peeters, L.W., 2003. A variable trip time model for cyclic railway timetabling. *Transportation Science* 37 (2), 198–212.
- Meng, L., Zhou, X., 2011. Robust single-track train dispatching model under a dynamic and stochastic environment: A scenario-based rolling horizon solution approach. *Transportation Research Part B* 45, 1080–1102.
- Rizvi, H., I., 2010. *Pakistan Railways Time Table, For Passenger Trains (Staff Copy)*, Pakistan Railways.
- Sato, K., Tamura, K., Tomii, N., 2013. A MIP-based timetable rescheduling formulation and algorithm minimizing further inconvenience to passengers. *Journal of Rail Transport Planning & Management* ,<http://dx.doi.org/10.1016/j.jrtpm.2013.10.007>.
- Schachtebeck, M., Schöbel, A., 2010. To wait or not to wait—and who goes first? Delay management with priority decisions. *Transportation Science* 44 (3), 307–321.
- Shafia M., A., Aghaee M., P., Sadjadi S.,J., Jamili A., 2012. Robust train timetabling problem: mathematical model and branch and bound algorithm. *Intell Trans Syst, IEEE Trans* 13(1):307–317.
- Szpigel, B., 1973. *Optimal train scheduling on a single track railway*. *Operations Research*'72. North-Holland, Amsterdam, Netherlands, pp. 343–352.
- Tomii, N., Tashiro, Y., Tanabe, N., Hirai, C., Muraki, K., 2005. Train rescheduling algorithm which minimizes passengers' dissatisfaction. In: Ali, M., Esposito, F. (Eds.), *Innovations in Applied Artificial Intelligence, Lecture Notes in Artificial Intelligence*, vol. 3533. Springer, Berlin, pp. 829–838. [http://dx.doi.org/10.1007/11504894\\_113](http://dx.doi.org/10.1007/11504894_113).
- Törnquist, J., 2006. Computer-based decision support for railway traffic scheduling and dispatching: a review of models and algorithms. In: Kroon, L.G., Möhring, R.H. (Eds.), *5th Workshop on Algorithmic Methods and Models for*



Optimization of Railways. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Dagstuhl. <http://dx.doi.org/10.4230/OASIS.ATMOS.2005.659>.

- Tornquist, J., Persson, J., 2007. N-tracked railway traffic re-scheduling during disturbances. *Transportation Research Part B* 41 (3), 342–362.
- Törnquist, J., 2007. Railway traffic disturbance management — an experimental analysis of disturbance complexity, management objectives and limitations in planning horizon. *Transportation Research Part A: Policy and Practice* 41, 249–266. <http://dx.doi.org/10.1016/j.tr.2006.05.003>.
- Törnquist Krasemann, J., 2012. Design of an effective algorithm for fast response to the re-scheduling of railway traffic during disturbances. *Transportation Research Part C: Emerging Technologies* 20, 62–78. <http://dx.doi.org/10.1016/j.trc.2010.12.004>.
- Zhou, X., Zhong, M., 2005. Bi-criteria train scheduling for high-speed passenger railroad planning applications. *European Journal of Operational Research* 167 (3), 752–771.
- Zhou, X., Zhong, M., 2007. Single-track train timetabling with guaranteed optimality: branch-and-bound algorithms with enhanced lower bounds. *Transp Res B* 41:320–341.