INFORMS Journal on Applied Analytics

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Station Dispatching Problem for a Large Terminal: A Constraint Programming Approach

Rajnish Kumar, Goutam Sen, Samarjit Kar, Manoj Kumar Tiwari

http://pubsonline.informs.org/journal/inte/

INTERFACES
Vol. 48, No. 6, November–December 2018, pp. 510–528
ISSN 0092-2102 (print), ISSN 1526-551X (online)

Station Dispatching Problem for a Large Terminal: A Constraint Programming Approach

Rajnish Kumar,[a] Goutam Sen,[b] Samarjit Kar,[a] Manoj Kumar Tiwari[b]

[a] Department of Mathematics, National Institute of Technology, Durgapur 713209, West Bengal, India; [b] Department of Industrial and Systems Engineering, Indian Institute of Technology Kharagpur, Kharagpur 721302, West Bengal, India

Contact: rajnishk435@gmail.com, https://orcid.org/0000-0003-1494-518X (RK); gsen@iem.iitkgp.ernet.in, https://orcid.org/0000-0002-2290-9114 (GS); kar_s_k@yahoo.com, https://orcid.org/0000-0002-2059-1755 (SK); mktiwari9@iem.iitkgp.ernet.in, https://orcid.org/0000-0001-8564-1402 (MKT)

Received: September 18, 2017
Revised: January 7, 2018
Accepted: March 12, 2018
Published Online in Articles in Advance: December 5, 2018

https://doi.org/10.1287/inte.2018.0950

Abstract. At Howrah Station, the largest railway station in India, planners manually determine the routing and scheduling of train movements (referred to as train dispatching) at present. This approach to train dispatching, which is a complex combinatorial optimization problem, can result in long delays because planners have many resource-allocation options in the spatial and temporal domains. In this paper, we discuss a computational model to provide the best feasible solution(s) for minimizing the dispatching delays within a planning horizon. We consider an integrated station dispatching problem that consists of allocating and scheduling platforms and routes for trains at Howrah Station, and we develop a constraint programming (CP)-based approach to model the problem. We show that the delay that the model generates is approximately half the delay currently observed in practice. The solutions have been verified by Howrah Station authority, and the model is being considered for implementation. The novelty of the solution is that it uses a CP-based model for the integrated station dispatching problem and successfully deals with a large station with specific operational protocols.

History: This paper was refereed.

Keywords: constraint programming • Indian Railway • Howrah Station • combinatorial optimization • integrated station dispatching problem • time-interval variable

Introduction

In a railway network, the movement of trains requires resources such as platforms and tracks. These resources are unary or disjunctive; that is, they are nonshareable in overlapping periods. This ensures safety in train operations by avoiding conflict situations in which two or more trains attempt to simultaneously occupy unary resources. A railway timetable is presumably designed such that trains move without any conflict. However, predetermined platform and route allocations based on a fixed timetable may not be useful in practice because of delays in train movements within the railway network. At Indian Railway's Howrah Station, planners use their judgement and experience to find a feasible solution, in real time, for the platform and route allocation of arriving and departing trains. The use of this approach, however, can result in long delays within the station, referred to herein as station dispatching delay, because manually examining a very large number of combinations of resource-allocation options in time and space is impossible. Therefore, the Howrah Station authority requires a computational model to provide feasible solutions that will result in reducing delay within a given decision time window.

A study of the literature on the dispatching problem shows the limitations of extant approaches in handling large-station instances such as Howrah. The dispatching problem belongs to a class of no-wait job-shop scheduling problems because it deals with the assignment of single-capacity (unary) resources to trains in an uninterruptible manner (Burkolter 2005). The dispatching problem is known to be NP-complete (Zwaneveld et al. 1996, Lamorgese and Mannino 2015). Job-shop scheduling problems have been traditionally modeled by disjunctive mixed-integer linear programming (MILP) formulations (Balas 1985). The possible conflict situations in disjunctive MILP formulations are either represented by big-M precedence constraints (Törnquist and Persson 2007, Mannino and Mascis 2009) or time-indexed formulations (Zwaneveld et al. 1996, Brännlund et al. 1998, Şahin et al. 2010, Caimi et al. 2011, Cacchiani et al. 2012). However, both approaches have limitations. The time-indexed formulations introduce large numbers of binary variables, whereas big-M formulations lead to poor bounds (Lamorgese et al. 2016).

A railway network consists of stations connected by tracks. Planners must control the movements within the station limits and on the track portion between stations. The literature on the station-level dispatching problem deals primarily with a simple layout of a single route from a station entry point to a platform and from a platform to a station exit point (Lamorgese et al. 2016). Thus, platform allocation has essentially been treated the same as route allocation, and vice versa. The train-platform-allocation problem has been studied extensively in the literature (Cardillo and Mione 1998, Billionet 2003, Caprara et al. 2011). In the Indian Railway context, Chakroborty and Vikram (2008) have used MILP formulations for the optimal assignment of platforms under partial schedule compliance for the Kanpur central station in Northern India. The problem of controlling train movements within a station has been referred to as the (station) traffic-control problem (Mannino and Mascis 2009), real-time railway traffic-management problem (Pellegrini et al. 2014), or station dispatching problem (SDP) (Lamorgese and Mannino 2015). In this paper, we combine the real-time decisions on route and platform allocation within the station limits and refer to it as the *integrated station dispatching problem* (ISDP).

Lamorgese and Mannino (2015) and Lamorgese et al. (2016) develop decomposition-based solution approaches to the dispatching problem for an entire railway network and model the station dispatching problem as a slave subproblem in the master–slave decomposition framework. The former paper uses appropriate feasibility cuts in the master problem, whereas the latter uses a Benders-like decomposition approach. However, both papers use two simplifying assumptions for the slave subproblem of station dispatching. First, the stations have a simple layout with a single route from entry point to platform and platform to exit point. Second, the travel time within the station is uniform and fixed for all trains. Mannino and Mascis (2009) solve the station dispatching problem for metro stations of relatively small size (i.e., with up to eight arriving trains and a maximum of six routes) by using a branch-and-bound algorithm.

A survey by Barták et al. (2010) indicates that constraint programming (CP) is aptly suited for scheduling problems. In a rare deviation from the traditional MILP approach, Rodriguez (2007) treats SDP as a scheduling problem and develops a CP-based solution approach. The author uses a branch-and-bound strategy to explore the solution search space. The search space is pruned by mechanisms of a timetable of variations of resource utilization, resource availability over time, disjunctive constraint propagation, and edge finding. The author arrives at a feasible solution using chronological backtracking in the search tree. Problem instances with 6 to 24 trains and four to eight alternative routes for each train are considered. The largest instance considered has 9,801 variables, 10,672 constraints, and 1.7×10^{13} possible route-assignment combinations. Because of the enormous size of the search space, Rodriguez (2007) recommends that a realistic goal should be to find a good solution, although it might not be optimal, within a reasonable time limit. In view of the reported limitations of MILP and the increasing evidence of the suitability of CP to handle large instances of scheduling problems, we develop a CP-based approach for our ISDP of Howrah Station. We can assess the complexity of the Howrah ISDP by realizing that the number of variables and constraints in many instances are 1–10 times higher than that of the largest instance that Rodriguez (2007) considers.

The contributions of this paper can be summarized in three points: (1) it studies an integrated station dispatching problem, which schedules both route allocation and platform allocation; (2) it develops a CP-based solution methodology; and (3) the solution is applied to Howrah Station, a large station with specific operational protocols. We organize the rest of the paper in the following sequence of sections: *Problem Description*, *Model Development*, *Solution Methodology*, *Computational Results*, and *Conclusions*. The appendix includes the mathematical CP model.

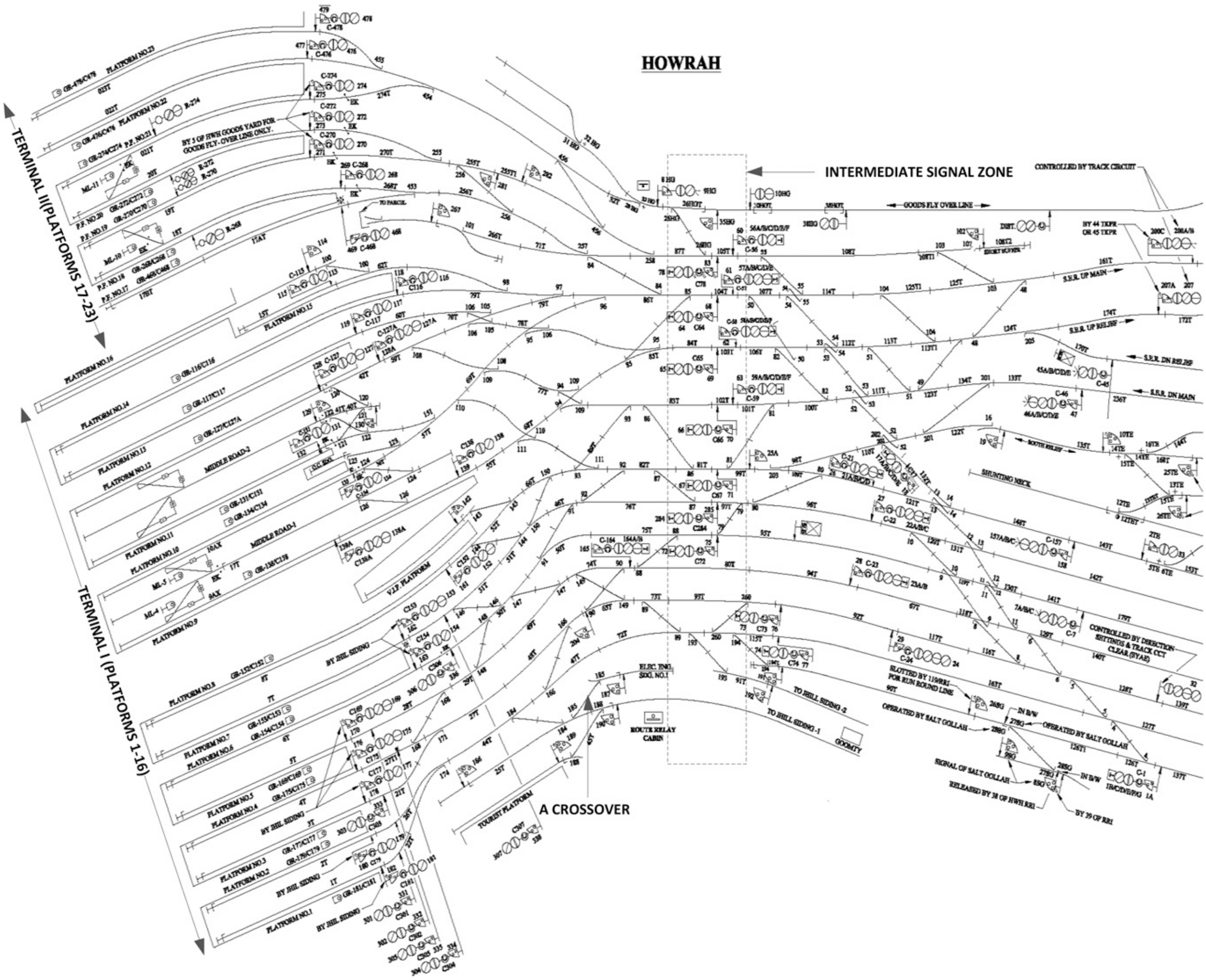
Problem Description

Howrah Station, the largest and one of the busiest railway stations in India, has two terminals, Terminal I and Terminal II, and 22 operating platforms (Figure 1). We describe related railway terminologies in the context of Howrah Station.

The Indian railway network is divided into separate control sections referred to as block sections. Howrah Station is one such block section, which starts from the entry-point signals of the station, covers the platforms, and ends at the exit-point signals. Trains approach and leave this station from three outstation directions: the Howrah main line coming from Bandel, the Howrah chord line coming from Bardhmann via Dankuni and the Kharagpur direction, and five yard directions—Eastern Railway (ER) car shed, South Eastern Railway (SER) car shed, Tikiapara yard, Sorting yard, and Santragachi yard (Figure 2). In a terminal station such as Howrah, the entry points and exit points of the block section are on the same side of the platform. As a result, some resources may be common for arrivals and departures. Howrah Station has one reversible track, which allows for bidirectional movement to and from Bandel. In addition, in specific outstation and yard tracks, reversible movements are allowed for predefined distances from platforms and yards, respectively.

A crossover is a joined pair of points, one each on two adjacent tracks (for example, crossover c for Track 1 and Track 2 in Figure 3). A crossover can be aligned in two ways: normal (N), in which the train is not allowed to change the track, and reverse (R), in

Figure 1. Howrah Station Layout Part I Shows Terminal I and II with Platforms (Left to Right) 1 to 16 and 17 to 23, Respectively, a Typical Crossover, and the Intermediate Signal Zone



Note. Platform 16 is not in use.

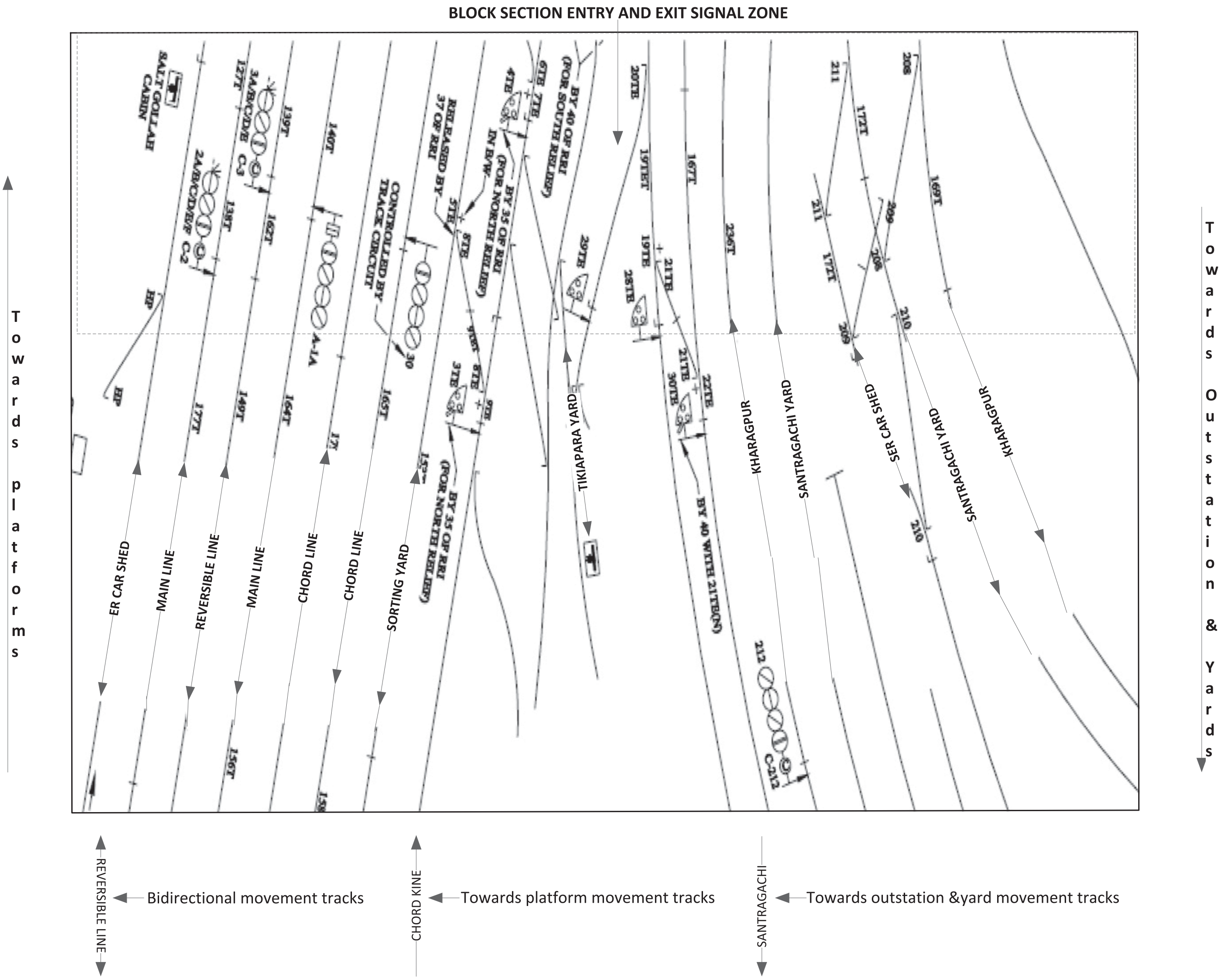
which the train movement is changed to the adjacent track.

A route is a portion of the track between its starting and stopping signal. It is uniquely characterized by a sequence of crossover points with their locking positions (N or R). However, if a crossover point (e.g., 88N in Figure 4) is locked for a train movement in either the N or R position on a given route, the movement control system blocks train movements on all the routes having this crossover. In view of this safety protocol at Howrah Station, we consider a route to be a sequence of crossovers regardless of its locking status. Sometimes a route may also include an overlapping portion (e.g., 100–120 meters) of the track with the subsequent route. This overlapping portion of the track is provided for safety purposes in case the train passes beyond the stopping signal.

At Howrah Station, only selected routes are allowed for train movement. These routes are updated annually in a permissible-movements route chart (PMRC). Table 1 and Figure 4 depict a typical permissible route in a PMRC.

A path is a sequence of routes between an origin–destination pair. At the station level, we consider two types of paths: an arrival path from an entry-point signal to a platform and a departure path from a platform to an exit-point signal. Control signals are preidentified starting signals, a subset of all starting signals of the constituent routes in a path, at which railway planners control train movements. At Howrah Station, an arrival or departure path typically consists of two to five routes. Train movements are controlled at two starting signals in the path: (1) the entry signal at the block section in the arrival path or the starting signal at the platform in

Figure 2. The Howrah Station Layout Part II Shows the Block Section Entry-Exit Signal Zone and Arrival-Departure Directions



the departure path and (2) an intermediate signal in the arrival or departure path (Figure 1). Accordingly, we consider two control signals and divided a path into subpaths 1 and 2, which meet at the intermediate signals. Rodriguez (2007) considers the control area at the smallest level of track circuits, which are portions of tracks where an electrical circuit detects the presence of trains. However, in our study, we consider the control area at the subpath level. Because subpaths are made up of single or multiple routes, a subpath can be uniquely expressed as a sequence of crossovers, as we describe in the appendix.

The design of the official train timetable is assumed to fulfill the nonconflicting resource-assignment criteria. However, in practice, we observe different types of delays relative to the timetable in a railway network. External delay refers to the delay in the arrival of the train at the entry-point signal of the station block section. Howrah Station authority has implemented two operational delay-handling mechanisms: (1) inclusion of train recovery time (TRT) in the official timetable, by keeping the difference between the time of arrival at the platform and the entry-point signal equal to the sum of the travel time in the arrival path and the TRT, and (2) platform stay compression, which refers to the reduction in the stopping time of a train at the platform to the extent possible. Despite the cushioning effect of these mechanisms, the railroad experienced significant delays. Therefore, we consider the problem of reducing the station dispatching delay, because external delay is beyond the station's control. Planners schedule the resources (i.e., platforms and routes) for arriving and departing

Figure 3. A Typical Crossover Shows Movements in Normal (N) and Reverse (R) Directions

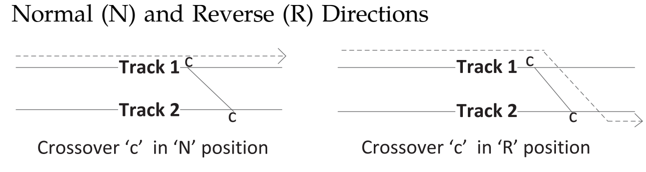
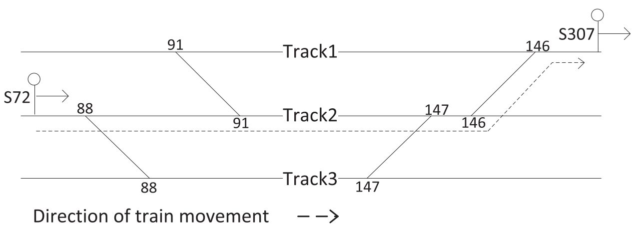


Figure 4. Typical Route Comprising Crossovers Set {88N, 91N, 147N, 146R} for Movement from Signal S72 to S307



trains within a planning horizon, which we call the dispatching time window (DTW). The duration of a DTW depends primarily on the station-level information available about the trains, and we determined the duration in consultation with the planners. Our objective is to minimize the station dispatching delay through a computational model that generates the best feasible solutions for a DTW.

Decision trains are the trains for which dispatching decisions are to be taken during a given DTW. To integrate platform allocation into the route allocation problem, we categorize decision trains into two classes—arrival decision trains, which require routing and platform allocation, and departure decision trains, which require only routing. System trains are the trains within the station block section at the start of a DTW. In an earlier DTW, arrival paths and platforms have already been determined for arrival system trains, and departure paths have already been determined for departure system trains. We note that departure decision trains are a subset of arrival system trains.

Howrah Station planners prepare a train platform preference list (TPPL), which is a list of preferable platforms for allocation to each arrival decision train. The platforms in the list for each train are given preference values based on factors such as length compatibility of both trains and platforms, direction of train arrival and departure, commuters' convenience, and other operational requirements.

Specifically, the following decisions must be made for the ISDP at Howrah: (1) allocation of platform to an arrival decision train, (2) start time and end time of stay at the platform for an arrival decision train, (3) time of release of an arrival decision train at the block section entry signal, (4) subpath of an arrival decision train from entry-point signal to intermediate signal, (5) time of release of an arrival decision train at an intermediate signal, (6) subpath of an arrival decision train from the intermediate signal to an allocated platform, (7) time of release of the departure decision train from the starting signal of the train's stopping platform, (8) subpath of the departure decision train from the platform to an intermediate signal, (9) time of release of the departure decision train from the intermediate signal, and (10) subpath of the departure decision train from the intermediate signal to an exit-point signal.

Model Development

We built the model on the interaction between trains and resources at the station block section level. An activity refers to the allocation of resources (i.e., platform and subpaths) to a train in the temporal domain. A pair of activities is forbidden or in conflict if they involve the simultaneous assignment of a unary resource. On the basis of these criteria, we classify a given railway resource or a pair of resources as in conflict (Inc). A given unary resource (e.g., a platform or subpath) is naturally in conflict with itself. A pair of subpaths, z_i and z_j, are said to be in conflict if they have any common crossover; see *Constraints* in the appendix.

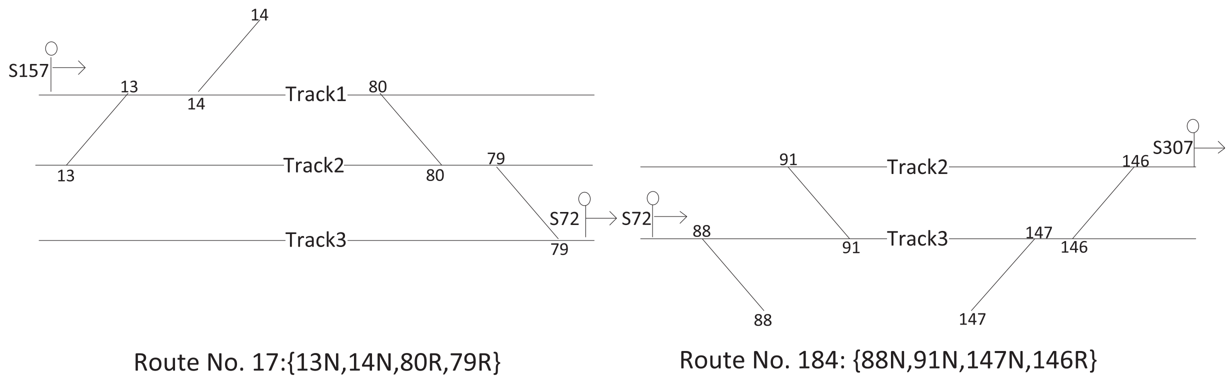
Howrah Station has multiple platforms (in the range of 5–7) in the TPPL for each train, each platform has multiple arrival and departure paths (in the range of 2–20), and each path has multiple routes (in the range of 2–5). Explicitly listing all possible arrival and departure paths within the block section is a tedious task. Hence, we generate all possible arrival or departure paths for a train, when required, using a computational framework. We obtain the paths from a directed graph of the adjacency list of 720 permissible routes in a PMRC, which is used for outstation and yard train movements. We exclude nonfunctional or minimally used routes, such as those to a visiting dignitary platform, tourist platform, and a railway yard (termed Jheel siding). If the stopping signal of a route is same as the starting signal of another route, there exists an edge between two nodes representing these routes in the adjacency list directed graph. We apply depth-first search on the directed graph to obtain all possible paths for a train.

As an illustration, we consider two routes from the set of routes for the movement of trains toward platforms (Figure 5). The stopping signal of Route 17 is the same as the starting signal of Route 184. Therefore, we introduce an edge between two nodes representing Route 17 and Route 184 in the adjacency list directed graph of the route set. The directed

Table 1. A Typical Route of Howrah PMRC from Starting Signal S72 to Stopping Signal S307 Defined by Crossover Set

Direction of movement	Starting signal	Stopping signal	Crossovers
S72 to S307	S72	S307	88N 91N 147N 146R

Figure 5. Generation of a Directed Edge in an Adjacency List–Directed Graph Between Nodes Representing Routes 17 and 184, Which Have a Common Ending and Starting Signal, Respectively



graph is completed by addition of all such possible edges.

We obtain 1,336 paths for all directions. The number of possible distinct path pairs is 891,780. Of these, 635,179 (~71%) pairs are in conflict; that is, they have one or more common crossovers. The high percentage of in-conflict path pairs is due to the reduction in the number of tracks near an intermediate-signal zone (Figure 1). This is one of the main infrastructural bottlenecks causing a large station dispatching delay at Howrah Station. Both railway planners and therefore ISDP aim to minimize the delay under the given station infrastructure. We established the magnitude of complexity of Howrah Station relative to the instances reported in Caprara et al. (2011) (Table 2).

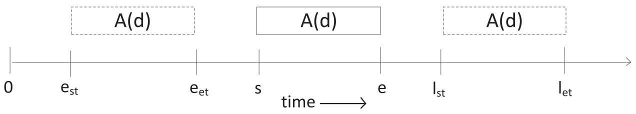
The ISDP described in the *Problem Description* section involves making many discrete decisions. We develop a CP approach using artificial intelligence to model and solve the ISDP. The model uses logical reasoning to reduce the large search space. CP allows the application of a wide variety of specialized constraints to effectively solve a scheduling problem. We model the ISDP using CP with variables, domains for the variables, and a set of constraints imposed on the possible assignment of values to the variables. A feasible solution is a solution that lies within the corresponding domains of the variables while satisfying all the constraints. CP is declarative; that is, it describes the solution search space by providing domains for the variables and restricts the assignments in the feasible solution by means of constraints. The problem of obtaining a feasible solution using CP is known as the constraint satisfaction problem (CSP). The addition of an objective function to the CSP converts it to a constraint optimization problem (COP). In this section, we discuss modeling the ISDP as a COP in which the objective is to minimize the station dispatching delay.

We use the conditional time-interval variable (or simply, the interval variable) of IBM's ILOG CPLEX CP optimizer, which is elegantly suited to model scheduling problems. An interval variable represents an interval of time characterized by its execution status, a start time, and an end time (Figure 6). The duration of the interval variable is the difference between the end time and start time. In the CP model, an interval variable is associated with each activity. An important aspect of the conditional time-interval variable is that it can be optional (i.e., including it in the solution is optional). Two optional interval variables are associated with each alternative platform from the set of platforms in the TPPL; they represent choices for two mandatory activities associated with arrival decision trains—the allocation of platform and allocation of arrival paths and subpaths. These variables are the platform choice for the arrival decision train (y'^{a}_{ij}) and the platform choice for the arrival subpath of the arrival decision train (z'^{a}_{ijl}); refer to notations for optional activities in *Interval Variables* in the appendix. Although the platform selected in the solution in both choices is essentially

Table 2. The Complexity of Howrah Railway Station (in Bold) by Comparing the Number of Paths, Platforms, and In-Conflict Path Pairs to European Stations in Caprara et al. (2011)

Station	No. of platforms	No. of paths	No. of in-conflict path pairs
Palermo Centrale	11	64	1,182
Genovo Piazza Principe	10	174	7,154
Bari Centrale	14	89	1,996
Milano Centrale	24	312	29,294
Howrah Terminal	**22**	**1,336**	**635,179**

Figure 6. Interval Variable for an Activity (A) with Start Time (s), End Time (e), Duration (d), Earliest Start Time (e_{st}), Earliest End Time (e_{et}), Latest Start Time (l_{st}), and Latest End Time (l_{et})



the same, we introduce the second variable to model two levels of alternative choices for an arrival path. The arrival path must be selected from a set of paths for a train–platform pair, and the platform in the train–platform pair must be selected from the set of platforms in the TPPL. One optional interval variable is associated with each path belonging to the set of all possible arrival and departure paths. These variables are the arrival subpath choice for the arrival decision train (z''^{a}_{ijkl}) and the departure subpath choice for the departure decision train (z'^{d}_{ikl}), respectively; as above, refer to notations for optional activities in Table A.4 in the appendix. A present interval variable represents a mandatory activity, such as the allocation of an arrival path, departure path, or halting platform to the train; this variable must be part of the solution. These variables are arrival subpath allocation for the arrival decision train (z^{a}_{il}), halting platform allocation for the arrival decision train (y^{a}_{i}), and departure subpath allocation for the departure decision train (z^{d}_{il}), respectively; refer to notations for mandatory interval variables in Table A.3 in the appendix.

The domain of an interval variable is defined by the ranges of its execution status, start time, end time, and the duration. The minimum and maximum values of the range of the execution status, which can assume binary values 0 or 1, are denoted by $x_{\min}$ and $x_{\max}$, respectively. We refer to the minimum and maximum values of the range of start time as the earliest start time (e_{st}) and latest start time (l_{st}), respectively. Similarly, we refer to the minimum and maximum values of the range of end time as the earliest end time (e_{et}) and latest end time (l_{et}), respectively (Figure 6). The maximum and minimum values of the ranges of the duration are represented as $d_{\max}$ and $d_{\min}$, respectively.

The range of execution status for optional interval variables is defined by $x_{\min} = 0$ and $x_{\max} = 1$. The expressions for the ranges of start time, end time, and duration of optional interval variables, which we show in *Domains* in the appendix, are formed using the following guidelines:

1. The earliest start time e_{st} of an interval variable associated with a subpath resource allocation activity is the time of train's actual arrival at the subpath entry point.
2. The latest start time l_{st} is the sum of actual arrival time and the delay in the release of the train at the subpath entry point.
3. The earliest end time e_{et} of an interval variable associated with a subpath resource allocation activity is the sum of the time of actual arrival at the resource entry point and the travel time in the subpath.
4. The latest end time l_{et} is the sum of time of actual arrival at the resource entry point, travel time within the subpath resource, and delay in the release of the train at the subpath entry and exit points.
5. The domain of the start time of subpath 1 is same as that of the domain of the end time of subpath 2. We assume zero transition time between two subsequent subpaths in the same travel direction because we have included the transition times in the travel times of the subpaths. The domain of the start-time platform occupancy is same as that of the domain of the end time of the arrival subpath 2.
6. The train once released from the last subpath in the arrival direction reaches the platform after the travel time without any delay.
7. The actual arrival time of the train at the entry-point signal is the sum of the scheduled arrival time at the entry-point signal and the train's external delay.
8. The scheduled departure time of the train from the platform is from the official timetable.
9. The departure readiness time at which the train is ready to depart from a platform is the maximum of the scheduled departure time and the time obtained from the sum of actual arrival time at entry-point signal, travel times in arrival subpaths, delays in the travel times in the arrival subpaths, and minimum-stay duration on the platform.
10. The model is provided with input data of the trains' scheduled arrival times at entry-point signals, external delays in train arrivals, scheduled departure times, departure readiness times, travel time in each subpath, recovery time for each train, and maximum- and minimum-stay duration of each train on the platform. The execution status, start time, end time, and duration for present interval variables are defined by model constraints.

We use four types of constraints in the CP model of the ISDP: logical execution and *if Then* constraints, temporal precedence constraints, resource-assignment alternative constraints, and unary-resource *noOverlap* constraints. Please refer to Laborie et al. (2009) for a detailed explanation of these constraints.

Logical execution constraints *setPresent* and setOptional in the CP optimizer define the execution status of the present and the optional interval variables, respectively. The *setPresent* constraints ensure that the allocation of platform, arrival, and departure paths for each train in the DTW must be present in the solution; see Equations (A.1)–(A.3) in the appendix. The *setOptional* constraints apply on optional interval variables that represent alternative choices; see Equations (A.4)–(A.7) in the appendix. The model includes three types of logical *ifThen* constraints. The first logical *ifThen* constraints impose the condition that the same platform is selected for the mandatory activities of allocating a platform and arrival path for a train. The arrival path shall be from the set of train and selected platform pairs; see Figure 7 and Equation (A.8) in the appendix. The second and third logical *ifThen* constraints impose the condition that if one subpath of a path is present in the solution, the second subpath of the same path also must be present in the solution; see Figure 7 and Equations (A.9) and (A.10) in the appendix.

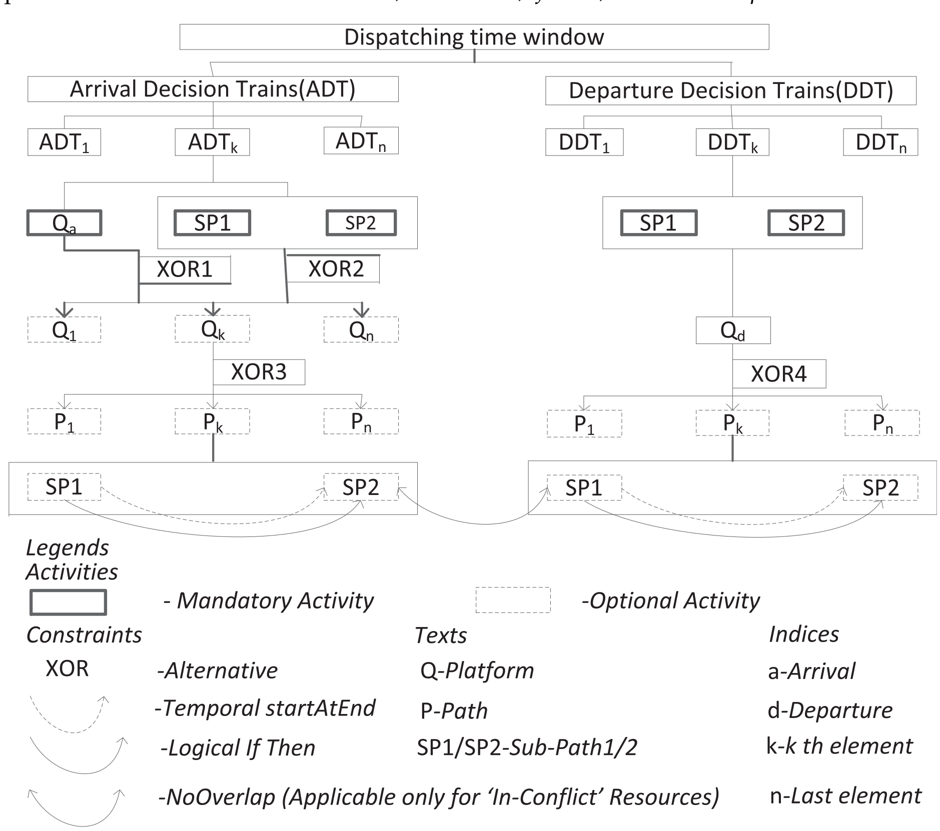
The *startAtEnd* constraints represent temporal precedence relationships between interval variables for a pair of resources. So, in two subsequent subpaths, the e_{st} of the second subpath in the direction of travel is same as the l_{et} of the preceding subpath; see Figure 7 and Equations (A.11)–(A.13) in the appendix. In addition, platform occupancy starts at the end of the second subpath occupancy in the arrival direction, indicating that the e_{st} of the platform occupancy interval variable is same as the l_{et} of the interval variable for the second arrival subpath; see Equation (A.14).

The resource-assignment alternative constraints represent the condition that a mandatory activity will select only one resource from a set of alternative resources. In an alternative constraint involving a present interval variable, referred to as a master variable, and a set of optional interval variables, only one optional interval variable from the set is selected for execution. For an alternative constraint between an optional master interval variable and a set of optional interval variables, only one optional interval variable from the set is selected for execution if and only if the master variable is executed. The start and end times of the master interval variable are the same as those of the selected optional interval variable.

The alternative constraints in our CP model represent four choices for a train. First, a choice of halting platform is made from a set of platforms in the TPPL; see XOR1 in Figure 7 and Equation (A.15) in the appendix. Second, a choice of platform is made from a set of platforms in the TPPL for selecting a set of alternative arrival paths corresponding to train–platform pair; see XOR2 in Figure 6 and Equation (A.16) in the appendix. Third, a choice of arrival path, and consequently the arrival subpaths 1 and 2, is made from the set of arrival paths corresponding to the selected train–platform pair; see XOR3 in Figure 7 and Equation (A.17) in the appendix. Fourth, a choice of departure path, and consequently the departure subpaths 1 and 2, is made from multiple departure paths; see XOR4 in Figure 7 and Equation (A.18) in the appendix.

The *noOverlap* constraints impose temporal disjunction on the use of unary resources. It specifies that a pair of activities demanding the use of in-conflict unary

Figure 7. Four Types of Model Constraints: *Alternative*, *startAtEnd*, *ifThen*, and *noOverlap*



resource(s) does not overlap in time; see Figure 7 and Equations (A.19)–(A.27) in the appendix.

The primary objective of Howrah Station authority is to obtain a best-feasible solution for the 10 decisions of the ISDP, which include platform allocation, as we describe in the *Problem Description* section. In the ISDP model, according to the directives of the station authority, the quality of a solution is measured by the preference value of the allocated platform and station dispatching delay. High preference values are assigned to platforms in the TPPL such that arrival and departure paths to these platforms have fewer crossovers and therefore contribute less to the station dispatching delay. Station dispatching delay is associated with four types of train movement—outstation departure, outstation arrival, yard arrival, and yard departure, with operational priorities in that order. To reflect the desired quality of the solution, weights are associated with platform allocation and station dispatching delay of four categories of train movement. The highest weight is attached to platform allocation; lower weights are attached to dispatching delays related to the four categories of train movement in the order of their importance; see Equation (A.28) in the appendix.

Solution Methodology

The CP finds a feasible solution in the large search space of combinatorial problems such as ISDP using techniques of constraint propagation, search strategies, and search refinement. Constraint propagators effectively do the pruning (i.e., reduction) of the large search space by removing infeasible or inconsistent solutions. Every constraint in the CP model is associated with one or more constraint propagators. Constraint propagators basically use different algorithms for domain reduction. The time-interval variable contributes to reducing the domain by maintaining consistency in the ranges of start time, end time, and duration, which are bound by the condition that the duration equals the difference between end time and start time If the ranges are inconsistent in this relationship, the execution status of the interval variable is set to 0. In addition, if the execution status is 1, the model generates an error (Laborie and Jerome 2008). The propagation algorithms associated with constraints prune the search space to an extent beyond which pruning is not possible. This limiting point of domain pruning is referred to as a fixed point.

Beyond the fixed point, CP uses various search strategies to reach a feasible solution. In the CP approach, the model and the search are two distinct components. A given model can apply different search strategies to reach a solution. Branching is one such search strategy that divides the original problem into a set of smaller subproblems. The subproblems are recursively divided into new subproblems and solved until a solution is obtained or is not found because of infeasibility. Multiple branching search strategies can be used depending on the order in which the subproblems are solved. Examples include depth first, multipoint, and restart. By contrast, a COP uses the branch-and-bound method, which constrains the objective value to be within the best bounds obtained as the search progresses. In case of a very large search space, a combination of metaheuristics, such as large neighborhood search (LNS) and failure-directed search (FDS), are used to obtain a good-quality solution by exploring a part of the search space. LNS uses the best solution obtained at each stage to find new and better solutions. FDS directs the exploration of the search space by eliminating possible assignments that are most prone to failure. In our model instances, the CP optimizer extensively uses FDS.

Search refinement involves variable and value ordering. *Variable ordering* refers to prioritized instantiation of critical variables. *Value ordering* refers to the preferred order of the values to be assigned to a variable from its domain. We use variable ordering to instantiate the departure decision variables on a priority basis to reduce departure delays. The inference level of a constraint is a measure of its strength in reducing the domain. The inference level for a constraint can be set to various levels. We use the stronger extended inference level in place of the default inference level in the model. Other than variable ordering and an extended inference level, we rely on the built-in search and constraint propagation mechanisms in the CPLEX engine to solve the model. We code our model using Java Concert Technology in IBM ILOG CPLEX 12.6.0 with a CP optimizer and run the model on a fifth-generation Intel core processor with 2.90 GHz CPU and 8 GB RAM.

Computational Results

We simulate traffic-congestion scenarios using the route and platform allocation decisions taken in earlier dispatching time windows. This process of simulating the traffic congestion starts at about 2 a.m., when there is no train in the Howrah Station block section limit; see Figure 8. We obtained this time by finding maximal cliques of the train–platform occupancy interval graph using the Bron–Kerbosch algorithm (Bron and Kerbosch 1973). The interval graph, which is constructed from the official timetable, is a network of nodes and edges, with nodes representing the trains and edges representing overlapping platform occupancy intervals. The graph is used to find maximal cliques, each of which represents a group of trains with overlapping platform occupancy intervals. The maximal cliques are used to find connected components of the graph, which include a set of trains connected by one or more edges in the interval graph. The three connected components of the graph are disjoint at around 2 a.m.; therefore, the graph represents a time at which no train is in the system. We note that although we consider external delay in the model, the assumption that no train is in the system at around 2 a.m. is appropriate because we observe negligible or no delay in the network at this time. The model simulation can start at any time provided we have inputs on the position of trains within the system at that point in time. In our case, we start the simulation at 2 a.m., when the system is empty, so that no real-time input is required. The optimal schedule thus computed is entered as input to subsequent model instances. However, in the actual implementation of this model, we can start the simulation at any time by mapping the current temporal and spatial positions of the trains as an input. So the model is equally applicable in cases where there is no down time or near down time.

The DTW is restricted to eight minutes or less because of the following factors.

1. Howrah Station's current information network provides 15 minutes of advance information about trains reaching the entry point of station.

2. A train takes a minimum of 11 minutes for travel in the arrival path to reach the platform from the entry point.

3. The stay time for the trains at Howrah Station platforms ranges from 5–300 minutes. The stay durations of most local trains are at the lower end of this range. The long-distance trains that arrive from or depart to yards typically stay at the platform for 30–60 minutes. Some local and long-distance trains stay at the platforms for longer durations and leave without going to yards. Therefore, a train reaching the entry-point signal can be ready for departure after a minimum of 16 minutes (11 minutes of travel time plus a minimum of 5 minutes of stay time).

4. In a terminal station such as Howrah, departure decisions and arrival decisions are intertwined because arrivals and departures share more resources; for example, some tracks are reversible for the entire length within the block section or for a fixed distance from platform or yard.

5. The time limit for the model to generate a solution is kept at 200 seconds (about three minutes). The CP model provides multiple feasible solutions with the same objective value; this time limit facilitates the planner's ability to analyze the outcomes and select the best one prior to the DTW's start time. In most problem instances that include more than 60,000 variables and constraints, a feasible solution could not be generated within 200 seconds, or it resulted in out-of-memory errors. A larger DTW duration would include more decision trains and, consequently, a larger number of variables and constraints.

6. Howrah Station authority gives priority to minimizing departure delays over arrival delays. To achieve this objective, we use the variable ordering for departure decision variables as we explain in the *Solution Methodology* section. However, the variable ordering does not reduce the departure delays to the desired level. So, in addition, we introduce the concept of blocking the critical railway resources for departure decisions train on a priority basis. We accomplish this by staggering the DTW for arrival and departure decision trains, and we determine the staggered time window to be eight minutes or less.

A DTW starting at 120 minutes (see DTW 2 in Figure 9), for example, considers arrival decision trains in intervals of 120–128 minutes and departure decision trains in intervals of 128–136 minutes. The decisions for departure trains corresponding to 120–128 minutes were taken in the preceding DTW; see DTW 1 in Figure 9. We note that staggering the DTW does not exclude any arrival decision train from being a part of departure decision train because a train requires a minimum of 16 minutes to be ready for departure after arriving at the entry point. The prioritized blocking of critical rail resources for departure trains closely resembles the current practice at Howrah Station.

We select 24 random instances from 24 hours representing varying levels of traffic congestion to demonstrate the results (Table 3). The external delays are generated randomly from a triangular probabilistic distribution as suggested by the station authority. The triangular distribution is characterized by parameters a (minimum value), b (maximum value), and c (peak value). We take three sets of values for a, b, and c: $\{0, 10, 5\}$, $\{0, 30, 15\}$, and $\{0, 60, 30\}$ corresponding to local trains, small-distance express trains, and long-distance express trains, respectively. The ISDP model

Figure 8. (Color online) Traffic Intensity in Terms of Number of Trains per Hour at Howrah Station

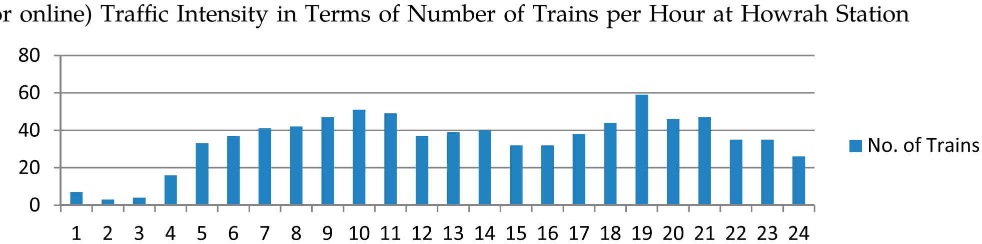
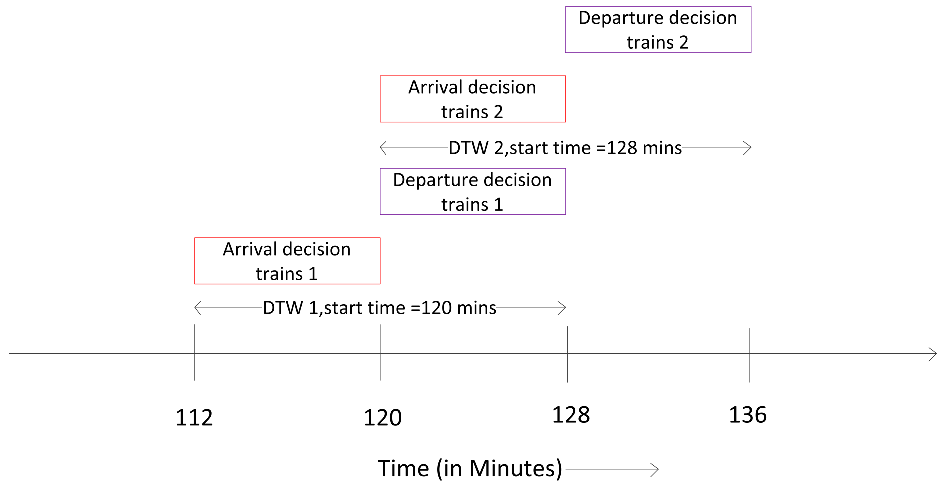


Figure 9. (Color online) Representation of Time Intervals for Arrival and Departure Decision Trains for Two Staggered Dispatching Time Windows: DTW 1 and DTW 2



provides best-feasible solution(s) within prespecified time limits (i.e., 200 seconds) for all instances. For example, a model instance for a DTW that starts at 1,200 minutes with a duration of 8 minutes, 2 arrival decision trains, 3 departure decision trains, and 16 system trains has 10,483 variables and 10,866 constraints. The model provides two best-feasible solutions with the same objective value for this instance. The delays given by the model, for outstation arrival, yard arrival, outstation departure, and yard departure are 16, 0, 4, and 0, respectively (Table 3).

In some model instances for a DTW of 8 minutes, the model generated time-limit- exceeded or out-of-memory errors. For example, a model instance with a DTW that starts at 568 minutes with a duration of 8 minutes, 6 arrival decision trains, 4 departure decision trains, and 20 system trains has 113,747 variables and 115,245 constraints and generates out-of-memory errors (Table 4).

Table 3. The Randomly Selected 24 Model: Start Time of the Instance, Duration of the Dispatching Time Window, Number of Decision and System Trains, Variables, Constraints, Number of Feasible Solutions, Model Solution Values of Total Arrival and Departure Outstation, and Yard Delays Obtained for the Selected Instances

		Decision trains						Arrival delays		Departure delays	
Start	Duration	Arrival	Departure	System trains	Variables	Constraints	No. of solutions	Outstation	Yard	Outstation	Yard
40	8	1	0	3	90	162	1	0	0	0	0
64	8	0	1	2	16	20	1	0	0	0	0
152	8	1	0	1	198	416	1	0	0	0	0
216	8	1	0	1	198	369	1	0	0	0	0
288	8	1	2	9	1,858	2,024	1	0	1	0	0
336	8	1	1	14	988	1,166	1	0	0	0	0
376	8	1	4	14	1,990	2,121	1	8	0	0	0
432	8	3	4	11	6,787	7,164	1	0	8	0	0
512	8	4	3	11	23,319	24,007	1	46	0	0	0
556	4	3	1	7	18,898	19,557	1	26	0	0	0
688	8	2	2	15	10,007	10,482	1	5	0	2	0
760	8	3	2	13	15,827	16,341	1	9	0	0	0
824	8	3	1	15	8,153	8,547	1	11	0	0	0
880	8	2	1	12	5,892	6,283	1	0	4	0	0
920	8	0	4	10	358	369	1	0	0	0	0
992	8	3	3	14	15,585	16,167	1	9	8	4	0
1,056	8	2	4	13	8,212	8,679	1	16	0	0	0
1,092	4	5	4	11	61,330	62,453	2	26	32	0	0
1,152	8	3	2	18	9,780	10,191	1	6	12	0	0
1,200	8	2	3	16	10,483	10,866	2	16	0	4	0
1,216	8	1	5	11	8,177	8,580	4	0	0	5	0
1,312	8	3	3	14	7,420	7,795	4	18	0	0	4
1,360	8	1	2	15	1,285	1,467	1	0	0	0	0
1,424	8	2	3	8	12,538	13,163	2	9	0	0	6

Table 4. Some Randomly Selected Model Instances: "Error" Output Because They Include a Larger Number of Decision and System Trains and a Substantially High Number of Model Variables and Constraints

		Decision trains				
Start	Duration	Arrival	Departure	System trains	Variables	Constraints
568	8	6	4	20	113,747	115,245
600	8	5	6	17	67,356	68,476
648	8	6	4	19	60,745	61,937
1,088	8	6	4	18	73,218	74,483
1,096	8	6	2	19	87,931	89,302

In these cases, we divide the instance such that each has a smaller DTW duration (Figure 10). The model instance with a DTW that starts at 1,088 minutes with a duration of 8 minutes, 6 arrival decision trains, 4 departure decision trains, and 18 system trains has 73,218 variables and 74,483 constraints generates a time-limit-exceeded error (Table 4). In this case, we divide the instance in two, with each having a DTW duration of 4 minutes.

The second instance DTW starts at 1,092 minutes and has 6 arrival decision trains, 4 departure decision trains, and 18 system trains. The number of variables and constraints for this instance decreases to 61,330 and 62,453, respectively, and the model provides two best-feasible solutions with the same objective value (Table 3).

To minimize the station dispatching delay for all trains in a DTW, the model finds the best-feasible solution. The outstation arrival delay per train in most instances is considerably less than the currently observed arrival delay, which is in the range of 30–45 minutes (Figure 11).

Staggering the dispatching time window in conjunction with variable ordering yields fewer departure delays compared with arrival delays. The average outstation arrival, outstation departure, yard arrival, and yard departure delays are obtained by dividing the total delay for each category by the respective number of corresponding category trains in the instance. The average delay in the outstation departure decision trains is one minute or less, which complies with the objectives of the station authority (Figure 12).

The delays correlate positively with traffic congestion represented by number of arrival decision trains, departure decision trains, and system trains (Figures 11 and 12). They also depend on other factors, such as the number of platforms and the number of paths for each platform in the TPPL of each arrival decision train, and the number of paths for each departure decision train. The yard arrival and departure delays are of little significance because they can be easily absorbed using operational methods and do not have much effect on commuter convenience. The total yard delays are in the range of 0–32 minutes (Table 3). The average yard delay is in the range of 0–16 minutes.

In most of the selected random instances, the first platform preference is chosen for all arrival decision

Figure 10. Model Execution, Including the Solution Method in the Case of a Time-Limit-Exceeded or Out-of-Memory Error

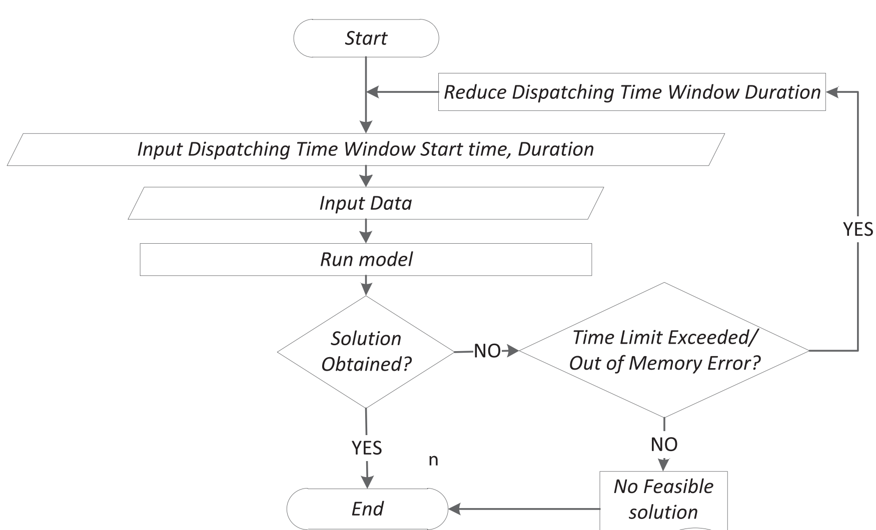
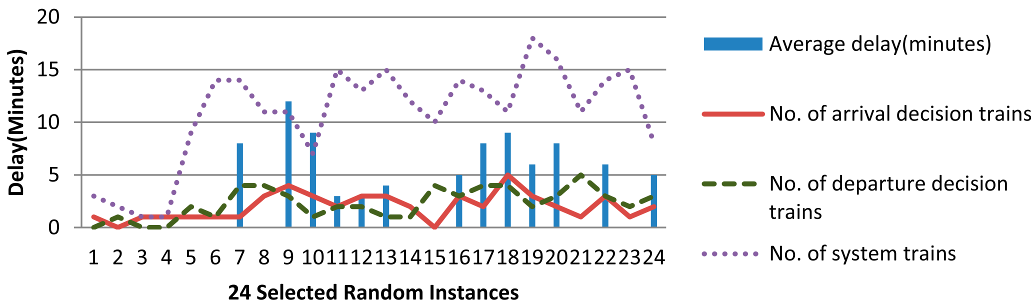


Figure 11. (Color online) Outstation Average Arrival Delays and Traffic Density Relative to the Numbers of Arrival Decision, Departure Decision, and System Trains for 24 Randomly Selected Instances



trains (100%); in the remaining instances, it is at least 60% (Figure 13). The weights in the objective function can be adjusted to obtain better preferential allocation at the cost of higher arrival and departure delays.

We illustrate the results for a DTW with the start time of 232 minutes and a duration of eight minutes. There are three system trains (trains 7, 6, and 9) at the start of this DTW (Table 5). The departure system train, train 6, departs from allocated platform 6 and moves through subpaths 1 and 2, which are represented by a corresponding set of crossovers. The occupancy intervals for platform and subpaths 1 and 2 for this train are [213–233), [233–237), and [237–243), respectively. The subpaths and their occupancy intervals for departure system trains, trains 6 and 9, represent decisions (A.7)–(A.10) of the ISDP taken in an earlier DTW.

In this DTW, we have six arrival decision trains and no departure decision train. The model provides platform allocation and subpaths 1 and 2 for arrival decision trains and their occupancy intervals (Table 6). For example, arrival decision train 12 has been allocated platform 10 and subpaths 1 and 2, which are represented by corresponding sets of crossovers.

The occupancy durations of subpath 1, subpath 2, and the platform are [245–251), [251–256), and [256–276), respectively. The platform, subpath allocation, and their occupancy intervals represent decisions (A.1)–(A.6) of the ISDP for arrival decision trains taken in the current DTW.

Model dispatching decisions are pictorially represented in a nonscaled version of the station layout, which includes only those resources allocated to decision or system trains (Figure 14). The platforms and subpaths of any pair of trains comprising decision trains, system trains, or both do not simultaneously overlap in time and space.

There are nine subpath overlaps in the space dimension. These involve two or three trains; however, the occupancy intervals are disjoint in the time domain (Table 7). For example, overlap "B" has a subpath overlap in space for arrival system train 7 and arrival decision trains 10 and 13. The subpath occupancy intervals of these trains are disjoint and are given by [229–235), [235–241), and [251–256), respectively.

The results demonstrate clearly that the ISDP model allocates unary resources to trains in spatial and temporal dimensions while respecting the limitations of unary resource assignments.

Conclusions

In this paper, we solved the station dispatching problem for a large-terminal station with specific operational protocols, such as controlling train movements at two signals, minimizing departure delays,

Figure 12. (Color online) Outstation Average Departure Delays and Traffic Density Relative to the Number of Arrival Decision, Departure Decision, and System Trains for 24 Randomly Selected Instances

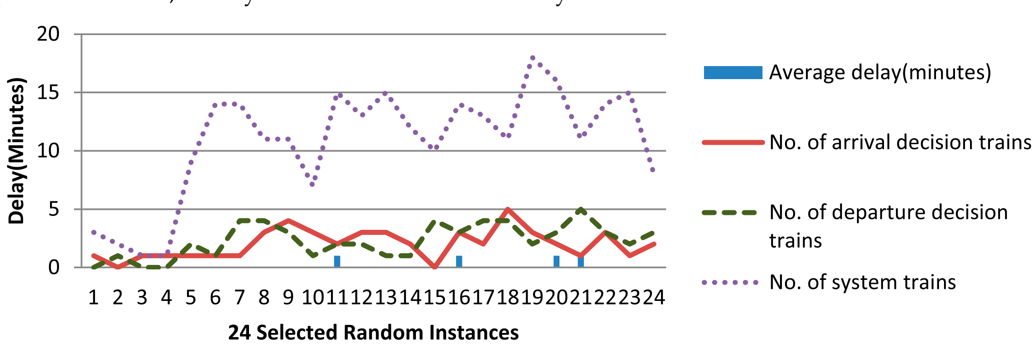
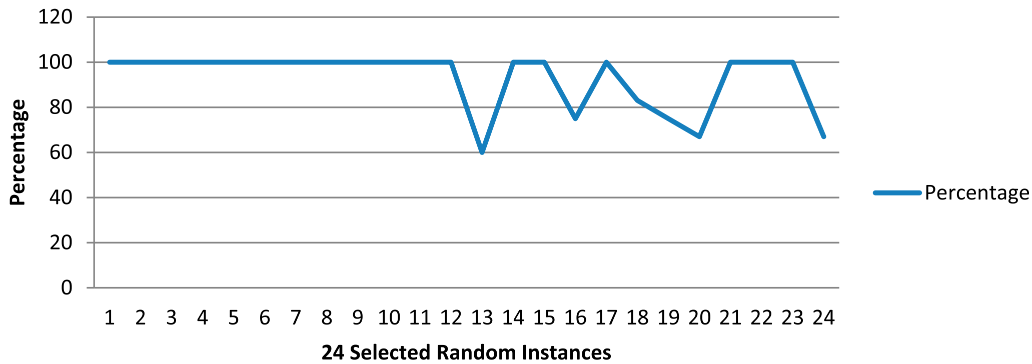


Figure 13. (Color online) Percentage of All Arrival Decision Trains Allocated First-Preference Platforms in the Model Solution for 24 Randomly Selected Instances



and incorporating operational methods to reduce the impact of external delays.

Our computational framework demonstrates the effective modeling of operational constraints using conditional time-interval variables and specialized constraints of CP. The model successfully achieves the objective of providing best-feasible solution(s) for our ISDP, thus resulting in a reduction of more than 50% in station dispatching delay. Although the delay cannot be eliminated completely because of infrastructural bottlenecks, the model has a potential to generate additional reductions if it is integrated into Howrah Station's information control system; however, doing so would require extensive modifications to the existing control system. The Howrah Station authority has reviewed and validated the model and its results for its suitability as a decision support tool.

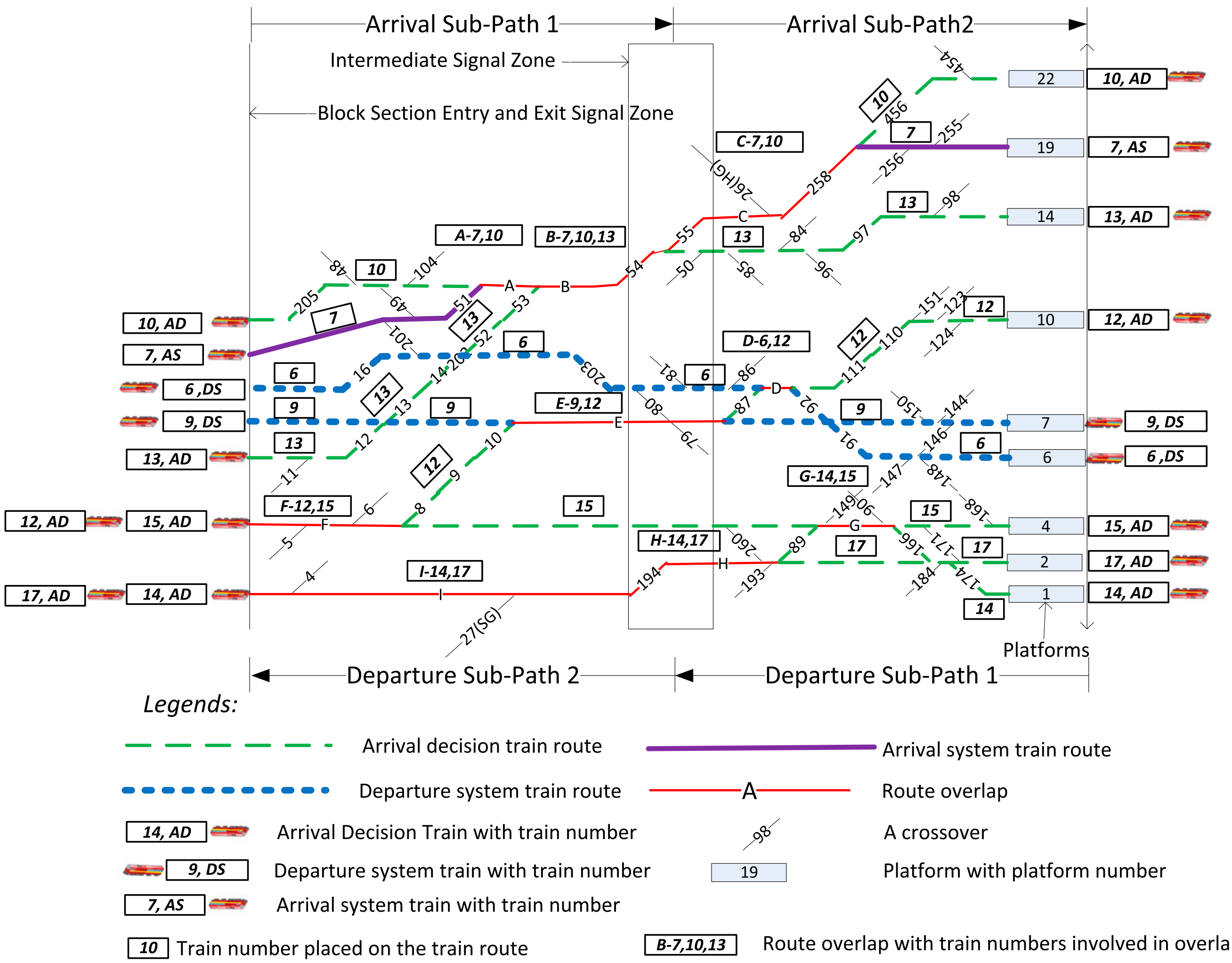
Table 5. Details of Arrival (Arr) and Departure (Dep) System Trains Show Allocated Platform Number, Platform Occupancy Interval, Subpaths 1 and 2, and Their Occupancy Intervals for a Dispatching Time Window Starting at 232 Minutes

Train no.	Train type	Platform	Platform occupancy interval	Subpath	Set of crossovers for subpath								Subpath occupancy interval
7	Arr	19	[240–260)	1	201N	49N	51R	53N	54R	55R			[229–235)
				2	26(Hg)N	258R	256N	255N	456N				[235–240)
6	Dep	6	[213–233)	1	148N	146N	147N	91R	92R	86N	87N	81N	[233–237)
				2	203R	52N	202N	201N	16R				[237–243)
9	Dep	7	[230–235)	1	146N	144N	91N	150N	92N	87N	79N	80N	[235–239)
				2	10N	13N	12N						[239–245)

Table 6. Details of Model Solution for Decision Trains Show Allocated Platform Number, Platform Occupancy Interval, Subpaths 1 and 2, and Their Occupancy Intervals for a Dispatching Time Window Starting at 232 Minutes

Train no.	Platform	Platform occupancy interval	Subpath	Set of crossovers for subpath										Subpath occupancy interval
10	22	[246–276)	1	48N	205R	49N	51N	53N	54R	55R	104N			[235–241)
			2	26(Hg)N	258R	456R	454N							[241–246)
12	10	[256–276)	1	5N	6N	8R	9R	10R	80N	79N				[245–251)
			2	87R	92N	93N	111R	110R	151N	123N	124N			[251–256)
13	14	[256–281)	1	11N	12R	13R	14R	202R	52R	53R	54R	55N	50N	[245–251)
			2	84N	85N	96N	97R	98N						[251–256)
14	1	[245–255)	1	4N	27(SG)N									[234–240)
			2	194R	260N	193N	89R	149N	90N	166R	184N	174R		[240–245)
15	4	[250–287)	1	5N	6N	8N								[239–245)
			2	260N	89N	90N	149N	166N	168N	171N				[245–250)
17	2	[255–262)	1	4N	27(SG)N									[240–250)
			2	194R	260N	193N	89N	166N	184N	174N				[250–255)

Figure 14. (Color online) Representation of a Model Solution for a DTW Starting at 232 Minutes Showing Train Route and Platform Allocation for Arrival Decision Trains, Routes for Arrival and Departure System Trains, and Route Overlaps for Two or More Trains



We considered fixed speed, and consequently, fixed travel times in the subpath resources. However, the ISDP model could also accommodate variable speeds and train travel times. Future research may consider CP modeling to solve the train dispatching problem between stations (also known as the line dispatching problem) and the integration of this solution to solve the station dispatching and railway timetabling problems.

Table 7. Details of Subpath Route Overlaps in Model Solution for a Dispatching Time Window Starting at 232 Minutes Show the Train Numbers Involved in the Overlap and the Disconnect in Their Subpath Occupancy Intervals

Overlap	Train nos.—Subpath occupancy interval
A	7—[229–235),10—[235–241)
B	7—[229–235),10—[235–241),13—[251,256)
C	7—[235–240),10—[241–246)
D	6—[233–237),12—[251–256)
E	9—[239–245),12—[245–251)
F	12—[245–251),15—[239–245)
G	14—[240–245),15—[245–250)
H	14—[240–245),17—[250–255)
I	14—[234–240),17—[240–250)

Acknowledgments

The authors thank the divisional railway manager of Howrah division, R. Badri Narayan, for his initiative and continued support and final solution verification of this work. They also thank Sachin Bansal, senior divisional operations manager, and Awdhesh Kumar Yadav, divisional signal and telecom engineer, for valuable inputs on the operational protocols. The authors acknowledge the efforts of the station master, Ajoy Pal, and the station staff for providing the data and explaining the current system in detail. They also express their gratitude to Sri Krishna Kumar, assistant professor, industrial systems and engineering, Indian Institute of Technology Kharagpur, for his ongoing support during this work.

Appendix. Integrated Station Dispatching Model

Please refer to the *Model Development* section for details regarding variables, domains, constraints, and the objective function. In Tables A.1 and A.2, we present the indices, subscripts, and notation we use throughout the paper.

Table A.1. Indices and Superscripts

Indices		Superscripts	
Letter	Index type	Letter	Definition
I	Trains	A	Arrival direction
J	Platforms	D	Departure direction
K	Paths	O	Outstation trains
L	Subpaths	Y	Yard trains

Table A.2. Notations Used in Our Mathematical Model

Symbol	Definition
T	Set of decision trains in a DTW
Q_i^a	Set of platforms in TPPL of the ith arrival decision train: $Q_i^a = \{q_{ij}^a\}, j \in 1, \ldots, \lvert Q_i^a \rvert$
$\underline{q}_{ij}^a$	Preference value of the jth platform in its TPPL of the ith arrival decision train
P_{ij}^a	Set of paths for the ith arrival decision train and jth platform in its TPPL
ρ_{ijkl}^a	The lth subpath of the kth path of the ith arrival decision train–q_{ij}^a pair
P_i^d	Set of paths for the ith departure decision train
ρ_{ikl}^d	The lth subpath of the kth path of the ith departure decision train
$\underline{g}_i^a$	Time of scheduled arrival of the ith arrival decision train at the entry-point signal
g_i^a, g_i^d	Times of arrival of the ith arrival decision train and of readiness for the departure of the departure decision train at the entry-point signal, respectively
h_l^a, h_l^d	Travel times in the lth arrival and departure subpath, respectively
r_i^a	Recovery time for the ith arrival decision train
$\Delta, \triangle$	Maximum and minimum allowable stay durations according to the schedule on the platform, respectively, for the arrival decision train
$\partial_{il}^a, \partial_{il}^d$	Dispatching delays in the release of a train at the starting signal of the lth subpath of the ith arrival decision train and departure decision train, respectively
$\bar{T}$	Set of system train at start of DTW
$\bar{q}_i^a$	Platform allocated to ith arrival system train
$\bar{\rho}_{il}^a, \bar{\rho}_{il}^d$	The lth subpaths allocated to the ith arrival system train and departure system train, respectively
$\bar{b}_{il}^a, \bar{b}_{il}^d$	Fixed time intervals of the lth subpath occupancy for the ith arrival system train and departure system train, respectively
$\bar{b}_i^q$	Fixed interval of platform occupancy for the ith arrival system train
w	Weight associated with dispatching delays and platform allocation

Variables

The three mandatory interval variables pertaining to present activities are listed in Table A.3.

Table A.3. Mandatory Interval Variables Pertaining to Present Activities

Notation	Description	Applicable for
z_{il}^a	Arrival subpath allocation for the arrival decision train	$i \in 1, \ldots, \lvert T^a \rvert, l \in \{1, 2\}$
y_i^a	Stopping platform allocation for the arrival decision train	$i \in 1, \ldots, \lvert T^a \rvert$
z_{il}^d	Departure subpath allocation for the departure decision train	$i \in 1, \ldots, \lvert T^d \rvert, l \in \{1, 2\}$

Interval Variables

The four optional interval variables pertaining to optional activities are listed in Table A.4.

Table A.4. Optional Interval Variables Pertaining to Optional Activities

Notation	Description	Applicable for	Alternative choices
$y_{ij}'^a$	Platform choice for the arrival decision train	$i \in 1, \ldots, \lvert T^a \rvert$	$j \in 1, \ldots, \lvert Q_i^a \rvert$
$z_{ijl}^{a\prime}$	Platform choice for the arrival subpath of arrival decision train	$i \in 1, \ldots, \lvert T^a \rvert, l \in \{1, 2\}$	$j \in 1, \ldots, \lvert Q_i^a \rvert$
$z_{ijkl}''^a$	Arrival subpath choice for the arrival decision train	$i \in 1, \ldots, \lvert T^a \rvert, l \in \{1, 2\}, j \in 1, \ldots, \lvert Q_i^a \rvert$	$k \in 1, \ldots, \lvert P_{ij}^a \rvert$
$z_{ikl}'^d$	Departure subpath choice for the departure decision train	$i \in 1, \ldots, \lvert T^d \rvert, l \in \{1, 2\}$	$k \in 1, \ldots, \lvert P_i^d \rvert$

Domains

The range of execution status $x(v)$ for all interval variables is defined by logical execution constraints (A.1)–(A.7) below. The range of the start and end times of optional variables is enumerated in Table A.5. The start time, end time, and duration of the mandatory interval variable are defined by alternative constraints (A.15)–(A.18).

Table A.5. Range of Start and End Times of Optional Interval Variables

	Range of start time (s)		Range of end time (e)	
Interval variable	e_{st}	l_{st}	e_{et}	l_{et}
$z''^{a}_{ijkl}\|_{l=1}$	g^a_i	$g^a_i + \partial^a_{il}\|_{l=1}$	$g^a_i + h^a_l\|_{l=1}$	$g^a_i + h^a_l\|_{l=1} + \sum_{l=1}^{l=2} \partial^a_{il}$
$z''^{a}_{ijkl}\|_{l=2}$	$g^a_i + h^a_l\|_{l=1}$	$g^a_i + h^a_l\|_{l=1} + \sum_{l=1}^{l=2} \partial^a_{il}$	$g^a_i + \sum_{l=1}^{l=2} h^a_l$	$g^a_i + \sum_{l=1}^{l=2} h^a_l + \sum_{l=1}^{l=2} \partial^a_{il}$
y'^{a}_{ij}	$g^a_i + \sum_{l=1}^{l=2} h^a_l$	$g^a_i + \sum_{l=1}^{l=2} h^a_l + \sum_{l=1}^{l=2} \partial^a_{il}$	$\underline{g}^a_i + \sum_{l=1}^{l=2} h^a_l + r^a_i + \Delta$	$g^a_i + \sum_{l=1}^{l=2} h^a_l + \sum_{l=1}^{l=2} \partial^a_{il} + \triangle$
z'^{a}_{ijl}	$g^a_i + \sum_{l=1}^{l=2} h^a_l$	$g^a_i + \sum_{l=1}^{l=2} h^a_l + \sum_{l=1}^{l=2} \partial^a_{il}$	$\underline{g}^a_i + \sum_{l=1}^{l=2} h^a_l + r^a_i + \Delta$	$g^a_i + \sum_{l=1}^{l=2} h^a_l + \sum_{l=1}^{l=2} \partial^a_{il} + \triangle$
$z'^{d}_{ikl}\|_{l=1}$	g^d_i	$g^d_i + \partial^d_{il}\|_{l=1}$	$g^d_i + h^d_l\|_{l=1}$	$g^d_i + h^d_l\|_{l=1} + \sum_{l=1}^{l=2} \partial^d_{il}$
$z'^{d}_{ikl}\|_{l=2}$	$g^d_i + h^d_l\|_{l=1}$	$g^d_i + h^d_l\|_{l=1} + \sum_{l=1}^{l=2} \partial^d_{il}$	$g^d_i + \sum_{l=1}^{l=2} h^d_l$	$g^d_i + \sum_{l=1}^{l=2} h^d_l + \sum_{l=1}^{l=2} \partial^d_{il}$

The range of duration of optional interval variables is enumerated in Table A.6.

Table A.6. Range of Duration of Optional Interval Variables

	Range of duration (d)	
Interval variable	$d_{\min}$	$d_{\max}$
$z''^{a}_{ijkl}\|_{l=1}$	h^a_l	$h^a_l + \partial^a_{il}\|_{l=2}$
$z''^{a}_{ijkl}\|_{l=2}$	$h^a_l\|_{l=2}$	$h^a_l\|_{l=2}$
y'^{a}_{ij}	$\triangle$	$r^a_i + \Delta$
y''^{a}_{ijl}	$\triangle$	$r^a_i + \Delta$
$z'^{d}_{ikl}\|_{l=1}$	h^d_l	$h^d_l + \partial^d_{il}\|_{l=2}$
$z^{d\prime}_{ikl}\|_{l=2}$	$h^d_l\|_{l=2}$	$h^d_l\|_{l=2}$

Constraints

Logical execution constraints impose conditions on the execution status of an interval variable v. Constraint *setPresent*(v) for present activities and *setOptional*(v) for optional activities impose the conditions $x(v)$=1 and $x(v) \in \{0,1\}$, respectively.

$$setPresent(y^a_i) \qquad \forall i \in 1,\dots,|T^a|; \tag{A.1}$$

$$setPresent(z^a_{il}) \qquad \forall i \in 1,\dots,|T^a|, l \in \{1,2\}; \tag{A.2}$$

$$setPresent(z^d_{il}) \qquad \forall i \in 1,\dots,|T^d|, l \in \{1,2\}; \tag{A.3}$$

$$setOptional(y'^a_{ij}) \qquad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|; \tag{A.4}$$

$$setOptional(z'^a_{ijl}) \qquad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|, l \in \{1,2\}; \tag{A.5}$$

$$setOptional(z''^a_{ijkl}) \quad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|, k \in 1,\dots,|P^a_{ij}|, l \in \{1,2\}; \tag{A.6}$$

$$setOptional(z'^d_{ikl}) \quad \forall i \in 1,\dots,|T^d|, k \in 1,\dots,|p^a_i|, k \in 1,\dots,|P^a_{ij}|, l \in \{1,2\}. \tag{A.7}$$

Logical constraint *ifThen* (v_1, v_2) between two conditional time interval variables v_1 and v_2 represents the condition$\neg exec(v_1) \vee exec(v_2)$; the term $exec(v)$ represents the execution of interval variable v.

$$ifThen(y'^a_{ij}, z'^a_{ijl}) \qquad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|, l \in \{1,2\}; \tag{A.8}$$

$$ifThen(z''^a_{ijkl_1}, z''^a_{ijkl_2}) \quad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|, k \in 1,\dots,|P^a_{ij}|, l_1 \in \{1,2\}, l_2 \in \{1,2\}, l_1 \neq l_2; \tag{A.9}$$

$$ifThen(z'^d_{ikl_1}, z'^d_{ikl_2}) \quad \forall i \in 1,\dots,|T^d|, k \in 1,\dots,|P^d_i|, l_1 \in \{1,2\}, l_2 \in \{1,2\}, l_1 \neq l_2. \tag{A.10}$$

The temporal precedence $startAtEnd(v_1, v_2, u)$ constraint between two interval variables $v_1$and v_2 imposes the condition on v_2 to start with a delay of u time units after the end of v_1. Mathematically, $startAtEnd(v_1, v_2, u) \Rightarrow start(v_1) + u == end(v_2)$.

$$startAtEnd(z''^a_{ijkl_2}, z''^a_{ijkl_1}, 0) \quad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|, k \in 1,\dots,|P^a_{ij}|, l_1 = 1, l_2 = 2; \tag{A.11}$$

$$startAtEnd(z'^d_{ikl_2}, z'^d_{ikl_1}, 0) \quad \forall i \in 1,\dots,|T^d|, k \in 1,\dots,|P^d_i|, l_1 = 1, l_2 = 2; \tag{A.12}$$

$$startAtEnd(z'^a_{ijl_2}, z'^a_{ijl_1}, 0) \quad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|, l_1 = 1, l_2 = 2; \tag{A.13}$$

$$startAtEnd(y'^a_{ij}, z'^a_{ijl}, 0) \qquad \forall i \in 1,\dots,|T^a|, j \in 1,\dots,|Q^a_i|, l = 2. \tag{A.14}$$

The *alternative* (v_0, V) constraint between an interval variable v_0 and a set of interval variables represented by V imposes the condition that if the interval variable $v_{0,}$ referred to as the master interval variable, is present in the solution, then only one of the interval variables from V is present in the solution, and the start time, end time, and duration of v_0 is same as the corresponding time of the present interval variable from V.

$$alternative\ (y_i^a, \{y_{ij}'^a \mid j \in 1, \ldots, |Q_i^a|\}) \qquad \forall i \in 1, \ldots, |T^a|; \tag{A.15}$$

$$alternative\ (y_{il}'^a, \{z_{ijl}'^a \mid j \in, 1 \ldots, |Q_i^a|\}) \qquad \forall i \in 1, \ldots, |T^a|, l \in \{1, 2\}; \tag{A.16}$$

$$alternative\ (z_{ijl}'^a, \{z_{ijkl}''^a \mid k \in 1, \ldots, |P_{ij}^a|\}) \quad \forall i \in 1, \ldots, |T^a|, j \in 1, \ldots, |Q_i^a|, l \in \{1, 2\}; \tag{A.17}$$

$$alternative\ (z_{il}^d, \{z_{ikl}'^d \mid k \in 1, \ldots, |P_i^d|\}) \qquad \forall i \in 1, \ldots, |T^d|, l \in \{1, 2\}. \tag{A.18}$$

The *noOverlap* constraint applies on interval variables associated with two activities A_i and A_j considered for allocation of in-conflict resources. It is mathematically expressed as $(e_i \le s_j) \vee (e_j \le s_i)$, where s_i, s_j and e_i, e_j represent the start and end times of variables, respectively. The *noOverlap* constraint is included for all possible allocations of a pair of in-conflict resources. Two platforms, $q_1$and q_2, are said to be in conflict if and only if $q_1 = q_2$. Two subpaths, $z_1$and z_2, are said to be in conflict if and only if they have any common crossover.

Let subpath $z = \{c_i \mid i \in 1, \ldots, m\}$, where c_i is a crossover and m is the number of crossovers. An in-conflict subpath pair is represented by $z_i Inc\ ; z_j \Leftrightarrow z_i \bigcap z_j \ne \varnothing$, where $z_i = \{c_i \mid i \in 1, \ldots, m_i\}$, $;z_j = \{c_j \mid j \in 1, \ldots, m_j\}$, and m_i and m_j are the number of crossovers in subpaths z_i and $;z_j$, respectively.

noOverlap Constraints for Allocation of In-Conflict Platforms. For a train pair consisting of two different arrival decision trains,

$$noOverlap(y_{i_1 j_1}'^q, y_{i_2 j_2}'^q)_{i_1 \ne i_2} \Leftrightarrow q_{i_1 j_1}^a\ Inc\ q_{i_2 j_2}^a \quad \forall i_1, i_2 \in 1, \ldots, |T^a|, j_1 \in 1, \ldots, |Q_{i_1}^a|, j_2 \in 1, \ldots, |Q_{i_2}^a|. \tag{A.19}$$

For a train pair consisting of an arrival decision train and an arrival system train,

$$noOverlap(y_{i_1 j_1}'^q, \bar{b}_i^q) \Leftrightarrow q_{i_1 j_1}^a\ Inc\ \bar{q}_i^a \quad \forall i_1 \in 1, \ldots, |T^a|, j_1 \in 1, \ldots, |Q_{i_1}^a|, i_2 \in 1, \ldots, |\bar{T}^a|. \tag{A.20}$$

noOverlap Constraints for Allocation of In-Conflict Subpaths. For a train pair consisting of two different arrival decision trains,

$$noOverlap(z_{i_1 j_1 k_1 l_1}''^a, z_{i_2 j_2 k_2 l_2}''^a)_{i_1 \ne i_2} \Leftrightarrow \rho_{i_1 j_1 k_1 l_1}^a\ Inc\ \rho_{i_1 j_1 k_1 l_1}^a \quad \forall i_1, i_2 \in 1, \ldots, |T^a|, j_1 \in 1, \ldots, |Q_{i_1}^a|, j_2 \in 1, \ldots, |Q_{i_2}^a|, k_1 \in 1, \ldots, |P_{i_1 j_1}^a|, k_2 \in 1, \ldots, |P_{i_2 j_2}^a|, l_1, l_2 \in \{1, 2\}. \tag{A.21}$$

For a train pair consisting of two different departure decision trains,

$$noOverlap(z_{i_1 k_1 l_1}'^d, z_{i_2 k_2 l_2}'^d)_{i_1 \ne i_2} \Leftrightarrow \rho_{i_1 k_1 l_1}^d\ Inc\ \rho_{i_2 k_2 l_2}^d \quad \forall i_1, i_2 \in 1, \ldots, |T^a|, k_1 \in 1, \ldots, |P_{i_1}^d|, k_2 \in 1, \ldots, |P_{i_2}^d|, l_1, l_2 \in \{1, 2\}. \tag{A.22}$$

For a train pair consisting of an arrival decision train and a departure decision train,

$$noOverlap(z_{i_1 j_1 k_1 l_1}''^a, z_{i_2 k_2 l_2}''^d) \Leftrightarrow \rho_{i_1 j_1 k_1 l_1}^a\ Inc\ \rho_{i_2 k_2 l_2}^d \quad \forall i_1 \in 1, \ldots, |T^a|, i_2 \in 1, \ldots, |T^d|, j_1 \in 1, \ldots, |Q_{i_1}^a|, k_1 \in 1 \ldots |P_{i_1 j_1}^a|, k_2 \in 1, \ldots, |P_{i_2}^d|, l_1, l_2 \in \{1, 2\}. \tag{A.23}$$

For a train pair consisting of an arrival decision train and an arrival system train,

$$noOverlap(z_{i_1 j_1 k_1 l_1}''^a, \bar{b}_{i_2 l_2}^a) \Leftrightarrow \rho_{i_1 j_1 k_1 l_1}^a\ Inc\ \bar{\rho}_{i_2 l_2}^a \quad \forall i_1 \in 1, \ldots, |T^a|, i_2 \in 1, \ldots, |\bar{T}^a|, j_1 \in 1, \ldots, |Q_{i_1}^a|, k_1 \in 1, \ldots, |P_{i_1 j_1}^a|, l_1, l_2 \in \{1, 2\}. \tag{A.24}$$

For a train pair consisting of an arrival decision train and a departure system train,

$$noOverlap(z_{i_1 j_1 k_1 l_1}''^a, \bar{b}_{i_2 l_2}^d) \Leftrightarrow \rho_{i_1 j_1 k_1 l_1}^a\ Inc\ \bar{\rho}_{i_2 l_2}^d \quad \forall i_1 \in 1, \ldots, |T^a|, i_2 \in 1, \ldots, |\bar{T}^d|, j_1 \in 1, \ldots, |Q_{i_1}^a|, k_1 \in 1, \ldots, |P_{i_1 j_1}^a|, l_1, l_2 \in \{1, 2\}. \tag{A.25}$$

For a train pair consisting of a departure decision train and an arrival system train,

$$noOverlap(z_{i_1 k_1 l_1}'^d, \bar{b}_{i_2 l_2}^a) \Leftrightarrow \rho_{i_2 k_2 l_2}^d\ Inc\ \bar{\rho}_{i_2 l_2}^a \quad \forall i_1 \in 1, \ldots, |T^d|, i_2 \in 1, \ldots, |\bar{T}^a|, k_1 \in 1 \ldots |P_{i_1}^d|, l_1, l_2 \in \{1, 2\}. \tag{A.26}$$

For a train pair consisting of a departure decision train and a departure system train,

$$noOverlap(z_{i_1 k_1 l_1}'^d, \bar{b}_{i_2 l_2}^d) \Leftrightarrow \rho_{i_1 k_1 l_1}^d\ Inc\ \bar{\rho}_{i_2 l_2}^d \quad \forall i_1 \in 1, \ldots, |T^d|, i_2 \in 1, \ldots, |\bar{T}^d|, k_1 \in 1, \ldots, |P_{i_1}^d|, l_1, l_2 \in \{1, 2\}. \tag{A.27}$$

Objective Function

The arrival delay for the ith arrival decision train at the platform is $\sum_{l=1}^{l=2} \partial_{il}^a$.

The departure delay for the ith departure decision train at the platform is $\partial_{il}^d|_{l=1}$.

The station authority wishes to minimize the arrival and departure delay at the platform only, which is also an appropriate measure of the total station dispatching delay. This naturally excludes the delay in releasing the train at the intermediate signal in the departure direction. The system can easily recover from this delay after the train moves out of the Howrah Station block section limit. Nevertheless, our model implicitly reduces this delay to minimize other significant delays, as we describe above.

The contribution of the ith arrival decision train platform allocation to the objective function is $w^q \times x(v_{ij}^q) \times \underline{q}_{ij}^a$.

The objective function is the weighted sum of delay and platform allocation terms over each category of train movement—outstation arrival and yard arrival for arrival delay, outstation departure, and yard departure for departure delay, and platform allocation.

$$O = \min\left(\left(w^{ao} \times \sum_{i=1}^{i=|T^{ao}|} (\partial^a_{il}|_{l=1} + \partial^a_{il}|_{l=2})\right)\right. + \left(w^{ay} \times \sum_{i=1}^{i=|T^{ay}|} (\partial^a_{il}|_{l=1} + \partial^a_{il}|_{l=2})\right) + \left(w^{do} \times \sum_{i=1}^{i=|T^{do}|} \partial^d_{il}|_{l=1}\right) + \left(w^{dy} \times \sum_{i=1}^{i=|T^{dy}|} \partial^d_{il}|_{l=1}\right) + \left.\left(w^{q} \times \sum_{i=1}^{i=|T^{a}|} \partial^q_{i}\right)\right). \quad \text{(A.28)}$$

References

Balas E (1985) On the facial structure of scheduling polyhedral. Cottle RW, ed. *Mathematical Programming Essays in Honor of George B. Dantzig*, Part I (Springer, Berlin), 179–218.

Barták R, Salido MA, Rossi F (2010) New trends in constraint satisfaction, planning, and scheduling: A survey. *Knowledge Engrg. Rev.* 25(3):249–279.

Billionet A (2003) Using integer programming to solve the train-platforming problem. *Transportation Sci.* 37(2):213–222.

Brännlund U, Lindberg O, Andreas N, Nilsson J-E (1998) Railway timetabling using Lagrangian relaxation. *Transportation Sci.* 32(4): 358–369.

Bron C, Kerbosch J (1973) Algorithm 457: Finding all cliques of an undirected graph. *Comm. ACM* 16(9):575–577.

Burkolter DM (2005) Capacity of railways in station areas using Petri nets. Doctoral dissertation, Swiss Federal Institute of Technology, Zurich.

Cacchiani V, Caprara A, Fischetti M (2012) A Lagrangian heuristic for robustness, with an application to train timetabling. *Transportation Sci.* 46(1):124–133.

Caimi G, Chudak F, Fuchsberger M, Laumanns M, Zenklusen R (2011) A new resource-constrained multicommodity flow model for conflict-free train routing and scheduling. *Transportation Sci.* 45(2):212–227.

Caprara A, Galli L, Toth P (2011) Solution of the train platforming problem. *Transportation Sci.* 45(2):246–257.

Cardillo DL, Mione N (1998) k L-list λ colouring of graphs. *Eur. J. Oper. Res.* 106(1):160–164.

Chakroborty P, Vikram D (2008) Optimum assignment of trains to platforms under partial schedule compliance. *Transportation Res. Part B: Methodology* 42(1):169–184.

Laborie P, Jerome R (2008) Reasoning with conditional time-intervals. Wilson D, Lane HC, eds. *Proc. 21st Internat. FLAIRS Conf.* (AAAI Press, Menlo Park, CA), 555–560.

Laborie P, Jerome R, Shaw P, Vilím P (2009) Reasoning with conditional time-intervals. Part II: An algebraical model for resources. Lane HC, Guesgen HW, eds. *Proc. 22nd Internat. FLAIRS Conf.* (AAAI Press, Menlo Park, CA), 201–206.

Lamorgese L, Mannino C (2015) An exact decomposition approach for the real-time train dispatching problem. *Oper. Res.* 63(1):48–64.

Lamorgese L, Mannino C, Piacentini M (2016) Optimal train dispatching by Benders'-like reformulation. *Transportation Sci.* 50(3):910–925.

Mannino C, Mascis A (2009) Optimal real-time traffic control in metro stations. *Oper. Res.* 57(4):1026–1039.

Pellegrini P, Marliere G, Rodriguez J (2014) Optimal train routing and scheduling for managing traffic perturbations in complex junctions. *Transportation Res. Part B: Methodology* 59(January):58–80.

Rodriguez J (2007) A constraint programming model for real-time train scheduling at junctions. *Transportation Res. Part B: Methodology* 41(2):231–245.

Şahin G, Ahuja RK, Cunha CB (2010) Integer programming based solution approaches for the train dispatching problem. Report, Sabancı University, Istanbul.

Törnquist J, Persson JA (2007) N-tracked railway traffic re-scheduling during disturbances. *Transportation Res. Part B: Methodology* 41(3):342–362.

Zwaneveld PJ, Kroon LG, Romeijn HE, Salomon M, Dauzere-Peres S, Van Hoesel SPM, Ambergen HW (1996) Routing trains through railway stations: Model formulation and algorithms. *Transportation Sci.* 30(3):181–194.

Verification Letter

Dr. R. Badri Narayan, Divisional Railway Manager, Howrah Division, Indian Railways, Howrah-711101, writes:

"This is to verify that the manuscript titled as 'Station Dispatching Problem for a Large Terminal: A Constraint Programming Approach' has considered a real problem currently being faced by Howrah Division of Eastern Railway. Presently, the trains do face delays as they approach the Howrah Railway Station. We had provided the authors with the required data to capture the real system and also to conform to our operational protocols. The model is promising in the sense that it has incorporated all realistic aspects of the system including safety features of route overlap and isolation, and the authors have demonstrated through their computational framework that delays could potentially be reduced.

"The work has potential for implementation after bringing in necessary changes in the present electronic and communication infrastructure in our division which may take some time. I congratulate the authors on having taken on this complex problem to provide a solution."

Rajnish Kumar is currently an executive at Indian Oil Corporation Limited (IOCL) and also pursuing a PhD in operations research from the National Institute of Technology, Durgapur. He has a BTech from the Indian Institute of Technology, Delhi and MBA from the Narsee Monjee Institute of Management Studies, Mumbai. He has designed a standardized solution for "Retail Outlet Canopy Illumination" for IOCL. This work resulted in huge savings in energy costs and a reduction in IOCL's carbon footprint.

Goutam Sen is an assistant professor (tenure track) in the Department of Industrial and Systems Engineering, Indian Institute of Technology (IIT) Kharagpur, India. He received his BTech in computer science and engineering from Kalyani Government Engineering College, West Bengal, India, and PhD jointly from IIT Bombay (India) and Monash University (Australia). He has keen research interest in network design and transportation, operations research, and combinatorial optimization techniques.

Samarjit Kar is currently an associate professor and HOD in the Department of Mathematics, National Institute of Technology, Durgapur, India. He received his MSc degree from Calcutta University, Kolkata, India, in 1995 and a PhD degree in Inventory Management in Uncertain Environment from Vidyasagar University, West Midnapore, India, in 2001. His current research interests include operations research and optimization, soft computing, uncertainty theory, and financial modeling.

Manoj Kumar Tiwari has research and teaching interests in modeling the manufacturing processes and operations analysis in supply chain networks. Optimization, simulation, and computational intelligence are the main techniques he has adopted to automate the decision support system for complex and large-scale problems in manufacturing and logistics systems. In 2016, he received the "Most Influential Researcher Award" in the domain of Operations and Supply Chain Management.