DEMAND RESPONSIVE SCHEDULING OF RAIL CONTAINER TRAFFIC

Nakorn Indra-Payoong, Raymond S K Kwan, Les Proll
Transport Scheduling Unit, School of Computing, University of Leeds,
Leeds, LS2 9JT, UK
nakorn@ comp.leeds.ac.uk, rsk@ comp.leeds.ac.uk, lgp@ comp.leeds.ac.uk

Abstract

Timetables are of utmost importance to a rail business because a rail carrier's profitability is heavily influenced by its service offerings. A need for demand responsive schedules is obvious not only because there is a risk that some potential customers may turn away if the customer's preferred itinerary is not attainable, but also because the take-up of some services in a fixed schedule may be low and therefore not profitable. We focus our attention on the step of constructing schedules that match, or respond to, customer demand, whilst maintaining the minimum operating costs for the rail carrier. We formulate the rail scheduling problem as a constraint satisfaction problem and propose a constraint-based search algorithm to solve the problem. The model is tested on data from the Royal State Railway of Thailand with 184 customers. The results show that the schedules generated have the potential to reduce the rail carrier's operating costs, enhances the customer satisfaction through a demand responsive schedule.

Keywords: Rail container transport; Freight rail scheduling; Demand responsive schedule
Topic Area: D2 Freight Transport Demand Modelling

1. Introduction

In the past, the transportation of rail freight was considered not to be an efficient mode of transport particularly in terms of physical accessibility and cargo handling. Since the advent of containerisation in the mid 1940s, rail carriers have gained higher profitability by tailoring containerised freight and have become more competitive with other inland transport providers.

Container rail service differs from conventional freight rail in several important aspects. Because of the high costs of container handling equipment, container rail networks have relatively few and widely spaced terminals. Networks with around ten terminals are common and the network flows are relatively simple. A typical container makes few or no stops and may be transferred between trains only up to a few times on its journey. In addition, small lot sizes of shipment, frequent shipment, and demand for flexible service are important characteristics in the transportation of rail containers.

Even though container traffic has increased, the increase in market share of rail transport, particularly in short-haul and medium-haul, has not been successfully achieved. Therefore, there have been efforts to investigate the factors influencing modal choice. The results have shown that the frequency and reliability of service are the main factors influencing shippers' decisions on the choice of transport mode.

A rail carrier's profitability is influenced by the railway's ability to construct schedules for which supply matches customer demands. For the transportation of containerised freight, shippers can often choose between rail and truck. A need for responsive flexible schedules may become obvious not only because there is a risk that some potential customers may turn away if the customer's preferred itinerary is not attainable, but also because the take-up of some services in a fixed schedule may be low and therefore not profitable. In order to construct a profitable schedule, a rail carrier needs to engage in a decision-making process with multiple criteria and a number of operational constraints, which is very challenging.

There is a large body of literature on freight rail scheduling, using diverse modelling structures and solution approaches. A recent survey by Cordeau et al (1998) suggests most of them cater for fixed schedules. However, our model incorporates challenging practical situations which involve: (i) non-uniform arrivals with distinct target times, i.e., not all containers are available at the beginning of the planning time horizon and must be treated as distinct customer bookings, (ii) a demand responsive service providing the flexible schedules, and (iii) a probabilistic decrease in customer satisfaction with deviation from target time. Table 1 contrasts our proposed model against those in the literature.

Table 1 Comparison of our proposed model to related literature

Literatures	Non-uniform Demand	Demand Responsive	Customer Satisfaction	Solution Approach
Crainic (1986)	No	No	No	Heuristics decomposition
Haghani (1989)	No	No	No	Heuristics decomposition
Keaton (1992)	No	No	No	Lagrangian relaxation
Huntley and et al (1995)	No	No	No	Simulated annealing
Branlund et al (1998)	No	No	No	Lagrangian relaxation
Gorman (1998)	Yes	No	No	Tabu-enhanced genetic
Brucker et al (1999)	No	No	No	Two-phase local search
Gualda and Murgel (2000)	No	No	No	Heuristics
Newman and Yano (2000)	No	No	No	Heuristics decomposition
Yano and Newman (2001)	Yes	No	No	Heuristics decomposition
Our paper	Yes	Yes	Yes	Constraint-based search

A few attempts have been made to generate flexible train schedules, which may be categorised into two types according to how the overall demand is met. Huntley et al. (1995), Gorman (1998), and Arshad et al. (2000) aggregate customer demands with minimum operating costs through flexible scheduling. They do not propose to meet individual demands. Newman and Yano (2000), Yano and Newman (2001), and Kraft (2002) share the same spirit of our study by being responsive to individual demand. Their models satisfy the operational constraints fully for each customer. In contrast, our framework models customer satisfaction, computed from preferred and alternative booking times, which is considered as one of the business criteria. Therefore, some customers might not be given their most preferred booking times. This framework is a natural one for supporting decisions as a rail carrier can measure how well their customers are satisfied and the implications of satisfying these customers in terms of cost.

From Table 1, the solution approaches for freight rail scheduling problem may be classified into two groups: mathematical approaches and heuristic methods. However, these approaches are complicated and fully rely on problem-specific knowledge. Hence they are not flexible nor convenient to implement, in particular, when the rail business criteria and operational constraints in the rail optimisation model need to be changed. In this paper, we present an alternative way to solve the freight rail scheduling problem. Our solution algorithm learns implicitly from a search history and targets for the optimal schedule.

The remainder of this paper is organised as follows. In section 1, we describe our rail scheduling model. Hard and soft constraints are defined and a constraint satisfaction representation of the problem is given. In section 2, we present a constraint-based search algorithm to solve the problem. In section 3, we introduce techniques that can improve the existing container rail schedules. In section 4, a case study and experimental results are given. Finally, we conclude the paper with a discussion of our results in section 5.

2. Problem formulation

Container rail services are independent of one another in the sense that demands for a container movement in a specific service do not interact with the demands in any other services. In addition, complex networks with many container terminals are capital-intensive and not practical for custom procedures when containers are moved within an international context.

We consider the container rail service from a container seaport to an inland container depot (ICD) in which the weekly schedule is provided and revised every week. Once containers arrive at the seaport, they can be transported to their final destinations by rail or truck via an inland container depot, or directly by truck. This paper assumes an advance booking scheme as illustrated in Figure 1. It also assumes that all containers are homogeneous in terms of their physical dimensions, and they will be loaded on trains ready for any scheduled departure times. Note also that we consider a standard container, which is measured in Twenty-Equivalent Unit (TEU).

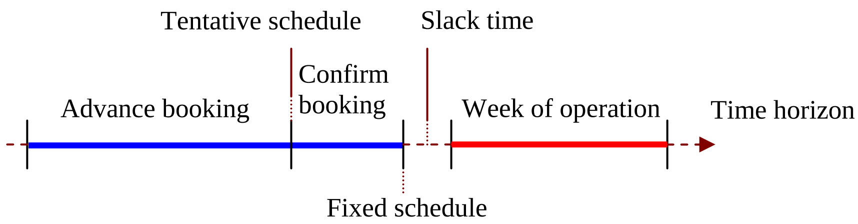

Figure 1 A short-term advance booking scheme

The day is divided into hourly slots for booking and scheduling. Customers are requested to state a preferred booking time or an earliest booking time in advance. A number of alternative booking times for each shipment may be specified, which might be judged from experience or estimated by the customer's delay time functions. These alternatives not only help a rail carrier consolidate customer demands to a particular train service with minimum total costs, but also provide flexible departure times for the customer's transport planning strategy.

It is noted that a preferred booking time and each alternative booking time may cover a few hours, which is illustrated in Table 2. This happens in practice because the service time needed to move containers from the loading point of a containership to the train container platform may vary. In addition, customers may have to allow more time for unexpected delays.

Table 2 Possible departure timeslots for customers

Departure	Customers (shippers)				
Timeslot	Customer 1	Customer 2	Customer 3	…	Customer *n*
.			P		
Mon: 0900 – 1000		P	A_1		
Mon: 1000 - 1100	P	P	A_1		A_3
.	P				A_3
Wed: 1800 - 1900		A_1	A_2		

P: preferred booking times, A: alternative booking times

It is also noted that there may be some customers that book the container rail service close to the end of a week; therefore their alternative booking times may fall into the following week. In this paper, the model takes preferred booking times for those customers and their alternatives are not considered.

2.1. Rail container scheduling model

A real-world optimisation problem tends to have a large number of constraints which may be hard or soft. In our problem, train capacity, service restrictions, and some customer requirements are hard constraints, whilst minimum number of trains, maximum customer satisfaction, minimum timeslot operating costs are soft constraints. Naturally, we can formulate the rail container scheduling problem as a constraint satisfaction problem (CSP) and then introduce a constraint-based search algorithm to solve this class of constraint satisfaction problem

We consider a problem faced by a typical large rail carrier; more than several thousands containers every week. To simplify our discussion, the following notation will be used:

Subscripts:

t : schedulable timeslot (departure time), $t = 1, 2, 3, \ldots, T$

j : customer, $j = 1, 2, 3, \ldots, M$

Set:

S_j : set of possible booking times for customer j

C_t : set of potential customers for departure timeslot t

R : set of service restrictions for departure timeslots

Decision variables:

x_t : 1, if a train departs in timeslot t, 0 otherwise

y_{tj} : 1, if customer j is served by the train departing in timeslot t, 0 otherwise

Parameters:

w_{tj} : customer j satisfaction score in departure timeslot t

r_t : train congestion cost in departure timeslot t

g_t : staff cost in departure timeslot t

P_2 : capacity of a train (number of containers)

N_j : demand of customer j (number of containers)

The optimisation criteria in the constraint satisfaction model are handled by transforming them into soft constraints. This is achieved by expressing each criterion as an inequality against a tight bound on its optimal value. As a result, such soft constraints are rarely satisfied.

A feasible solution for a CSP representation of the problem is an assignment to all decision variables in the model that satisfies all hard constraints, whereas an optimal solution is a feasible solution with the minimum total soft constraint violation. For a linear constraint satisfaction problem, the violation v_i of constraint i is defined as follows:

$$\sum_j a_{ij} x_j \begin{matrix} \geq \\ \text{or} \\ \leq \end{matrix} b_i \Rightarrow v_i = \max\left(0, \left| b_i - \sum_j a_{ij} x_j \right|\right) \quad (1)$$

where a_{ij} are coefficients, b_i is a numeric value and x_j are decision variables.

When all decision variables are assigned a value, the violation of hard and soft constraints can be tested. A quantified measure of the violation can be used to evaluate local moves for our solution algorithm (described in section 2).

2.1.1. Soft constraints (optimisation criteria)

The number of trains - the aim is to minimise the number of trains on a weekly basis, which is defined as:

$$\sum_{t=1}^{T} x_t \leq \theta; \tag{2}$$

where: θ is a lower bound on the number of trains, e.g., $\left\lceil (\sum_j N_j)/P_2 \right\rceil$

Customer satisfaction **–** the objective is to maximise the total customer satisfaction score. The satisfaction score is assigned values from a function of the delay time related costs. Each customer holds the highest score at a preferred booking time, the score then decreases probabilistically to the lowest score at the last alternative booking time, i.e., later than preferred departure times would cause a decrease in the future demand, and the rail carrier is expected to take a loss in future revenue. For the evaluation of a schedule, the probability of customer satisfaction is then multiplied by the demand. The customer satisfaction constraint can be expressed as:

$$\sum_{t=1}^{T} \left(\sum_{j \in C_t} \frac{w_{tj}}{100} y_{tj} N_j \right) \geq \Omega; \tag{3}$$

where: Ω is an upper bound on customer satisfaction, i.e., $\sum_j (\frac{W_j}{100} N_j)$; W_j is the maximum satisfaction score on a preferred booking time for customer *j*

Timeslot operating costs **–** the aim is to minimise the operating costs. A rail carrier is likely to incur additional costs in operating a demand responsive schedule, in which departure times may vary from week to week. This may include train congestion costs and staff costs. The train congestion cost reflects an incremental delay resulting from interference between trains in a traffic stream. The rail carrier calculates the marginal delay caused by an additional train entering a particular set of departure timeslots, taking into account the speed-flow relationship of each track segment. The over-time costs for crew and ground staff would also be paid when evening and night trains are requested. The time slot operating costs constraint is defined as:

$$\sum_{t=1}^{T} (r_t + g_t) x_t \leq (\lambda + \delta); \tag{4}$$

where: $(\lambda + \delta)$ is a lower bound on the timeslot operating costs, $\lambda = \sum_{t \in T_a} r_t$; T_a is the set of θ least train congestion costs, $\delta = \sum_{t \in T_b} g_t$; T_b is the set of θ least staff costs, θ is a lower bound on the number of trains.

2.2. Hard (operational) constraints

Maximum capacity **–** this constraint ensures the demand must not exceed the capacity of a train. The maximum capacity constraint is defined as:

$$x_t\left(\sum_{j=1}^{M} y_{tj} N_j\right) \leq P_2 \quad ;\forall t \tag{5}$$

Coverage constraint **–** it is a reasonable assumption that in practice customers do not want their shipment to be split in multiple trains, this constraint ensures that customer can only be served by one train. The coverage constraint is given as:

$$\sum_{t \in S_j} y_{tj} = 1; \quad \forall j \tag{6}$$

Timeslot consistency – this ensures that if timeslot *t* is selected for customer *j*, a train does depart at that timeslot. On the other hand, if departure timeslot *t* is not selected for customer *j*, a train may or may not run at that time. The constraint is defined as:

$$x_t \geq y_{tj}; \quad \forall j \in C_t \tag{7}$$

Service restriction - this is a set of banned departure times. The restrictions may be pre-specified so that a railway planner schedules trains to achieve a desirable headway or to avoid congestion at the container terminal. The constraint is defined as:

$$x_t = 0; \quad \forall t \in R \tag{8}$$

2.2.1. Implied constraints

The soft and hard constraints completely reflect the requisite relationships between all the variables in the model, i.e., the operational requirements and business criteria. Implied constraints, derivable from the above constraints, may be added to the model. Whilst implied constraints do not affect the set of feasible solutions to the model, they may have computational advantage in the solution algorithm as they reduce the size of the search space.

Timeslot covering – a timeslot covering constraint can be thought of as a set covering problem in which the constraint is satisfied if there is at least one departure timeslot x_t serving customer *j*; otherwise the algorithm assigns a fixed violation penalty v_s to that constraint. This constraint favours a combination of selected departure timeslots that covers all customers. Any quantification of v_s could be used. The timeslot covering constraint is defined as:

$$\sum_{j \in S_j} x_t \begin{cases} \geq 1, \text{violation} = 0 \\ \\ = 0, \text{violation} = v_s \end{cases} , \forall j \tag{9}$$

2.3. Customer satisfaction

In a highly competitive market, assessing customer satisfaction for the transport service is of great importance to a rail container carrier. A rail carrier could take advantage of the knowledge of customer satisfaction to improve its service and to strengthen its competitive position with respect to the other transport services. A rail carrier could increase the quality of service and market share by tailoring a service that satisfies individual customers. The rail schedule may be just one of the factors including cost, travel time, reliability, safety, and so forth. As customers have different demands, it is hard to find a single definition of what a good quality of service is. For example, some customers are willing to tolerate a delayed service in return for sufficiently low total shipping costs.

In this paper, we only investigate customer satisfaction with respect to the rail schedule. To acquire the customer satisfaction data, face-to-face interviews were carried out. The outline of the survey interview is tabulated in Table 3.

Table 3 The outline of the survey interview

Shipping Information
Type of company (shipping line, freight forwarder, MTO, .etc)
Type of container and commodity value per ton
Container density measured
Shelf life of commodity in days
Annual container volume shipped
Period of advance booking regularly used.
Modal Characteristics for Each Mode (Shipping Time)
Arrival time at seaport
Discharging time at seaport
Waiting time at the discharging point
Haulage time from the discharging point to main terminal
Waiting time at seaport terminal
Loading time at train/truck terminals
Travel time
Modal Characteristics for Each Mode (Shipping Costs)
Freight rate (TEU-ton-km) and Commodity rate factor
Terminal storage cost at seaport terminal/shipside (TEU-ton/day) (day = a consecutive 24-hour period)
Free time storage period at seaport terminal (days)
Reduction rate if containers are removed from terminal/shipside within a free time storage (day-percent)
Terminal handling charge per TEU-ton
Overhead cost for waiting time at seaport terminal/shipside (TEU-ton/day)
Departure times
Preferred train departure time
Alternative departure time I
Alternative departure time II
Others alternatives

This survey includes 184 customers currently using both rail and trucking services or using only rail but with a potential to switch their shipment to truck in the future. The containerised cargo is classified into four categories as follows:

1. Cargo type I (perishable consumer goods): food and beverages, dairy products, fruits and vegetables (24 customers)
2. Cargo type II (durable consumer goods): household products, and furniture (52 customers)
3. Cargo type III (intermediate products and raw materials): textile fibres, tobacco leaves, paper and paperboard, chemicals (67 customers)
4. Cargo type IV (capital goods and others): iron and steels, metal manufacture, non-electrical machinery and parts, construction materials (41 customers)

To quantify customer satisfaction, customer satisfaction functions are developed. These use customer characteristics, shipping information and modal characteristics as primary data. Total shipping costs associated with movement by modes are calculated as a percentage of commodity market price or value of containerised cargo, expressed in price per ton. Average shipping costs of the containerised cargo and the market price are summarised in Table 4.

Table 4 Modal cost percentages for each mode (×1000 Baht /ton)

Cargo Types -Cost	Cost /Unit Price		Market	Modal Cost Percentages		
	Truck	Rail	Price*	Truck	Rail	Δ
Freight rate (FR)						
Type I	2.21	1.55	25.00	8.84	6.20	2.64
Type II	6.71	2.96	68.00	9.87	4.35	5.52
Type III	10.45	7.56	87.20	11.98	8.67	3.31
Type IV	0.95	0.21	13.00	7.30	1.62	5.68
Terminal handling charge (THC)						
Type I	0.28	0.51	25.00	1.12	2.04	-0.92
Type II	0.57	1.04	68.00	0.84	1.53	-0.69
Type III	1.18	2.06	87.20	1.35	2.36	-1.01
Type IV	0.03	0.08	13.00	0.23	0.61	-0.38
Terminal storage charges (TC)						
(Within free time storage)	0	0		0	0	0
Overhead cost (OC)						
(Within free time storage)	0	0		0	0	0
Total shipping costs (TSC)						
Type I	2.49	2.06	25.00	9.96	8.24	1.72
Type II	7.28	4.00	68.00	10.70	5.88	4.82
Type III	11.63	9.62	87.20	13.34	11.03	2.31
Type IV	0.98	0.29	13.00	7.54	2.23	5.31

* Estimated by the Department of Business Economics, Ministry of Commerce, Thailand (2002)

We assume that all customers know a full set of shipping costs and can justify the modal preferences on a basis of accurately measured and understood costs. The freight rate may be adjusted by the relative costs that a customer may be willing to pay to receive superior service. For example, some customers may have higher satisfaction using a trucking service even if the explicit freight rate is higher; speed and reliability of the service may be particularly important if the containerised cargo has a short shelf life.

To determine customer satisfaction between modes, we assume that the difference between modal cost percentages is a normal distribution. The customer satisfaction is then derived from cumulative probability functions. Figure 2 illustrates the customer satisfaction function of cargo type I.

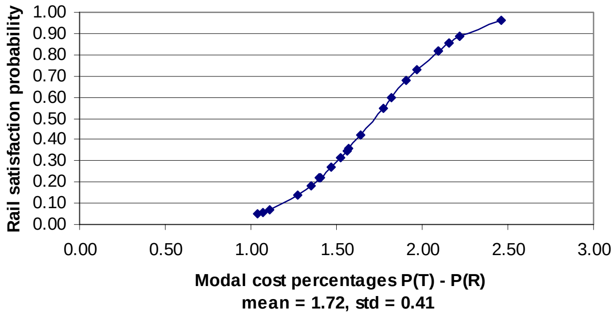

Figure 2 The customer satisfaction function of cargo type I

Figure 2 shows that if there is no difference between the modal cost percentages, customers tend to state their satisfaction on the service between rail and truck equally. The commodity that has a small difference between the modal cost percentages causes a high sensitivity in the total shipping costs. For instance, the commodity type I has an average of the difference between modal cost percentages 1.72; when the transport of containers are delayed by rail, it will result in an increase in the total shipping costs. For this type of commodity even a small cost increase can lower the customer satisfaction using container rail service quite substantially. This is due to a high sensitivity in the total shipping costs. An average difference between the cost percentages (mean value) could imply the lowest satisfaction for the container rail service. If the satisfaction is below this level, customers may turn to use a trucking service instead; otherwise, they would tolerate with the rail service. Nowadays, it is obvious that rail carriers would try to keep the customer satisfaction above this level.

Once the satisfaction function has been developed, a customer satisfaction score can be obtained from the modal satisfaction probability. This probability could be used to predict the market share between transport modes and to test the modal sensitivity when the rail schedule is changed.

The customer satisfaction score is on a percentage scale and all customers have a rail satisfaction score ranging from 0 to 100. Note that all customers currently using container rail service may already hold a certain level of satisfaction score regardless of taking the quality of rail schedule into account. Once the rail carrier has been chosen as a choice of transport mode and later the schedule is delayed, customers incur additional total shipping costs, i.e., terminal storage and overhead costs involved at the seaport. This would result in a decrease in customer satisfaction. An example of the calculation of customer satisfaction score is shown in Table 5.

Table 5 Customer satisfaction score

Shipping Lists	**Unit**	**Value**			
Commodity type	Type			IV	
Ship arrival time	Day: Time			Mon: 0900	
Discharging time	Hour			4	
Free time storage allowance	Day			3	
Reduction rate on THCs					
- Scheme (I)	Day (%)			1 - 25%	
- Scheme (II)	Day - %			2 - 20%	
- Scheme (III)	Day - %			3 - 10%	
Departure times	Day: Time				
- Preferred timeslot (P)				Mon: 1500-1700	
- Alternative (I)				Tue: 0900-1600	
- Alternative (II)				Wed: 0900-1600	
- Alternative (III)				Thu: 1600-2200	
Modal cost percentages (%)		$\mathbf{P^*}$	$\mathbf{A_1}$	$\mathbf{A_2}$	$\mathbf{A_3}$
- Trucking		9.51	9.51	9.51	9.51
- Rail		3.78	3.84	4.09	4.33
- Δ		5.73	5.67	5.42	5.18
The customer satisfaction score		86	75	60	37

P^* is a modal cost percentage when the first reduction rate scheme is applied

2.4. Generalised cost

For the evaluation of a schedule, a cost function taking into account the number of trains can be expressed in terms of operating costs; but it is hard to express customer satisfaction using a monetary unit. In this paper, we express the customer satisfaction on a rail scheduling service in terms of shipping costs related to the delay time. We introduce the term "virtual revenue loss" as a unit cost. This term is derived from the difference in probability of choosing the container rail service between the preferred timeslot and the alternatives. The probability is then multiplied by a demand and freight rate per demand unit. Therefore, a generalised cost function is the sum of the operating costs and the virtual revenue loss:

$$G = (\theta + v_1)FC + v_2 FR + (\lambda + \delta + v_3) \quad (10)$$

where: FC is a fixed cost of running a train; FR is a freight rate per ton-container; v_1, v_2, v_3 are penalty costs (soft violations) for the number of trains, customer satisfaction, and timeslot operating costs constraints respectively.

3. Solution algorithm

As the size of the rail container scheduling problem that needs to be routinely solved in industry is large, i.e., there are several thousand decision variables in our rail scheduling model, this makes an optimal train schedule unattainable within a reasonable run-time on a standard personal computer. In this section, we propose a constraint-based search algorithm that can generate near optimal schedules within a reasonable time.

3.1. Constraint-based search

The constraint-based search algorithm works with a complete assignment to all decision variables in the optimisation model, starting with an initial random assignment, in which some operational constraints in the model can be violated or some constraints are relaxed. In the iteration loop, the algorithm randomly selects a violated constraint. For example, the assigned train timeslot for which the demands exceed a train capacity or the assigned timeslot that is not consistent with the customer booking times. Having selected a violated constraint, the algorithm randomly selects one variable in that constraint and another variable from the search space. Then, two flip trials are performed, i.e., changing the current value of the variable into the other binary value. Suppose that G_j takes the value g_j at the start of the iteration so $A = (g_1, g_2, \ldots, g_m \mid h)$, where m is the total number of variables and h is the total violation of all hard constraints. Suppose further that G_1, G_2 are selected and that their flipped values are $\bar{g}_1, \bar{g}_2$ respectively. We look at the assignments $A_1 = (\bar{g}_1, g_2, \ldots, g_m \mid h_1)$, $A_2 = (g_1, \bar{g}_2, \ldots, g_m \mid h_2)$ and select the alternative with the smaller total hard violation. The total hard violation is compared between iterations.

```
proc constraint-based search
        input soft and hard constraints, nbIterations or timeLimit
        A := initial random assignment
        for i := 1 to nbIterations do
            C := select-violated-hard-constraint (A)
            flip C, A1, A2
                if (h1 < h2) then (A ← A1)
                else (A ← A2)
                if h = 0 then A is feasible
                   record solution A
                end if
                if h = 0 then reinitialise A
                end if
        end for
        output feasible solutions found
end proc
```

Figure 3 The constraint-based search procedure

Whenever all operational (hard) constraints are not violated, the algorithm stores the soft violation penalties as feasible objective values, together with the associated variable values. The algorithm continues until the iteration limit, *nbIterations* or the time limit, *timeLimit* is reached. The procedure of the constraint-based search is given in Figure 3.

3.2. Hierarchical constraint violation

The hierarchical violation scheme allows the search algorithm to apply specific weights to constraints and/or sets of decision variables. In our rail container scheduling model, constraints carry different weights.

We distinguish between soft and hard constraints. The search algorithm works solely on the hard constraints. Whenever all hard constraints are satisfied, soft constraint violations present feasible objective values in terms of the generalised cost.

3.2.1. Hard violations

Hard violation weights allow the search to give priority to satisfying particular sets of hard constraints and/or variables that lead to feasible schedules. Hard violation weights are applied to maximum capacity, timeslot consistency, and timeslot covering constraints.

For the violation of maximum capacity constraint, we consider that any number of containers in timeslot *t* exceeding train capacity of is penalised with the same violation v_m. An attempt to apply different violation weights to different number of exceeded containers on a train makes little sense because one can never guarantee whether the lower number of exceeded containers would be more likely to lead to feasible schedules. The violation for maximum capacity constraints is defined as:

$$x_t\left(\sum_{j=1}^{M} y_{tj} N_j\right)\begin{cases} \leq P_2, \text{violation} = 0 \\ > P_2, \text{violation} = v_m \end{cases} ;\forall t \qquad (11)$$

For the timeslot consistency constraint, the algorithm assigns the violation v_c if the train does not depart at timeslot *t* but there are some customers assigned to that timeslot. The violation for timeslot consistency constraints is defined as:

$$\sum_{l=1}^{L} y_{tl}\begin{cases} \leq x_t L, \text{violation} = 0 \\ > x_t L, \text{violation} = v_c \end{cases} ;\forall t \qquad (12)$$

where: *L* is the number of potential customers for timeslot *t*

3.2.2. Timeslot violations

The timeslot violation weights influence the search towards minimising a generalised cost whilst satisfying the operational constraints. The timeslot violation (v_t) depends on the possibility of assigning a particular timeslot on a schedule with a minimum generalised cost. It is noted that the total hard violation *h* is the sum of hard violations (v_m, v_c, v_s) and timeslot violation (v_t). When total hard violations (v_m, v_c, v_s) are satisfied, the timeslot violation (v_t) is set to zero.

An attempt to derive the timeslot violation weights in a monetary unit by trading off between objectives is not possible. This is because the train schedule is not a single departure timeslot but is a set of timeslots used; thus considering only a single timeslot separately from the others cannot represent a total cost for the rail carrier. However, as in practice some business criteria play more important roles than others, the way to solve the tradeoff between soft constraints for a single timeslot is to apply hierarchical violation weights to the rail business criteria.

4. The improved schedule

In this section, we describe the construction and use of the probabilistic learning method that steer the constraint-based search by inferring, from its recent violation history, the likely optimal value of chosen variable of interest. The quality of train schedules is improved using this technique.

4.1. Minimum train loading

The minimum train loading is used to ensure satisfactory revenue for the rail carrier and spreads out the size of shipment on train services. The carrier may need to set the minimum train loading as high as possible, ideally equal to the capacity of a train. The minimum train loading is directly related to the number of trains expected (serviced), which is defined as:

$$T_{\exp} = \left\lceil \sum_{j} N_j / P_1 \right\rceil \tag{13}$$

where: T_{exp} is the number of trains expected, N_j is the demand of customer *j*, P_1 is the minimum train loading

Apart from ensuring satisfactory revenue, the minimum train loading is a key feature of the performance of our solution algorithm. The higher the minimum train loading, the more constrained the problem is and hence the number of feasible schedules decreases. This increases the usefulness of the probabilistic learning method (described in section 3.2).

From the computational point of view, it is hard to guarantee that the train schedule constrained by the highest minimum train loading, derivable from a lower bound on the number of trains θ, would be feasible. If such a feasible schedule exits, it would imply that the schedule is approximately optimal. With this information, our search algorithm can be advised to move towards a near optimal schedule directly. However, it is still very hard to find the optimal schedule without the aid of the probabilistic learning method.

Few techniques for proving the existence of feasible solutions within a given integer optimisation problem have been proposed, such as the interval Newton method (see Kearfott (1998)). Nevertheless, practical implementations of these techniques have not yet been described, and for a large-scale optimisation problem, the methods are awkward and fail to prove feasibility.

In this paper, we can derive the minimum train loading from the problem specific domain. If the minimum train loading is relatively low the possibility of having no feasible schedule is also low; however, if the minimum train loading is so high a feasible train schedule would unlikely exist. Ideally, suppose that there are ten customers with the uniform demands of 20 containers, and the capacity of a train is 60. Therefore, the averaged demand is 20 and the standard deviation is 0, and this standard deviation indicates the degree of schedule having no feasible solution *r*. Suppose further that, if the demands are consolidated with the maximum train utilisation, the minimum train loading P_1 = 50 will ensure the existence of feasible schedule, in the other words, *r* is zero.

However, the uniform customer demand never occurs in practice; in addition, non-uniform arrival times on the demands and different customer’s booking preferences could make the consolidation by averaged demand underestimated. This is because some customer’s booking times may only be consistent with a particular set of departure timeslots. In this paper, we partition customer demands into a day timeslot and the demand is averaged by each day. Then, the mean for averaged demand for the total classified days is calculated. Note that, each classified day can have the same demands due to the number of customer's booking preferences given within a week of operation. Now, we can find the minimum train loading by first defining a certain level of *r*, e.g., r = 20% means that the chance of schedule having no feasible solution is 20%, in other words, the confident level is 80% on the existence of a feasible schedule. The minimum train loading P_1 is defined as follows:

$$P_1 = \sum_j N_j / \lceil M / T \rceil \tag{14}$$

where: M is the total number of customers, $T = \lfloor P_2 / \mu_r \rfloor$, μ and σ are mean and standard deviation of averaged demand for the total classified days, μ_r is the mean with a percentage of r, $\mu_r = [(\mu + \sigma) \times (100 - r)] / 100$

It is noted again that the algorithm assigns the number of trains departing in timeslots according to the number of trains expected T_{exp}, and this rule is maintained during the search process. With P_1, whenever all hard constraints are satisfied (a feasible schedule is obtained), the minimum train loading is increased by removing one train from the current state of the feasible solution, i.e. $T_{exp} = T_{exp} - 1$.

4.2. Probabilistic learning method

In this section, we develop the value choice model that can predict a likely optimal value for a decision variable in the optimisation model. The choice model is based on the logit method and the proportion method.

For the container rail optimisation model, decision variables may take several different values across the set of feasible schedules. Thus, it is very difficult to predict a good consistent value for the variables. However, when the problem is severely constrained and has few feasible solutions, this can be done by setting a high value of minimum train loading as described in section 3.1, it may well be that some decision variables would take more consistent values in all the feasible schedules during the search.

Once a variable has been selected, the search algorithm has to choose a value that is likely to lead to a smaller total degree of hard violation in a complete assignment. In our solution algorithm, two variables are considered at each trial iteration. The first variable is chosen in a violated constraint and considered as a variable of interest, the second variable is randomly selected from the search space and used for a comparison. Flipping the first variable might result in a reduction in total hard violation. However, it might be that flipping the second variable would result in even more reduction in the constraint violation. Therefore, the flipped value of the first variable is not accepted and the unflipped value is recorded for the lowest total hard violation.

As illustrated in Table 6, the history of constraint violation is recorded whilst the algorithm performs trials of flip assignments. Each time a decision (binary) variable is trial flipped in its value, the total degree of hard constraint violation associated with its two possible values is compared.

Table 6 Constraint violation history

Flip Trial	Variable x_1				Hidden Combination					x_1^*
	Current		Flipped		Variable	Current		Flipped		
	Value	h_1'	Value	h_1	x_i	Value	h_2	Value	h_2	
1	1	26	0	22	15	1	26	0	36	0
2	1	20	0	12	9	0	20	1	6	1
3	1	15	0	14	30	0	15	1	10	1
N	0	46	1	53	8	0	46	1	31	0

h : total degree of hard constraint violation, x_1^* : value of x_1 chosen in the flip trial

note that, only h_1', h_1, and x_1^* are recorded for the violation history of x_1

Clearly, the interdependency of the variables implies that the effect of the value chosen for any particular variable in isolation is uncertain. This observed inconsistency is taken to be a result of the uncertainty or random behaviour on the part of a given value choice decision. Therefore, at each flip trial, the algorithm is enforced to select the choice of value with lowest total hard violation (disutility) for a variable of interest.

4.2.1. Logit method

We can derive a choice probability by assuming a joint probability distribution for the two random disutilities. At each flip trial n the algorithm selects a choice value for $x = 0$ or $x = 1$ with the lowest total hard violation (disutility). Under the assumption that the non-deterministic total hard violation is logistically distributed, applying a standard logistic distribution function and probability theory, a specific probabilistic choice model, the *logit model* (see Ben-Akiva and Lerman (1985)), can be obtained as follows:

$$P_{0n} = \frac{e^{V_{0n}}}{e^{V_{0n}} + e^{V_{1n}}} \tag{15}$$

where: P_{0n} is a probability for flip trial n choosing value 0.

4.2.2. Proportional method

The proportional choice model is also probabilistic but only depends on the number of occurrences of choice values in x^* (Table 6), i.e., the total hard violation h is not considered. This choice model is designed to enhance the prediction when the violations $\bar{h}_0$ and $\bar{h}_1$ are close, in this case, the logit model may not perform well. In addition, as the proportional model requires less computation, the logit model may be called only when an occurrence of any value in x^* is not obviously dominating, i.e., the proportion of any one value in x^* is less than a preset parameter D, e.g., D = 70%. A probability of choice value 0 for the total number of flip trials N, P_0 is defined as:

$$P_0 = \frac{\sum_{n=1}^{N} \left| x_n^* - 1 \right|}{N} \tag{16}$$

As an outcome of the choice models is a probability of choosing a value for the variable, the variable is fixed at its predicted value for a number of iterations determined by the magnitude of the probability. When the number of fixing iterations F is reached, the variable is freed and its violation history is refreshed. During these iterations, other decision variables may become fixed. This sequence of fixing values for the variables helps intensify the search and targets the optimal schedule.

5. Experimental results

The experiments are compared with results from a case study. The description of the case study is given first. The real case problem is then solved in order to demonstrate the performance of the model and solution algorithm proposed.

5.1. Case study

The eastern line container rail service in Thailand is selected for a case study because it has a very high volume of container traffic and serves the main import-export activities between Bangkok and the eastern region, the gateway of the country.

Recently, it has been roughly estimated that more than a million containers a year or several thousand containers a day are transported between the two regions. About thirty percent of container movement is shared by rail. In 2000, because of traffic congestion and environmental concerns, the government of Thailand decided to limit the number of containers through Bangkok port to 1.2 million containers a year. As a result, Laem Chabang port, located in the eastern region of the country, has to serve an increasing number of container flows between two regions.

At present, the Royal State Railway of Thailand (RSRT) provides a weekly fixed schedule in which a certain number of trains are provided in fixed departure timeslots. The real life data is colleted from four successive weeks with 184 shipping companies.

5.2. Performance of the proposed model and algorithm

We present results for our solution algorithm on a test of four realistic data sets described in Table 7. In fact, more extensive tests have been conducted; however, we present here only four problems that typify the problem size and performance characteristics.

Table 7 Characteristics of the test problems

Test Case	Customers	Containers	θ	RSRT's schedules		Supply - Demand (Δ)	
				Trains	Capacity	Capacity	Trains
W1	134	2907	43	57	3876	969	14
W2	116	2316	35	42	2856	540	7
W3	84	1370	21	28	1907	537	7
W4	109	2625	37	50	3400	775	13

θ: a lower bound on number of trains; a capacity of a train (P_2) is 68 containers

We present the results yielded by the implemented framework of our rail container scheduling model and solution algorithm, and compare them with current practice. Several constraint violation schemes are evaluated in order to investigate their robustness and sensitivity. The probabilistic learning method is also tested by comparing the results with the constraint-based search algorithm.

From all test cases, we set the limitation of 5 minutes of CPU time, value choice-specific parameter in the probabilistic learning method β_1 = 0.05, number of flip trials (violation history) N = 20, percent decision method D = 80, the number of fixing timeslots *nbFixX* = 50, the number of fixing customers *nbFixY* = 100. Note that *nbFixX* and *nbFixY* are the upper limits which the number of variables in the model can be fixed at their predicted values. However, when *nbFixX* or *nbFixY* is reached, the first variable in the fixing list is suddenly released, and a new predicted variable is inserted and fixed. Each test case is run five times. The results for the test cases are shown in Tables 8 - 11.

The abbreviations displayed in Tables 3 - 6 have the following meanings: v_c, v_m, v_s are hard violations for timeslot consistency, maximum capacity, and timeslot covering constraints respectively; f_i is the optimality index factor, which allows the algorithm to give priority between timeslot and hard violations; *F* is a preset number of fixing iterations; T^{*}_{exp} is the number of trains expected at the last feasible schedule found; *OC* is the average operating costs (× 10^6 Baht); *VC* is the average virtual revenue lost (× 10^6 Baht); *GC* is a generalised cost and is the sum of *OC* and *VC*.

Table 8 Computational results for test problem W1

Test No.	Violation (v_c, v_m, v_s)	f_i	*F*	T^{*}_{exp}	Proposed schedule's costs *OC*	*VC*	*GC*	Reduction *OC* (%)
1	(1, 1, 0)	0	0	53	4.77	0.85	5.62	12.16
2	(1, 10, 0)	0	0	56	5.32	0.75	6.07	2.03
3	(10, 1, 0)	0	0	-	-	-	-	-
4	(1, 1, 10)	0	0	54	5.05	0.85	5.90	7.00
5	(1, 1, 50)	0	0	50	4.50	0.82	5.32	17.13
6	(1, 1, 100)	0	0	50	4.50	0.82	5.32	17.13
7	(1, 1, 200)	0	0	51	4.61	0.82	5.43	15.10
8	(1, 1, 100)	1	0	49	4.36	1.20	5.56	19.70
9	(1, 1, 100)	5	0	50	4.50	0.82	5.32	17.13
10	(1, 1, 100)	10	0	-	-	-	-	-

RSRT schedule's cost (*OC*) = 5.43, minimum train loading (P_1) = 52, T_{exp} = 56

Table 9 Computational results for test problem W2

Test No.	Violation (v_c, v_m, v_s)	f_i	*F*	T^{*}_{exp}	Proposed schedule's costs *OC*	*VC*	*GC*	Reduction *OC* (%)
1	(1, 1, 0)	0	0	45	4.43	0.75	5.18	-9.93
2	(1, 10, 0)	0	0	45	4.43	0.75	5.18	-9.93
3	(10, 1, 0)	0	0	45	4.43	0.75	5.18	-9.93
4	(1, 1, 10)	0	0	47	4.60	0.62	5.22	-13.26
5	(1, 1, 50)	0	0	45	4.43	0.62	5.18	-9.93
6	(1, 1, 100)	0	0	39	3.35	1.14	4.49	16.04
7	(1, 1, 200)	0	0	47	4.60	0.62	5.22	-13.26
8	(1, 1, 100)	1	0	40	3.48	1.05	4.89	12.78
9	(1, 1, 100)	5	0	-	-	-		-
10	(1, 1, 100)	10	0	-	-	-		-

RSRT schedule's cost (*OC*) = 3.99, minimum train loading (P_1) = 50, T_{exp} = 47

In Table 8 - 9, we perform a sensitivity analysis to test parameters in the constraint violation scheme. For instance, test no. 1, the violations of timeslot consistency and maximum capacity constraints are given the same weight (v_c, v_m = 1), and the violation v_s for timeslot covering constraint is not included. The results of test no.1-3 show that the violations v_c and v_m are not sensitive as they both represent infeasible schedules. This indicates the robustness of the solution algorithm, which does not require special tuning parameters between hard constraint violations.

The violation scheme with v_s shows a superior performance with respect to the quality of the schedule produced. This indicates that positive values of v_s have a computational advantage as it influences the search for the feasible schedules. However, choosing too large a value for v_s can decrease its effectiveness. Our experiments indicate that v_s = 100 is a reasonable setting.

The results also show the effect of introducing the optimality index factor f_i. With the existence of f_i, we obtain a slightly better quality of schedule. This is because the desirable schedule is mostly controlled by the minimum number of trains constraint as in practice it plays a more important role than other objectives. However, a high value of f_i provides a low quality of results. This may be because the search is dominated by the timeslot violation and hard constraints have not been satisfied.

Table10 Computational results for test problem W3

Test No.	Violation (v_c, v_m, v_s)	f_i	F	T^*_{exp}	Proposed schedule's costs *OC*	*VC*	*GC*	Reduction *OC* (%)
1	(1, 1, 0)	0	50	27	2.40	0.42	2.82	9.43
2	(1, 1, 0)	0	100	25	2.38	0.45	2.83	10.19
3	(1, 1, 0)	0	200	-	-	-	-	-
4	(1, 1, 0)	0	300	-	-	-	-	-
5	(1, 1, 100)	1	50	26	2.39	0.45	2.84	9.81
6	(1, 1, 100)	1	100	24	2.14	0.61	2.75	19.25
7	(1, 1, 100)	1	200	25	2.38	0.45	2.83	10.19
8	(1, 1, 100)	1	300	-	-	-	-	-

RSRT schedule's cost (*OC*) = 2.65, minimum train loading (P_1) = 51, T_{exp} = 27

When the probabilistic learning method is incorporated, i.e., a preset number of fixing iterations F is set more than zero, the algorithm targets near optimal schedules quicker than constraint-based search alone, whilst containing computational effort within the limitation of 5 minutes of CPU time. From test no. 1- 4, violation scheme (v_c, v_m, v_s) = (1, 1, 0), and the optimality index factor f_i is not applied. The performance of algorithm is still good enough. When the violation scheme and the optimality index are used, we get rather better schedules. This is because the change in constraint violations affects the input of the value choice models. The prediction is improved as the weights are tuned by the problem specific knowledge. We also examine the performance of algorithm having different numbers of fixing iterations F. The large number of fixing iterations affects the quality of schedule both positively and negatively. However, setting F between 100- 200 iterations performs well in general.

Table 11 Computational results for test problem W4

Test No.	Violation (v_c, v_m, v_s)	f_i	F	T^*_{exp}	Proposed schedule's costs *OC*	*VC*	*GC*	Reduction *OC* (%)
1	(1, 1, 0)	0	50	46	4.21	0.83	5.04	11.55
2	(1, 1, 0)	0	100	43	3.93	1.08	5.01	17.44
3	(1, 1, 0)	0	200	48	4.42	0.67	5.09	7.14
4	(1, 1, 0)	0	300	-	-	-	-	-
5	(1, 1, 100)	1	50	47	4.35	0.69	5.04	8.61
6	(1, 1, 100)	1	100	41	3.82	1.23	5.05	19.75
7	(1, 1, 100)	1	200	48	4.42	0.67	5.09	7.14
8	(1, 1, 100)	1	300	-	-	-	-	-

RSRT schedule's cost (*OC*) = 4.76, minimum train loading (P_1) = 55, T_{exp} = 48

Comparing the model results with current practice, in all test cases, there are some reductions in rail operating costs, but these are not considerable. This is because in practice the RSRT's schedule is not fixed at the same service level everyday. The rail carrier normally cuts down the number of train services with a short notice if the train supply is a lot higher than the customer demand. This is done by delaying some customer's departure times according to its demand consolidation strategy.

As noted in test case W2 (Table 9), some feasible schedules based on our proposed model incur higher operating costs than RSRT's schedule as more trains are required. This is one of the reasons for the RSRT to rely on its demand's consolidation strategy. However, our best feasible schedule requires slightly lower number of trains to operate than the RSRT's practice. In addition, our model maximises customer satisfaction, in other words, minimises the virtual loss of future revenue within a generalised cost framework. Therefore, the near optimal schedule could reflect the maximum degree of customer satisfaction within the restrictions of other rail business criteria through a demand responsive schedule.

6. Conclusions

With the proposed demand responsive rail scheduling model, we are able to find a profitable schedule whilst satisfying customer demand within a generalised cost function. The schedule leads to some reductions in total operating costs and enhances the customer satisfaction through a demand responsive schedule.

We also propose a novel method to solve the rail container scheduling problem. As the existing ways of solving the problem rely on complex heuristics and problem specific knowledge, our solution approach is more flexible and convenient to implement. The problem is modelled as a constraint satisfaction problem and solved using a constraint-based search algorithm that learns implicitly from the constraint violation history and tries to fix a likely optimal value for the decision variables adaptively. It helps the search quickly target the optimal schedule.

References

Arshad, F., EL-Rhalibi, A., Kelleher, G. 2000. Information management within intermodal transport chain scheduling, European project PISCES report, Liverpool John Moores University, UK.

Ben-Akiva, M. Lerman, S. R. 1985. Discrete choice analysis: theory and application to predict travel demand, MIT Press.

Brannlund, U., Lindberg, P.O., Nou, A., Nilsson, J.E. 1998. Railway timetabling using Lagrangian relaxation, Transportation Science 32 358-369.

Brucker, P., Hurink, J.L., Rolfes, T. 1999. Routing of railway carriages: a case study, Memorandum No. 1498, Faculty of Mathematical Sciences, University of Twente.

Cordeau, J., Toth, P., Vigo, D. 1998. A survey of optimisation models for train routing and scheduling, Transportation Science 32 380-404.

Crainic, T. 1986. Rail tactical planning: issues, models and tools, Proceeding of the International Seminar on Freight Transportation Planning and Logistics, Bressanone, Italy.

Gorman, M.F. 1998. An application of genetic and tabu searches to the freight railroad operating plan problem, Annals of Operations Research 78 51-69.

Gualda, N.D.F., Murgel, L.M.S.F. 2000. A model for the train formulation problem, Third International Meeting for Research in Logistics, Trois-Rivieres, May 9-10.

Haghani, A.E. 1989. Formulation and solution of combined train routing and makeup, and empty car distribution model, Transportation Research 23B 433-452.

Huntley, C.L., Brown, D.E., Sappington, D.E., Markowicz, B. P. 1995. Freight routing and scheduling at CSX transportation, Interface 25(3) 58-71.

Kearfott, R.B. 1998. On proving existence of feasible points in equality constrained optimisation problems, Mathematical Programming 83 89-100.

Keaton, M.H. 1992. Designing railroad operating plans: a dual adjustment method for implementing Lagrangian relaxation, Transportation Science 26 263-274.

Kraft, E. R. 2002. Scheduling railway freight delivery appointments using a bid price approach, Transportation Research 36A 145-165.

Ministry of Commerce. 2002. Thai commodity market price, the Department of Business Economics, Ministry of Commerce, Thailand.

Newman, A.M., Yano, C.A. 2000. Scheduling direct and indirect trains and containers in an intermodal setting, Transportation Science 34 256-270.

Yano, C. A., Newman, A.M. 2001. Scheduling trains and containers with due dates and dynamic arrivals, Transportation Science 35 181-191.