RAIL CONTAINER SERVICE PLANNING: A CONSTRAINT-BASED APPROACH

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

Abstract This paper considers a container rail service planning problem, in which customer demands are known in advance. The existing rail freight optimisation models are complex and not demand responsive. This paper focuses on constructing profitable schedules, in which service supply matches customer demands and optimises on booking preferences whilst satisfying regulatory constraints. A constraint satisfaction approach is used, in which optimisation criteria and operational requirements are formulated as soft and hard constraints respectively. We present a constraint-based search algorithm capable of handling problems of realistic size. It employs a randomised strategy for the selection of constraints and variables to explore, and uses a predictive choice model to guide and intensify the search within more promising regions of the space. Experimental results, based on real data from the Royal State Railway of Thailand, have shown good computational performance of the approach and suggest significant benefits can be achieved for both the rail company and its customers.

Keywords: rail container service planning, local search, constraint-based approach.

1. INTRODUCTION

The transportation of rail freight is a complex domain, with several processes and levels of decision, where investments are capital-intensive and usually require long-term strategic plans. In addition, the transportation of rail freight has to adapt to rapidly changing political, social, and economic environments. In general, the rail freight planning involves four main processes: path formulation, fleet assignment, schedule production, and fleet repositioning. This paper addresses an issue in schedule production, constructing profitable schedules for the container rail service, using a constraint-based approach.

1.1 Container Rail Service Planning Problem

Container rail service differs from conventional freight rail in several important aspects. Because of the high costs of container handling equipment,

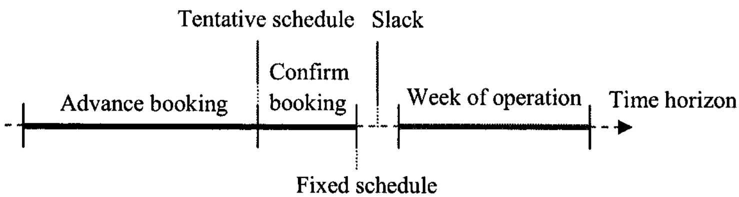

Figure 1. A short-term advance booking scheme.

container rail networks have relatively few, and widely spaced, terminals. Networks with a small number of 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 of rail container transportation.

This paper considers the container rail service from a container port to an inland container depot (ICD). Once containers arrive at the port, there is a need to move them to their final customers, which can be done by rail or truck via ICD, or by truck direct to the final destinations.

A rail carrier's profitability is influenced by the railway's ability to construct schedules for which service supply matches customer demand. The need for flexible schedules is obvious because the take-up of some services in a fixed schedule may be low and not profitable. In order to create a profitable timetable, a container rail carrier needs to engage in a decision-making process with multiple criteria and numerous constraints, which is very challenging.

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.

Customers are requested to state a preferred departure timeslot or an earliest departure time in advance. A number of alternative departure timeslots 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.

1.2 Literature Review

There are two principal areas of work relating to the application domain and the solution approach.

Application domain. There is an increasing interest in flexible rail freight schedules in the literature, which may be distinguished into two types according to how the overall demand is met. Huntley *et al.* (1995), Gorman (1998) and Arshad *et al.* (2000) aggregate customers 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 demands.

The models proposed by Newman and Yano (2000), Yano and Newman (2001) and Kraft (2002) satisfy the operational constraints fully for each customer. In contrast, our framework models customer satisfaction, computed from preferred and alternative departure times, which is then maximised as one of the business criteria. Hence, some customers might not be given their most preferred departure times. This framework is more natural for supporting decision-makers, in which a rail carrier can measure how well their customers are satisfied and the implications of satisfying these customers in terms of cost.

Solution approach. As the size of the problem that needs to be routinely solved in the rail freight industry is large, local search methods, such as simulated annealing, tabu search, genetic algorithms, etc, have been employed to produce near-optimal solutions. For instance, Huntley *et al.* (1995) applied simulated annealing to the problem of rail freight routing and scheduling, Marin and Salmeron (1996) evaluated a descent method, simulated annealing, and tabu search for solving large size rail freight networks, Gorman (1998) used a tabu-enhanced genetic algorithm for solving the freight railroad operating plan problem, and Arshad *et al.* (2000) combined constraint programming with a genetic algorithm to solve the multi-modal transport chain scheduling problem. A recent survey of optimisation models for train routing and scheduling is given by Cordeau *et al.* (1998). However, these approaches are complex and highly domain specific; thus they lose flexibility and the sustainability to solve rail container optimisation models in which the rail business strategy keeps changing.

The concept of domain-independent algorithms is always attractive and may be appropriate for our problem. There are many algorithms in this class; for example, Connolly (1992) introduced general purpose simulated annealing (GPSIMAN), Abramson and Randall (1999) extended GPSIMAN to solve integer programs (INTSA), and Nonobe and Ibaraki (1998) proposed a tabu search approach as a general solver for a constraint satisfaction problem.

In contrast, our approach is inspired by SAT local search for the satisfiability (SAT) problem (Gomes *et al.*, 1998; Selman *et al.*, 1992, 1994). SAT is a problem of deciding whether a given Boolean formula is satisfiable. When the problem is not solvable in polynomial time by exact algorithms, SAT local search might be employed. An attractive framework of SAT local search is that the structure of the local move is simple. GSAT and WalkSAT are well-known local search techniques for SAT problems (Selman *et al.*, 1992, 1994). SATz-Rand, introduced by Gomes *et al.* (1998) is a recent solver for SAT problems. Walser (1999) extended WalkSAT to WSAT(OIP) for solving integer programming problems.

However, our problem encoded into a Boolean formula or 0–1 integer constraints would be large and the solution structure may be difficult to maintain by simple local move with a randomised strategy, as performed by SAT local search and other domain-independent algorithms. One way to enhance the algorithm whilst maintaining a simple structure of local move is to build in learning capabilities in the algorithm. We propose a predictive choice model that learns from the search history in order to fix locally some variables, and enforce the consistency between different sets of variables. The model addresses the case in which all decision variables are binary. Our predictive learning algorithm has some similarities to other algorithms based on probabilistic models (Horvitz *et al.*, 2001; Larraaga and Lozano, 2002; Marin and Salmeron, 1996; Resende and Ribero, 2001). All these algorithms attempt to draw inferences specific to the problem and therefore can be regarded as processes for learning domain knowledge implicitly. However, the formulation and application of the models are different. Our predictive model is based on a discrete choice theory of human behaviour for choosing a particular value for variables in a probabilistic way, whilst the others are based on different theories and use the probabilistic models to limit the search space at different points of process.

The paper is organised as follows. We first define the hard and soft constraints and model the problem as a constraint satisfaction problem. The solution algorithm will be described next. Experimental results, based on real data from the Royal State Railway of Thailand will be presented. Finally, conclusions are discussed.

2. CONSTRAINT-BASED MODELLING

Real-world problems tend to have a large number of constraints, which may be hard or soft. Hard constraints require that any solutions will never violate the constraints. Soft constraints are more flexible, constraint violation is tolerated but attracts a penalty. Naturally, a real-world problem can be thought of as a constraint satisfaction problem (CSP).

There are two critical advantages of using constraint-based modelling. Firstly, it is a clean separation between problem modelling and solution technique. If new problem conditions are introduced, we only need to model such conditions as constraints. Secondly, problem-specific knowledge can influence the search naturally. This is done by applying problem-specific weights, reflecting their relative importance, directly to constraints in order to enhance a solution algorithm within a CSP framework.

We model the demand-responsive container rail scheduling problem as a CSP and introduce a constraint-based search for solving this class of CSP. We consider problems in which the day is divided into hourly slots for weekly booking and scheduling. The following notation will be used.

Subscripts:

- t schedulable timeslot (departure time), $t = 1, 2, 3, \ldots, T$.
- j customer, $j = 1, 2, 3, \ldots, M$.

Sets:

- S_j set of possible departure timeslots 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 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).

Our problem may be thought of as analogous to a capacitated facility (warehouse) location problem (CFLP) (Aardal, 1998; Beasley, 1988), with the container shippers being the customers, and the train departure timeslots being analogous to the possible warehouse locations. The CFLP class of problems also include location and distribution planning, capacitated lot sizing in production planning, and communication network design (Boffey, 1989; Kochmann and McCallum, 1981), etc. However, in contrast to the CFLP, our problem is recurrent with complex interaction between the possible warehouse locations. We handle non-uniform demands that arrive at the container port dynamically with distinct target times to their final destinations. In addition, we include in our model a probabilistic decrease in customer satisfaction as the schedulable timeslots deviate from the customer target time.

In a CSP-model, optimisation criteria and operational requirements are represented as soft and hard constraints respectively. The criteria 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 constrained variables in the model that satisfies all hard constraints, whereas an optimal solution is a feasible solution with the minimum total soft constraint violation (Henz *et al.*, 2000; Lau *et al.*, 2001; Walser, 1999). For a constraint satisfaction problem, the violation ν_i of constraint i is defined as follows:

$$\sum_j a_{ij}x_j \leq \; b_i \Rightarrow \nu_i = \max\left(0, \sum_j a_{ij}x_j - b_i\right) \tag{1}$$

where a_{ij} are coefficients, b_i is a tight bound and x_j are constrained variables. Note that violations for other types of linear and non-linear constraints can be defined in an analogous way.

When all constrained variables are assigned a value, the violation of the hard and soft constraints can be tested and quantified for evaluating local moves.

2.1 Soft Constraints

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. $\left\lceil \left(\sum_j N_j\right) / P_2 \right\rceil$.

Customer satisfaction. This constraint aims to maximise the total customer satisfaction. The satisfaction is assigned values from a customer satisfaction function (e.g., Figure 2). Each customer holds the highest satisfaction at a preferred booking timeslot, the satisfaction then decreases probabilistically to the lowest satisfaction at the last alternative booking timeslot, i.e. later than preferred booking timeslots 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 demand N_j. The customer satisfaction constraint can be expressed as

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

where Ω is an upper bound on customer satisfaction, i.e. $\sum_j W_j N_j$, with W_j the maximum satisfaction on the preferred booking timeslot for customer j.

Timeslot operating costs. This constraint aims 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 constraint is defined as

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

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 Constraints

Train capacity. This constraint ensures the demand must not exceed the capacity of a train, which is defined as

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

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 a customer can only be served by one train. The constraint is given as

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

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

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

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

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

2.3 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. While implied constraints do not affect the set of feasible solutions to the model, they may have computational advantage in search-based methods as they reduce the size of the search space (Proll and Smith, 1998; Smith *et al.*, 2000).

Timeslot covering. A 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. This constraint favours a combination of selected departure timeslots that covers all customers. The covering constraint is defined as

$$\sum_{t \in S_j} x_t \geq 1 \quad \forall t \tag{9}$$

2.4 Customer Satisfaction

A rail carrier could increase the quality of service by tailoring a service that satisfies 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. This survey includes 184 customers currently using both rail and trucking services or using only rail but with the potential to ship by truck in the future. To quantify customer satisfaction, customer satisfaction functions were developed. Total shipping costs associated with movement by different 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 cargo from survey data and the market price (Ministry of Commerce, 2002) are summarised in Table 1.

We assume that all customers know a full set of shipping costs, and can justify modal preferences on the basis of accurately measured and understood costs. The freight rate may be adjusted by the relative costs that a customer

Table 1. Modal cost for a transport mode.

Cargo types/cost	Cost /unit price Truck	Rail	Market price	Modal cost (%) Truck, C_T	Rail, C_R	ΔC
Freight rate						
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						
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						
(within free time storage)	0	0		0	0	0
Overhead cost						
(within free time storage)	0	0		0	0	0
Total shipping costs						
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

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, modal cost percentages are then applied to an assumed normal distribution (Indra-Payoong *et al.*, 1998) and the difference between modal cost percentages, i.e. $\Delta C = C_T - C_R$. The customer satisfaction derived from the cumulative probability density function is illustrated in Figure 2.

Once the satisfaction function has been developed, a customer satisfaction score can be obtained from the modal satisfaction probability. This probability could also 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 is a probability of choosing rail service; hence satisfaction ranges from 0 to 1. Note that all customers currently using container rail service may already hold a certain level of satisfaction regardless of taking the quality of rail schedule into account. Once the rail carrier has been chosen as transport mode and later the schedule is delayed, customers incur additional

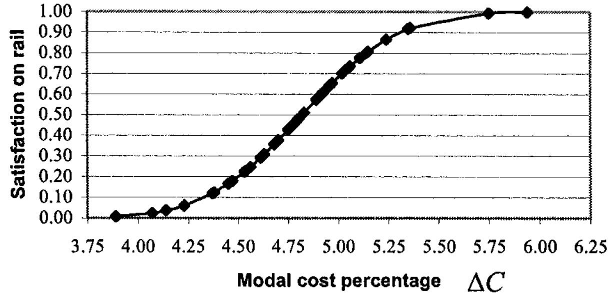

Figure 2. Customer satisfaction function of cargo type II.

total shipping costs, i.e. terminal storage and overhead costs involved at the seaport. This would result in a decrease in customer satisfaction.

2.5 Generalised Cost Function

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. We shall 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 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, GC, is the sum of the operating costs and the virtual loss of revenue:

$$GC = (\theta + s_1)FC + s_2 FR + (\lambda + \delta + s_3) \tag{10}$$

where FC is a fixed cost of running a train, FR is a freight rate per demand unit (ton-container), s_1, s_2 and s_3 are soft constraint violations for the number of trains, customer satisfaction, and timeslot operating costs constraints respectively.

3. SOLUTION ALGORITHM

We propose a constraint-based search algorithm (CBS) for solving the constraint satisfaction problem. The algorithm consists of two parts: CBS based on a randomised strategy and a predictive choice learning method, which guides and intensifies the search.

3.1 Constraint-Based Search Algorithm

Points in the search space correspond to complete assignment of 0 or 1 to all decision variables. The search space is explored by a sequence of sample randomised moves which are influenced by the violated hard constraints at the current point.

The CBS starts with an initial random assignment, in which some hard constraints in the model can be violated. In the iteration loop, the algorithm randomly selects a violated constraint: e.g., the assigned train timeslot for which the demands exceed train capacity. Although different constraint selection rules have been studied for SAT local search (McAllester *et al.*, 1997; Parkes and Walser, 1996; Walser, 1999), for instance choosing the violated constraint with maximum or minimum violation, none have been shown to improve over random selection.

Having selected a violated constraint, the algorithm randomly selects one variable in that constraint and another variable, either from the violated constraint or from the search space. Then, two flip trials are performed, i.e. changing the current value of the variable to its complementary binary value. Suppose that V_i takes the value v_i at the start of the iteration, so that $A = (v_1, v_2, \ldots, v_m|h)$, where m is the total number of variables and h is the total violation of all hard constraints. Suppose further that V_1,V_2 are chosen and that their flipped values are $\bar{v}_1$, $\bar{v}_2$ respectively. We then look at the assignments $A_1 = (\bar{v}_1, v_2, \ldots, v_m|h_1)$, $A_2 = (v_1, \bar{v}_2, \ldots, v_m|h_2)$ and select the alternative with the smaller total hard violation. Whenever all hard constraints are satisfied, the algorithm stores the soft violation penalties as feasible objective values, together with the associated variable values. The algorithm continues until the stopping criterion is met, i.e. a feasible solution is found or if no improvement has been achieved within a specified number of iterations. The procedure of CBS is outlined in Figure 3.

The procedure can be readily modified to give a set of feasible solutions and to make more use of the soft constraints, which in the procedure of Figure 3 are largely ignored. We do not do so here but do in the context of an enhanced procedure incorporating the predictive choice model in the following section.

3.2 Predictive Choice Model

The first development of choice models was in the area of psychology (see Marder, 1997). The development of these models arose from the need to explain the inconsistencies of human choice behaviour, in particular consumer choice in marketing research. If it were possible to specify the causes of these inconsistencies, a deterministic choice model could be easily developed.

These causes, however, are usually unknown or known but very hard to measure. In general, these inconsistencies are taken into account as random

proc CBS
  input soft and hard constraints
  $A \quad :=$ initial random assignment
  while not stopping criterion **do**
    $C :=$ *select-violated-hard-constraint* (A)
    $P :=$ *select-two-variables* (C, A)
    $A_1, \quad A_2 \quad :=$ flip(A, P)
    if $(h_1 < h_2)$ **then** $(A \leftarrow A_1)$
    else $(A \leftarrow A_2)$
    if $h = 0$ **then** A is feasible, record solution A
    end if
  end while
  output a feasible solution found
end proc

Figure 3. The constraint-based search procedure.

behaviour. Therefore, the choice behaviour could only be modelled in a probabilistic way because of an inability to understand fully and to measure all the relevant factors that affect the choice decision.

Deciding on a choice of value for a variable in a CSP is not obviously similar to the consumer choice decision. However, we could set up the search algorithm to behave like the consumer behaviour in choice selection. That is, we consider the behavioural inconsistencies of the algorithm in choosing a good value for a variable.

For general combinatorial problems, a particular variable may take several different values across the set of feasible solutions. Thus it may never be possible to predict a consistently good value for the variable during the search. However, when the problem is severely constrained and has few feasible solutions, it may well be that some variables take a more consistent value in all the feasible solutions during the search. The predictive choice model is intended to discover such values and to use them to steer the search. Note that for the container rail scheduling model in Section 2, the problem becomes more severely constrained as the value of minimum train loading increases (Section 3.3).

Violation history. Once a variable has been selected, the algorithm has to choose a value for it. The concept is to choose a good value for a variable: e.g. the one that is likely to lead to a smaller total hard constraint violation in a complete assignment. In our constraint-based search algorithm, two variables are considered at each flip trial. The first variable is randomly chosen

Table 2. Violation history.

	Variable of interest x_1				Compared variable x_j					
Flip	Current		Flipped			Current		Flipped		
trial	Val	h_1	Val	h_1'	j	Val	h_2	Val	h_2'	x_1^*
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

from those appearing in a violated constraint and considered as the variable of interest; the second variable is randomly selected, either from that violated constraint or from the search space, and is to provide a basis for comparison with the variable of interest.

Clearly, the interdependency of the variables implies that the effect of the variable value chosen for any particular variable in isolation is uncertain. Flipping the first variable might result in a reduction in total hard constraint violation. However, it might be that flipping the second variable would result in even more reduction in the violation. In this case, the flipped value of the first variable is not accepted.

In Table 2, the variable of interest is x_1 and the compared variable is x_j; the two variables are trial flipped in their values; the violations associated with their possible values are recorded and compared. In this table, h is the total hard violation, $x_1^*$is the 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.

In flip trial 1 the selected variables are x_1 (current value 1) and, separately, x_{15} (current value 1). The current assignment has violation = 26. Flipping x_1, with x_{15} fixed at 1, gives violation = 22; flipping x_{15}, with x_1 fixed at 1, gives violation = 36. Hence in this trial the algorithm records $x_1 = 0$ as the better value. At some later iteration the algorithm chooses to flip x_1 again, this time (flip trial 2) with compared variable x_9. Flipping x_1, with x_9 fixed at 0, gives violation = 12; flipping x_9, with x_1 fixed at 1, gives violation = 6. Although flipping x_1 to 0 gives a better violation than the current assignment, in this flip trial the algorithm records $x_1 = 1$ as the better value as there is an assignment with $x_1 = 1$ which gives an even better violation. If we view the results of these flip trials as a random sample of the set of all assignments, we can build up a predictive model to capture the behavioural inconsistency in the choice selection and to predict what would be a "good" value for x_1.

Utility concept. The predictive choice model is based on the random utility concept (Anderson *et al.*, 1992; Ben-Akiva and Lerman, 1985). Choosing a good value for a variable in each flip trial is considered as a non-deterministic task of the search algorithm. The algorithm is designed to select a choice of value for a variable that has a maximum utility.

However, the utility is not known by the algorithm with certainty and is therefore treated as a sum of deterministic and random utilities. The utility is defined as follows:

$$U_0 = V_0 + \varepsilon_0 \tag{11}$$

where U_0 is an overall utility for the algorithm choosing value $0, V_0$ is a deterministic utility for the algorithm choosing value 0, ε_0 represents inconsistencies (uncertainties) in the choice selection, measurement errors and unobserved choice decision factors, and is a random utility for the algorithm choosing value 0.

For each flip trial, the algorithm selects value 0, when flipping a variable to 0 is preferred to 1. This can be written as follows:

$$0 \succ 1 \Rightarrow U_0 > U_1 \Rightarrow (V_0 + \varepsilon_0) > (V_1 + \varepsilon_1) \tag{12}$$

The random utilities ε_0 and ε_1 may cause uncertainty in the choice selection, i.e. U_0 might be greater than U_1 or U_1 might be greater than U_0, even if the deterministic utility satisfies $V_0 > \quad V_1$. From this point, the probability for the algorithm choosing value 0 is equal to the probability that the utility of choosing value 0, U_0, is greater than the utility of choosing value 1, U_1. This can be written as follows:

$$P_0 = \mathrm{Prob}[U_0 > U_1] \tag{13}$$

where P_0 is the probability for the algorithm choosing value 0.

Thus,

$$P_0 = \mathrm{Prob}\left[(V_0 - V_1) > (\varepsilon_1 - \varepsilon_0)\right] \tag{14}$$

To derive a predictive choice model, we require an assumption about the joint probability distribution of the random utilities ε_1 and ε_0.

Joint probability distribution. To derive the joint probability distribution of the random utilities ε_1 and ε_0, the difference between the random utilities, i.e. $\varepsilon' = \varepsilon_1 - \varepsilon_0$, is used. However, ε' is unknown by the algorithm. We use the difference between deterministic utilities, i.e. $V' = V_1 - V_0$, to investigate the probability distribution of ε' because the deterministic and random utilities are the components of the overall utility U.

We can now look for an appropriate functional form for the distribution of V'. From the central limit theorem (Trotter, 1959), whenever a random sample of size n ($n > 30$) is taken from any distribution with mean μ and variance

σ^2, then the sample would be approximately normally distributed. We perform the Shapiro–Wilk test and Kolmogorov–Smirnov test (Patrick, 1982; Shapiro and Wilk, 1965) to find out whether the non-deterministic component appears to follow any specific distribution. From our experiments and the central limit theorem, we are first encouraged to assume normality of the distribution.

Although the normal distribution seems reasonable based on the central limit theorem, it has a problem with not having a closed probability function form, i.e. the PDF is expressed in terms of an integral; thereby it is computationally intractable. The logistic function is therefore chosen instead because its distribution is an approximation of the normal law (Kallenberg, 1997). Under the assumption that V' is logistically distributed, applying a standard logistic distribution function and probability theory, a specific probabilistic choice model, the *logit model* (Anderson *et al.*, 1992; Ben-Akiva and Lerman, 1985), can be obtained as follows:

$$P_0 = \frac{e^{V_0}}{e^{V_0} + e^{V_1}} \tag{15}$$

where P_0 is the probability for the algorithm choosing value 0.

Violation function. For any flip trial, the deterministic utility V may be characterised by many factors. In this research, the utility is determined by the total hard constraint violation h. This is because it can easily be measured by the algorithm and gives a reasonable hypothesis to the choice selection. In other words, we would like to use a function of deterministic utility for which it is computationally easy to estimate the unknown parameters.

We define a function that is linear in its parameters. A choice specific parameter is introduced so that one alternative is preferred to the other when the total hard violation is not given, i.e. the choice decision may be explained by other factors. The deterministic utility functions for V_0 and V_1 are defined as

$$V_0 = \beta_1 + \beta_2 h_0 \tag{16}$$

$$V_1 = \beta_2 h_1 \tag{17}$$

where β_1 is the choice specific parameter, β_2 is the violation parameter, and h_0 and h_1 are the total hard violations when a variable is assigned a value to 0 and 1 respectively.

We could now use the predictive choice model to predict a value for a particular variable from an individual flip trial. However, the predictions for an individual flip trial may not reliably help the algorithm make a decision on what a good value for a variable would be. Instead, we use an aggregate quantity, i.e. a prediction for the value choice based on a set of trials. We use the arithmetic mean of the total hard violation to represent the aggregate of N flip trials, which can be written as

$$\bar{h}_0 = \sum_{n=1}^{N} \frac{h_{0,n}}{N} \quad \text{and} \quad \bar{h}_1 = \sum_{n=1}^{N} \frac{h_{1,n}}{N} \tag{18}$$

where $\bar{h}_0$ and $\bar{h}_1$ are the average total hard violations when a variable is assigned a value 0 and 1 respectively.

Logit method. When an occurrence of any choice value x^* is not obviously dominating (Table 2), the logit method is called. As a statistical way to estimate the utility's parameter values requires a significant computational effort, we introduce a simplified estimation, in which logit method only accounts for the constraint violation. We set the choice-specific parameter β_1 to any small value, e.g. $\beta_1 = 0.05$, so that the utility of one alternative is preferred to the other. This is because an equal utility lies outside the assumption of the choice theory. Then, the relative difference between $\bar{h}_0$ and $\bar{h}_1$, $\Delta\bar{h}$, is used in order to characterise the value choice selection. $\Delta\bar{h}$ is defined as follows:

$$\Delta\bar{h} = \left| \frac{\bar{h}_0 - \bar{h}_1}{\bar{h}_0 + \bar{h}_1} \right| \tag{19}$$

where $\bar{h}_0$ and $\bar{h}_1$ are the average total hard violations when a variable is trial flipped or assigned a value 0 and 1 respectively.

From (19), when the value of $\Delta\bar{h}$ is large, the probabilities of two alternatives (value 0 and 1) would be significantly different, and when $\Delta\bar{h} = 0$, the probabilities of the two alternatives would tend to be equal. $\Delta\bar{h}$ is shown in a proportional scale so that the formulation could be generalised for a combinatorial problem in which the total hard violation and a number of flip trials can be varied. Then, we use a simplified estimation of β_2 as follows:

$$\beta_2 = -\Delta\bar{h} \tag{20}$$

where β_2 is the violation parameter.

Proportional method. This method is also based on a probabilistic mechanism in the sense that the algorithm may select the current value of the variable even though flipping that variable to the other value gives a lower violation.

The proportional method is more straightforward than the logit method. The choice selection is only affected by the number of occurrences of choice values in x^*, i.e. the constraint violation is not explicitly considered. This method is developed and used to enhance the choice prediction by the simplified estimation of the utility's parameters when $\bar{h}_0$ and $\bar{h}_1$ are close. In this case the logit method may not perform well. In addition, this method requires less computation than the logit method. The proportional method is defined as

$$P_0 = \frac{x_0^*}{N} \tag{21}$$

where P_0 is the probability for the algorithm choosing value 0, x_0^* is the number of occurrences of value 0 in x^*, N is the number of flip trials.

An outcome of the predictive choice model is the probability of choosing a particular value for a variable. The timeslot and customer's booking variables (x_t and y_{tj}) are chosen for flip trials, but propagation of their values for consistency may not be carried out fully all the time. At the beginning, constraint propagation is only carried out within each of the sets x_t and y_{tj}, but not across the two sets of variables, in order to promote wider exploration of the search space.

After a specified number of iterations, the trial violation history is analysed. Some variables may have high probability of a particular value given by the predictive choice model. These variables will be fixed at their predicted value for a number of iterations determined by the magnitude of the associated probability. At this point, consistency between timeslots and customer's bookings variables is enforced, leading to intensified exploration of the neighbouring search space. When the fixing iteration limit, F, is reached, the variable is freed and its violation history is refreshed.

3.3 Minimum Train Loading

The constraint-based search assigns a fixed number of trains according to the number of trains expected, which is derived from the minimum train loading. In other words, a fixed number of timeslots used is maintained during the search process, which can be written as

$$\sum_{t=1}^{T} x_t = T_{\text{exp}} \tag{22}$$

where T_{exp} is the number of trains expected.

Setting a minimum train loading ensures satisfactory revenue for a rail carrier and spreads out the capacity utilisation on train services. The carrier may want to set the minimum train loading as high as possible, ideally equal to the capacity of a train. Note that the minimum train loading is directly related to T_{exp}, which is defined as

$$T_{\text{exp}} = \left\lceil \sum_j N_j / P_1 \right\rceil \tag{23}$$

where N_j is the demand of customer j, and P_1 is the minimum train loading.

Apart from ensuring satisfactory revenue, minimum train loading is a key factor in the performance of the search algorithm. The higher the minimum train loading, the more constrained the problem is and hence the number of feasible solutions decreases. Using a high minimum train loading allows the algorithm to focus on satisfying the hard constraints more than the soft constraints. In addition, it increases the usefulness of the value choice prediction mechanism, i.e. the variables in the container scheduling model would take more consistent values in all the feasible solutions during the search.

However, it would be very hard to prove whether there exists a feasible solution to the problem constrained by a high minimum train loading. If we could prove the existence of a feasible solution for the highest possible minimum train loading, it would imply that the solution is approximately optimal. A good setting of the minimum train loading helps limit the size of the search space. Although a few techniques for proving the existence of feasibility have been proposed (Hansen, 1992; Kearfott, 1998), implementations of these techniques for practical problems have not yet been achieved. In this research, the minimum train loading is derived from some heuristic rules. We estimate the minimum train loading by defining a risk parameter R. For example, $R = 20\%$ means that the estimated chance of the problem having no feasible solution is 20%. An initial value of P_1 is defined as follows:

$$P_1 = \frac{\sum_j N_j}{\lceil M/T \rceil} \tag{24}$$

where $T = \left\lfloor \frac{P_2}{\mu_R} \right\rfloor$, $\mu_R = \frac{(\mu_g+\sigma_g)\times(100-R)}{100}$, P_2 is a capacity of a train, N_j is the demand of customer j, M is the total number of customers, and μ_g and σ_g are the mean and standard deviation of the total average demand.

Whenever all hard constraints are satisfied (a feasible train schedule is obtained), the minimum train loading is implicitly increased by removing one train from the current state of the feasible solution, i.e. $T_{\exp} = T_{\exp} - 1$, and CBS attempts to find a new feasible schedule.

4. HIERARCHICAL CONSTRAINT SCHEME

In SAT local search and its variants, the number of violated constraints (unsatisfied clauses) is used to evaluate local moves without accounting for how severely individual constraints are violated. In CBS, a quantified measure of the constraint violation to evaluate local moves is used. In this case, the violated constraints may be assigned different degrees of constraint violation. This leads to a framework to improve the performance of the solving algorithm. The constraints can be weighted in the formulation of train measures of violation in order to allow the search to give hierarchical priority to satisfying some subsets of the constraints.

For the container rail service planning, soft and hard constraints in the model are treated separately. When all hard constraints are satisfied, the soft constraint violations are calculated and used as a measure of the quality of the solution. Nevertheless, whilst the hard constraint have not yet been fully satisfied, our scheme incorporates an artificial constraint, and its weighted violation measure is designed to exert some influence over the search process based on an estimation of the soft constraint violation (Section 4.2).

4.1 Feasibility Weights

The principal goal of the CBS is to find feasible solution to the problem, i.e. points at which the total violation of the hard constraints is zero. For the container rail service planning model, all sets of hard constraints use weighted measures of violation according to some heuristic rules.

From (5), any number of containers in a potential timeslot exceeding a train capacity is penalised with the same violation h_m. An attempt to use different measures of the violation to different number of exceeded containers on an assigned train makes little sense because one can never guarantee whether the lower number of exceeded containers is more likely to lead to feasible schedules. The violation penalty for a set of capacity constraints is defined as

$$x_t\left(\sum_{j=1}^{M} y_{tj}N_j\right)\begin{cases}\le P_2, & \text{violation} = 0\\ > P_2, & \text{violation} = h_m\end{cases} \quad \forall t \tag{25}$$

where h_m is the violation penalty for a capacity constraint, and P_2 is the capacity of a train.

From (7), the violation penalty for a set of consistency constraints is defined as

$$\sum_{l=1}^{L} y_{tl}\begin{cases}\le x_tL_t, & \text{violation} = 0\\ > x_tL_t, & \text{violation} = h_c\end{cases} \quad \forall t \tag{26}$$

where h_c is a violation penalty for a consistency constraint, and L_t is number of potential customers for timeslot t. From (9), the algorithm allocates a penalty if the assigned trains do not serve all customer demands. In other words, the covering constraint favours a combination of selected timeslots that covers all customers' bookings. The violation penalty within a set of covering constraints uses the same quantification, which is defined as

$$\sum_{t\in S_j} x_t\begin{cases}\ge 1, & \text{violation} = 0\\ = 0, & \text{violation} = h_s\end{cases} \quad \forall t \tag{27}$$

where h_s is a violation penalty for a covering constraint.

4.2 Timeslot Weights

Timeslot violation h_t has been introduced artificially so that the search considers an estimation of total soft constraint violation, whilst some other hard constraints are still to be fully satisfied. The timeslot violation is regarded as if it were a hard violation until the capacity, consistency, and covering constraints have all been satisfied, then the timeslot violation is set to zero.

The algorithm assigns a penalty if a timeslot t is selected as a train departure time. This can be written as

$$x_t \begin{cases} = 1, & \text{violation} = h_t \\ = 0, & \text{violation} = 0 \end{cases} \quad \forall t \tag{28}$$

In contrast to (25)–(27) which imply a fixed penalty for each member of the associated set of constraints, a violation penalty for the timeslot violation h_t varies from timeslot to timeslot. The timeslot violation penalty depends on the possibility of assigning a particular timeslot on a train schedule with a minimum generalised cost.

An attempt to derive the timeslot violation in monetary units by trading off between the business criteria is not possible. This is because a train schedule is not a single timeslot, but is a set of the timeslots. Therefore, 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 a more important role than others, the relative weights for the criteria could be applied.

A rail carrier may assign a relative weight to the number of trains, customer satisfaction, and operating costs criteria in which the one with the lower weight is more important. In practice, given the relative weights 0.2, 0.5 and 0.3, the timeslot violation h_t is therefore obtained as

$$h_t = 0.2N_t + 0.5S_t + 0.3E_t \quad \forall t \tag{29}$$

where h_t is the timeslot violation if timeslot t is chosen ($x_t = 1$), N_t is the violation cost for the number of trains in timeslot t, S_t is the violation cost for the customer satisfaction in timeslot t, and E_t is the violation cost for the carrier's operating costs in timeslot t.

The violation cost N_t. We first assume that the higher the number of potential customers in timeslot t, the more likely that timeslot would lead to the minimum number of trains used. However, it is also necessary to consider the distribution of customer shipment size. Although there are a large number of potential customers in a timeslot, each customer shipment may be large. Therefore, such a timeslot could allow only a few customers to be served on a train so giving a high violation cost (or a priority) to this timeslot is no longer reasonable. N_t is defined as

$$N_t = \frac{n_t}{n_{t(\max)}} \times 100 \quad \forall t \tag{30}$$

where $n_t = \frac{a_t}{\sum_{t=1}^{T} a_t} \quad \forall t$, with $a_t = \frac{\mu_t + \sigma_t}{C_t} \quad \forall t$, with μ_t the mean of customer shipment sizes in timeslot t, σ_t the standard deviation of the customer shipment sizes in timeslot t, and C_t the number of customers in timeslot t.

The violation cost S_t. Although the virtual loss of revenue in the generalised cost function could represent customer satisfaction in terms of a monetary unit, it is an indirect cost. In practice, the indirect cost is not obvious for rail expenditure as it affects the long-term financial plan. Therefore, in a competitive transport market, the direct cost that affects the short-term cash flow is regarded as more important. Since satisfaction probability represents customer satisfaction, we can sum up the satisfaction weight for the violation cost of each timeslot. S_t is defined as

$$S_t = \frac{s_t}{s_{t(\max)}} \times 100 \quad \forall t \tag{31}$$

where $s_t = \frac{b_t}{\sum_{t=1}^{T} b_t} \quad \forall t$, with $b_t = \frac{\sum_{t=1}^{T} W_t}{W_t} \quad \forall t$, and W_t a total customer satisfaction weight in timeslot t.

The violation cost E_t. A rail carrier may have different operating costs for different timeslots. The operating costs comprise train congestion cost and staff cost. Although a train schedule is a set of timeslots, we could consider E_t for the operating costs of each timeslot directly. This is because the operating cost is a cost unit and does not affect the number of timeslots in the optimal train schedule. The lower the operating costs for the timeslot, the higher the chance that the timeslot would lead to a schedule with the minimum generalised cost. E_t is defined as

$$E_t = \frac{e_t}{e_{t(\max)}} \times 100 \quad \forall t \tag{32}$$

where $e_t = \frac{U_t}{\sum_{t=1}^{T} U_t} \quad \forall t$.

5. COMPUTATIONAL RESULTS

The container rail scheduling model was tested on two sets of four successive weeks data from the eastern-line container service of the Royal State Railway of Thailand (SRT) and 184 shipping companies (customers). Each train has a capacity of 68 containers. The problem instances are summarised in Table 3.

Table 3. Problem instances (θ is a lower bound on number of trains).

Test case	Customer	Container	θ	SRT schedules Trains	SRT schedules Capacity	Supply–Demand Capacity	Supply–Demand 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
W5	225	4115	61	73	4964	816	12
W6	198	3350	50	59	4012	612	9
W7	126	2542	38	49	3332	748	11
W8	286	4731	70	86	5848	1088	16

Table 4. Comparative results. The unit cost $\times 10^6$ Baht (Thai currency), OC: operating costs, VC: virtual loss of revenue, GC: generalised cost.

Test case	SRT cost	CBS cost OC	CBS cost VC	CBS cost GC	OC Reduction (%)
W1	5.17	4.51	1.05	5.56	12.77
W2	3.74	3.66	0.83	4.49	2.13
W3	2.19	1.86	0.33	2.19	15.07
W4	4.48	3.66	0.72	4.38	18.30
W5	6.49	5.88	2.18	8.06	9.40
W6	5.25	5.11	1.62	6.73	2.66
W7	4.66	4.25	0.87	5.12	8.79
W8	7.65	6.48	2.79	9.27	15.29

These eight instances were solved with the CBS algorithm described in Section 3.1 on a Pentium III 1.5 GHz. Each test case is run ten times using different random number seeds at the beginning of each run. If no improvement has been achieved within 2000 iterations, the search will terminate. For all test cases, we set $R = 20$, $(h_m, h_c, h_s) = 1, 1, 100$ respectively. Table 4 compares the model results with current practice.

Table 4 shows, in all the test cases, there are some reductions in terms of the number of trains and operating costs, but these are not considerable. This is because in practice the SRT schedule is not fixed at the same service level everyday. The rail carrier normally cuts down the number of train services with 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.

Table 5. Results obtained by CBS and PCM.

Test case	CBS schedule Train Avg.	Cost* Avg.	Time (s)	PCM schedule Train Avg.	±	Cost* Avg.	SD.	Time (s)
W1	51	5.56	230	47	2	4.18	0.56	105
W2	39	4.49	117	39	2	3.84	0.43	83
W3	24	2.19	74	24	1	2.09	0.27	61
W4	43	4.38	96	41	1	3.90	0.29	79
W5	70	8.06	882	66	3	7.02	0.72	351
W6	59	6.73	310	54	2	5.99	0.61	150
W7	46	5.12	145	43	1	4.93	0.25	92
W8	82	9.27	1170	75	3	8.15	0.66	509

* The generalised cost ($\times 10^6$ Baht, Thai currency)

However, the proposed model maximises customer satisfaction—in other words, minimises the virtual loss of future revenue within a generalised cost framework. Therefore, the schedule obtained by CBS could reflect the maximum degree of customer satisfaction with the minimum rail operating costs through a demand responsive schedule.

In addition, we demonstrate the performance of the constraint-based search incorporating the predictive choice model (PCM), and compare the results with CBS alone. The same test cases are run ten times using different random numbers at the beginning of each run. If no improvement has been achieved within 2000 iterations, the search will terminate. For all test cases, we set $R = 20$, $(h_m, h_c, h_s) = 1, 1, 100$ respectively, the number of flip trials $N = 20$, choice specific parameter $\beta_1 = 0.05$, decision method parameter $D = 75$, the number of fixing iterations $F = 100$, the number of fixing departure timeslots $x_t = 50$, the number of fixing selected booking timeslots $(y_{tj} = 1) = 50$, and the number of fixing unselected booking timeslots $(y_{tj} = 0) = 200$. The last three parameters govern the maximum number of variables which are allowed to be fixed at a point in the search and are necessary to allow some flexibility in the search. The results for the test cases are shown in Table 5.

Table 5 shows that the results of PCM are better than that of CBS alone. On average, the PCM schedule is 6.04% better in terms of the number of trains, and gives a reduction of 12.45% in generalised cost. Although in PCM learning from the search history implies a computational overhead over CBS, it is offset against a lower run-time required to find near optimal schedules, in particular for large test cases.

6. CONCLUSIONS

The ability to find a profitable train schedule along with satisfying customer demand using demand consolidation leads to some reductions in total operating costs, and enhances the level of customer service through demand responsive schedules.

The viability of using a CSP representation of the problem and solving this problem by CBS has been shown. CBS only relies on a simple local move and could accommodate new conditions without any change in the algorithm's structure. To achieve effective performance, problem-specific violation on a generalised scale is simply applied to constraints. The problem is first severely constrained so that few feasible solutions are likely to exist, the variables therefore would take consistent values in most of the feasible solutions. A new learning model is then introduced to predict a likely optimal value for those variables in order to help CBS target optimality in a probabilistic way. The predictive choice learning model is developed on theoretical grounds, using an analogy with discrete choice theory. The experimental results for the container rail service planning problem have demonstrated that CBS is a convenient and effective tool in producing good solutions, particularly when the predictive choice model is incorporated.

References

Aardal, K. (1998) Capacitated facility location: separation algorithm and computational experience, *Mathematical Programming*, **81**:149–175.

Abramson, D. and Randall, M. (1999) A simulated annealing code for general integer linear programs, *Annals of Operation Research*, **86**:3–24.

Anderson, P. S., Palma, A. and Thisse, J. (1992) *Discrete Choice Theory of Product Differentiation*, MIT Press, Princeton, NJ.

Arshad, F., Rhalibi, A. and Kelleher, G. (2000) *Information Management within Intermodal Transport Chain Scheduling*, European Project PISCES Report, Liverpool John Moores University.

Beasley, J. E. (1988) An algorithm for solving large capacitated warehouse location problems, *European Journal of Operational Research*, **33**:314–325.

Ben-Akiva, M. and Lerman, S. R. (1985) *Discrete Choice Analysis: Theory and Application to Predict Travel Demand*, MIT Press, Princeton, NJ.

Boffey, T. B. (1989) Location problems arising in computer networks, *Journal of the Operational Research Society*, **40**:347–354.

Connolly, D. (1992) General purpose simulated annealing, *Journal of the Operational Research Society*, **43**:495–505.

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

Gomes, C. P., Selman, B. and Kautz, H. (1998) Boosting combinatorial search through randomisation. In *Proceeding of AAAI-98*, AAAI Press, Menlo Park, CA.

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.

Henz, M., Lim, Y. F., Lua, S.C., Shi, X. P., Walser, J. P. and Yap, R. (2000) Solving hierarchical constraints over finite domains. In *Proceedings of the 6th International Symposium on Artificial Intelligence and Mathematics* (Fort Lauderdale, FL).

Hansen, E.R. (1992) *Global Optimisation Using Interval Analysis*, Dekker, New York.

Horvitz, E., Ruan, Y., Gomes, C., Kautz, H., Selman, B. and Chickering, M. (2001) A Bayesian approach to tackling hard computational problems. In *Proceedings of UAI01*, pp. 235–244.

Huntley, C. L., Brown, D. E., Sappington, D. E. and Markowicz, B. P. (1995) Freight routing and scheduling at CSX transportation, *Interfaces*, **25**:58–71.

ILOG (2000) ILOG OPL Studio Version 3.5.1 Reference Manual, ILOG Inc.

Indra-Payoong, N., Srisurapanon, V. and Laosirihongthong, T. (1998) Factors influencing modal choice for freight transportation. In *Proceedings of the Civil and Environmental Engineering Conference, New Frontiers and Challenges (Bangkok, Thailand)*, **4**19–26.

Johnson, R. and Wichern, D. (1996) *Applied Multivariate Statistical Analysis*, Cambridge University Press, Cambridge.

Kallenberg, O. (1997) *Foundations of Modern Probability*, Springer, New York.

Kearfott, R. B. (1998) On proving existence of feasible points in equality constrained optimisation problems, *Mathematical Programming*, **83**:89–100.

Kochmann, G. A. and McCallum, C. J. (1981) Facility location models for planning a transatlantic communications network, *European Journal of Operational Research*, **6**:205–211.

Kraft, E. R. (2002) Scheduling railway freight delivery appointments using a bid price approach, *Transportation Research*, **36A**:145–165.

Larraaga, P. and Lozano, J. A. (2002) *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*, Kluwer, Dordrecht.

Lau H. C., Lim, Y. F. and Liu, Q. (2001) Diversification of neighbourhood via constraint-based local search and its application to VRPTW. In *Proceedings of CP-AI-OR*, pp. 361–374.

Marder, E. (1997) *The Laws of Choice: Predicting Customer Behavior*, The Free Press, Simon and Schuster, New York.

Marin, A. and Salmeron, J.(1996) Tactical design of rail freight networks. Part II: local search methods with statistical analysis, *European Journal of Operational Research*, **94**:43–53.

McAllester, D., Selman, B. and Kautz, H. (1997) Evidence for invariants in local search. In *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI-97)*, pp. 459–465.

Ministry of Commerce (2002) *Thai Commodity Market Price*, Department of Business Economics, Ministry of Commerce, Thailand.

Newman, A. M. and Yano, C. A. (2000) Scheduling direct and indirect trains and containers in an intermodal setting, *Transportation Science*, **34**:256–270.

Nonobe, K. and Ibaraki, T. (1998) A tabu search approach to constraint satisfaction problem as a general problem solver, *European Journal of Operational Research*, **106**:599–623.

Parkes, A. and Walser, J. (1996) Tuning local search for satisfiability testing. In *Proceedings of AAAI-96*, pp. 356–362.

Patrick, R. (1982) Algorithm AS 181: The *W* test for normality, *Applied Statistics*, **31**:176–180.

Proll, L. and Smith, B. (1998) ILP and constraint programming approaches to a template design problem, *INFORMS Journal of Computing*, **10**:265–277.

Resende, G. C. and Ribero, C. C. (2001) *Greedy Randomized Adaptive Search Procedures, State-of-the Art Handbook in Metaheuristics*, F. Glover and G. Kochenberger (Eds.), Kluwer, Dordrecht.

Selman, B., Levesque, H. and Mitchell D. (1992) A new method for solving hard satisfiability problems. In *Proceedings of AAAI-92*, pp. 440–446.

Selman, B., Kautz, H. and Cohen, B. (1994) Noise strategies for improving local search. In *Proceedings of AAAI-94*, pp. 337–343.

Shapiro, S. S. and Wilk, M. B. (1965) An analysis of variance test for normality (complete samples), *Biometrika*, **52**:591–611.

Smith, B., Stergiou, K. and Walsh, T. (2000) Using auxiliary variables and implied constraints to model non-binary problems. In *Proceedings of AAAI-2000*, Austin, TX.

Trotter, H. F. (1959) An elementary proof of the central limit theorem, *Archives of Mathematics*, **10**:226–234.

Walser, J. P. (1999) *Integer Optimisation by Local Search: A Domain-Independent Approach*, Lecture Notes in Artificial Intelligence, Vol. 1636, Springer, Berlin.

Yano, C. A. and Newman, A. M. (2001) Scheduling trains and containers with due dates and dynamic arrivals, *Transportation Science*, **35**:181–191.