Abstract

The project is addressing the problem faced by the aviation industry while scheduling its operations. The crew scheduling problem is often divided into crew pairing and crew assignment due to the large size of the problem and the complexity of constraints formed due to safety standards enforced by regulating authorities and labor unions. There are two types of formulations commonly used for these problems, known as a set covering problems and set partitioning problems. We have explored the set covering and set partitioning formulation in this project and tried to solve the problem using the column generation approach and multi-label shortest path algorithm to generate suitable schedules which satisfy all constraints.

Contents

1. Introduction

1.1 Motivation

The aviation industry is receiving extensive attention from many operation researchers. The aviation industry operations are observing high growth due to various technological advances, and it has a significant impact on other industries like tourism and aircraft manufacturing [1]. The primary source of revenue for the aviation industry is via passenger tickets, and expenses include fuel costs, labour cost, repairing and maintenance cost and other operational costs. According to A. *Kasirzadeh et al.* (2017), the significant expenses in the aviation industry are fuel cost and labour cost. Data from the Air transport association (2008) indicate that the most significant expense is due to fuel costs and labour cost are the second largest cost, approximately 23.4% of the total cost. Due to such high cost, there is an immense cost-saving potential for generating optimum schedules. Such schedules can provide an industry edge over other competitors in the field. Due to expanding air travel, many industries are emphasizing low-cost flight tickets to attract more customers. To compete with this, minimizing operation costs has been critical to many industries. Hence this airline crew scheduling is receiving much attention from researchers as well as industries.

1.2 Terminologies

Kasirzadeh et al. (2017) have provided some terminologies while addressing the crew scheduling problem, and we have used the same terminologies in this project. The terminologies include:

Air leg: A non-stop flight segment and contains information about arrival time and origin airport, departure time and destination airport and flight number [1].

Duty: A sequence of airlegs that forms a working day for crew [1].

Deadhead: Air leg in which crew flies as a passenger to complete next duty [1].

Crew members: There are two types of crew members present in an aircraft: cockpit crew and cabin crew. Cockpit crew consists of pilot, co-pilot and flight engineer and cockpit crew can be assigned to specific aircraft only [1].

The cabin crew consists of cabin captain and flight attendants, and the cabin crew can be assigned to any aircraft. In this project, we have only considered cockpit crew scheduling as the cockpit crew receives more incentives and salary due to their skillset. Optimizing their schedule can be very beneficial [1]. Crew members indicate cockpit crew members only in this report from this point forward .

Pairings: A sequence of duties and layovers for a crew member that starts and ends at a base [1].

1.3 What makes Crew scheduling difficult?

According to M.R. Garey and D.S. Johnson (1979), The crew scheduling problems are combinations of complex combinatorial optimization sub-problems which belong to class NP-Hard and NP-Complete [2]. A domestic problem on a hub-and-spoke network with several hundred flights typically has billions of pairings. Solving set partitioning formulation requires the explicit enumeration of all pairings, making this problem one of the most challenging optimization problems. Also, there are various constraints placed by regulatory authorities like DGCA (Directorate General of Civil Aviation) for India and ICAO (International Civil Aviation Organization) to ensure the safety of passengers and crew. In addition, many labour unions also have some contracts with industries to ensure the well-being of employees (in our case, the crew). This constraint increases the complexity of the problem. These rules include the restriction on the minimum number of duty assigned to a crew member, the maximum number of flights taken in a period, minimum rest period to be provided before the next duty to each crew member.

1.4 Approaches to solving the problem

Generally, the crew assignment problem is modelled using either set covering formulations or using Set partitioning formulations. More constraints are added depending on the rules enforced by authorities, unions and the crew's preference.

These are large scale mixed-integer problems, and solution methods include, Lagrangian relaxation (Geoffrion,1974), Benders decomposition (Benders 1962) and branch-and-price (Desaulniers et al. 1998).

2. Problem Statement

The airline crew scheduling problem is often divided into crew pairing problem and crew assignment problem due to the large size of the problem and very high complexity. Crew pairing problems are associated with finding a set of minimum-cost pairings such that each scheduled flight over the time horizon is included in precisely one pairing [1]. Crew assignment combines the generated pairings with vacations, training schedules and other breaks for the crew members [1].

We have found out good crew pairings satisfying all constraints and having minimum cost over a timeline considering ten flights. We generated the crew schedules for individual crew member based on their vacation preferences, pre-scheduled training and satisfying all rules imposed by regulatory authorities. There are two types of schedules possible in crew assignment :

1. Bidline schedules: In these type of schedules, pairings are generated anonymously and then announced to the crew. Based on their preference, the crew bids on these schedules and according to the bids, the final schedule allocation is completed [1].

2. Personalized schedules: In these individual crew member's preference for a particular air leg is taken into consideration, and also their vacations and other activities are considered before generating final schedules [1]. There are two types of personalized schedules :

   A. Rostering: the aim is to achieve maximum crew members satisfaction for generated schedules.

   B. Seniority-Based: priority is given to the satisfaction of senior crew members.

To solve the set partitioning problem, we used a column generation approach with a multi-label shortest path algorithm to find valid crew pairings. We have generated a schedule based on bid line schedules and studied an extended formulation of the set covering problem from *Kasirzadeh et al.* (2017).

3. Crew Pairing

Crew pairing models are usually formulated as set partitioning problems, in which we want to find a minimum cost subset of the feasible pairings such that every flight segment is included in exactly one chosen pairing.

3.1 Basic Set Partitioning Problem

$$\min \sum_{p \in P} c_p y_p$$

$$\sum_{p: i \in p} y_p = 1 \qquad i \in F$$

$$y_p \in \{0,1\} \qquad p \in P\,.$$

F: Set of flight segments to be covered.

P: Set of all feasible pairings.

$\boldsymbol{c_p}$ **:** Cost of pairing p ∈ P

y_p **:** Decision variable is equal to 1 if pairing p is included in the solution, and 0 otherwise

This formulation requires the explicit enumeration of all pairings.

Enumerating pairings can be challenging because of the numerous work rules that must be checked to ensure legality and, more importantly, because of the huge number of potential pairings. In fact, for most real instances, explicit enumeration of the constraint matrix is not possible. For example, a domestic problem on a hub-and-spoke network with several hundred flights typically has billions of pairings.

Thus, heuristic local optimization approaches or column generation methods are used to solve all but the smallest of problem instances. This basic set partitioning model is used for all three phases of crew pairing optimization. The models differ in the set of flights F that define the constraints of the problem. For the daily problem, there is one constraint for each flight that is repeated four or more times per week. The underlying assumption in solving this problem is that each pairing in the solution will be flown starting each day of the week. Recall that this presupposes that pairings are constrained to cover a flight leg at most once.

3.2 Personalized Assignment Problem

Given a minimum cost set of pairings that covers all the definite flights in the planning horizon, the personalized pilot assignment problem finds minimum cost schedules such that the pairings are covered precisely once and at least given numbers of the flight and vacation preferences are fulfilled [1].

C_s : Cost of personalized schedule s for pilot l

$$C_s^l = \sum_{p \in P} e_p^{s,l} C_p + n_s^l . c_f^l + \sum_{v \in V_l} (1 - v_v^{s,l}). c_v^l$$

Minimize

$$\sum_{l \in L} \sum_{s \in S_l} C_s^l x_l^s + \sum_{f \in P} e_p^- C_f^- \quad \textbf{(i)}$$

Subject to

$$\sum_{l \in L} \sum_{s \in S_l} e_p^{s,l} x_l^s + e_p^- = 1, \forall p \in P \quad \textbf{(ii)}$$

$$\sum_{l \epsilon L} \sum_{f \epsilon B} \sum_{s \epsilon S_l} \sum_{p \epsilon P} e_f^p e_p^{s,l} x_l^s >= u \quad \textbf{(iii)}$$

$$\sum_{l \in L} x_l^s <= l \, \forall \, l \, \epsilon \, L \quad \textbf{(iv)}$$

$$\sum_{l \epsilon L} \sum_{s \epsilon S_l} \sum_{v \epsilon V_l} v_v^{s,l} x_l^s >= w \quad \textbf{(v)}$$

$$x_l^s \epsilon \{0, 1\} , \forall \, l \in L, \forall s \in S_l \quad \textbf{(vi)}$$

The main objective function **(i)** minimizes the total cost of the schedules and penalty costs for uncovered flights. Constraint **(ii)** ensures that each and every pairing is covered exactly once. Constraint **(iii)** is a global constraint on the minimum number of favored flights.

Constraint **(iv)** is a global constraint on the minimum number of satisfied vacation preferences. Constraint **(v)** ensures that at most one schedule is chosen for each pilot, and constraint **(vi)** is the integrality condition (Atoosa Kasirzadeh,2017)

$$e_p^{s,l} = \begin{cases} 1 & \text{if pairing } p \in P \text{ is covered by pilot } l \in L \text{ in personalized schedule } s \in S_l \\ 0 & \text{otherwise;} \end{cases}$$

$$\bar{e}_p = \begin{cases} 1 & \text{if flight } f \in P \text{ is not covered} \\ 0 & \text{otherwise;} \end{cases}$$

$$e_f^p = \begin{cases} 1 & \text{if flight } f \in F \text{ is covered by pairing } p \in P \\ 0 & \text{otherwise;} \end{cases}$$

$$v_v^{s,l} = \begin{cases} 1 & \text{if vacation } v \in V_l \text{ for pilot } l \in L \text{ is covered by schedule } s \in S_l \\ 0 & \text{otherwise;} \end{cases}$$

Variables

$$x_l^s = \begin{cases} 1 & \text{if schedule } s \in S_l \text{ is chosen for pilot } l \in L \\ 0 & \text{otherwise;} \end{cases}$$

(image source : *Kasirzadeh et al.* (2017))

Sets

F is the set of all scheduled flights to be covered

L is the set of all pilots

V_l *is the* set of preferred vacations for pilot l

P is the set of pairings

S_l is the set of all feasible personalized schedules for pilot l

B_l is the set of preferred flights for pilot $l \in L$

Parameters

C_p is the cost of pairing p

C_f^- is the penalty cost for not covering flight f

C_s^l is the cost of schedule s for pilot l

n_s^l is the number of preferred flights in schedule s

C_f^l is the bonus cost for covering preferred flight $f \in B_l$

C_s^l is the penalty cost for not covering preferred vacation v

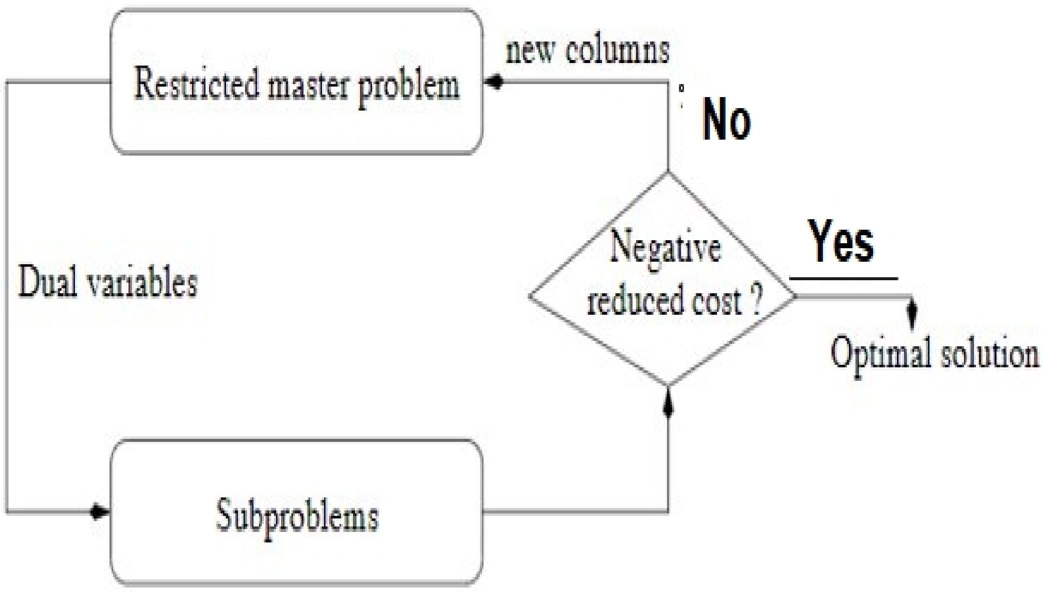
u is the minimum number of preferred flights in schedule

w is the minimum number of preferred vacations to be covered

4. Column Generation

Column Generations a technique for solving (mixed) integer programming problems with larger number of variables or columns. This technique was first applied to large real-life cutting stock problem by Gilmore and Gomory. The problem is split into two problems: master problem and subproblem. Master problem contains only a subset of variables. The subproblem is created to decide which variable/column will enter the master problem. The objective function of subproblem is the reduced cost of new variable w.r.t to the current dual variables, and constrain require variable to satisfy original constraints.

Once the master problem is solved, we find the dual variables corresponding to each constraint. These dual variables are used in objective function of sub-problem. If the objective value of the sub-problem is negative then a variable has a negative reduced cost. This variable is entered in master problem and again master problem is resolved. Now we get new dual variables and we repeat the process until no negative reduced cost variable is found and we reach optimality.

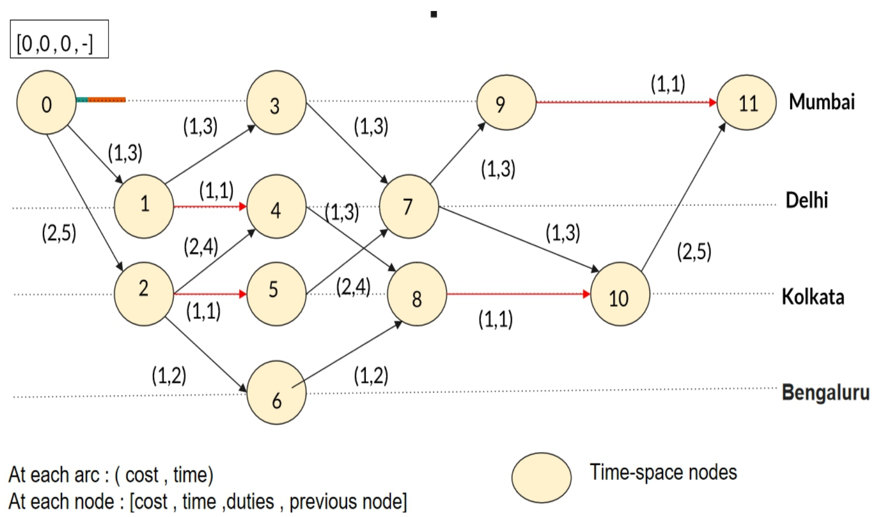


5. Multi-label Shortest path Algorithm

We have to solve a column generation subproblem for each pilot to obtain a valid crew assignment that satisfies all legal constraints imposed by regulatory authorities and obeys labour unions contracts with industries. This sub-problem can be represented on a directed acyclic time-space graph with each arc indicating duties, and nodes are transition moments between duty periods located at a given airport at a given time.

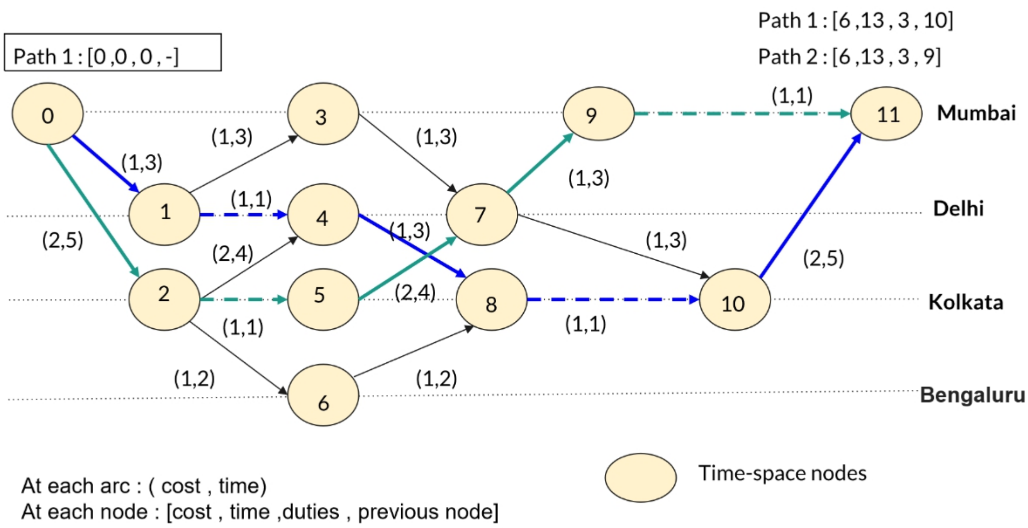
To find duties that can be combined following all constraints, we need to use the shortest path algorithm, but a simple shortest path algorithm will not work as we need to find the shortest path considering some constraints. Multi-label shortest path algorithm considers finding shortest path having minimum cost while keeping bound on some other parameters. We only consider the path that has labels within the limits of parameters. If two-path have the same labels, we use the label dominance rule to discard suboptimal paths. We say label one is dominated by label two if the cost component is lowest at label one and all other parameters are also within bounds given.

Consider an example of 5 days pairing problem with four airports, say, Mumbai, Delhi, Kolkata and Bengaluru, where we have to obtain the minor cost schedules with the constraint that no more than three duties are allowed per pairing and total flying time should not be longer than 15 hours, with Mumbai as the base airport. Suppose we can represent the possible pairing as below graph :



At each arc, we are given the cost of following that path and the flying time required to follow the path.

At each node, we are given a label that includes total cost up to that node, total duties covered till that node, total flying time required to reach that node and previous node information. Label for starting node is set to all 0 values. We update new labels using arc values, and the label dominance rule is used to select the next node. After Solving, we obtain results as followed :



We obtained following two paths : path1 - node{0,1,4,8,10,11} path2 - node{0,2,5,7,9,11}.For both these path by multi-label shortest path algorithm, we can check that final label at destination have same minimum cost of going from node 0 to node 11. Also, both nodes are at the same airport: Mumbai; therefore, the base airport constraint is also satisfied. The total flying time is also less than 15 hours for both path, and total duties are also less than 3. Hence we can declare these two paths as the valid schedule. In bid line schedules, these are made available to all crew members, and depending on bids, each schedule can be assigned to them. In contrast, in a personalized schedule, an additional label of holidays can be included to generate the feasible schedule.

6. Results and discussion

Example1

Flight #	Orig	Dest	Start	End	Frequency
1	A	B	08:00	09:00	not 67
2	B	C	10:00	11:00	
3	C	D	13:00	14:00	not 7
4	C	A	15:00	16:00	
5	D	A	15:00	16:00	not 6
6	A	B	17:00	18:00	
7	B	C	11:00	12:00	not 67

Table shows the schedule of 7 flights. Airports are represented as A, B, C, and D. Frequency represents the frequency of flights in a week (e.g., not 67 means no flight on Saturdays and Sundays).

Valid duty periods are -

$$D_1 = \{1\}, D_2 = \{2\}, D_3 = \{3\}, D_4 = \{4\},$$
$$D_5 = \{5\}, D_6 = \{6\}, D_7 = \{7\}\ D_8 = \{1, 2\}$$
$$D_9 = \{1, 7, 3\}\ D_{10} = \{2, 3\}$$

A valid duty consists of flights such that-

- Two adjacent flights will have common arrival and departure airports.
- No clash in time

Airport A, C, and D are crew bases, we can find valid pairings. Each pairing must start and end at same crew base.

$$P_1 = \{D_4, D_8\}, P_2 = \{D_9, D_5\},$$
$$P_3 = \{D_5, D_6, D_{10}\}, P_4 = \{D_4, D_6, D_7\},$$
$$P_5 = \{D_1, D_7, D_4\}, P_6 = \{D_5, D_7, D_9\}$$

Additional constraints are :

$$3 <= 4y_2 + 3y_5 <= 6,$$
$$0 <= 3y_1 + 3y_4 <= 5,$$
$$3 <= 4y_3 <= 6$$

Optimal pairings are P3 and P5.

To extend this solution for weekly problem we need to solve the weekly exception problem to repair broken paring.

	P3	P5
Friday	5	1
Saturday	6	
Sunday	6,2	4
Monday	2,3	7,4
Tuesday		4

First row of the table shows that flight 5 in P3 cannot be scheduled on Friday because flights 3 cannot be scheduled on Sunday.

<u>Example-2</u>

To understand the set covering problem, we have tried an example on python from a textbook **[8]**

	Activity (Feasible Sequence of Flights)							
Requirement (Flight)	1	2	3	4	5	6	7	8
1 New York to Buffalo	1			1			1	
2 New York to Cincinnati		1			1			
3 New York to Chicago			1			1		1
4 Buffalo to Chicago	2			2				
5 Chicago to Cincinnati			2	3		2		
6 Cincinnati to Pittsburgh		2		4		3		
7 Cincinnati to Buffalo			3		2			
8 Buffalo to New York			4		3		2	
9 Pittsburgh to New York		3		5		4		
10 Chicago to New York	3							2
Cost ($1000) for each sequence	5	4	4	9	7	8	3	3

(Image source : Chen and Der-San (1940) [8])

Input parameters: 0-1 incident matrix(A), cost for each feasible sequence of flights (c_j, j = 1,2,..,8), requirement vector (b_j= 1, I=1,2,…,10).

Decision variable: one 0-1 variable for each sequence of flight (y_j = 1 or 0, j=1,2,…,8).

Constraint: one constraint for each requirement or flight.

Objective: minimize the total cost of assigning crews to the selected sequence of flights.

Let y_j = 1 if a crew is assigned to jth sequence of flights and 0 otherwise. To ensure that at least one crew is assigned to the first flight, we have constraint

$y_1 + y_4 + y_7 \geq 1$ (requirement 1 constraint)

Doing this for all 10 flights. Get the IP model as-

Minimize z =$5y_1 + 4y_2 + 4y_3 + 9y_4 + 7y_5 + 8y_6 + 3y_7 + 3y_8$

An optimal solution to the problem solved in python using GLPK(GNU Linear Programming Kit) solver uses pairings ***1 ,2 and 3***, for a total cost of ***$13000***

Link to our ipython notebook is attached here for reference (Google Colab IDE) : https://colab.research.google.com/drive/1sOgYA3eQvEw6s1xRZni8rOllzo1MIPsg?usp=sharing

7. Future Aspects

- Meta-heuristics like Tabu search, Genetic algorithms, Simulated Annealing can be used.

- Benefits can be gained by developing more efficient schedules for cabin crews. This problem has received less attention than cockpit crew scheduling, both because cabin crews are significantly less costly and also because it is a much larger problem.

- Finally, and perhaps most challenging, is the integration of the crew pairing, fleet assignment, and schedule planning problems, especially since these problems are difficult to solve individually.

Conclusion

A. We obtained the constraints for all 10 flights. Combining these constraints with the objective function, we obtained the 0-1 Integer programming model.

B. An optimal solution to the problem solved in python using GLPK(GNU Linear Programming Kit) solver uses pairings ***1 ,2 and 3***, for a total cost of ***$13000***

C. We obtained following two paths: path1 – node {0,1,4,8,10,11} path2 – node {0,2,5,7,9,11}.For both these path by multi-label shortest path algorithm, we can check that final label at destination have same minimum cost of going from node 0 to node 11.

D. Hence, we can declare the found out paths as the valid schedule. In bid line schedules, these are made available to all crew members, and depending on bids, each schedule can be assigned to them. In contrast, in a personalized schedule, an additional label of holidays can be included to generate the feasible schedule.

References

[1] Kasirzadeh, Atoosa, Mohammed Saddoune, and François Soumis. "Airline crew scheduling: models, algorithms, and data sets." *EURO Journal on Transportation and Logistics* 6.2 (2017): 111-137.-

[2] Garey, Michael R., and David S. Johnson. "Computers and intractability." *A Guide to the* (1979).

[3] Geoffrion, Arthur M. "Lagrangean relaxation for integer programming." *Approaches to integer programming*. Springer, Berlin, Heidelberg, 1974. 82-114.

[4] Benders, J.F. Partitioning procedures for solving mixed-variables programming problems. *Numer. Math.* **4,** 238–252 (1962)

[5] Desaulniers, Guy, et al. "A unified framework for deterministic time constrained vehicle routing and crew scheduling problems." *Fleet management and logistics*. Springer, Boston, MA, 1998. 57-93.

[6] Reference : Barnhart, C., Cohn, A.M., Johnson, E.L., Klabjan, D., Nemhauser, G.L. and Vance, P.H., 2003. Airline crew scheduling. In *Handbook of transportation science* (pp. 517-560). Springer, Boston, MA.

[7] Wu, Wei, et al. "A column generation approach to the airline crew pairing problem to minimize the total person-days." *Journal of Advanced Mechanical Design, Systems, and Manufacturing* 10.3 (2016): JAMDSM0040-JAMDSM0040

[8] Chen, Der-San, 1940- Applied integer programming : modeling and simulation / Der-San Chen, Robert