This article was downloaded by: [175.35.111.175] On: 26 October 2025, At: 21:54
Publisher: Institute for Operations Research and the Management Sciences (INFORMS)
INFORMS is located in Maryland, USA



Operations Research

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

Integrated Simulation, Heuristic and Optimisation Approaches to Staff Scheduling

Andrew J. Mason, David M. Ryan, David M. Panton,

To cite this article:
Andrew J. Mason, David M. Ryan, David M. Panton, (1998) Integrated Simulation, Heuristic and Optimisation Approaches to Staff Scheduling. Operations Research 46(2):161-175. https://doi.org/10.1287/opre.46.2.161

Full terms and conditions of use: https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

© 1998 INFORMS

Please scroll down for article—it is on subsequent pages

With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org

INTEGRATED SIMULATION, HEURISTIC AND OPTIMISATION APPROACHES TO STAFF SCHEDULING

ANDREW J. MASON and DAVID M. RYAN

University of Auckland, New Zealand

DAVID M. PANTON

University of South Australia, Australia

(Received February 1996; revision received May 1997; accepted January 1998)

This paper details a new simulation and optimisation based system for personnel scheduling (rostering) of customs staff at the Auckland International Airport, New Zealand. An integrated approach using simulation, heuristic descent, and integer programming techniques has been developed to determine near-optimal staffing levels. The system begins by using a new simulation system embedded within a heuristic search to determine minimum staffing levels for arrival and departure work areas. These staffing requirements are then used as the input to an integer programming model, which optimally allocates full- and part-time staff to each period of the working day. These shifts are then assigned to daily work schedules having a six-day-on, three-day-off structure. The application of these techniques has resulted in significantly lower staffing levels, while at the same time creating both high-quality rosters and ensuring that all passenger processing targets are met. This paper charts the development of this system, outlines failures where they have occurred, and summarises the ongoing impacts of this work on the organisation.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

New Zealand Customs is responsible for, amongst other tasks, the processing of arriving and departing passengers at the Auckland International Airport. Two main activities are undertaken in processing passengers. The primary processing activities include the immigration function (being the examination of passports and travel documentation) and the simultaneous "profiling" of passengers. (Note that New Zealand is unusual in combining the immigration and customs functions in this manner.) These activities occur at the so-called primary lines on both arrivals and departures. Secondary processing includes the examination and searching of baggage, which in Auckland is undertaken for selected passengers after the primary line processing is completed.

The aim of this work is to investigate the staffing levels that are required at the primary line to meet the target performance specified in the New Zealand Customs Statement of Service Performance. Such performance is measured in terms of so-called "facilitation targets," which specify maximum permitted times for the processing of a flight's passengers. For the recommended staffing levels necessary to meet facilitation targets, a set of shift patterns involving both full-time and part-time shifts must be developed prior to the generation and publication of a complete roster.

The Customs staffing problem can be divided into two subproblems. The first is to determine the minimum number of staff required during each day to satisfy the service objectives. Once these work requirements have been determined, the second problem is to construct individual full-time and part-time stretches of work for each day, which, when taken together, simultaneously meet both the daily work requirements determined for the first problem and, when viewed as a work sequence, provide high quality rosters for the Customs staff.

Personnel scheduling problems have attracted considerable attention in the literature. The task of determining staff numbers and shift starting times throughout the day in order to provide minimum coverage has been well understood for many years, and is referred to as the *workforce allocation* problem. An early survey giving a detailed discussion of this problem is given in Baker (1976). While the workforce allocation problem in its simplest form can be modelled as a network flow, as in Segal (1974) and Bartholdi and Ratliff (1978), the addition of side constraints, such as break requirements, demands more complex procedures. A variety of techniques employing networks, heuristics, and integer programming have been employed. A brief survey of these methods, together with a method for handling breaks, can be found in Bechtold and Jacobs (1990). The second task of constructing individual full-time and part-time stretches of work for each day (roster construction) has also been studied by a large number of authors. A detailed survey of work carried out up until

Subject classifications: Organizational studies, manpower planning: modeling of customs staffing requirements. Labor: efficient scheduling of customs staff.
Area of review: OR PRACTICE.

0030-364X/98/4602-0161 $05.00
© 1998 INFORMS

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

the early 1980s can be found in Tien and Kamiyama (1982). In this survey the authors describe the personnel scheduling problem in terms of a five-stage process in which stage 1 is the workforce allocation problem discussed above. Stage 2 involves the determination of the total workforce requirement, given off-weekend and other recreational constraints; stage 3, the identification of recreation blocks; stage 4, the placement of these blocks to create a single shift schedule; and finally stage 5, the assignment of shifts to the single shift schedule. A variety of methods, including many heuristic procedures, have been described for the construction of personnel schedules. Many such techniques involve the creation of *cyclic* schedules in which personnel rotate from one line (week) to the next over a period of time equal to the number of lines (weeks) in the schedule. Methods for generating such cyclic schedules in relation to this work will be discussed in Sections 3 and 4.

1. DETERMINING THE WORK REQUIREMENTS

In this section we examine the problem of determining minimal customs staffing requirements during a day to meet the service requirements. The lack of flights in the early hours of each day allows the problem to be partitioned into distinct days. The approach taken in this work was to develop a new simulation model that permits specified staffing levels throughout the day to be quickly tested for feasibility. This simulation model then forms the core of a new greedy sequential heuristic that determines near-minimum staffing levels.

1.1. Staffing Specification

For the purposes of this work, each day is discretised into m 15-minute intervals, with staffing levels for the arrivals primary line and departures primary line being specified for each 15-minute period as m-dimensional vectors $A = (a_1, a_2, \ldots, a_m)$ and $D = (d_1, d_2, \ldots, d_m)$, respectively. These vectors form the input to the simulation model which then characterises the staffing plan as feasible or otherwise, where feasibility is defined in terms of the following requirements.

1.1.1. *Facilitation Targets.* The first requirement, for a staffing plan to be feasible, is that sufficient staff are available to complete the processing of each flight's passengers within the so called "facilitation targets." For arrivals, the Government-specified service requirements given in the Customs Statement of Service Performance specify that 85 percent of the passengers on each arriving flight must be processed through the primary line within one hour of the aircraft's arrival, with the remaining 15 percent being processed within the next 15 minutes. For departures, a staffing plan D is deemed feasible if the processing of each flight's passengers is completed no later than 10 minutes before that flight's scheduled departure time. It is the function of the simulation model to determine if these goals are met for some given staffing plan. A day's arrivals or departures staffing plan is deemed infeasible (or, more precisely, "flight infeasible") if it fails to meet the respective facilitation targets for any flight.

1.1.2. *Minimum Staffing Presence.* As well as satisfying the service requirements, it is also necessary to allocate Customs officers in order to maintain a minimum staffing level during the day. In particular, Customs management requires that at least one (possibly idle) Customs officer be stationed at the arrival processing booths from the first scheduled arrival time through to a fixed time after the last scheduled arrival time. Similarly, Customs requires that at least one customs officer be present at the primary departure line from a specified time before the first scheduled departure time of the day until the departure of the last flight.

1.1.3. *Re-Queuing Restrictions.* In addition to the facilitation targets and minimum staffing levels, New Zealand customs also requires that staffing levels at arrivals should not be reduced during busy periods. Because passengers queue in front of individual customs booths, the reduction of staff and associated closure of a processing booth entails moving queued passengers from one queue to the end of another booth's queue. The re-queuing of large numbers of passengers in this way is deemed unacceptable by customs management. However, booths can be closed when queues are short, so this is permitted in the model. In general, if $Q(A) = (q_1, q_2, \ldots, q_m)$ gives the number of passengers queued at the end of each 15-minute period under some arrivals staffing plan A, then q_i/a_i gives the approximate number of passengers queued in front of *each* booth at the end of period i. If the staffing level is reduced at the end of period i, i.e., for some i, $a_{i+1} < a_i$, then the total number of passengers forced to re-queue at another booth is given by $r_i = (q_i/a_i) \cdot (a_i - a_{i+1})$. An arrivals staffing plan A is deemed infeasible (or, more precisely, "re-queuing infeasible") if $r_i > r_{\text{limit}}$ for any $i: a_{i+1} < a_i$, where r_{limit} is a constant giving the maximum permitted re-queuing passenger count.

1.2. Simulation

The new simulation model developed in this work is able to determine whether some given staffing level A (or D) meets the above feasibility requirements. The development of a simulation model such as this requires a detailed understanding of passenger behaviours at the airport for both arrivals and departures. This information is required to predict the times at which arrival and departure passengers join the customs queues, and thus make themselves available for processing by customs.

Unlike most international airports, passengers arriving at Auckland collect their luggage before they join the customs primary line queues. Thus, to predict passenger behaviour it is first necessary to model the unloading of baggage from the aircraft and the transfer of that baggage to the baggage carousels. The collection process is further complicated by the tendency of families and friends to wait

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

until they have all collected their baggage before the group proceeds to the Customs processing area. On the departures side, passengers typically arrive at the airport two hours before their flight's scheduled departure time, proceed through the associated airline's check-in procedure, and then perhaps spend time shopping or dining before proceeding through the Customs departures primary line. Thus, the time at which departing passengers become available for processing by Customs depends on the distribution of passenger arrival times at the airport, the rate of airline check-in, and the length of time passengers spend shopping and dining. Fortunately, simulation models describing these arrivals and departures behaviours have been developed by Auckland International Airport Limited (AIAL) (Kerridge 1993). This system has been made available to New Zealand Customs and proved invaluable in accelerating the development of the Customs staffing model. (A joint project between AIAL, Customs, and Dr. David Whitaker of Waikato University, New Zealand, has recently upgraded these simulation models; see Whitaker 1995.) Given a daily schedule, the AIAL simulation models produce deterministically calculated minute-by-minute records for each flight, giving the number of passengers presenting themselves at the Customs arrivals and departures primary lines. These data are independent of Customs staffing levels, and thus a single run of the AIAL simulation is used to generate the input data for the new Customs simulation model. The new Customs simulation can then be run repeatedly to rapidly test different staffing levels for feasibility; the value of this will become apparent in Section 1.3.

Once the passenger queuing behaviour has been determined, the system needs to simulate the processing of these passengers by Customs. This simulation requires data on the mean processing rates on arrivals and departures. In the model used by AIAL, arriving passengers were classified as either New Zealand/Australian (nonrestricted) or Restricted, so two processing rates were used in the new Customs simulator for arriving passengers. Similarly, the availability of a fast Blue Lane facility at departures required the specification of two departures processing times. When matched with passenger data, a mean per-passenger processing time for each flight can be calculated for use in the Customs simulation model. (Note that these figures are increased by some fraction of their standard deviations to add robustness to the system.)

Given these input data, the new Customs simulation program uses an underlying deterministic model to step through 24 hours of passenger processing for some given staffing level. Where facilitation targets are missed or excessive re-queuing is detected, this is reported back to the user. This simulation model forms the core of a new optimisation heuristic. Figure 1 shows the steps this heuristic follows in converting a flight schedule into a daily work requirement. These steps are discussed in detail below.

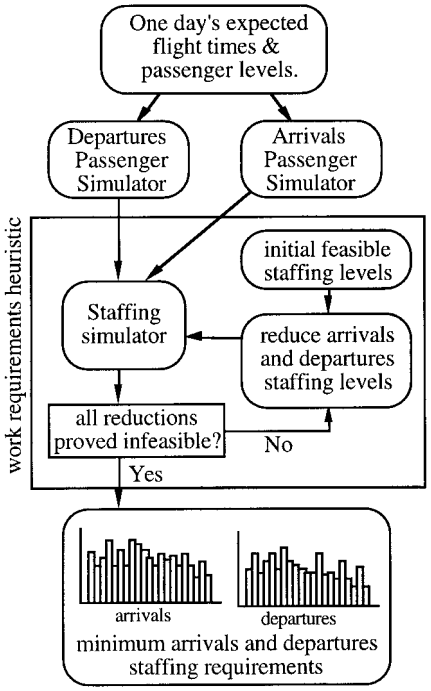


Figure 1. Overview of the new work requirements heuristic.

1.3. Work Requirements Heuristics

Because Customs staff are readily able to move between the arrivals and departures primary lines, the new work requirements heuristic seeks to determine two feasible staffing plans, one for arrivals and one for departures, that when taken together give a minimum staffing requirement for the day under consideration. These minimum staffing plans give the day's effective staffing requirements.

Because officers must work some minimum shift length, it is likely that the greatest inefficiencies will occur in the staffing of short peaks in demand. Therefore, the heuristic seeks solutions that are optimal in the lexicographical sense of first minimising the number of intervals in which the greatest number of officers are required, and then the number of intervals requiring one fewer officers, then two fewer officers, and so on. To be more specific, if $g_j(A + D) = |i : a_i + d_i = j, i \in \{1, 2, \ldots, m\}|$ denotes the number of times a total of j officers are on duty (i.e., the number of intervals in which $a_i + d_i = j$), and $p = \max_{i=1,2,\ldots,m}(a_i + d_i)$ is the maximum number of officers on duty in the solution, then we seek a solution which minimises firstly g_p, then g_{p-1}, then g_{p-2} and so on. Note that there will, in general, be many optimal solutions to this problem.

1.3.1. *Departure Work Requirements Heuristic.* Consider first the optimising of departures staffing levels in isolation, i.e., without consideration of the same day's arrivals requirements, given a fixed passenger queuing distribution generated by the AIAL simulation model. Now, a departures plan D is feasible if sufficient staff are available to ensure all flights meet their facilitation targets and at least

ALGORITHM 1. Departures Work Requirement Algorithm

Let D be a starting feasible solution.

Form a candidate list $C = \{1, 2, 3, \ldots, m\}$. (As the algorithm proceeds, C will be updated by deleting the indices of those periods in which further staff reductions have been shown to give infeasible solutions. Thus, C will contain indices of periods in which further staffing decreases are likely to give feasible solutions.)

repeat

  Let $c = f_D(D, C)$ be the index of the next period to reduce
  If the simulation model shows $D'(c) = (d_1, d_2, \ldots, d_{c-1}, d_c - 1, d_{c+1}, \ldots, d_m)$ to be feasible then
    Let $D \; 4 \; D'(c)$
  else
    Remove c from C, i.e. let $C \; 4 \; C\backslash c$

until $C = \{\}$

Staffing plan D is a dominant minimum.

one staff member is available throughout the operational hours. (Note that there are no re-queuing restrictions at departures.) Furthermore, we say that a staffing plan $D = (d_1, d_2, \ldots, d_m)$ is a *dominant minimum* if there is no alternative feasible plan $D' = (d'_1, d'_2, \ldots, d'_m)$ with $d'_i < d_i$ for any i. Now, for any staffing plan D, the time at which the last passenger on some flight k is processed increases (or, more accurately, doesn't decrease) as the number of staff d_i in any period i is decreased. Therefore, decreases in staffing levels can only worsen the feasibility of a solution. Consequently, a sufficient condition for a departures staffing plan D to be a dominant minimum is that every plan $D'(i) = (d_1, d_2, \ldots, d_{i-1}, d_i - 1, d_{i+1}, \ldots, d_m)$, $i = 1, 2, \ldots, m$ is infeasible. (We will see later that this is not the case for arrivals.)

The algorithm to determine a locally minimum staffing plan D starts with any initial feasible solution and then descends to a dominant minimum by repeatedly removing staff until no further improvements can be made to the solution. This procedure is shown in Algorithm 1. Notice that if a decrease in interval i has been rejected because it generated an infeasible solution, then no further staffing decreases in that period need be considered.

This algorithm is greedy sequential, with the function $f_D(D, C)$ determining from a set $C \subseteq \{1, 2, \ldots, m\}$ of candidate periods that period in which a staffing reduction is to be next attempted. For the lexicographical objective mentioned before, $f_D(D, C)$ simply returns some period with the greatest staffing level:

$$f_D(D, C) = c : d_c = \max_{i \in C} d_i.$$

Where there is a tie, it is advantageous to choose the smallest index of the possible candidate periods, thereby biasing low staffing levels towards the beginning and high staffing towards the end of a flight's processing, and consequently giving a more robust solution should delays occur. Note that because ties need to be broken in an arbitrary way, no guarantee can be given that a lexicographically optimal solution will be found.

The starting feasible solution is determined by calculating the smallest staffing level d_0 that is feasible when applied in all periods across the day. Thus, the starting solution is given by the uniform staffing $D_0 = (d_0, d_0, \ldots, d_0)$, where d_0 solves

$$\begin{array}{ll} \text{minimise} & d_0 \\ \text{subject to} & (d_0, d_0, \ldots, d_0) \text{ is feasible.} \end{array}$$

The reader will note that if the above starting solution is used, then $f_D(D, C)$ can be simplified to return, on successive calls, the next index of those remaining in C. However, the more general definition will be required for the arrivals case considered next.

1.3.2. *Arrival Work Requirements Heuristic.* The minimisation of the arrivals staffing requirements is complicated by the booth-closing requirement that prevents large numbers of passengers from being moved between queues. As before, let $r_i(A) = (q_i/a_i) \cdot (a_i - a_{i+1})$ denote the number of passengers forced to re-queue at the end of period i under staffing plan A, and let r_{limit} be the maximum such re-queuing permitted. (We put $r_i(A) = 0$ whenever $a_{i+1} \geqslant a_i$.) As in the departures procedure, let $A'(c) = (a_1, a_2, \ldots, a_{c-1}, a_c - 1, a_{c+1}, \ldots, a_m)$ be a new staffing plan formed from a feasible plan A by removing a staff member in period c. Unlike the departures case, $A'(c)$ may be infeasible because $r_i(A'(c)) > r_{\text{limit}}$ for some i, i.e., because a booth is being shut during a busy period i. Typically, closing a booth at the start of period c may lead to a re-queuing infeasibility at the end of the previous period $i = c - 1$, i.e., the change may give $r_{c-1}(A'(c)) > r_{\text{limit}}$. Alternatively, by delaying the processing of passengers, a staffing decrease during period c may lead to greater queuing at some later time, and thus risks generating $r_i(A'(c)) > r_{\text{limit}}$ for some period $i \geqslant c$. For example, reducing the staffing in period $c = 2$ of solution $A = (4, 4, 4, 2, \ldots)$ to give $A'(2) = (4, 3, 4, 2, \ldots)$ may introduce a re-queuing infeasibility at the end of period $c - 1 = 1$ where a new booth closure has been introduced. Alternatively, the reduced staffing in period 2 may increase the queue length at

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

the end of period 3 beyond the maximum permitted for the two-booth closure that occurs at the end of this period. In this case, a re-queuing infeasibility may occur at the end of period $i = 3 \geqslant c$.

In the Departures Work Requirements Heuristic, a staffing decrease in period c is rejected immediately if it creates an infeasible solution. However, the arrivals situation is more complex. When a re-queuing infeasibility occurs at the end of period i, it may be possible to repair the requeuing infeasibility by reducing the number of booth closures $a_i - a_{i+1}$ occurring at that point. While this could be achieved by increasing a_{i+1} (i.e., worsening the staffing solution by increasing the staffing in period $i + 1$), it could also be achieved by reducing a_i, i.e., by reducing the staffing in the period i where the re-queuing infeasibility occurs. Considering again the previous example, the staffing solution $A = (4, 3, 4, 2, \ldots)$ may be infeasible while an improved solution such as $A = (4, 3, 3, 2, \ldots)$, $A = (3, 3, 4, 2, \ldots)$ or $A = (3, 3, 3, 2, \ldots)$ may be feasible because it has fewer booth closures. Clearly, the requirement for being a dominant minimum is more complex, and thus the algorithm becomes more complicated.

Let A be some feasible staffing plan, and let $A'(c) = (a_1, a_2, \ldots, a_{c-1}, a_c - 1, a_{c+1}, \ldots, a_m)$ be some new plan that meets the flight facilitation targets but is re-queuing infeasible. For notational convenience, we will denote the individual staffing levels in $A'(c)$ by $A'(c) = (a'_1, a'_2, \ldots, a'_m)$. Let e be an index for which $r_e(A'(c)) > r_{\text{limit}}$; we say e is *blocking* the reduction of staffing in period c. Now, consider the set of all the solutions that can be formed from $A'(c)$ by decreasing one or more staffing levels. Because these staff reductions can only increase, not decrease, queue lengths, the re-queuing infeasibility can be removed only by decreasing the change in staffing level that occurs at the end of period e, i.e., by reducing a'_e and hence reducing $a'_e - a'_{e+1}$. Thus, $A'(c)$ may become feasible if a'_e is reduced, i.e., plan $(A'(c))'(e) = (a'_1, a'_2, \ldots, a'_{e-1}, a'_e - 1,' a_{e+1}, \ldots, a'_m)$ may be feasible. It is also possible that $(A'(c))'(e)$ is flight infeasible because it fails to meet some flight facilitation target; in this case the original reduction at c can be deemed infeasible and can be skipped in any future reduction attempts. Lastly, $(A'(c))'(e)$ itself may be blocked, so the procedure needs to be repeated recursively.

Note that it is important to drive through the infeasibility in this way, and not simply delay the reduction in a_c until after a_e has been reduced. This latter approach is unable to resolve the case of deadlocks occurring within the blocking periods, i.e., where, for example: period 5 is blocked by the previous period, period 4; period 4 is blocked by its predecessor, period 3; but period 3 is blocked by the later period, period 5. This last case arises when the reduction in a_3 is sufficient to turn period 5 into a "busy period," i.e., reducing a_3 increases the queue length at the end of period 5 above the re-queuing limit. Such cases have been observed during actual simulation runs.

The full arrivals procedure is shown in Algorithm 2. Note that the uniform starting solution A_0 is the arrivals equivalent of D_0, and that $f_A(A, C)$ operates identically to (the general version of) $f_D(D, C)$.

1.4. Simulation Implementation

The above algorithms require that repeated calls be made to the simulation model. There are a number of important implementation details that allow the time taken in each call to be reduced.

First, we note that the work requirements problem naturally partitions into independent subproblems separated by intervals during which no passengers are processed. These idle periods introduce independence in the sense that the passenger processing (and hence staffing) occurring after some idle interval is unaffected by the processing (and staffing) before that idle interval. Idle intervals are observed to occur frequently throughout the day at the Auckland airport. Second, we note that most calls to the simulation model are made after changing the staffing level in just one period, period c, and thus passenger processing needs to be recalculated only for periods c and later. However, processing outside idle intervals will be unaltered by the change. In particular, if $t_c[A]$ is the start time of the first idle interval occurring after period c under staffing plan A, and $t_c[A'(c)]$ is the corresponding time for the reduced plan $A'(c)$, then processing occurring before period c and after $\max(t_c[A], t_c[A'(c)])$ will be the same under both A and $A'(c)$. Furthermore, because staff decreases (and not increases) are being tested, $t_c[A] \leqslant t_c[A'(c)]$. Thus, if the staffing level in period c is reduced, and the time $t_c[A'(c)]$ occurs in period j, then the passenger processing in periods $1, 2, \ldots, c - 1$ and $j + 1, j + 2, \ldots, m$ is unchanged by the staffing decrease, and need not be re-simulated. If sufficient information is retained from the previous simulation run, the simulation model need only update that processing occurring between periods c and j, and use this updated information to test for feasibility. (Note that re-queuing infeasibility also needs to be checked for period $c - 1$.) This reuse of previous simulation solutions reduces solution times by about a factor of 10.

The above two heuristics simply consider the staffing of either arrivals or departures in isolation, i.e., the two staffing requirements are assumed to be met independently. While New Zealand Customs have occasionally operated rosters like this, it is more common for staff to move between arrivals and departures as the work requirements shift between these two tasks. Thus, it is important that this feature of the problem be incorporated within the solution procedures.

The approach taken to the joint staffing problem is to independently minimise staffing on one primary line, and then minimise the joint staffing requirement by adjusting staffing on the other primary line. For this discussion, let us assume that the departures staffing has been minimised independently using the procedure in Algorithm 1. We now wish to determine an arrivals staff requirement that minimises (in the lexicographic sense discussed earlier) the

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

ALGORITHM 2. Arrivals Work Requirement Algorithm

Let A be a starting feasible solution.

Form a candidate list $C = \{1, 2, 3, \ldots, m\}$. (As the algorithm proceeds, C will be updated by deleting the indices of those periods in which further staff reductions have been shown to give infeasible solutions. Thus, C will contain indices of periods in which further staffing decreases are likely to give feasible solutions.)

repeat

  Let $c = f_A(A, C)$ be the index of the next period to reduce
  If the simulation model shows $A'(c) = (a_1, a_2, \ldots, a_{c-1}, a_c - 1, a_{c+1}, \ldots, a_m)$ to be feasible then
    Let $A = A'(c)$
  else if $A'(c)$ is flight infeasible then
    Remove c from C, i.e. $C = C \backslash c$
  else $A'(c)$ must be flight feasible, but re-queuing infeasible, so
    Let $\tilde{A} = A'(c)$
    repeat
      Let e be the (smallest) blocking index for $\tilde{A}$
      Let $\tilde{A} = \tilde{A}'(e)$
    until the simulation model shows $\tilde{A}$ to be flight infeasible or $\tilde{A}$ is feasible
    if $\tilde{A}$ is feasible, then
      Let $A = \tilde{A}$
    else $\tilde{A}$ must be flight infeasible
      Remove c from C, i.e. $C = C \backslash c$

until $C = \{\}$

Staffing plan A is a dominant minimum.

Note: The staffing plan formed by $\tilde{A} = \tilde{A}'(e)$ will be flight infeasible if $e \notin C$, and so no run of the simulation model needs to be made in this case.

total arrivals-plus-departures staffing requirements. We can achieve this by simply modifying the order in which the staffing reductions at arrivals are attempted. If D represents the independently minimised departures solution, and A is some feasible arrivals solution that is to be further minimised, then the next period c in which a decrease in arrivals staffing level a_c should be attempted is determined by considering the joint staffing levels $A + D$. Thus, when minimising arrivals for a given departures requirements, the function $f_A(A, C)$ in Algorithm 2 is given by

$$f_A(A, C) = (c : a_c + d_c = \max_{i \in C} a_i + d_i),$$

where $D = (d_1, d_2, \ldots, d_m)$ is the (now effectively constant) departures staffing plan.

Recall that when minimising arrivals independently, the algorithm commenced from a minimal uniform staffing solution $A = A_0 = (a_0, a_0, \ldots, a_0)$. However, when minimising arrivals given some departure solution, it may be advantageous to increase the number of arrivals staff on duty above the level a_0, thus shortening the time required to complete the processing of some arrivals flight and thereby increasing the availability of staff for some immediately following departures processing requirement. For example, if the departures heuristic starts from $D_0 = (8, 8, 8, \ldots)$ to give the solution $D = (1, 1, 8, \ldots)$, then an arrivals solution $A = (8, 8, 1, \ldots)$ giving a total staffing $A + D = (9, 9, 9, \ldots)$ may be desirable even if $A_0 = (6, 6, 6, \ldots)$ is the minimum uniform feasible arrivals solution. Thus, in minimising the number of periods in which the maximum total number of staff, say y, are required, it may be advantageous to increase the number of periods in which $y - 1$ total staff are required. An upper bound on y is given by $a_0 + d_0$, and so for a given departures solution (with a maximum staffing of d_0), a total staffing of $y - 1 = a_0 + d_0 - 1$ should be permitted in any period by allowing up to $a_0 + d_0 - 1$ staff on arrivals. This can be achieved by using an initial arrivals staffing level $A = A_0 + D_0 - (1, 1, \ldots, 1)$ in the joint arrivals.

This discussion has so far assumed that the arrivals staffing is minimised after determining the minimum departures requirement. In fact, our implementation of the joint staffing algorithm actually determines two solutions, firstly that given by minimising departures, and then jointly minimising arrivals, and secondly that given by minimising arrivals, and then jointly minimising departures. The better of these two solutions is then kept.

It is important to note that while the independent arrivals and departures heuristics produce dominant optima, no such claim can be made for the joint staffing heuristic.

1.5. Typical Work Requirements

A typical staffing requirement generated by the heuristic work requirements process is shown in Figures 2 and 3,

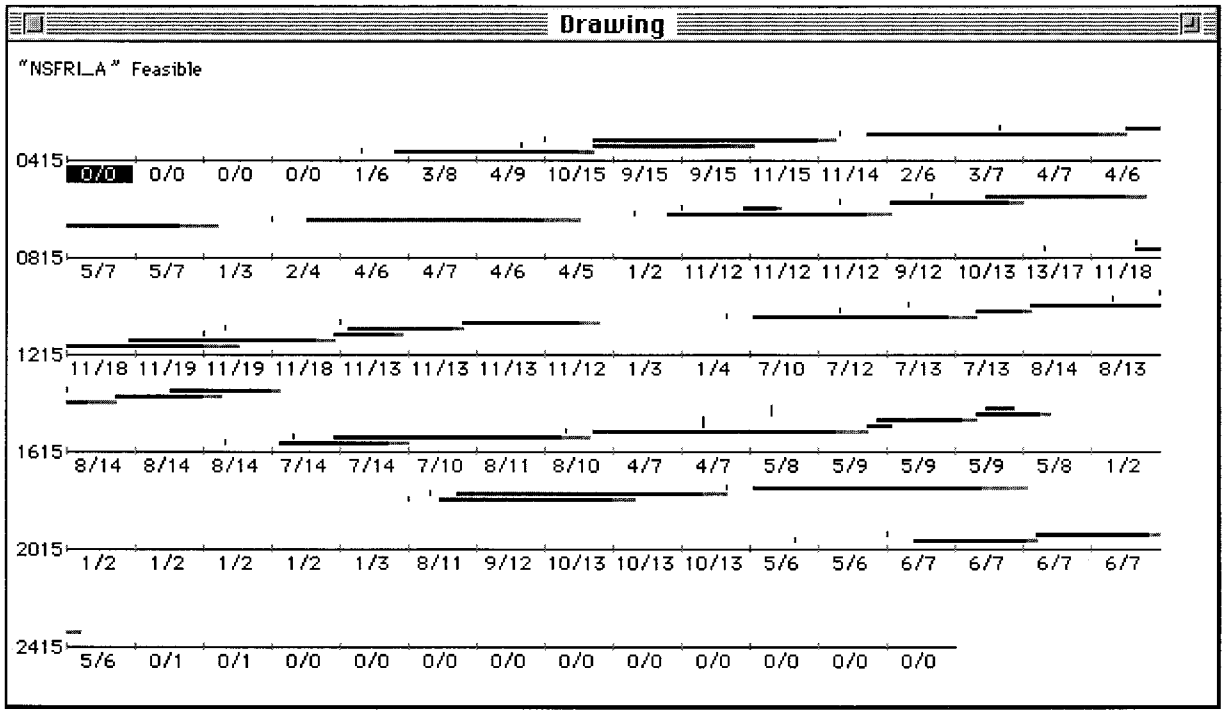


Figure 2. Flight processing for arrivals on a Friday during the northern summer.

which illustrate the flight processing of both arrivals and departures, respectively, for a Friday during the Northern summer. The figures under each 15-minute segment show the minimum number of Customs officers working the relevant primary line (either arrivals or departures) and the total number of officers required for both arrivals and departures. Each horizontal line segment represents the duration of a flight's processing from the arrival of the passengers at the primary line through to the completion of the processing of the last passenger aboard that flight. Note that this particular solution is one of many that achieve the same objective function value for the heuristic.

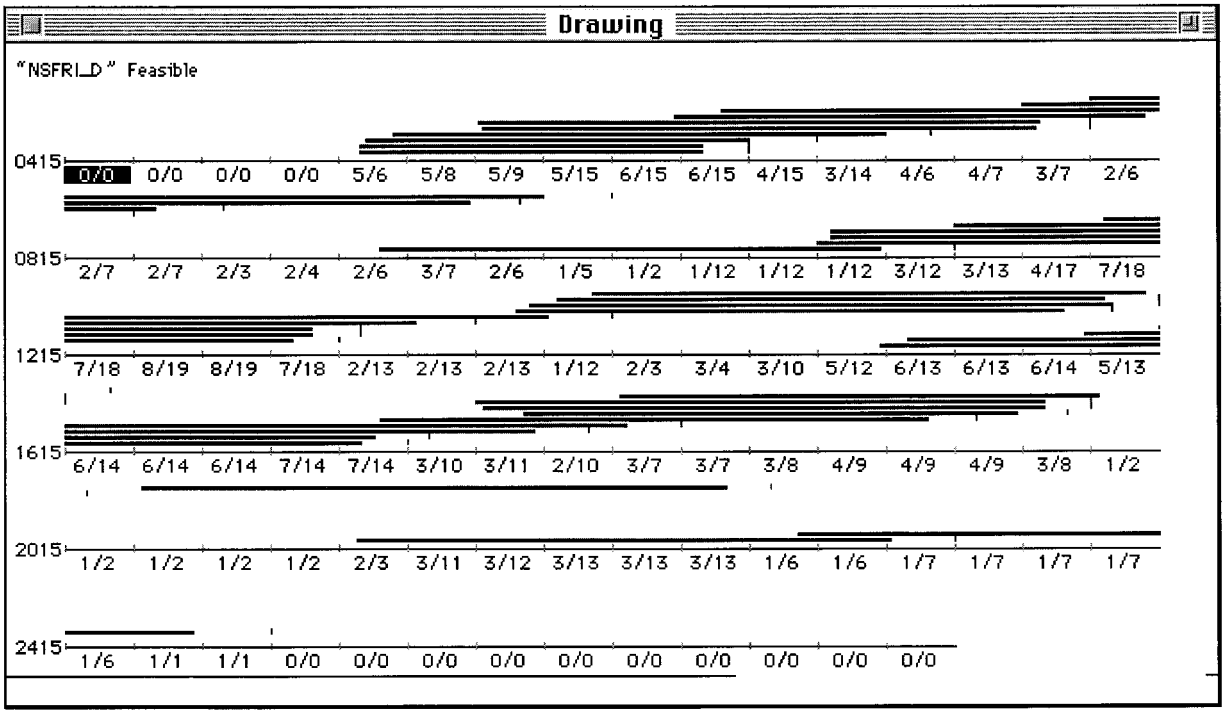


Figure 3. Flight processing for departures on a Friday during the northern summer.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

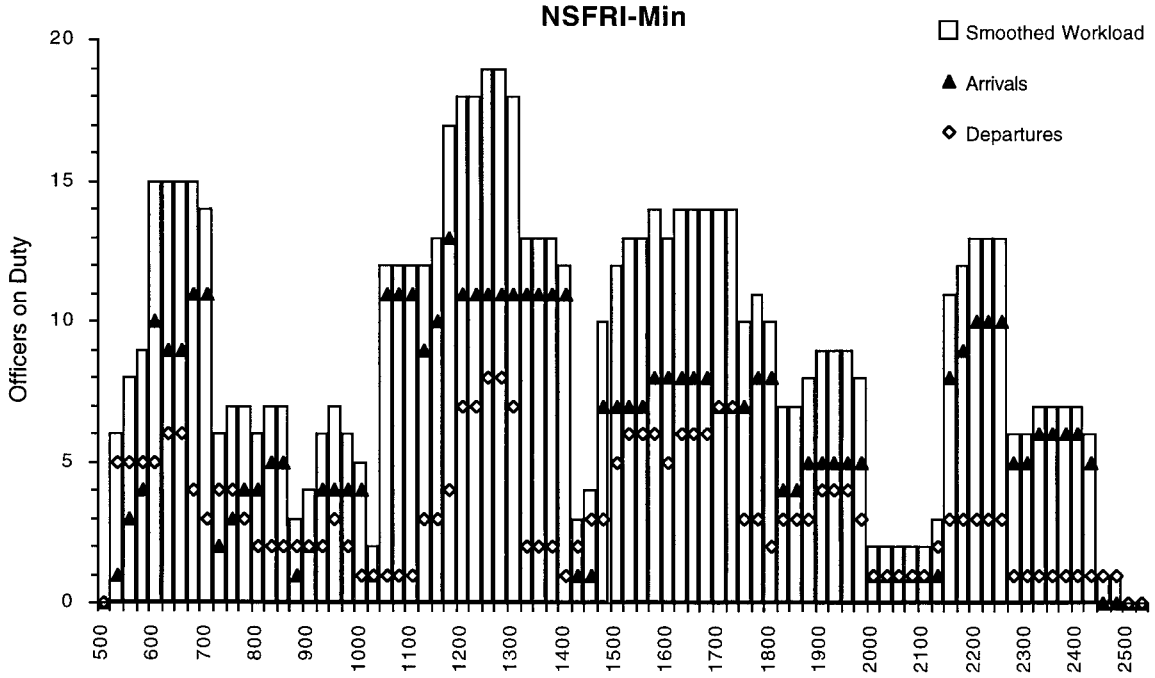


Figure 4. Work requirements on a Friday during the northern summer.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

While it is useful to consider the detailed flight processing that occurs under a plan, the principal use of such a solution is to generate the daily work requirements that can then be used in the generation of roster shifts. Such a work requirement is best viewed using a plot such as that shown in Figure 4. The next section considers how shift start times and lengths can be generated that ensure sufficient staff are available to cover this work requirement.

2. WORKFORCE ALLOCATION

The Customs staffing requirements are met using a mixture of full- and part-time staff. The full-time staff operate an essentially fixed roster during each successive six-month period corresponding to the summer and winter season flight schedules. Therefore, a roster needs to be built that meets the work requirements associated with a typical week in the season. Such a typical week is identified by Customs staff, and then passenger numbers for this week are forecast in consultation with the airlines. (Where airline forecasts differ from those made by Customs staff, the higher of these two figures is used, thus introducing some robustness into the system.) The work requirements algorithm described in Section 1.3 is then used to generate quarter-by-quarter hour work requirements for each day of the week.

An integer programming optimisation model is used to determine the best allocation of personnel within each day that provides sufficient coverage during each 15-minute period. Since there are no overlapping shifts from one day to the next, the model considers the allocation of personnel for each day of the week as an independent problem. We therefore describe the allocation model for a single day.

First, it is required to achieve coverage using as few people as possible, and second, it is desirable to obtain a coverage in which the surpluses of personnel in any period are as small as possible. Both of these goals might be achieved by minimising paid or total duty hours. As a subsidiary objective we could also favour solutions in which shift start times occur on the hour and half hour. This later preference makes it easier to manage and operate the resulting roster. It is also possible to constrain shift start times to a preferred subset of times.

A large number of alternative shift combinations are possible for a given set of requirements. A typical day will involve as many as 80 15-minute periods (20 hours of coverage), and shifts could start at a large number of these times. A manual search for the best allocation (combination of shift start times, and numbers who start at those times) is not practical. The set partitioning optimisation model used in this study enables this search to be performed quickly and efficiently, and guarantees an optimal solution for a given staffing requirement. Such models have been used extensively in rostering problems; see Ryan (1992).

As before, let m be the number of 15-minute periods over which staff coverage is required as determined by the work requirements for some day (e.g., if personnel are required from 0545 to 2330 then $m = 71$). The available shifts are of four types: a full-time shift (type f) typically of length $n(f) = 30$ 15-minute periods (corresponding to 8 hours on duty less two 15-minute changing times), and

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

3-, 4-, and 5-hour part-time shifts of $n(3) = 12$, $n(4) = 16$, and $n(5) = 20$ 15-minute periods in length, respectively. (Union rules prevent planned overtime from being included in the full-time roster, so no full-time shifts longer than the standard length are currently considered in this model.) In general, each shift type $s \in \{f, 3, 4, 5\}$ can start in each of $m - n(s) + 1$ 15-minute periods. We can represent the coverage of all possible shift types at all possible shift times using the matrix $A = (a_{ij})$, $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$ where

$$a_{ij} = \begin{cases} 1 & \text{period } i \text{ is covered by shift } j, \\ 0 & \text{otherwise.} \end{cases}$$

Note that an analysis of the idle periods occurring within shifts showed that these provided ample opportunities for meal breaks. Consequently, meal breaks have not been explicitly incorporated within the optimisation model. An extension to incorporate such breaks can be made if required but will increase the solution times.

The cost of a shift j of type $s_j \in \{f, 3, 4, 5\}$ is given by the length of that shift (in 15-minute periods):

$$c_j = n(s_j).$$

Let x_j be the number of staff working shift j. The following optimisation model will determine an allocation of full-time and part-time personnel satisfying the staffing requirements for each period with the minimum number of paid hours.

$$\text{minimise} \quad \sum_{j=1}^{n} c_j x_j$$

$$\text{subject to} \quad \sum_{j=1}^{m} x_j a_{ij} \geq b_i \quad \forall\, i = 1, 2, \ldots, m,$$

$$x_j \geq 0,\ x_j \text{ integer} \quad \forall\, j = 1, 2, \ldots, n,$$

where $b_i = a_i + d_i$ is the personnel requirement for period i as determined earlier.

A number of variations of this integer program are also used as appropriate. First, it is possible to ban all full-time shifts that do not start on the hour or half-hour; or alternatively, such starts can be costed against by adding a penalty to the appropriate cost terms, giving

$$c_j = \begin{cases} n(s_j) + c_{\text{poorstart}}, & j \in F_{\text{poorstart}}, \\ n(s_j), & \text{otherwise,} \end{cases}$$

where $F_{\text{poorstart}}$ is the set of all full-time shifts starting 15 or 45 minutes past the hour, and $c_{\text{poorstart}}$ is a small penalty term, e.g., $c_{\text{poorstart}} = 0.01$.

To meet union demands, Customs management can specify that a certain proportion p_{fulltime} of the work requirement in each quarter hour be met by full time staff. This requires the addition of the constraint

$$\sum_{j \in F_{\text{fulltime}}} x_j a_{ij} \geq \lceil p_{\text{fulltime}} \cdot b_i \rceil \quad \forall\, i = 1, 2, \ldots, m,$$

where F_{fulltime} is the set of all full-time shifts, and $\lceil p_{\text{fulltime}} \cdot b_i \rceil$ gives the rounded-up minimum number of full-time staff required in the quarter hour. Note that if only full-time staff are required, then $p_{\text{fulltime}} = 1$, while $p_{\text{fulltime}} = 0$ is used if there is no lower requirement on the number of full-time staff required.

As well as specifying minimum full-time staffing requirements for each 15-minute period, the total number of full-time staff can also be constrained using the following constraint:

$$\sum_{j \in F_{\text{fulltime}}} x_j = n_{\text{fulltime}},$$

where n_{fulltime} is the total number of full-time staff required.

This fully developed model can be solved for each day using the Zip integer programming package developed by Ryan (1980). An integer solution is obtained by using variable branching. Solution times of 30 to 40 seconds are required to solve all 7 days on a Sun workstation. Note that these problems do not have 0–1 variables, but instead x_j are general integer variables, thus the Zip package utilises standard variable branching and not the constraint branching techniques often found in rostering problems (Ryan 1992). However, the problems are generally small, so solution times are not excessive.

The solution to this model gives the number of 3-hour, 4-hour, 5-hour, and full-time staff starting duty in each 15-minute period during the day. Figure 5 shows a solution generated to cover the Friday work requirements in Figure 4 using $p_{\text{fulltime}} = 0.3$ and $n_{\text{fulltime}} = 28$. The actual shift start times are shown in Table I. Note that this run used $c_{\text{poorstart}} = 0$, thus there is no penalty on shift starts on quarter hour periods. The impact of the full-time staffing constraint required by the union is readily visible in the resulting overcoverage. Management are now very aware of the cost associated with the longer full-time shifts and are examining ways in which this can be reduced.

3. CYCLIC ROSTER CONSTRUCTION

The roster pattern in operation at the commencement of this work was a cyclic 6/3 pattern in which each officer worked six continuous days followed by three days of recreation. Nine individual troops consisting of a superior and 13 officers each worked this 6/3 pattern, covering a total of 6 fixed shift starting times each day of the week. Although such a roster was easily implementable and highly predictable, it was also very inefficient in terms of matching personnel appropriately to peaks and troughs in demand. Having addressed the core of this problem in terms of the appropriate number and placement of personnel on a day-to-day basis via the simulation and workforce allocation models, the task now was to devise work rosters for the allocated officers. Whereas the workforce allocation model had generated optimal combinations of full-time *and* part-time officers, work rosters were required only for the full-time personnel.

From the outset it was agreed that an alternative *cyclic* roster would be devised, at the same time maintaining the

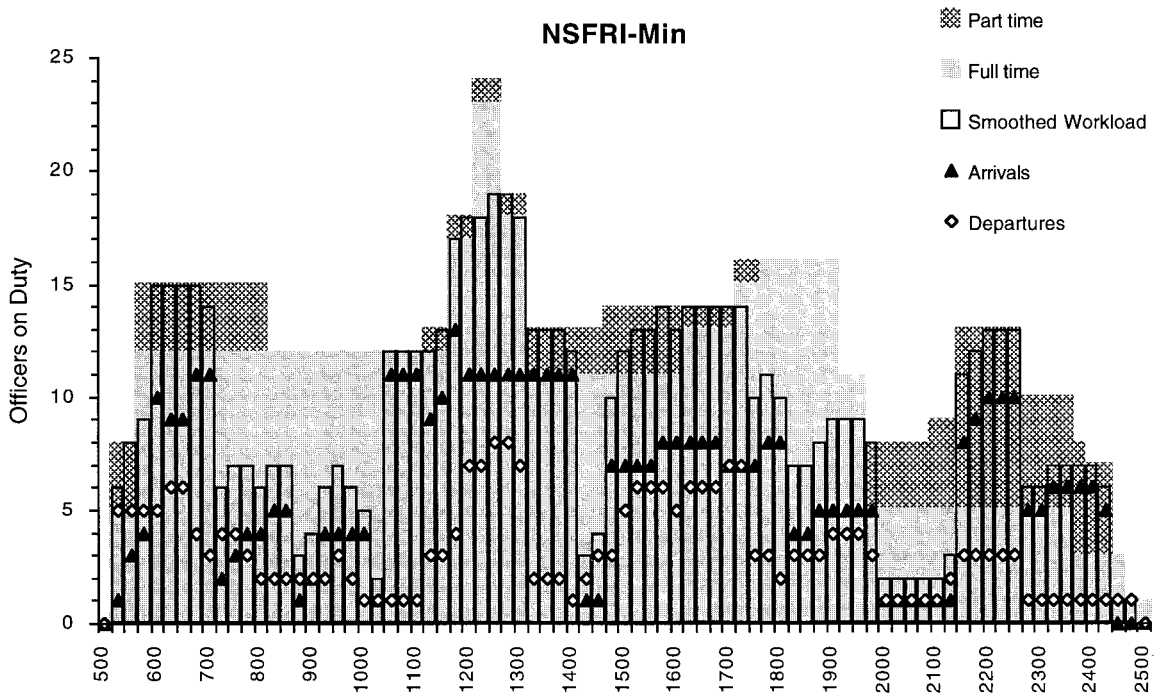


Figure 5. Work requirements and staff coverage on a Friday during the northern summer. (Note that 15 minute changing times for the full-time staff at the beginning and end of each shift are not included in the plot.)

working week average of 40 hours over the roster cycle. In addition, the inclusion of 15 minutes paid "dressing time" at the start and finish of their shifts needed to be taken into account. In the 6/3 pattern this resulted in a total shift length of 8 hours and 35 minutes in order to maintain a 40-hour week average over the 9-week cycle. It was decided that, rather than restrict the cyclic roster structure by maintaining this shift length, a shift length of eight hours would be used (including dressing time), with an average of five working days per week over the roster cycle. Coverage guaranteed from the workforce allocation model was then based on an actual time on duty of 7.5 hours.

Table I
Shift Start Times and Number of 3-, 4-, and 5-Hour Part-Time and Full-Time Staff Required On Each Shift

Shift Start Time	3 Hour Part-Time Starts	4 Hour Part-Time Starts	5 Hour Part-Time Starts	Full-Time Starts
515	3	0	0	5
545	0	0	0	7
1130	0	0	1	0
1145	0	0	0	5
1215	0	0	0	6
1300	1	0	0	0
1530	1	0	0	0
1615	0	0	0	2
1715	0	0	0	2
1745	0	0	0	1
1945	3	0	0	0
2000	1	0	0	0
2115	1	0	0	0
2130	1	0	0	0
2145	1	0	0	0
2200	1	0	0	0
Totals	13	0	1	28

Note: Because of a 15-minute changing period, full-time shifts actually start 15 minutes before the start-time shown here.

A number of factors influence roster construction. It is assumed that the planning horizon is one week, since demand, based on flight schedules, is approximately replicated on a weekly basis for each six month period in either the northern summer or northern winter. The workforce allocation model produces a set of full-time shifts that need to be sequenced with days off to form a roster. (Note that rosters for the part-time staff are easily constructed by hand.) The roster must guarantee that the prescribed work week structure of a 40-hour week average is satisfied over the roster cycle. A further binding constraint on the structure of the roster is that it contains as many lines of work (weeks) as is consistent with workforce allocation requirements. Table II shows the shift requirements for the northern winter in which shift starting times have been grouped into three broad categories, namely early (E: 0500 to

Table II
Shift Requirements for a Typical Northern Winter

	M	T	W	Th	F	Sa	Su
E	10	16	14	14	12	16	13
D	9	7	11	9	11	12	11
N	9	4	3	5	5	—	4
Total	28	27	28	28	28	28	28

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

0700), day (D: 1130 to 1415), and night (N: 1600 to 1800). The total weekly shift requirement is 195 full-time shifts, which for a 5-day week average equates to 39 full-time officers. It should be noted that because of externally driven changes in the division of work, the new staffing levels cannot be compared directly with the original 6/3 roster.

In addition to the binding constraints on the roster structure discussed above, a number of factors of a non-binding nature, which relate to the quality of the roster must be considered. Issues such as the minimum and maximum length of work-stretches, length of recreation blocks, frequency and spacing of off-weekends, and shift rotation within work-stretches, all impinge on roster quality. It was agreed from the outset that work-stretches should be between three and seven days in length and that recreation blocks should ideally be consecutive. Although off-weekends were not a major issue, it was agreed that as many evenly spaced off-weekends as possible should be obtained. On the other hand, the rotation of shifts within work-stretches was an issue that raised some concern and will be discussed at more length shortly.

Complete cyclic rosters for both the northern winter and northern summer schedules were generated using software based on techniques discussed in Panton (1991). Using this method, single-shift schedules that determine which days are worked and which days are not, are created using a heuristic which generates good feasible schedules over the roster cycle by the end-to-end matching of good feasible schedules generated for work stretches of up to four weeks in length. Once a good quality single-shift schedule satisfying all the hard constraints is determined, shifts are assigned to the days worked in order to maximise the occurrence of continuous work-stretches in which the starting times of consecutive shifts remain the same. Conditions under which this is guaranteed are discussed in van den Berg and Panton (1994). In both the northern winter and northern summer schedules built in this way, recreation days were in blocks of length two or three with the majority of work-stretches being of length five or six. Table III shows a portion of the cyclic roster built for the northern winter.

Where possible, adjacent shifts are of the same type (E, D, or N). Rotations within early, day, or night groupings are of minor consequence since time spans within each group are at most two or three hours. Where possible however, rotations are assigned in a forward direction, e.g., *D*1, *D*3. Backward rotations between different shift groupings are less desirable and indeed may be illegal if they do not allow sufficient rest time before the next shift. The *N*3, *D*3 backward rotation in line 13 is not illegal but nevertheless is undesirable. Note, however, that this could be removed by exchanging the *D*3 in line 9 for the *N*3 in line 13, if it were deemed that the *D*3, *N*3 forward rotation from lines 8 to 9 was acceptable. Alternative methods of removing undesired rotations are often available. For example, compromises can often be made with shift starting times

Table III
First 14 Days of the Northern Winter Roster for Full-Time Staff

Line No.	M	T	W	Th	F	Sa	Su	Key to Shift Starting Times	
1			D3	D3	D3	D4	N2	E1	5.30
2	N3			D1	D3	D4	N2	E2	6.00
3	N3	N4			D2	D4	D3	E3	7.00
4	D3	D3	D3			E1	E1		
5	E1	E1	E1	E1				D1	11.30
6	N3	N4	N1	N3	N1			D2	12.00
7		E1	E1	E1	E1	E2	D3	D3	13.00
8			D3	D3	D3	D4	D3	D4	13.30
9	D3			E1	E1	E1	E1	D5	14.15
10	E1	E1			E1	E1	E1		
11	E1	E1	E1			E1	E1	N1	16.00
12	E1	E1	E1	E1				N2	16.30
13	N3	D3	D3	N3	N1			N3	17.00
14		E1	E1	E1	E1	E1	E1	N4	18.00

Notes: Each shift is of 8 hours duration with the starting time given in the key; the lack of an entry indicates a day off. During the first week of the roster, staff member 1 works row 1, staff member 2 works row 2 and so on. At the end of the week, everybody drops down a line, with staff member 1 now working row 2, staff member 2 working row 3, and so on down to staff member 14 who now works row 1. This pattern repeats until after 14 weeks, staff member 1 is once again working row 1.

when surpluses occur at certain times. In most cases however there is a cost associated with such changes, usually resulting in an increase in surpluses.

This cyclic roster was first introduced for the northern winter schedule in 1993. Part-time staff were employed, and the full-time staff were removed from their teams and allocated to the cyclic roster. Unfortunately, staff reaction to the new roster was very poor for a number of reasons. First, the introduction of the new roster coincided with record passenger levels for a Christmas period. Analysis showed that the data provided on predicted passenger numbers had significantly underestimated the actual passenger levels, and consequently the calculated staffing levels were not sufficient to meet demand. Although additional staff were recruited to the Primary lines from other work areas, queue lengths increased and overall service levels dropped. Second, staff reacted poorly to the drastic change to their roster. These staff had spent the last decade working in teams following a very simple roster structure. Now they were being asked to leave their teams and simultaneously learn a new, more complex roster pattern. Their response to this included significantly increased levels of sick leave, further compounding the staffing shortage. It soon became apparent that further changes had to be made.

4. A SECOND ROSTER

Following the experience gained during implementation of the first roster, it was decided to develop a second roster based more closely on the "6 on, 3 off" roster pattern that had been used historically. Management also saw this

Table IV
Typical Roster Lines for the 6/3 Roster

Line No.	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	500	500	500	500	500	500	500
2	500	500	500	500	500	500	500
3	500	500	500	500	500	500	500
4	500	500	500	500	500	500	500
5	500	500	500	500	500	500	500
6	500	500	500	500	500	500	500
7	500	500	500	500	500	500	500
8	500	500	500	500	500	500	500
9	500	500	500	500	500	500	500
10	930	930	930	930	930	930	930
11	930	930	930	930	930	930	930
12	930	930	930	930	930	930	930
13	930	930	930	930	930	930	930
14	930	930	930	930	930	930	930
15	930	930	930	930	930	930	930
16	1230	1230	1230	1230	1230	1230	1230
17	1230	1230	1230	1230	1230	1230	1230
18	1230	1230	1230	1230	1230	1230	1230
19	1230	1230	1230	1230	1230	1230	1230
20	1230	1230	1230	1230	1230	1230	1230
21	1230	1230	1230	1230	1230	1230	1230
22	1230	1230	1230	1230	1230	1230	1230
23	1230	1230	1230	1230	1230	1230	1230
24	1700	1700	1700	1700	1700	1800	1700
25	1700	1700	1700	1700	1700	1800	1700
26	1700	1700	1700	1800	1700	1800	1700
27	1700	1700	1700	1800	1700	1800	1700
28	1700	1700	1700	1800	1700	1800	1700

Note: Shift start times are built using the optimisation system for the 6/3 roster. Staff work a succession of 6 shifts from within 1 row; each such 6-day work-stretch can start in any column, and then finishes 6 shifts later after wrapping back to the start of the row. Note the regularity in shift start times across each row.

change as an opportunity to reintroduce a team structure. The new roster required longer (and consequently less efficient) shifts and an increased number of staff. However, management expected these up-front costs to be offset by reduced sick leave and improved morale.

A 6/3 roster requires that 2 out of 3 staff be working on any day. After discussions with management, this requirement was met by forming the staff into 3 teams of 16 with members of the same team following the same 6/3 pattern, but each team being staggered 3 days behind the previous team. After allowing for leave requirements, this pattern gives 28 full-time staff members on duty each day.

Table IV shows the shift start times for each roster line in the 6/3 roster for each day of the week. (These times were generated using modifications to the optimisation outlined below.) These lines are combined to form a roster using the 6/3 pattern shown in Table V. In this pattern, each staff member is permanently allocated to their own row in the table, where the values in each row give the roster line being worked by a staff member. The actual shift start time is found by looking up the appropriate roster line and day of week in Table IV. Thus, despite the cycle length of 6 + 3 = 9 days, the full-time staffing on each day will be identical from week to week. Note that although the day's total full-time staffing level is fixed at 28, adjustments in start times across the week allow staff levels to more accurately reflect the actual work requirements during each day. The use of part-time staff then ensures that each day's requirements are fully covered.

Perhaps the most significant change introduced by the 6/3 roster is the need to closely coordinate start times across successive work days to ensure a highly regular roster is produced. In the original cyclic roster with its individual lines of work, the shifts and nonwork periods could be carefully arranged to minimise variation in the start times of successive work days. Consequently, the number of times each shift start time appeared in the work requirements did not need to be constrained in any way. However, because the 6/3 roster operates with a period of 9—not 7—days, it is not possible to say a priori where nonwork days will appear within the weekly pattern. Indeed, over a nine-week period, every successive pair of weekdays (i.e., Mon–Tue, Tue–Wed, . . . , Sun–Mon) will be worked at least once without a break by the staff. That is, if there are, for example, five 6:00 AM starts on Monday, then to avoid excessive variation when progressing to the next day's shifts, there should be a similar number of such early starts on Tuesday, and then Wednesday, and so on. Therefore, if excessive variation in the staff members' shift start times is to be avoided, the broad distribution of shift start times across the day must not vary during the week.

To ensure that the full-time rosters maintain the desired continuity in shift start times across the week, the set covering integer programming model discussed earlier was modified to include constraints forcing a fixed number of full-time shift starts within any user specified time period. Each of these constraints has the form

$$\sum_{j \in F_{\text{specifiedstarts}}} x_j = q_{\text{specifiedstarts}},$$

where $F_{\text{specifiedstarts}}$ is the set of all full-time shifts that start within some user-specified time interval, and $q_{\text{specifiedstarts}}$ is the total number of full-time shift starts required in that interval. These constraints allow for manual coordination of shift start times across the week. For example, it can be specified that five shift starts occur between 5:00 AM and 6:00 AM on Monday. This facility allows the roster builder to experimentally determine appropriate shift start bands and their associated number of full-time shifts. Experience has shown that the degeneracy of the set covering problems allows the number of full-time shift start times within one day to be significantly reduced with no increase in cost. However, when the flight patterns on successive days vary significantly, applying restrictions on shift start times to ensure the grouping of start times across the week typically incurs some small cost (being increased part-time requirements). This cost is considered minor given the improved quality of the resulting full-time rosters.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

Table V
The First 15 Day's Pattern for the 6/3 Roster

Staff Member	M	T	W	Th	F	Sa	Su	M	T	W	Th	F	Sa	Su	M
1	1	1	1	1	1	1				15	15	15	15	15	15
2	2	2	2	2	2	2				16	16	16	16	16	16
3	3	3	3	3	3	3				17	17	17	17	17	17
4	4	4	4	4	4	4				18	18	18	18	18	18
5	5	5	5	5	5	5				19	19	19	19	19	19
6	6	6	6	6	6	6				20	20	20	20	20	20
7	7	7	7	7	7	7				21	21	21	21	21	21
8	8	8	8	8	8	8				22	22	22	22	22	22
9	9	9	9	9	9	9				23	23	23	23	23	23
10	10	10	10	10	10	10				24	24	24	24	24	24
11	11	11	11	11	11	11				25	25	25	25	25	25
12	12	12	12	12	12	12				26	26	26	26	26	26
13	13	13	13	13	13	13				27	27	27	27	27	27
14	14	14	14	14	14	14				28	28	28	28	28	28
15				15	15	15	15	15	15				14	14	14
16				16	16	16	16	16	16				1	1	1
17				17	17	17	17	17	17				2	2	2
18				18	18	18	18	18	18				3	3	3
19				19	19	19	19	19	19				4	4	4
20				20	20	20	20	20	20				5	5	5
21				21	21	21	21	21	21				6	6	6
22				22	22	22	22	22	22				7	7	7
23				23	23	23	23	23	23				8	8	8
24				24	24	24	24	24	24				9	9	9
25				25	25	25	25	25	25				10	10	10
26				26	26	26	26	26	26				11	11	11
27				27	27	27	27	27	27				12	12	12
28				28	28	28	28	28	28				13	13	13
29	15	15	15				14	14	14	14	14	14			
30	16	16	16				1	1	1	1	1	1			
31	17	17	17				2	2	2	2	2	2			
32	18	18	18				3	3	3	3	3	3			
33	19	19	19				4	4	4	4	4	4			
34	20	20	20				5	5	5	5	5	5			
35	21	21	21				6	6	6	6	6	6			
36	22	22	22				7	7	7	7	7	7			
37	23	23	23				8	8	8	8	8	8			
38	24	24	24				9	9	9	9	9	9			
39	25	25	25				10	10	10	10	10	10			
40	26	26	26				11	11	11	11	11	11			
41	27	27	27				12	12	12	12	12	12			
42	28	28	28				13	13	13	13	13	13			

Notes: Each row in this table is worked by one staff member (with no rotation between lines). Each value in the table identifies a roster line being worked by the staff member. The actual start time of the shift is found by looking up the appropriate roster line and weekday in the previous Roster Lines table. This pattern continues beyond day 15 by slowly rotating each staff member through the successive roster lines. A quick check down the columns will confirm that on every day each line is worked exactly once. Note that no leave requirements are shown in this roster.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

While the new roster was being developed, a project was undertaken by Chen (1993) to analyse flight arrival and departure times. This showed that about 55 percent of arriving flights were more than 15 minutes early or late. To improve the robustness of the roster in covering this variation, a "peak stretching" was used to inflate the work requirements. In particular, if $b_i = a_i + d_i$ is the original work requirement in period i as determined earlier, then the peak-stretched work requirement b_i' is given by $b_i' = \max(b_{i-2}, b_{i-1}, b_i, b_{i+1})$. The outcome of this is that each peak work requirement is stretched one period earlier and two periods later. Figure 6 shows how the work requirements are modified by the peak stretching. This peak stretching is applied to the work requirements before staff coverage is generated with the integer programming model.

The new 6/3 roster was implemented for the northern summer schedule in 1994. Shift lengths were increased to 8 hours 30 minutes and extra "payback" days for training added to the roster to bring the hours worked up to 40 per week. Although the modelled cost of the 6/3 roster was higher than that of the cyclic roster, the actual implementation cost was much reduced, with an improved team morale and reduced levels of sick leave. The result of peak stretching was similar, with a higher up-front cost but

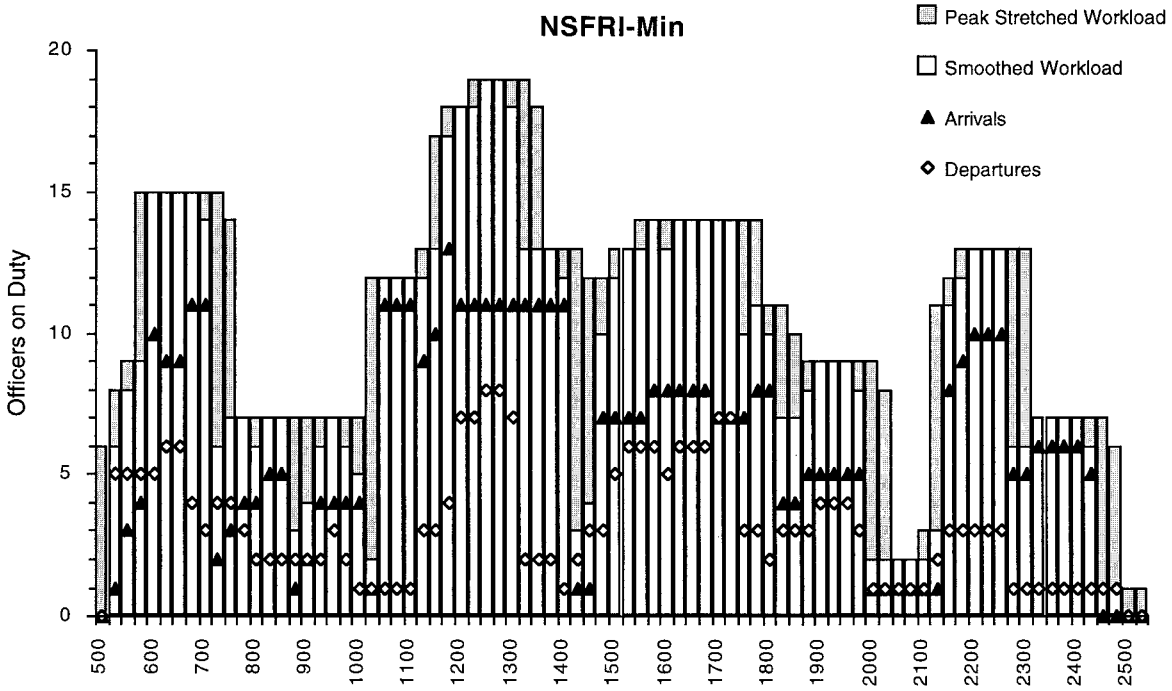


Figure 6. The work requirements both before and after peak stretching.

reductions in actual overtime costs and improved passenger facilitation. (Note that while no attempt has been made to model overtime, the robustness introduced through peak stretching acts as a buffer to limit overtime costs arising from small variations in flight and passenger movements.) These improvements to the roster were considered a success by both staff and management.

5. ORGANISATIONAL IMPACT

One of the most important organisational impacts of this work has been the introduction of part-time staff to the workplace. While full-time staff initially felt threatened by the new part-time staff, there has been a gradual acceptance by all staff of the need to meet peaks in workload through shortened shift lengths. There is no longer any pressure to return to a purely full-time staffing regime.

Since the initial implementation, the airport has experienced annual increases in passenger levels of between 7 percent and 10 percent per annum. The introduction of part-time staff has been important in allowing Customs to keep staffing costs relatively steady during this time. This has been possible partly because capacity limitations at the airport have lead to the scheduling of new flights within off-peak periods, thus providing load when staff previously sat idle. Where passenger growth has required more staffing, this has been provided primarily by part-time staff. In addition, a reduction in full-time numbers through natural attrition has seen further increases in the proportion of work being undertaken by part-time staff, helping to keep costs down. One difficulty associated with the increasing numbers of part-time staff is the increased complexity of manually building their rosters; the authors are currently working to automate this process.

While the initial focus of this work was the development of new rosters, the impact of this collaboration between Customs and the authors has extended beyond the initial brief. For example, the authors have recently been working with Customs management to verify queue lengths and system capacities as part of an on-going airport expansion plan. The rostering project has also provided less direct benefits by illustrating the value of enhanced communication between airlines and Customs and indeed all parties to the airport's facilitation targets. These benefits have been recognised by AIAL who are now developing a centralised flight information database to provide up to date information on flight times and passenger levels. This will eventually allow dynamic simulation of each day as it unfolds, allowing Customs to foresee difficulties several hours in advance and organise any appropriate staffing changes.

Over the last few years, New Zealand has undergone a period of unprecedented reform aimed at privatisation and efficiency improvements. A number of reviews of the Customs operations have been undertaken by the Government's Treasury Department, with the modelling tools developed by the authors being used extensively by Customs management to address issues raised by Treasury. Indeed, management believe that the reports generated from this modelling have been pivotal in demonstrating the efficiency of the Customs processing system, thus reducing the pressure for the removal of the immigration function from Customs and its allocation to the private sector. In related work, Treasury are developing a unit

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.

pricing model that gives a fixed payment for each passenger processed by Customs assuming "an optimally efficient service provider." While Treasury's initial analysis calculated costs by simply multiplying average processing times by passenger levels, Treasury's acceptance of the University of Auckland's expertise has been important in ensuring the cost of uncertainty (both in passenger levels and arrival times) is also included in this new Treasury pricing model.

There have recently been a number of initiatives to improve the processing of passengers travelling between Australia and New Zealand. These initiatives seek to improve data communication between the New Zealand and Australian Customs authorities, allowing data captured at departures in New Zealand to be transmitted to the Australian authorities, and vice versa. This change will slightly increase the processing times for departing passengers in both countries but will allow significant reductions in the arrival processing times. A number of alternative scenarios have been evaluated using the new modelling systems, with expected costings from these being used by the Government's Ministry of Customs to assist in their decision making.

6. CONCLUSIONS AND ONGOING WORK

Before this work began, New Zealand Customs was viewed as a traditional public sector organisation with entrenched inefficiencies and a reluctance to implement change. This work has demonstrated the role of operations research as both a modelling and problem-solving tool, and also as an instrument for driving the wide-reaching institutional change required in Customs. Attitudes at the airport have undergone fundamental change, with staff and management now embracing the challenge of matching staffing to work requirements. Customs are also enthusiastic about the role computer-based decision tools can play in this management process.

The modelling and optimisation tools developed in this project are now used on a regular basis to generate each season's new rosters. They have also formed the core of a number of studies into procedural change at the airport, with the resulting reports being presented at ministerial level. Recent work has focused on the conversion of these programs into an integrated PC-based package that makes these new technologies available to Customs management on a day-by-day basis. The scope of this work is being continually expanded as management demand that increasingly more of the current manual staffing, attendance, and payroll systems be incorporated within one automated environment. Indeed, this work has seen the number of personal computers (PCs) used at the airport increase from zero just three years ago to about six today. The systems are also undergoing modification for implementation at the other international airports in New Zealand.

ACKNOWLEDGMENT

The authors thank all the Customs staff involved in this exercise, from the original review team through to the current management and staff, for their willingness to implement change and their commitment to this project. Glenys Hoffman, Paul Campbell, Kevin Donovan, and particularly Fiona Joe deserve special mention for their ongoing enthusiasm and commitment. Peter Maxwell has played a pivotal role in both coding, development analysis, and also on-the-spot support for the new systems and approaches.

REFERENCES

BAKER, K. 1976. Workforce Allocation in Cyclical Scheduling Problems: A Survey. *Opnl. Res. Quarterly,* **27,** 1, 155–167.

BARTHOLDI, J. AND H. RATLIFF. 1978. Unnetworks, with Applications to Idle Time Scheduling. *Mgmt. Sci.* **24,** 8, 850–858.

BECHTOLD, S. AND L. JACOBS. 1990. Implicit Modeling of Flexible Break Assignments in Optimal Shift Scheduling. *Mgmt. Sci.* **36,** 11, 1339–1351.

CHEN, M. 1993. Analysis of the Arrival Times of Flights at Auckland International Airport. DIP (Stats) Dissertation, University of Auckland.

KERRIDGE, P. 1993. The AIAL Simulation Model of Auckland International Airport. Private Communication.

PANTON, D. 1991. On the Creation of Multiple Shift Continuous Operation Rosters under General Workforce Conditions. *Asia Pacific J. Opns. Res.* **8,** 189–210.

RYAN, D. 1980. Zip—A Zero-One Integer Programming Package for Scheduling. Technical Report, Report C.S.S. 85, AERE Harwall, Oxfordshire, UK.

RYAN, D. 1992. The Solution of Massive Generalized Set Partitioning Problems in Aircrew Rostering. *J. Opnl. Res. Soc.* **43,** 5, 459–467.

SEGAL, M. 1974. The Operator-Scheduling Problem: A Network Flow Approach. *Opns. Res.* **22,** 808–823.

TIEN, J. AND A. KAMIYAMA. 1982. On Manpower Scheduling Algorithms. *SIAM Review,* **24,** 3, 275–287.

VAN DEN BERG, Y. AND D. PANTON. 1994. Personnel Shift Assignment: Existence Conditions and Network Models. *Networks,* **24,** 385–394.

WHITAKER, D. 1995. An Improved Airport Simulation for AIAL. Private Communication.

Downloaded from informs.org by [175.35.111.175] on 26 October 2025, at 21:54 . For personal use only, all rights reserved.