Minimum Cost Assignment of Crews to Meet Tracking Requirements

C. A. Greenhall[1]
TDA Planning Section

A model of the tracking constraints, maintenance constraints, labor constraints, and labor costs of a DSN complex is made. The problem of minimizing the labor cost while satisfying the constraints is solved. Minimum cost schedules for all cases of interest are given. Modifications of the model are suggested.

I. Introduction

This report gives a solution, subject to simplified assumptions, to the management science problem of scheduling the spacecraft tracking, station maintenance, and crew shifts at a DSN tracking complex. Section II defines the problem precisely, but here is an overview: assume we have a complex with 1 to 4 stations. There are 0 to 5 spacecraft to be tracked at that complex. Each spacecraft pass lasts 12 hours and must be tracked in its entirety by one and only one station, or not at all. (Four hours of pre- and post-calibration are included.) In this version of the problem, rises and sets are synchronized with each other modulo 12 hours and are constant from day to day. Each station requires 16 hours per week of maintenance subject to certain constraints, and must be open at least 40 hours a week. Work crews (whose number is not predetermined) are to be assigned the above duties. Their schedules are governed by constraints imposed by labor laws, sound personnel practice, and the mechanics of the situation. The problem is to schedule tracking, maintenance, and crew assignments in such a way that the labor cost is minimized, while meeting the various constraints.

The general case of the problem is labeled Case $(i, j), n$; this means that there are n stations, i spacecraft up during one 12-hour period of each day, and j spacecraft up during the other 12-hour period. We can always assume $j \leq i$. The restriction $0 \leq i + j \leq 5$ gives rise to 12 spacecraft configurations: (0,0), (1,0), (2,0), (1,1), (3,0), (2,1), (4,0), (3,1), (2,2), (5,0), (4,1), (3,2). Since there are 1, 2, 3, or 4 stations, the problem has $4(12) = 48$ different cases. In order to generalize, however, we will also consider cases with more than 5 spacecraft or 4 stations. From now on, the original cases of the problem will be called the "lower 48 cases."

Minimum cost schedules for all the lower 48 cases are given in Appendix C. But the purpose of this report is not just to solve these particular problems, for the constraints are perhaps not yet realisic enough for these schedules to be usable in the field without modification. Rather, the

[1]Consultant.

solution techniques will prove adaptable to similar DSN scheduling problems, but in realistic situations.

After stating the problem (Section II), we show that solutions always exist (Section III). Section IV gives an algorithm that computes a lower bound for the cost of a most economical schedule. The bound is actually attained in the lower 48 cases by the schedules in Appendix C. Construction of schedules is still partly *ad hoc*, but it is possible to give some guidelines (Section V). In any case, *some* schedule can be made, whether or not it is a cheapest one; its cost then certainly provides an upper bound for the minimum cost.

II. Description of the Model

A. Constraints

There is a DSN complex with n stations, $n \geq 1$. There are $i + j$ spacecraft up, $0 \leq j \leq i$. The first i spacecraft rise at midnight and set at noon. The other j rise at noon and set at midnight. We will call this Case (i, j), n.

(1) All schedules are periodic with period one week.

(2) The complex is manned by an indeterminate number of crews, each crew to be treated as a single indivisible unit. A crew works only 8- or 10-hour shifts. Possible work weeks are 40, 42, 44, 46, or 48 hours. Any crew can work at any station, but a crew must not switch stations during a shift. No more than two crews can be at one station at the same time. Starting times for consecutive shifts of the same crew must be at least 24 hours apart.

If work (tracking or maintenance) is being done at a station, then at least one crew is present. We allow crews to be on duty at a station but not working (in this mathematical sense).

(3) Each station must be open, with a crew, at least 40 hours a week.

(4) If a particular spacecraft is tracked at all during a pass, then it is to be tracked by one, and only one, station during its entire 12-hour pass. Any one station is allowed to track no more than 13 passes a week. (Fourteen are available but are regarded as an overload.)

(5) Each station must receive at least 16 "units" of maintenance a week. (The notion "unit" will be identified subsequently.) At each station, at least 12 hours must be spent on maintenance while the station is not tracking.

If the station is not tracking, and a single crew does maintenance (another crew could be present but idle), then each crew hour accomplishes 1 unit of maintenance. We say that the crew works with efficiency 1.

If tracking and maintenance are simultaneous at a station, then two crews are present, one tracking, the other doing maintenance. The crew that is doing maintenance has efficiency 2/3, that is, each crew hour accomplishes 2/3 units of maintenance. (The crew doing the tracking has priority, and is allowed to interfere with the maintenance crew; the latter's efficiency is therefore reduced.)

If two crews are at a station and both are doing maintenance, then no tracking occurs, and each crew has efficiency 2/3 again, so that both crews working for an hour accomplish 4/3 unit of maintenance. (The crews work at less than double efficiency because they interfere with each other.)

(6) Maintenance on each station must be done in "blocks." A block is a time interval of uninterrupted maintenance composed of x hours at efficiency 1, y hours by a single crew at efficiency 2/3, and z hours by two crews at efficiency 2/3, where $x + (2/3)\,y + z \geq 4$. (Observe that we have z, not $(4/3)z$. In this case, $x + (2/3)y + (4/3)z$ units of maintenance get done. (It has been found that this job cannot adequately be done in short time blocks. The multiplier 4/3 is removed from z in order that a block always be at least 4 *hours*, not just 4 *units*.)

This completes the list of constraints.

B. Costs

Each crew is paid time and one quarter for work beyond 40 hours. (We consider that half the crew is "exempt" and gets straight time, while the other half gets time and a half.) Thus a $(40 + 2k)$-hour work week is assigned a cost $40 + (5/2)k$, $k = 0, 1, 2, 3, 4$.

C. The Problem

Given the spacecraft configuration and the number of stations. If it is possible to track all spacecraft passes while satisfying all constraints, devise a minimum cost schedule of tracking, maintenance, and crew assignments that does so. If not all the passes can be tracked, find the maximum number of passes that can be tracked with the constraints still satisfied. Then find a minimum cost schedule that achieves this maximum.

III. Existence of Solutions

Let there be given n stations and $i + j$ spacecraft. From now on, a schedule of tracking, maintenance, and crew assignments that meets the constraints will be called simply a "schedule." We will soon see that schedules always exist. Any schedule tracks a whole number of passes; consider the nonempty set S of schedules that track the maximum possible number of passes. Since the cost of a $(40 + 2k)$-hour work week is $(5/2)(16 + k)$, the cost of any schedule is an integer multiple of 5/2. Therefore, there exist schedules in S that are cheapest.

It is *a priori* possible that the problem of finding the maximum possible amount of tracking is mixed inextricably with the maintenance and labor constraints, that a given tracking schedule cannot necessarily be completed to a schedule. Fortunately, this is not the case.

PROPOSITION 1. *Given a tracking schedule that satisfies Constraints 1 and 4. There exists a schedule of maintenance and crews that satisfies all the other constraints.*

Proof. If a given station is tracking fewer than 13 passes a week, then there are at least two 12-hour gaps in its tracking schedule. In these gaps, place a total of 16 hours of maintenance in blocks of at least 4 hours. Then schedule 4 crews, a, b, c, d, to this station as in Fig. 1. (We fill up the whole week even if the duty schedules do not demand it; since we are only proving existence we can be very wasteful—temporarily!) Call this a "row" of crews.

If a given station is tracking 13 passes a week, there is only one 12-hour gap in its schedule. Fill this gap with maintenance. Place a 6-hour block of maintenance anywhere else, to be performed simultaneously with tracking. This will yield 4 more units of maintenance. Now assign two separate rows of crews, each scheduled as in Fig. 1. Thus at all times there are two crews at the station. (The second crew does nothing, except during the 6-hour block of maintenance at efficiency 2/3; we said we would be wasteful.)

After carrying out this procedure for all stations, the reader can verify that all constraints are met.

It follows that to determine the maximum number of passes that can be tracked, we need consider only Constraints 1 and 4; this part of the problem can be solved first, without considering maintenance and crews.

IV. Computation of Cost Lower Bounds

Suppose that we find a lower bound for the costs of all schedules that maximize tracking. Suppose that we can construct a schedule that costs exactly this much. Then we have a minimum cost schedule. This we were able to do for all the lower 48 cases. If, for any reason, we had been unable to make a schedule whose cost equals the lower bound, but were able to make a higher-cost schedule, then at least we would have had both upper and lower bounds for the minimum cost.

We have no algorithm for finding minimum cost schedules in the general Case (i, j), n. However, we have prepared a structured flow chart (Figs. 2, 3, 4) that includes an algorithm for computing the maximum number of passes that can be tracked and a cost lower bound. This bound is valid for the general case and is sharp for the lower 48 cases.

The top of the flow chart, Boxes 1, 2, 3, 4, computes the maximum number p of passes that can be tracked. From then on, we consider only schedules that track this many passes. The total number of tracking hours is then $12p$.

Next comes the task of finding a lower bound for the number of crew hours that have to be paid for. Given a schedule that tracks p passes, let p_k be the number of passes tracked by Station k, $k = 1, 2, \cdots, n$. Then $p = \Sigma p_k$. Each station needs at least 16 units of maintenance. Since each crew hour results in 1 hour of tracking, 1 unit of maintenance, 2/3 units of maintenance, or nothing, the number of crew hours spent at Station k is at least $12p_k + 16$. Therefore, a lower bound for total crew hours is $\Sigma(12p_k + 16) = 12p + 16n$. Let us call this the *basic* hours lower bound.

Often, an hours lower bound greater than the basic one can be found. If at least $12p_k + 16 + c_k$ crew hours are worked at Station k, then the whole schedule has at least $12p + 16n + \Sigma c_k$ crew hours. In some cases, we can show that Σc_k is bounded below by some positive number.

After getting an hours lower bound, we compute the cost of the cheapest collection of work weeks such that the total hours worked is not less than the hours lower bound. The resulting cost must be a lower bound for the cost of any actual schedule.

The following paragraphs, keyed to the flow chart box numbers, explain the algorithm in detail.

Box 1. There are only n stations, so no more than n passes can be tracked at one time. If either i or j is greater than n, we can replace it by n and solve this new case. Then we will have done our best for the original case; we need only say which passes go untracked. So from now on, assume $j \leq i \leq n$.

Boxes 2, 3, 4. By Constraint 4, a total of no more than $13n$ passes a week can be tracked. The number of spacecraft passes is $7(i + j)$. Hence, the largest number p of passes that can be tracked cannot exceed the smaller of $13n$ and $7(i + j)$. Fortunately, this upper bound is attained.

PROPOSITION 2. *Assume $j \leq i \leq n$. There exists a schedule that tracks exactly minimum $(13n, 7(i + j))$ passes.*

Proof. By Proposition 1, we need only make a tracking schedule that meets Constraints 1 and 4.

Assume $7(i + j) \leq 13n$. For the first half of each day, we can relax a set of $n - i$ stations; for the second half, a set of $n - j$ stations. By the end of the week we have relaxed 14 sets of stations. Since the above inequality can be written $7(n - i) + 7(n - j) \geq n$, we can determine the sets so that their union is the set of all n stations. Then each station has been relaxed for at least one half-day, and all passes have been tracked.

Assume $7(i + j) > 13n$. Then $7(n - i) + 7(n - j) < n$. We can make the sets of relaxed stations disjoint. Then $7(n - i) + 7(n - j)$ stations have been relaxed exactly once, and tracking has been scheduled for all $7(i + j)$ passes. But there remain $n - 7(n - i) - 7(n - j) = 7(i + j) - 13n$ stations that need to be relaxed. Their relaxation periods can be stolen from the $7(i + j)$ passes that have just been scheduled. This leaves $7(i + j) - [7(i + j) - 13n] = 13n$ passes tracked.

If $7(i + j) \leq 13n$ then $p = 7(i + j)$. If $7(i + j) > 13n$, then $p = 13n$, and we will immediately establish an hours lower bound better than the basic one, which is $156n + 16n = 172n$. Each station has only one 12-hour period, or "gap," in which it is not tracking. By Constraint 5, there are no more than 12 crew hours spent on maintenance at efficiency 1, for such must take place when there is no tracking. To reach 16 units of maintenance, at least 6 crew hours must be spent at efficiency 2/3. Therefore, at least 18 crew hours per station are spent on maintenance. Since tracking takes up $12(13) = 156$ crew hours, each station accounts for at least 174 crew hours. Therefore, $174n$ is an hours lower bound.

Boxes 5, 6. At this point, all the $7(i + j)$ passes can be tracked, and the basic hours lower bound is $84(i + j) + 16n$. Even so, some stations may still have only one relaxation period. Assume $12n < 7(i + j) \leq 13n$. The number of relaxation periods is $14n - 7(i + j)$. If s is the number of stations with only one relaxation period, then the other $n - s$ stations have at least 2 relaxation periods. Therefore, $14n - 7(i + j) \geq s + 2(n - s)$, i.e., $s \geq 7(i + j) - 12n$. Let $s' = 7(i + j) - 12n$. There exist s' stations with one relaxation period. Each of these stations accounts for at least 174 crew hours, but contributes only $12(13) + 16 = 172$ hours to the basic hours lower bound. It follows that $2s'$ can be added to the basic hours lower bound. This gives a new hours lower bound of $84(i + j) + 16n + 14(i + j) - 24n = 98(i + j) - 8n$.

Boxes 7, 8. Here, $84(i + j) + 16n$ is an hours lower bound. By Constraint 3, so is $40n$. Hence, if $84(i + j) + 16n < 40n$, i.e., $7(i + j) < 2n$, we use $40n$ as our hours lower bound. Otherwise, we try to improve on the basic bound.

Boxes 9, 10, and Fig. 3. If $j = 0$ when we reach Box 9, then it may be possible to improve on the basic bound $84i + 16n$. There are $7i$ passes, and all are tracked during the first half of the day. Each 12-hour tracking interval is isolated from the others on that station's schedule. Since shifts are 8 or 10 hours, each interval must touch at least 2 shifts, and no shift touches more than one interval. It follows that each of the $7i$ tracking intervals accounts for at least 16 crew hours. Accordingly, $(7i)(16) = 112i$ is an hours lower bound. If $7i \leq 4n$, then $84i + 16n \geq 112i$; we keep the basic bound. If $7i > 4n$ then $112i$ is a better bound.

Boxes 11, 12, and Fig. 3. If $i = n$ at Box 11, then the tracking schedule may still be forced to have so many isolated intervals that the basic hours lower bound $84(i + j) + 16n = 100n + 84j$ can again be improved upon. The argument is more complicated; it is necessary to consider each station separately. All stations track during the first half of the day, and $n - j$ of them have a "gap" (are not tracking) during the second half of the day. The entire tracking schedule has $7(n - j)$ gaps, and if this number is large enough compared with n, some stations are forced to have 6 or 7 gaps: If $7(n - j) > 6n$, i.e., $n > 7j$, then at least one station has 7 gaps, otherwise the number of gaps would be at most $6n$. If $7(n - j) > 5n$, i.e., $2n > 7j$, then at least one station has at least 6 gaps. We will elaborate on this later.

Let us investigate the consequences of 7 gaps in the tracking schedule of a station. The schedule is the same as the right side of Fig. 5, except for a rotation. There are 7 isolated tracking intervals. As the argument for Box 10 shows, these intervals touch at least 14 crew shifts, which account for at least $8(14) = 112$ crew hours. This is 12 more than the $7(12) + 16$ hours contributed by this station to the basic hours lower bound.

If a station has 6 gaps in its schedule, then this schedule looks like the left side of Fig. 5. There are 5 isolated tracking intervals, which touch at least 10 shifts, which account for at least 80 crew hours. There are 3 other tracking intervals, which account for at least 36 more crew hours. Accordingly, this station accounts for at least 116 crew hours, which is 4 more than the $8(12) + 16$ hours contributed to the basic hours lower bound.

Consider now Case (n,j), n. There are $7(n - j)$ gaps; let the kth station have g_k gaps, where $1 \leq g_k \leq 7$. Without losing generality, we can assume $g_1 \geq g_2 \geq \cdots \geq g_n$. If $g_k = 7$, then set $c_k = 12$ (12 crew hours over the basic bound). If $g_k = 6$ set $c_k = 4$. If $g_k = 5$ set $c_k = 0$. There is a choice of the g_k that minimizes $\Sigma\, c_k$ subject to the constraint $\Sigma\, g_k = 7(n - j)$. This minimum we call "extra," and the new hours lower bound is $100n + 84j$ + extra.

Proposition 3. *Let $i = n$, $7(i + j) \leq 12n$. Then*

$$\begin{aligned} &\text{extra} = 0 && \text{if } 7j \geq 2n \\ &\text{extra} = 8n - 28j && \text{if } n \leq 7j \leq 2n \\ &\text{extra} = 12n - 56j && \text{if } 7j < n. \end{aligned}$$

We have relegated the proof to Appendix A.

The only case in the lower 48 such that $i = n$, $j > 0$, and extra > 0 is Case (4,1), 4. There are $7(4 - 1) = 21$ gaps, and the distribution of gaps that minimizes $\Sigma\, c_k$ is 6,5,5,5. Therefore, extra = 4.

Box 13. The basic hours lower bound is sharp for the lower 48 cases that reach this place in the flow chart.

Box 14. Let h be an hour's lower bound. To compute a cost lower bound, we must find a string of work weeks $40 + 2x_1, 40 + 2x_2, \cdots, 40 + 2x_m$ ($x_i = 0,1,2,3,4$; m not predetermined), such that the total crew time $\Sigma\,(40 + 2x_i) = 40m + 2\Sigma x_i$ is *at least h*, while the cost $\Sigma\,(40 + (5/2)\,x_i) = 40m + (5/2)\,\Sigma x_i$ is minimized. At this point we need only work with the two numbers m and $2\Sigma x_i$. It will suffice to give two examples from the lower 48 cases.

Case (3,1), 3. Here $h = 384$. Since $8(48) = 384$, at least 8 crews are needed. If 8 crews are used, the cost is $8(50) = 400$. If 9 crews are used, then there are $9(40) = 360$ regular hours and at least 24 overtime hours. The minimum cost of 9 crews is then $360 + (5/4)\,24 = 390$. If 10 or more crews are used, the cost is at least $10(40) = 400$. Therefore, a cost lower bound is 390.

Case (2,1), 4. Here $h = 316$. Since $6(48) < 316$, at least 7 crews are needed. If 7 crews are used, the cost is at least $7(40) + (5/4)\,36 = 325$. If 8 crews are used, the cost is at least $8(40) = 320$, which is therefore a cost lower bound. Notice that it pays to waste 4 crew hours.

Occasionally, the solution for m and Σx_i is not unique. See, for example, Case (1,1), 4.

Box 15. We have obtained the number of crews m and the overtime hours $2\Sigma x_i$ that achieve the cost lower bound. Before one attempts to construct a schedule that costs just this much, it may be helpful to list all the ways that these overtime hours can be distributed among the m work weeks (with the work weeks in non-increasing order, for example). For example, in Case (1,1), 1 we have $m = 4$, $x_1 + x_2 + x_3 + x_4 = 7$. The list of ways to write 7 as the sum of four non-increasing integers between 0 and 4 is $4 + 3 + 0 + 0$, $4 + 2 + 1 + 0$, $4 + 1 + 1 + 1$, $3 + 3 + 1 + 0$, $3 + 2 + 2 + 0$, $3 + 2 + 1 + 1$, $2 + 2 + 2 + 1$. These yield the following work week splits:

48, 46, 40, 40

48, 44, 42, 40

48, 42, 42, 42

46, 46, 42, 40

46, 44, 44, 40

46, 44, 42, 42

44, 44, 44, 42

V. Construction of Schedules (Figure 4, Box 16)

If this is the first time we have come to this box, then we have a list of work-week splits with cost equal to a cost lower bound. We can do no more than give some imprecise guidelines for constructing a weekly schedule that uses one of these splits.

First, we make a tracking schedule that satisfies Constraint 4 and tracks p passes. In doing this, we avoid isolated tracking intervals as much as possible, for as we saw in the discussion of Boxes 10 and 12, each interval forces at least 4 hours of non-tracking crew hours, which are either spent on maintenance or wasted. If there are 4 isolated intervals at a station, then we can adjoin a 4-hour block of maintenance to each and cover the resulting 16 hours of work by two 8-hour shifts. If there are fewer than 4 isolated intervals at a station, then there is more freedom in assigning maintenance. If the algorithm has gone through Boxes 10 or 12, then we know how many isolated intervals we have to handle. Otherwise, we hope that we can get by with 4 or fewer per station. This is so for the lower 48 cases, but if we run into a case that requires more isolated intervals (and can prove that it does), then we can add something to the basic hours lower bound and go back to Box 14.

Next, we assign 16 hours of maintenance to each station while observing Constraints 5 and 6. The proof of Proposition 1 tells how to start. If a station is tracking fewer than 13 passes, so that maintenance can be disjoint from tracking, then, as a general rule, we try to assign maintenance so that the duty intervals of tracking plus contiguous maintenance have lengths which are multiples of 8 hours. Such an interval can be covered tightly by 8-hour shifts, which are easier to work with. If the length of a duty interval is an odd multiple of 4 hours and is at least 20 hours, then two 10-hour shifts plus some 8-hour shifts will cover it tightly.

Finally, we assign crew shifts. There is a list of work week splits (Box 15). Each work week can be split in turn into 8- and 10-hour shifts, perhaps in more than one way. We try to choose a work week split and shift splits so that there are just enough 10-hour shifts to suit the tracking and maintenance schedule. We give names a, b, c,··· to the crews, and show the shift split for each. On the schedule, we show where the 8- and 10-hour shifts are to go. Then the shifts are labeled with crew names such that Constraint 2 (especially the 24-hour part) is satisfied. For all the lower 48 cases, this can be done by labeling from top down, then left to right. (We must make sure that the 10-hour shifts are labeled correctly.) It may then happen that the end of this week and the beginning of the next week violate the 24-hour constraint. If so, we try to remedy the situation by juggling the labels. For the lower 48 cases, this works.

At any point, it may be necessary or convenient to go back and choose a different shift split, work week split, maintenance schedule, or tracking schedule, and proceed again from there.

The scheduling process is really quite easy, for most of the thinking has been done by the time a good cost lower bound is derived. There seems to be considerable leeway in the construction of schedules; the first or second choice usually works. We were initially successful in all lower 48 cases (except for Case (4,1), 4; see the remark at the end of Section VI), and thus made it to Box 17 (cheapest schedule found).

VI. Improving the Bound (Figure 4)

Figure 4 is a guide to follow in case we cannot construct a schedule whose cost equals the cost lower bound we have on hand. The reason for this failure may be either that no such schedule exists, or that we have not been persistent or clever enough. By Proposition 1, it is possible to make *some* schedule that tracks the greatest possible number of passes. We do this as cheaply as we are able (Box 18). The cost of the schedule so made is an upper bound for the cost of a cheapest schedule.

If we cannot prove that our cost lower bound can be increased, then we leave the flow chart by way of Box 21 with upper and lower bounds for the cost of a cheapest schedule. If we *can* prove that there is a greater lower bound, we do so (Box 20). Then we try to make a schedule that achieves this new bound (Box 16). Thus we go around the loop (a finite number of times) until we achieve an exit through Box 17 or are forced out through Box 21.

Case (4,1), 4 drove us once around the loop; the cheapest schedule we could make exceeded our lower bound by 4 hours. This forced us to make the argument associated with Box 12; the lower bound increased by 4 hours; so we made it to Box 17. Of course, the improvement is now part of the algorithm.

The FLAG device is merely a way of avoiding a double exit from the loop; the entire flow chart follows the rules of structured algorithms, the proposed DSN standard for software.

VII. Proposed Changes in the Model

We have assumed that rises and sets of spacecraft are synchronized with each other modulo 12 hours. If we remove this constraint, the spacecraft passes could have lengths other than 12 hours, and rises would no longer need to coincide with the starts of half-days. Perhaps some randomness could be built in. This modification would introduce more parameters into the problem.

Some of the labor constraints may have to be tightened. The International Brotherhood of Electrical Workers/ Philco-Ford Corporation Labor Agreement requires that each shift have a regular start time, that overtime be equalized among the crews, and that days off be consecutive. Exceptions to these rules can occur, but *each* such exception must be negotiated between the company and the union.

The cost function of the problem may need modification. For example, we have assumed that no overtime be paid for a work week of four 10-hour shifts. The Walsh-Healey Act requires that any business with a government contract pay overtime to non-exempt employees for hours worked in excess of 8 in a given work day. Philco-Ford Company policy requires that time and a half be paid to exempt employees for scheduled time in excess of 8 hours in a given work day. (A bill has been introduced in the California Legislature to repeal the State overtime requirement on 10-hour days, but the Federal would still control.)

a: 6(8). b, c, d: 5(8)

MON	TUE	WED	THU	FRI	SAT	SUN	MON
b a c	d a b	c d a	b c d	a b c	d a b	c a d	b a c

Fig. 1. A crew schedule that fills the week

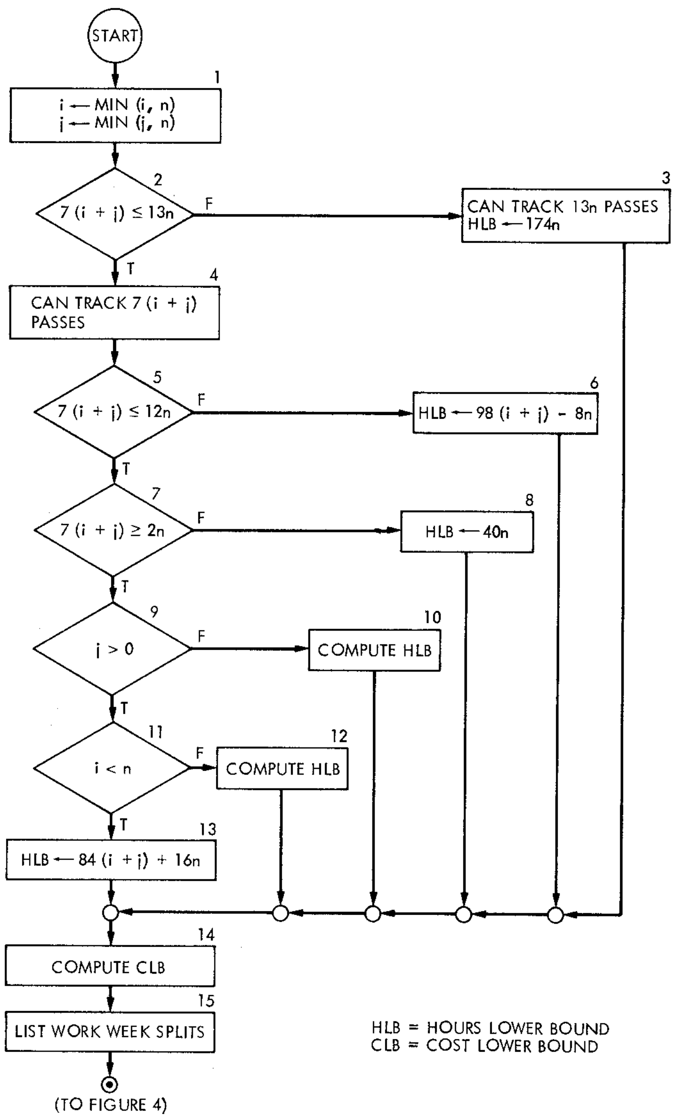


Fig. 2. Cost lower bound algorithm for Case *(i, j) n*

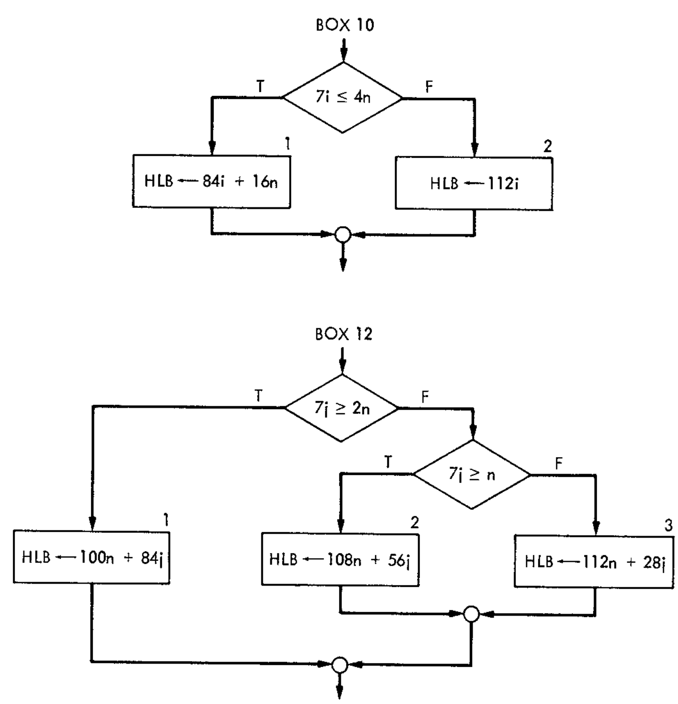


Fig. 3. Boxes 10 and 12 (Fig. 2)

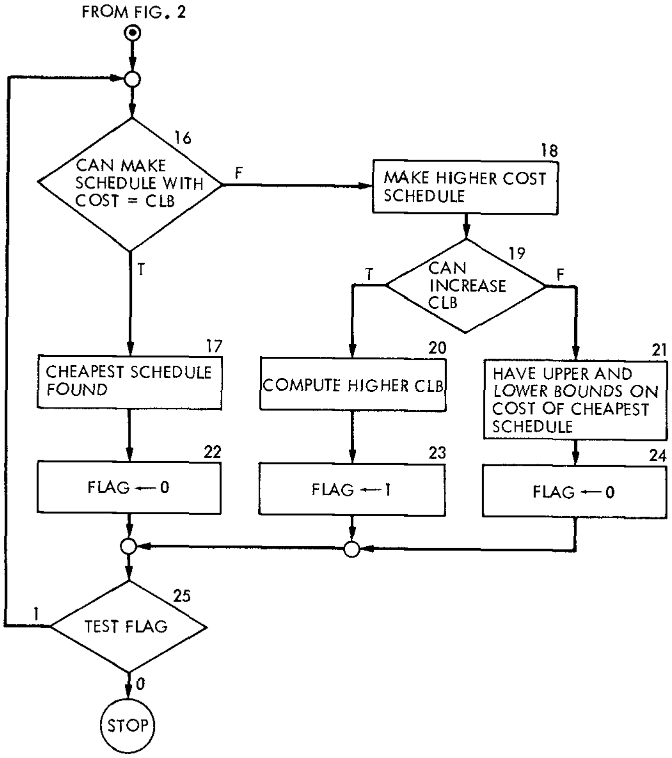


Fig. 4. Improving the bound from Fig. 2

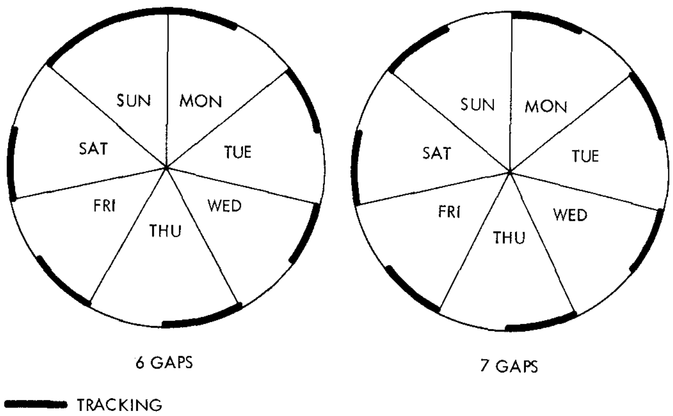


Fig. 5. Tracking schedules for Cases $i = n$

Appendix A
Proof of Proposition 3

Let n be a positive integer, j a nonnegative integer such that $7j \leq 6n$. (Actually, we know $7j \leq 5n$.) Define a function c by

$$\begin{aligned} c(g) &= 12, && \text{if } g = 7 \\ &= 4, && \text{if } g = 6 \\ &= 0, && \text{otherwise} \end{aligned}$$

Consider the problem

$$\text{Find } M = \text{minimum} \sum_{k=1}^{n} c(g_k)$$

subject to the constraints

(1) g_k an integer, $1 \leq g_k \leq 7 \qquad (k = 1, \cdots, n)$

(2) $\sum_{k=1}^{n} g_k = 7(n-j)$

The solution is

$$\begin{aligned} M &= 0 && \text{if } 7j \geq 2n \\ &= 8n - 28j && \text{if } n \leq 7j \leq 2n \\ &= 12n - 56j && \text{if } 7j < n \end{aligned}$$

Proof

Let $7j \geq 2n$. Then

$$n \leq 7(n-j) \leq 5n$$

Accordingly, $g_1, \cdots, g_n$ can be chosen between 1 and 5 to add up to $7(n-j)$.

Let $n \leq 7j < 2n$. Then $7(n-j) > 5n$; we need some 6's or 7's. Suppose that m of the g_k are 7. Then the remaining $n - m$ g_k add up to $7(n-j-m)$ and are between 1 and 6. Suppose

$$7(n-j-m) > 5(n-m)$$

Then at least $7(n-j-m) - 5(n-m) = 2n - 7j - 2m$ 6's are needed, for otherwise the sum of the $n - m$ terms is less than $7(n-j-m)$. In this case (with $c_k = c(g_k)$),

$$\begin{aligned} \Sigma c_k &\geq 12m + 4(2n - 7j - 2m) = 8n - 28j + 4m \\ &\geq 8n - 28j = 4(2n - 7j) \end{aligned}$$

Suppose on the other hand that

$$7(n-j-m) \leq 5(n-m)$$

Then $2m \geq 2n - 7j$, no 6's are needed, and

$$\Sigma c_k \geq 12m \geq 12n - 42j = 6(2n - 7j) > 4(2n - 7j)$$

since $2n - 7j > 0$. In either case, $\Sigma c_k \geq 8n - 28j$, and this bound can be achieved by using no 7's, $2n - 7j$ 6's, and the remaining $7j - n$ g_k equal to 5.

Let $7j < n$. Then at least $n - 7j$ 7's are needed. Suppose in fact that $n - 7j + r$ 7's are used. The remaining $7j - r$ g_k add to $7(n-j) - 7(n - 7j + r) = 42j - 7r$. Suppose $42j - 7r > 5(7j - r)$. Then at least $42j - 7r - 5(7j - r) = 7j - 2r$ of these g_k must be 6. In this case,

$$\begin{aligned} \Sigma c_k &\geq 12(n - 7j + r) + 4(7j - 2r) \\ &= 12n - 56j + 4r \geq 12n - 56j \end{aligned}$$

On the other hand, if $42j - 7r \leq 5(7j - r)$, then $2r \geq 7j$, no 6's are needed, and

$$\Sigma c_k \geq 12(n - 7j + r) \geq 12n - 84j + 42j = 12n - 42j$$

In either case, $\Sigma c_k \geq 12n - 56j$, and this bound is achieved by using $n - 7j$ 7's and $7j$ 6's.

Appendix B

Table of Minimum Costs for the Lower 48 Cases

Figure B-1 gives the minimum cost for each of the lower 48 cases. For each case we enter the minimum cost and the "slack," defined by slack = minimum cost − $(12p + 16n)$, where p is the maximum number of passes that can be tracked, and $16n$ is the required maintenance. If it is not possible to track all spacecraft passes, then a "1" or a "3" is entered. A "1" means that either i or j is greater than n. (See Box 1 of Fig. 2.) A "3" means that the case runs into the constraint that no station can track 14 passes a week (Box 3 of the flow chart).

ENTRIES: MINIMUM COST/SLACK[a]

n \ (i, j)	(0, 0)	(1, 0)	(2, 0)	(1, 1)	(3, 0)	(2, 1)	(4, 0)	(3, 1)	(2, 2)	(5, 0)	(4, 1)	(3, 2)
1	40 24	120 20	120 1 20	177 1/2 3 5 1/2	120 1 20	177 1/2 3 1 5 1/2	120 1 20	177 1/2 3 1 5 1/2	177 1/2 3 1 5 1/2	120 1 20	177 1/2 3 1 5 1/2	177 1/2 3 1 5 1/2
2	80 48	120 4	230 30	200 0	230 1 30	285 1	230 1 30	285 1 1	355 3 11	230 1 30	285 1 1	355 3 1 11
3	120 72	135 3	230 14	220 4	340 40	305 5	340 1 40	390 6	390 6	340 1 40	390 1 6	475 7
4	160 96	160 12	240 8	240 8	340 24	320 4	450 50	400 0	400 0	450 1 50	490 6	485 1

[a]SLACK = COST - (TRACKING + MAINTENANCE)

Fig. B-1. Minimum cost of lower 48 Cases

Appendix C

Minimum Cost Schedules for the Lower 48 Cases

For each case, the stations are named A, B, C, D and the crews a, b, c, $\cdots$. The shift split of each crew is given. The days of the week run from Monday to Sunday, although this is arbitrary. Monday is repeated at the right end of the schedules. An interval of tracking is shown by a solid line; we do not bother to state which spacecraft is being tracked. Maintenance is shown by dashed lines. Below the tracking and maintenance schedules for a station we put the crew shifts. Each 10-hour shift is indicated by a superscript; otherwise it is 8 hours. Occasionally the minimum cost can be achieved with more than one value for crew hours. When this happens, we give an alternate crew schedule by using Greek letters $\alpha, \beta, \gamma, \cdots$. An example is Case (1,1), 4.

A flow chart box number, which refers to Fig. 2, shows which path the cost lower bound algorithm takes. If Box 10 or 12 is cited, then the next number in parentheses refers to the box numbers in Fig. 3.

The cases with $i > n$ or $j > n$ are omitted because they reduce to other cases. (See Box 1, Fig. 2.)

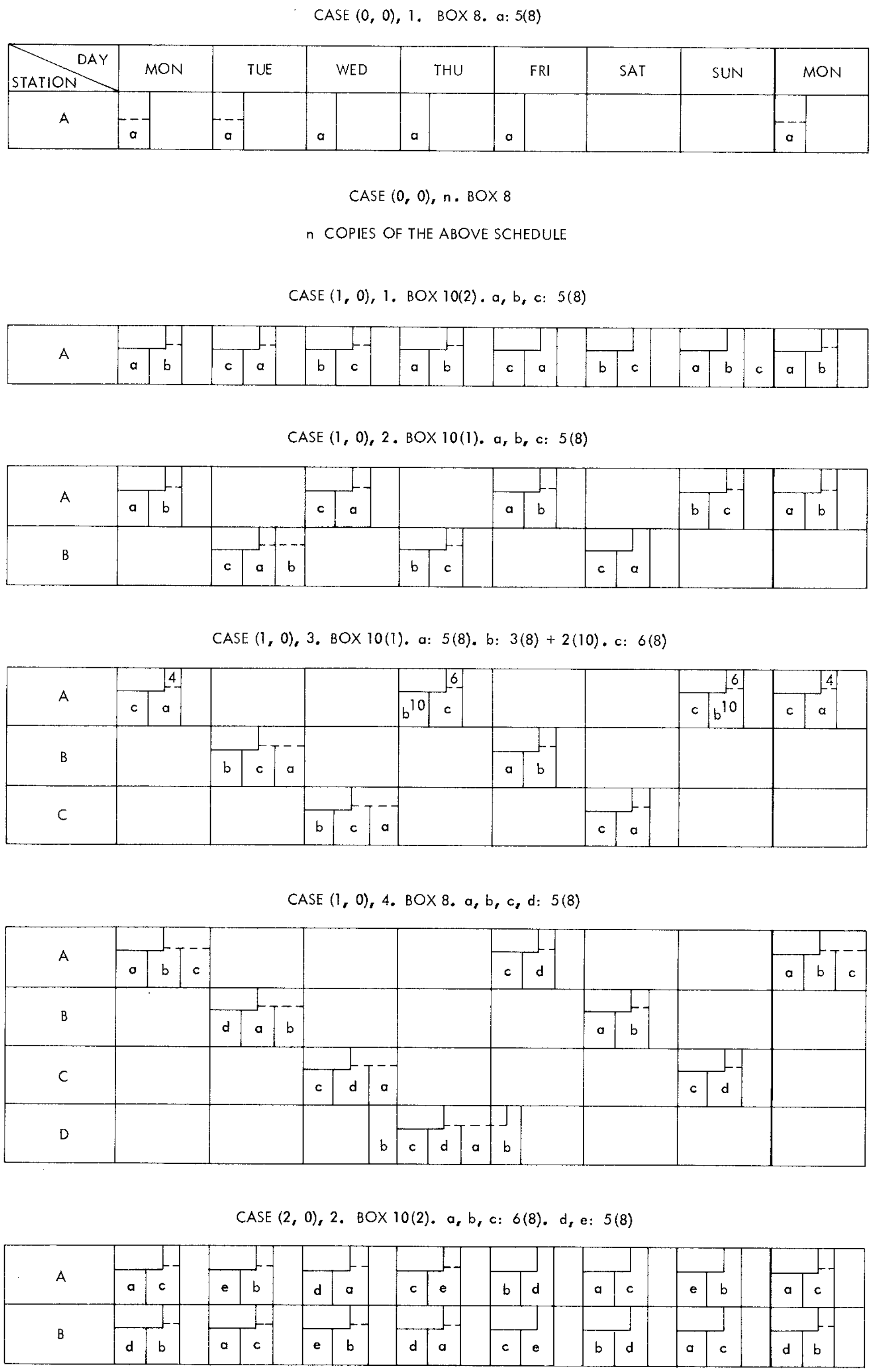
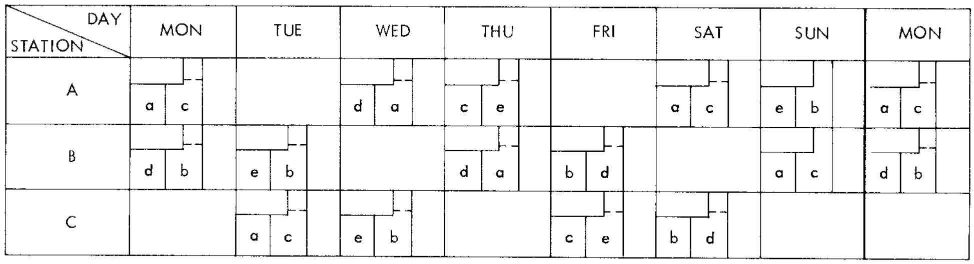
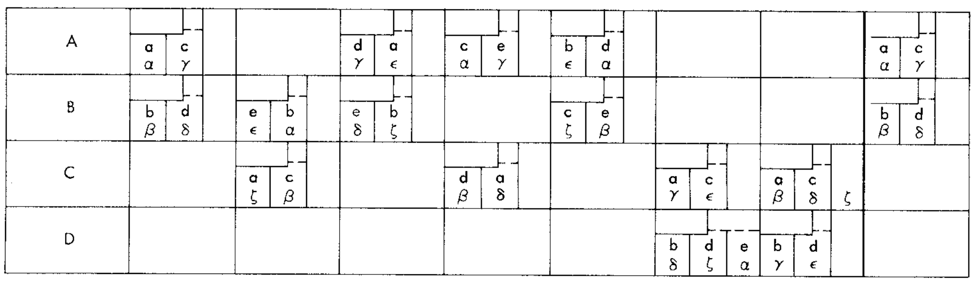


Fig. C-1. Minimum cost schedules for the lower 48 Cases

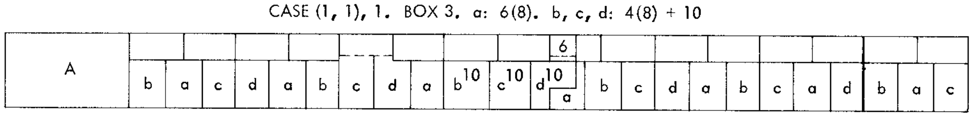
CASE (2, 0), 3. BOX 10(2). a, b, c: 6(8). d, e: 5(8)



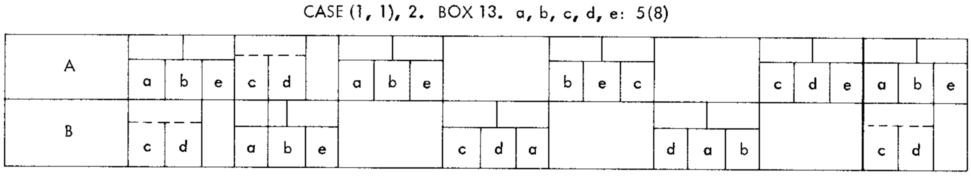
CASE (2, 0), 4. BOX 10(1). a, b, c, d: 6(8). e: 5(8)
α, β, γ, δ, ϵ, ζ : 5(8)



CASE (1, 1), 1. BOX 3. a: 6(8). b, c, d: 4(8) + 10



CASE (1, 1), 2. BOX 13. a, b, c, d, e: 5(8)



CASE (1, 1), 3. BOX 13. a, b: 6(8). c, d, e: 5(8)

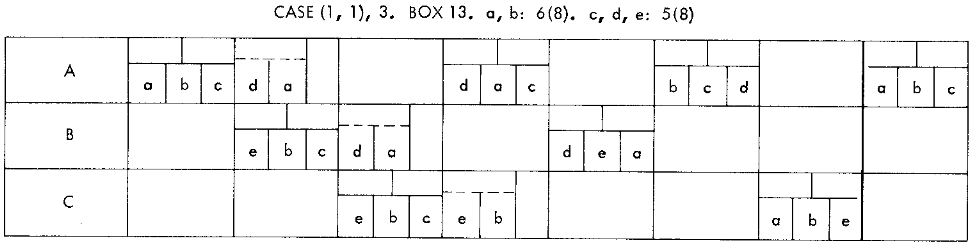
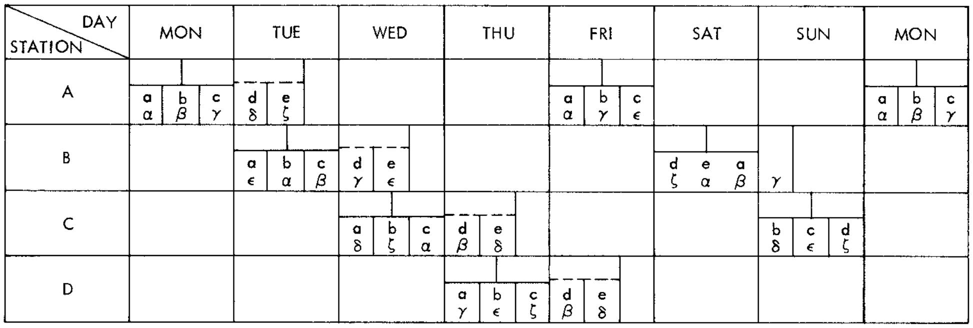
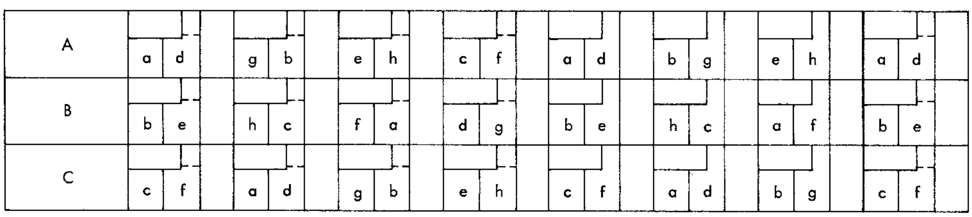


Fig. C-1 (contd)

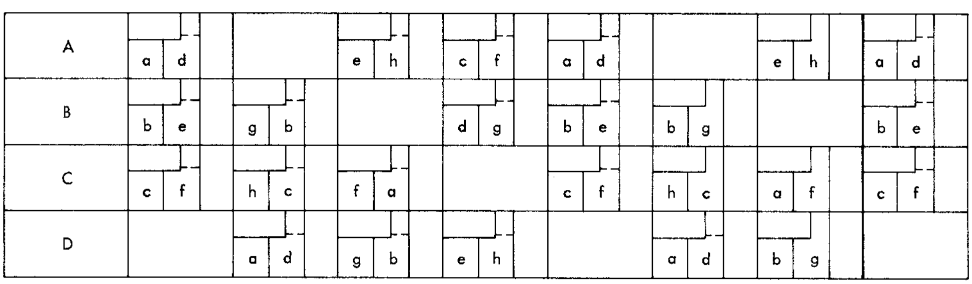
CASE (1, 1), 4. BOX 13. a, b, c, d: 6(8). e: 5(8)
α, β, γ, ε, δ, ζ : 5(8)



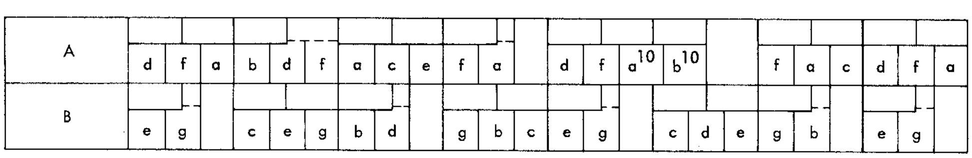
CASE (3, 0), 3. BOX 10(2). a, b: 6(8). c, d, e, f, g, h: 5(8)



CASE (3, 0), 4. BOX 10(2). a, b: 6(8). c, d, e, f, g, h: 5(8)



CASE (2, 1), 2. BOX 12(1). a, b: 4(8) + 10. c, d, e, f, g: 5(8)



CASE (2, 1), 3. BOX 13. a, b: 6(8). c: 3(8) + 2(10). d, e, f, g: 5(8)

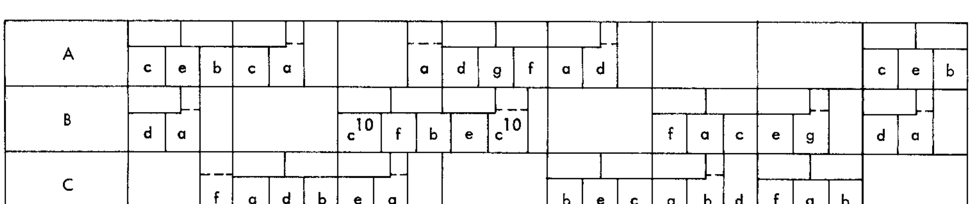


Fig. C-1 (contd)

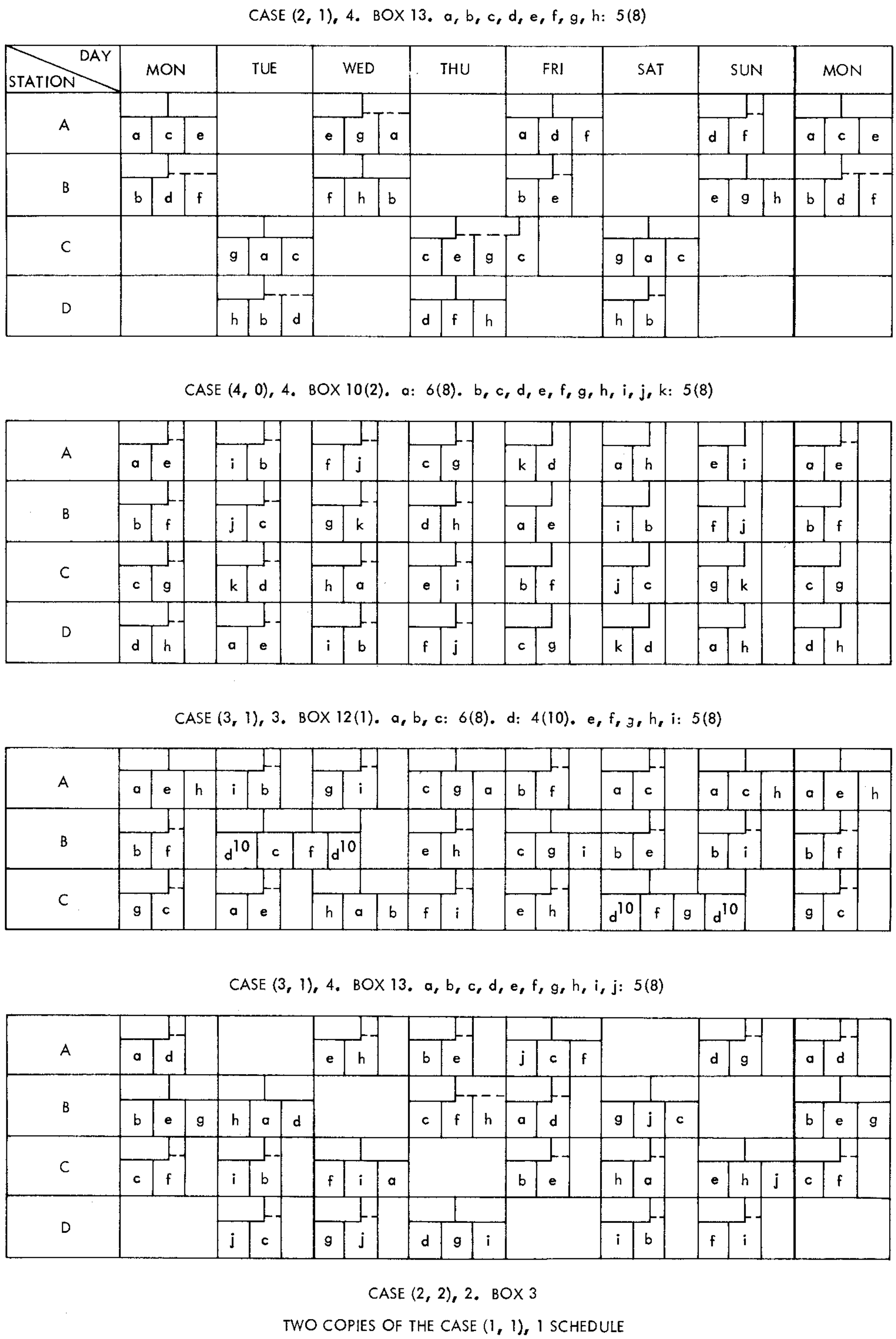

CASE (2, 1), 4. BOX 13. a, b, c, d, e, f, g, h: 5(8)
DAY
STATION
MON
TUE
WED
THU
FRI
SAT
SUN
MON
A
B
C
D
CASE (4, 0), 4. BOX 10(2). a: 6(8). b, c, d, e, f, g, h, i, j, k: 5(8)
CASE (3, 1), 3. BOX 12(1). a, b, c: 6(8). d: 4(10). e, f, g, h, i: 5(8)
CASE (3, 1), 4. BOX 13. a, b, c, d, e, f, g, h, i, j: 5(8)
CASE (2, 2), 2. BOX 3
TWO COPIES OF THE CASE (1, 1), 1 SCHEDULE

Fig. C-1 (contd)

CASE (2, 2), 3. BOX 13. a, b, c: 6(8). d, e, f, g, h, i: 5(8)

STATION \ DAY	MON	TUE	WED	THU	FRI	SAT	SUN	MON
A	a c e	g a	f i c	e h b		a c e	b g i	a c e
B	b d f	h b d	g a	f i c	d f h		a c h	b d f
C		i c e	h b d	g a	e g i	b d f		

CASE (2, 2), 4. BOX 13

TWO COPIES OF THE CASE (1, 1), 2 SCHEDULE

CASE (4, 1), 4. BOX 12(2). d: 4(10) + 8. l: 4(10). a, b, c, e, f, g, h, i, j: 5(8)

A	a e h	i l^{10} d^{10}	i	b f	k c	g k	e i	a e h
B	b f	j b	f j l^{10}	c g j	l^{10}	h a	f j	b f
C	c g	k c	g k	d h	a d^{10} f	i b d^{10}	h	c g
D	d^{10}	a e	h a	e i	b e	j c	g k l^{10}	d^{10}

CASE (3, 2), 3. BOX 12(1). a, b, c: 6(8). d: 3(8) + 2(10). e, f, g, h, i, j, k: 5(8)

A	a d g	j b e	h k c	f i k	b e	j b e	g b j	a d g
B	h b e	k c f	i a e	g j a	c f h	k c	h c k	h b e
C	f c i	a d^{10} g	j b d^{10}	h	d g i	a d f	a i	f c i

CASE (3, 2), 4. BOX 13. b: 2(10) + 3(8). a, c, d, e, f, g, h, i, j, k, l: 5(8)

A	a d g		g j a	d g j	a d g		e h	a d g
B	b^{10} f i	l b^{10} f		e h k	b e h	i l c		b^{10} f i
C	c e h	j a d	h k b		c f	j a d	f i k	c e h
D		k c e	i l c	f i l		k b	g j l	

Fig. C-1 (contd)