FAA-RD-72-7

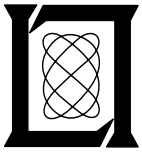
Project Report
ATC-8

Interrogation Scheduling for the Discrete Address Beacon System

E. J. Kelly

24 January 1972

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS



Prepared for the Federal Aviation Administration,
Washington, D.C. 20591

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FAA-RD-72-7		

4. Title and Subtitle	5. Report Date
Interrogation Scheduling for the Discrete Address Beacon System	24 January 1972
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
E. J. Kelly	ATC-8

9. Performing Organization Name and Address	10. Work Unit No.
Massachusetts Institute of Technology Lincoln Laboratory P.O. Box 73 Lexington, Massachusetts 02173	Proj. No. 034-241-012
	11. Contract or Grant No. IAG DOT-FA72WAI-242
	13. Type of Report and Period Covered Project Report
12. Sponsoring Agency Name and Address Department of Transportation Federal Aviation Administration Office of Systems Engineering Management Systems Research and Development Service Washington, D.C. 20590	14. Sponsoring Agency Code

15. Supplementary Notes
The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology under Air Force Contract F19628-70-C-0230.

16. Abstract

This report is an attempt to define the interrogation scheduling problem which arises in the implementation of the discrete address beacon idea. The interfaces of this problem with other parts of the beacon system design are discussed, and several specific algorithms for scheduling are analyzed for both arrays and rotating antennas.

17. Key Words	18. Distribution Statement
Discrete Address Beacon Interrogation Scheduling Algorithm	Availability is unlimited. Document may be released to the National Technical Information Service, Springfield, Virginia 22151, for sale to the public.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	60	3.00 HC .95 MF

Form DOT F 1700.7 (8-69)

TABLE OF CONTENTS

I. INTRODUCTION

The FAA is currently undertaking the development of a new surveillance system, the Discrete Address Beacon System (DABS), which will complement the introduction of automation in air traffic control to form the so-called upgraded third generation system. The basic system concept was spelled out in the ATCAC Report [1], which recommended its development, and a detailed discussion of the system and its planned development may be found in the FAA Technical Development Plan [2] for DABS. The basic idea, which gives DABS its name, is to assign to each aircraft a unique code, and to direct all interrogations to a given aircraft by using that aircraft's unique address. Each aircraft is interrogated in turn, and responds only to interrogations containing its address. This scheme provides the mechanism for eliminating overlapping replies from targets close in range, by scheduling interrogations in time so that the replies do not overlap, and at the same time establishes a data-link between aircraft and the ground which can be used for traffic control purposes.

The interrogation scheduling problem itself is the subject of this note, in which we attempt to define the problem, discuss some specific algorithms, and study their capacity in terms of the number of targets a DABS can handle. In the following sections we give a brief discussion of the DABS system concept, which provides the background for the interrogation scheduling problem, and then discuss the scheduling problem in general terms, with emphasis on its interaction with other elements of the DABS system design, such as surveillance and communication requirements, modulation technique, antenna structure and method of azimuth measurement. A discussion of ATCRBS serves as the basis for an analysis of the DABS-ATCRBS interlace problem. Specific algorithms are discussed first in the context of a phased array. Classes of algorithms are introduced and examples given, followed by a treatment of the effect of

constraints imposed by other parts of the system design. A similar discussion is then given of algorithms suitable to the rotator. Finally, target motion and tracking requirements are analyzed.

The entire subject is new, and the treatment given here is introductory in nature. Much room remains for the development of new algorithms, suited to various distributions of targets in space and external constraints on the scheduler.

2. THE DABS SYSTEM CONCEPT

When the beacon system (ATCRBS) was added to primary radar, the ATC surveillance system was changed from a non-cooperative, pure surveillance system to a cooperative system, providing surveillance and a limited amount of one-way communication (on the down-link). The down-link "messages" of identity and altitude are extremely useful to the ground control system, but their long duration (compared to a primary radar echo) results in greatly reduced range resolution and frequent failure to decode messages correctly due to the overlapping of replies ("synchronous garble"). The next step, to DABS, provides a more complete communication system, since each aircraft is discretely accessible, and at the same time provides garble-free surveillance by ranging and azimuth angle measurement. Altitude is reported by the aircraft, as before, along with message acknowledgment and, possibly, the readings of various other on-board sensors. Initiation of a transmission by an aircraft is not a part of the present system concept, although aircraft-initiated messages can be sent to the ground as part of ordinary replies to surveillance interrogations. Ground-to-air messages can also be sent as part of ordinary surveillance interrogations, although the system should provide the capability of

breaking into the normal surveillance cycle in order to transmit messages of high priority (e. g., collision-avoidance commands) to specified aircraft. Thus, the task of the interrogation scheduler is two-fold:

a. to provide garble-free surveillance at some regular refresh rate,

b. to meet the message-transmission demands of the (automated) traffic control system.

At present the first of these tasks is better understood than the second, and hence it will be treated in more detail in this report.

In addition to providing surveillance and communications access to targets already in the discrete address roll call, the DABS sensor must be capable of acquiring (or re-acquiring) DABS-equipped aircraft not presently being discretely addressed, and it must provide surveillance on targets equipped only with ATCRBS transponders for the duration of an extended transition period. The second point, the need to provide data on ATCRBS targets, has a major influence on the interrogation scheduling problem for DABS. Aside from the possibility of using separate antennas for DABS and ATCRBS interrogations (and replies), this means that a DABS sensor must be time-shared between a DABS mode and an ATCRBS mode. In certain cases, discussed in more detail below, it is advantageous to alternate rapidly between these modes, so that the channel is available for DABS interrogations and reply only during successive intervals of time, which may be no more than a few milliseconds in length. Such a chopping of the DABS channel occupancy time has a profound effect on the utility of different scheduling algorithms.

The need to provide an acquisition mode is less serious since this mode is virtually identical in function to ATCRBS scanning, and it can

probably be included in the ATCRBS mode by some modification of the standard ATCRBS interrogation. DABS transponders must also have the capability to reply to ATCRBS interrogations, hence a "pop-up" DABS aircraft would automatically be seen in the ATCRBS mode, and the special nature of its ATCRBS replies would lead to its eventual inclusion in the DABS roll call, after which it could be "locked out" from ATCRBS, i.e., told to reply no longer to ATCRBS interrogations. These operational questions have only indirect bearing on the DABS interrogation scheduling problem, and a more detailed discussion may be found in Ref. 2. It is not necessary for DABS sensors to operate with ATCRBS transponders in exactly the same way as do contemporary ATCRBS interrogators, so long as the resulting surveillance performance equals or exceeds that presently attained. Thus, a certain freedom of design remains in scheduling ATCRBS interrogations as well as in processing ATCRBS replies. This problem is discussed below along with the general question of DABS/ATCRBS interlacing.

3. THE INTERROGATION SCHEDULING PROBLEM

In this section the general definition of the interrogation scheduling problem given above will be sharpened by considering, in turn, the interaction of interrogation scheduling with the other major parts of DABS, in terms of interface parameters, requirements and constraints.

a. Surveillance Requirements

The requirements of the air traffic control system with respect to surveillance are expressed in terms of data accuracy, reliability, and data refresh rate. In order to find the requirements placed on the interrogation scheduling function, we must also know the traffic demand, in terms of the number of targets to be handled, and the variation of this number with position and time. Hard, numerical values for these parameters are not available, hence our discussion will have to be qualitative.

In the simplest case, data updates will be required on every target at nearly regular intervals, determined by the scan time, or data update interval, T. It is likely that this parameter will have different values for enroute and terminal sensors and the possibility exists that DABS will employ a variety of data rates, which vary with environment (enroute, terminal area, final approach), and with target, depending on whether or not the target is currently involved in a conflict, or is listed as the addressee of an urgent message. In this latter case, the data rate would be assigned each target by a central facility, which also assigns targets to sensors, thus controlling the degree of data redundancy in the system. In any event, from the point of view of the scheduler, a list of targets is provided every T seconds, where T is the smallest data update interval employed by the system. Targets being handled at a data rate of 1/2T reports per second will appear on the list every other time, and so on. This list will contain target identity, position, and messages to be transmitted. Positional data will contain enough history to permit rough tracking and prediction through the course of the next T seconds, so that the scheduler can predict range delay and azimuth. As targets enter and leave the operating ranges of various sensors, they appear and disappear from these lists. All that matters is that the scheduler is provided with a list of targets to be handled during the next scan (data update interval), and the important characteristics of that list will be the total number of targets and their distribution in range and azimuth. It is assumed that once each scan a list is formulated and that the scheduler immediately computes the detailed roll call order and schedule of interrogation times for the forthcoming scan according to some fixed procedure, or algorithm.

An individual discrete transmission to an aircraft, together with the aircraft's reply, will be referred to as a "call", whether it is a standard surveillance interrogation or a special transmission whose purpose is to

ensure delivery of an ATC message. It is possible that several calls will be made, per scan, to each target on the list for that scan. The average number of calls per target per scan, denoted by m, is an important parameter in the DABS system in general, and is a basic parameter of interrogation scheduling. The number of calls per scan is closely related to the surveillance requirements for data accuracy and reliability. Multiple calling would increase the probability of detection and correct decoding of information in the down-link message. Also, it is anticipated that some form of monopulse will be used for angle measurement, and the requirements of the monopulse technique may require m to exceed unity. The implications of the use of monopulse are discussed further below. The requirement of range accuracy will affect the interrogation scheduling in two ways. First, it determines the accuracy of the transponder reply delay, and second, it affects the accuracy of prediction of range for each forthcoming scan. Both of these uncertainties contribute to the buffer intervals that must be added to the down-link message length to determine the time interval which must be reserved for a given reply.

Surveillance demand is expressed as the number of targets which must be included in the roll call of a given scan for the purpose of obtaining position updates. If the system features a variable data rate, and if N_1 targets must be handled every T seconds, N_2 targets every 2T seconds, etc. then the average demand per scan will be

$$N = N_1 + N_2/2 + N_3/3 + \ldots . \qquad (3.1)$$

In any case, N will fluctuate from scan to scan, minute to minute, and from place to place. Some number will have to be chosen for each environment (enroute, high-density terminal, medium density terminal, etc.), and imposed as a peak demand on the surveillance system. These bounds

may occasionally be exceeded in practice, but not so often that the resulting saturation of the surveillance system has a limiting effect on safety or traffic flow.

It is desirable to have the surveillance system capable of accommodating a given peak number of targets per scan, irrespective of the distribution of these targets in range and azimuth. Later on we will show that scheduling algorithms exist which can handle arbitrary distributions in range, but the capability of dealing with various distributions in azimuth depends entirely on the nature of the antenna. This question is discussed below, but it may be anticipated here that an important factor in the capacity of a system based on a conventional rotating antenna is the degree of peaking, or bunching, of targets in azimuth.

b. Data-Link Requirements

From the point of view of interrogation scheduling, the data-link requirements will be expressed in terms of the frequency and urgency with which messages must be sent over the channel, and the required probability of correct delivery within some fixed time delay. Message types and the question of message length are discussed below. In order to meet the surveillance requirements, the DABS will provide periodic access to all targets; specifically, some minimum number of calls will be made each scan. The question to be answered regarding data-link requirements is the extent to which these requirements will exceed the capability of this periodic access either in volume or in urgency. We assume that the aircraft cannot independently address the ground, but must rely on normal interrogations for opportunities to communicate. This is so because it is thought that most down-link messages will either consist of routine identification and sensor readout data (e.g., altitude)or of acknowledgements of up-link messages, probably in the form of simple parroting. However, there will be ample opportunity to include pilot-originated messages in the replies to standard surveillance interrogations and one can easily imagine

an arrangement whereby a special bit in the reply is used to request additional calls to accommodate an urgent or extended pilot-initiated message to the ground.

The sources of ground-to-air messages are incompletely understood at present, although it is thought that some form of Intermittent Positive Control (IPC) will be implemented, and that "tactical" ATC commands, associated with a partially automated traffic control system, will be transmitted via the DABS data-link. Messages requiring delivery within some nominal time (currently taken to be 15 seconds) are to be considered "tactical". The probability of successful message delivery on a single call will depend upon the modulation/coding scheme employed and the interference environment in which it will operate. When this probability is known, it can then be determined how many calls are required, for a single message, to meet the given delivery probability requirement. Thus, the message delivery requirements, like monopulse, tend to set a minimum value to the number of calls per scan. If the message remains unacknowledged, or if other urgent messages remain to be transmitted, the scheduler should be able to address more calls to the target without waiting for the next scan, subject to the pointing limitations of the antenna. The interrogation scheduling problem is much simplified if we can convince ourselves that the needs of both monopulse and data-link can be met with a fixed number, m, of calls per scan, and the specific algorithms discussed below will be based on this assumption. It causes no difficulty, of course, if it is known at the start of a scan that a given target should be called 2m or 3m times, to accommodate a longer, urgent message, since the target in that case can simply be included two or three times in the roll call for that scan. The problem arises when readdressing is required because of a failure to deliver a message in the normal course of a surveillance cycle, and the schedule must be dynamically interrupted and rearranged. The issues here are quite complex, and represent a point where the areas of scheduling, surveillance and data-link

requirements, angle measurement, antenna type, and communications link performance all interact closely. One compromise, which might avoid the need for schedule interruption for readdressing purposes, would be the use of an interrogation cycle several times shorter than the surveillance data update interval, for example, by a factor K. The result is a smaller access delay, with normal surveillance interrogations to a given target scheduled every K cycles. This scheme can easily be combined with a variable data rate, as discussed above.

c. Modulation, Coding, and Message Structure

Up to this point we have discussed messages and replies in rather vague terms, with undefined message structure and duration. From a scheduling viewpoint, the important quantity is obviously message duration. The modulation/coding scheme will be the link that connects message content with message duration, and this scheme will be designed chiefly to cope with the interference and multipath effects expected on the channel. The DABS system will probably employ various interrogation modes, depending upon the purpose of the interrogation (surveillance only, surveillance plus IPC command, etc.) and this may be reflected in a variable message duration. This possibility again complicates the scheduling problem, unless all possible up-link messages have durations which are multiples of a basic message, with replies on the down-link which are corresponding multiples of the basic reply. In this case, the use of a longer message is equivalent to inclusion of the target in the roll call more than once, with the several interrogations scheduled in immediate sequence (back-to-back).

Some of the more promising scheduling algorithms work most efficiently when up-link message and reply are of equal length. This choice also seems to be compatible with considerations of message content and modulation capabilities in the two directions on the channel. For definiteness, we assume for the remainder of this discussion that the basic message and reply share a common duration, τ seconds. The possibility of sending messages 2τ

or 3τ in length is tacitly recognized, along with the resultant reduction in overall target capacity.

To get a feeling for the numerical values involved, it may be assumed that message content will fall in the range 25 to 150 bits, and that modulation/coding schemes of practical interest for DABS will likely be capable of delivering in the range one to five bits per microsecond. This allows a wide range of message durations, within which 50 μsec seems, at present, to represent a nominal choice. In this case, each call would occupy a minimum of 100 μsec of channel time for interrogation and reply. It will be necessary to add buffer intervals to these durations to allow time for antenna and transmitter control and also to allow for error in the prediction of range delay. These effects should be accommodated by an allotment of an extra five to ten microseconds, at most, to each interrogation and reply, and we now assume that such allotment is included in the "effective" message duration, still denoted by τ. Many considerations enter into the design of the DABS waveform which are not discussed here, including some, like the problem of false triggering of ATCRBS (and consequent use of a deliberate suppression technique) and the use of redundant coding, which directly affect message duration.

We assume that the DABS transponder will be designed to begin replying after a fixed, brief delay, following receipt of the end of the up-link message. This message ending will be clearly tagged in the waveform, and the reply delay tolerance will be more stringent than that of contemporary ATCRBS, in order to achieve an increased range accuracy in the system. If a longer message is occasionally to be sent, say 3τ in duration, then the additional reply delay will be accounted for by the interrogation scheduler by means of an appropriate increase in apparent target range.

d. Antenna and Monopulse

At present, the candidates for the DABS antenna subsystem range from the conventional fixed-beam rotator to a phased array with essentially

complete beam agility in azimuth. An intermediate case is represented by the rotating planar array, which has beam agility over a limited field of view. The choice, or choices (different antennas may be used in different environments), which are made here will profoundly affect the interrogation scheduling problem, and conversely, the difficulties of achieving high target capacity with a rotator may rule out this antenna option in high-density environments. The one property of the antenna of overriding importance to interrogation scheduling is the degree of simultaneous accessibility of targets at different azimuths. For the phased array, all targets are accessible at all times, while for the rotator targets are accessible only when they lie within some angular distance, $\pm\Theta/2$, of boresight (this "angle of accessibility", Θ, could be different for surveillance and communications purposes). The hybrid antenna, or rotating array, is limited in the same way as the rotator, but the angle Θ will, in general, be larger.

To see how this affects scheduling and capacity, consider a simple surveillance system which uses one call to each target per scan ($m = 1$), with a fixed scan time, or data update interval, T. With a phased array, we can lump all targets into one list, regardless of azimuth, and arrange a roll-call sequence according to one of the algorithms (based on some sort of range-ordering) discussed below. The entire interval, T, can be devoted to this roll call, which generally results in the minimum wasted time, or channel inefficiency. The target capacity of the system is the number of targets which can be handled in time T, and limits only the total number of aircraft in the surveillance area of the sensor, irrespective of their distribution in space (the more interesting roll call scheduling algorithms are characterized by capacity bounds independent of range distribution). The situation with a rotator is quite different, since each target must be addressed during the time interval in which it is illuminated by the rotating beam. A given target is illuminated for a time equal to the time required for the antenna boresight to sweep out the angle Θ. We define the "number of beamwidths", N_b, by the equation,

$$N_b = 2\pi/\Theta, \tag{3.2}$$

and note that each target is accessible for only T/N_b seconds. One way of organizing the scheduling process for the rotator is to divide the azimuth circle into N_b wedges, each measuring Θ in angular width. The targets in the k^{th} wedge are arranged into a roll call sequence in some way, and addressed during the time when the antenna boresight lies within the k^{th} wedge. Thus one roll call, occupying T seconds, is replaced by N_b smaller roll calls, each occupying T/N_b seconds. This change degrades the system performance in two different ways. First, as will be shown later, many algorithms waste a fixed amount of time for every pass through a target roll call, particularly if the targets in the roll span the full operating range of the sensor. This wasted time (or at least part of the wasted time) is multiplied by the factor N_b for the rotator. Since N_b will certainly be larger than 100, this represents a potentially serious loss. The second effect is a more restrictive bound on target capacity for a system using a rotator. The roll call algorithm will provide a limit to the number of targets which can be addressed each T/N_b seconds hence target capacity will be expressed by a bound on the permissible number of targets in any angular wedge of width Θ. If we ignore target range and think of the cumulative histogram of targets as a function of azimuth, from zero to 2π, then with a rotator we have a bound on the amount this histogram can increase over any angular interval, Θ, while the array imposes a bound only on the total target number, or maximum value of the histogram. If we postulate that the system must handle target distributions with a peaking factor of $P(\Theta)$ in azimuth (defined as the ratio of the maximum number of targets in any angular wedge of width Θ to the average number in these wedges), then the rotator must have an inherent capacity, in terms of targets addressed per second, which is $P(\Theta)$ times larger than the required inherent capacity of a phased array handling the same total target load, not even accounting for the effects of channel inefficiency, which also favor the array. Again, the hybrid antenna performs like a rotator, but the larger angle Θ implies a

smaller penalty on capacity due to limited target accessibility.

The requirements of monopulse angle measurement techniques are, in a sense, extensions of the constraints imposed on the scheduler by the degree of target accessibility provided by the antenna. Instead of deriving the requirements of each of a series of specific monopulse schemes, we introduce several general classes of scheduling requirements with the belief that they will be adequate to represent the actual requirements of any of the monopulse techniques which are considered for DABS.

i. The simplest requirement is that the scheduler provide m calls to each target in the roll call, so placed that each call finds the target within a wedge of width Θ, centered on boresight. No other conditions are imposed and it is assumed that, in the interests of scheduling efficiency (see Section 5), these calls will be made in very rapid succession. The reason for the multiple calls in this case is mainly to enhance accuracy and reliability (in the face of interference) by means of redundancy. This requirement might be appropriate to a monopulse scheme which employs rapid nulling during the course of each reply.

ii. A second class requires m calls per scan, all within an angular wedge (width Θ), but spaced out in time by a minimum amount which is long compared to the message duration, τ. The motivation is again to obtain redundancy, this time in an interference environment characterized by a relatively long correlation time.

iii. The third class requires m calls, with no timing restrictions, but with a requirement that the calls find the target at various different positions, relative to boresight, spaced out across the beam. An off- boresight monopulse scheme might fit this case, if it is required to obtain measurements at several points along the "monopulse error curve". For definiteness, we assume that the basic antenna beam (e.g., sum beam), of width Θ, is divided into m equal parts, and the scheduler must provide one call to the target in each of these parts. The requirements of classes ii and iii can be

combined and it can be seen that the rotator, in meeting one of these requirements, automatically meets a form of the other.

iv. The final, and most demanding class, involves the adaptive scheduling and pointing of calls, depending on the results of the angle measurements on preceding calls of the current scan. This would apply to a nulling technique which reacts to each reply, attempting to direct an interrogation and reply very close to the monopulse null (boresight), without changing the beam pointing angle during the course of a reply.

We make only some general comments on these requirements in this section, reserving more detailed discussion to the sections on specific algorithms.

For the phased array, classes i and iii are equivalent, since the array can direct m calls to a target in immediate succession (back-to-back) and with m different pointing angles as easily as directing all the calls to the same azimuth. Class ii can be accommodated by the array, but will inevitably cause an increased channel inefficiency, due to the need to perform repeated roll calls during the course of one data update interval. Case iv also requires spacing out calls in time, but with the additional complication of a variable number of calls per scan.

The rotator can handle case i as easily as the array, by scheduling m calls in immediate succession, but spacing out calls in azimuth can only be achieved by spacing them out in time, with the disadvantages associated with repeated roll calling. Thus, in case iii, we divided the azimuth circle into mN_b wedges, each Θ/m radians in width and arrange each of the resulting mN_b groups of aircraft into order for roll calling. During the time that boresight dwells within each of these smaller wedges, m separate rolls must be called, so that each small wedge is ultimately roll-called m times. The total scan now entails m^2N_b roll-calls, hence the wasted time associated with each roll call is multiplied by a large factor. The rotator is not capable of dealing with requirements of class iv at all.

e. Summary

From the foregoing discussion it appears that interrogation scheduling for a given scan breaks loosely into two parts: (1) The organization of the total target load for each scan into a set of rolls to be separately called, and (2) the formulation of a roll-call sequence (target ordering and interrogation timing) for each roll within the scan. The division of a scan into "sub-scans", implied in part (1), may result from a need to reduce access time by having an interrogation cycle faster than the surveillance cycle, a specific requirement of the monopulse technique employed, or as a simple consequence of the limited accessibility of targets to a rotator.

The scheduling algorithms themselves, which formulate the interrogation sequence for a given roll call, can be characterized in many cases by the following simple equation, which expresses the time, T(N), required to complete a roll call, to the number, N, of targets in the list:

$$T(N) = aN + b \tag{3.3}$$

Since each call necessitates at least 2τ seconds, the wasted time in a given roll call will equal

$$W(N) = T(N) - 2N\tau = (a - 2\tau)\, N + b \tag{3.4}$$

The channel efficiency, η, may be defined as the ratio of the number of targets actually handled in a given time, to the number which could be handled if no time were wasted. It is easily shown that

$$\eta = \frac{2\tau}{a}\left(1 - \frac{b}{T}\right). \tag{3.5}$$

Certain algorithms can be described by these equations with $a = 2\tau$, i.e., they waste only a fixed time for each roll-call, and hence become more

efficient with larger loads.

From Eq. (3. 3) it is easy to see how the division of a target list into sub-lists, which must each be ordered and called, can reduce efficiency. Let (3. 3) express the time required to call the entire list and let us compare this to the result of breaking the list into K equal parts, and calling each part by means of the same algorithm. The total time required then becomes

$$K\ T(N/K) = aN + Kb \quad , \tag{3. 6}$$

and the constant term, b, has been increased by the factor K.

4. ATCRBS INTERROGATION SCHEDULING AND INTERLACING OF ATCRBS AND DABS SCHEDULES

In the DABS system, interrogators will continue to provide surveillance on targets equipped only with transponders meeting specifications essentially identical to the present ATCRBS National Standard. As background to a discussion of this problem, we present a brief review of the ATCRBS system, with emphasis on its scheduling and scanning properties.

An ATCRBS interrogator transmits interrogations at a regular rate, which we denote by F (interrogations per second), while the antenna rotates, completing a scan in T seconds. The basic interrogation is a simple pulse pair, whose spacing determines the "mode" of the interrogation. Different modes request different information in the aircraft's reply: civil identification, military identification, or altitude. All aircraft which receive this interrogation, above a minimum power level, reply with the requested information in a standard format. The reply duration is about 25 microseconds, which is also approximately the duration of the longest interrogation. We denote the maximum range to be processed

in the system by R_o. Since the total propagation delay for a target at maximum range is $2\,R_o/c$, it is clear that the interrogation rate cannot exceed the value

$$F_o \equiv c/2R_o \quad . \tag{4.1}$$

We refer to F_o as the "nominal interrogation rate," since it is matched to the processed range of the system. If F is less than F_o, there will be a "dead time" preceding each interrogation, during which replies (from targets beyond R_o) are not processed. For example, in the present ATCRBS system in terminal areas, interrogation rates in the vicinity of 400 ips (interrogations per second) are used, corresponding to 2500 μsec between interrogations, while the processed range of about 60 nautical miles implies a maximum propagation delay of some 750 μsec (F_o = 1333 ips). This low channel efficiency is forced on the (terminal) ATCRBS system by other constraints, but we shall see below how this long dead time can be used to advantage in a DABS system.

Let the beamwidth over which ATCRBS replies are elicited (roughly the width between 10 dB points of the interrogator antenna pattern) be denoted by Θ, and let

$$N_b = 2\pi/\Theta \quad . \tag{4.2}$$

As before, N_b stands for the number of beamwidths in the azimuth circle. Since a given taget will be illuminated for T/N_b seconds, the number of replies, or "runlength," called N_r, achieved in the system will be given by

$$N_r = F\ T/N_b \quad .$$

This basic result, in the form

$$N_r N_b = FT \quad , \tag{4.3}$$

is called the scanning equation, and represents a basic constraint on the design of any system which provides surveillance on ATCRBS-equipped aircraft. The parameters in this equation are all constrained by far-reaching system considerations. Thus F is limited by F_o and the system range, as mentioned above, and T determines the basic surveillance data refresh rate, which affects tracking and traffic control in a direct way. The parameter N_b, dependent upon beamwidth, determines the angular resolution of the system and, together with N_r, fixes the angular accuracy that can be attained. Detection probability and code readout reliability likewise depend upon N_r. Thus, accuracy, reliability, and resolution performance requirements tend to drive the left side of (4.3) up, while coverage and tracking requirements drive the right side down.

The processing of replies in the ATCRBS system, in particular the method of azimuth measurement by "beamsplitting", is rigidly tied to the rotating character of the antenna scan. Since a DABS interrogator may employ a phased array and use a monopulse technique to determine azimuth on ATCRBS as well as DABS targets, the scanning constraint may not have the form of Eq. (4.3). However, a slightly more general relation holds, of which (4.3) is a special case, and this relation may be obtained by the following line of reasoning. Each interrogation requires a listening interval, $1/F_o$ seconds in length, during which the antenna (array or rotator) must remain pointed in a nearly fixed direction. If the detection and angle-measurement processing (monopulse or beamsplitting) requires N_r replies on each target, and if each of these replies must find the target within $\pm\Theta$ radians of boresight, then the antenna is committed to spending N_r/F_o seconds illuminating a fraction $\Theta/2\pi$, or $1/N_b$, of the azimuth circle. Thus, to accommodate an arbitrary target distribution, all N_b of these azimuth wedges must be processed, requiring a total time, T_a, equal to

$$T_a = N_b \cdot N_r/F_o$$

In the form

$$N_r . N_b = F_o T_a, \qquad (4.4)$$

we regard this as the general "scanning equation" for ATCRBS processing. The sequencing of the ATCRBS interrogations in time and azimuth is arbitrary, so long as each azimuth is included in the beamwidth, Θ, N_r times during each scan. The time devoted to ATCRBS, T_a, may therefore be composed of many non-contiguous intervals, or be a single interval of time. Equation (4.3) is a special case in the sense that the "dead time" need not be included in the useful "ATCRBS time", T_a. The useful time is clearly

$$T_a = \frac{F}{F_o} . T$$

hence

$$FT = F_o T_a,$$

and (4.3) is equivalent to (4.4).

In a DABS system configuration which shares the time of a single beam between a DABS mode and an ATCRBS mode, the total data update interval (or interrogation cycle length), T, will be divided into two parts, T_a and T_d, which represent the respective times devoted to ATCRBS and DABS scheduling. This assumes a complete separation in time, of the scheduling of both interrogations and replies for DABS-equipped and ATCRBS-equipped aircraft. During the time T_a, which is either one block of continuous time or the total of many blocks, ATCRBS interrogations are scheduled in any sequence compatible with the capabilities

of the antenna and the scanning and angular coverage requirements included in Eq. (4.4). For example, in the terminal we could use a rotator with contemporary ATCRBS parameters, as discussed above, and attempt to meet the DABS requirements by scheduling DABS interrogations and replies during the dead time periods which regularly interleave ATCRBS interrogation/reply intervals. A variation would be to send out ATCRBS interrogations at the nominal rate, F_o, (roughly three times the rate now used) in short bursts of, say, M interrogations. Each burst would be followed by a DABS interval which would last M times as long as the present terminal dead time. With a phased array one could complete an entire ATCRBS scan at F_o ips, which would require roughly one-third the present scan time of four seconds, followed by an uninterrupted DABS interval. In these variations, each aircraft would intermittently be subjected to high interrogation rates from individual sensors although the average number of interrogations, over a scan period, would be unchanged. Other variations avoid high local ATCRBS interrogation rates by using the beam agility of an array to spread the interrogations out in azimuth. For instance, the antenna pointing angle could advance by $(2\pi + \delta)/M$ radians with each ATCRBS interrogation, where M is an integer and δ is a small angle. This method carries out a "simultaneous scan" of M sectors, which make up the azimuth circle, in the time it takes to send out $2\pi/\delta$ interrogations. Assuming the nominal rate, F_o, is used, each target sees interrogations at the lower rate, F_o/M.

Thus, much flexibility exists for time-sharing ATCRBS and DABS operation, and there are many possibilities for arranging the ATCRBS portion of the scheduling task. In any case, we are, of course constrained by the "time-sharing equation"

$$\frac{T_a}{T} + \frac{T_d}{T} = 1, \qquad (4.5)$$

in which it is assumed that any wasted time is included in either T_a or T_d. We note, from (4.4) that N_r, the ATCRBS runlength, is proportional to T_a. Let $(N_r)_o$ be the runlength that would result if the full scan-time could be devoted to ATCRBS, using the same values of ® and F_o. Then we have

$$\frac{T_a}{T} = \frac{N_r}{(N_r)_o} \tag{4.6}$$

Recalling Eq. (3.3), let N_d be the number of DABS targets which can be handled by a given algorithm in the available T_d seconds:

$$T_d = a N_d + b.$$

If the same algorithm were employed (with the same resulting efficiency) for the full T seconds, the capacity would be larger, say $(N_d)_o$ targets:

$$T = a(N_d)_o + b$$

Dividing, we obtain

$$\frac{T_d}{T} = \frac{a N_d + b}{a(N_d)_o + b} = \frac{N_d + b/a}{(N_d)_o + b/a} \tag{4.7}$$

When (4.6) and (4.7) are substituted in (4.5) we obtain the interesting result

$$\frac{N_r}{(N_r)_o} + \frac{N_d + b/a}{(N_d)_o + b/a} = 1 \tag{4.8}$$

For large DABS loads and efficient algorithms, the term b/a will be small compared to N_d, and in this case we obtain a very simple trade-off relation

between ATCRBS hits and DABS targets:

$$\frac{N_r}{(N_r)_o} + \frac{N_d}{(N_d)_o} \approx 1 \qquad (4.9)$$

Equation (4.9) is very useful as a starting point in the design of a time-sharing interrogation scheduling algorithm to meet stated requirements in terms of DABS targets and ATCRBS hits. From it we can see how to apportion time between the two modes, and then try various interlace schemes and scheduling algorithms to complete the design. A few examples quickly show how costly, in terms of DABS target capacity, are ATCRBS hits. In fact the cost can be approximately obtained by differentiating (4.9):

$$-\frac{dN_d}{dN_r} = \frac{(N_d)_o}{(N_r)_o} \qquad (4.10)$$

The DABS capacity, $(N_d)_o$, can be expressed in terms of its "target rate", R_d, so that

$$(N_d)_o = R_d T.$$

In other words, the target rate for the algorithm in question when operating over the full scan time, yields the capacity $(N_d)_o$. Combined with (4.4), we obtain

$$-\frac{dN_d}{dN_r} = N_b R_d / F_o \qquad (4.11)$$

The relation expressed here is obvious: each ATCRBS hit takes $1/F_o$ seconds, and an addition of one hit to the runlength adds N_b hits to the total scan. During the total increased ATCRBS time, the DABS algorithm could

have been handling $(N_b/F_o)R_d$ more targets. In the enroute case, with a range of 200 nautical miles, $1/F_o$ is 2.5 milliseconds. The present ATCRBS beamwidth of 4^o corresponds to $N_b = 90$, and a new interrogator might well have a narrower beam, thus N_b/F_o will be of the order of 1/4 second or more. With a nominal message length of 50 μsec, DABS algorithms will have target rates of the order of one thousand targets per second (see Section 5), hence ATCRBS hits may be valued at hundreds of possible DABS targets. This provides strong motivation for implementing monopulse, on ATCRBS, along with other changes in reply processing techniques, in order to reduce the runlength needed for adequate performance.

5. DISCRETE ADDRESS INTERROGATION SCHEDULING FOR THE PHASED ARRAY

We begin the discussion of DABS scheduling algorithms with the relatively simple hypothetical case of a system which uses a single fixed data rate whose data update interval, T, is divided into two intervals, T_d and T_a, devoted to DABS and ATCRBS scheduling respectively. Thus, one long interval, T_d, is available for DABS scheduling each scan, and the scheduler is provided with a target list, including targets at all azimuths and ranges, to be called during each forthcoming scan. Suppose, at first, that m = 1, i.e., only one call per target per scan is required and, as usual, assume that each call consists of an interrogation and reply of equal length, τ.

The ideal scheduler would be capable of filling the interval, T_d, completely with interrogations and replies, given sufficient targets, regardless of their distribution in range. This scheduler would have the maximum possible target capacity, $T_d/2\tau$, which corresponds to a target rate of $1/2\tau$. For a target at range R, the time interval from the end of an interrogation to the start of its reply is $2R/c$ (the fixed transponder delay can be included by a slight increase in R). Since this delay will not usually be very close to a multiple of τ, there will generally be some gaps between successive

messages (interrogations or replies), and this will reduce channel efficiency.

In principle, an interrogation schedule can be devised starting with an arbitrary arrangement of targets in the list. The DABS interval is opened by interrogating the first target on the list, after which the scheduler enters a loop in which the following steps are taken:

1. a list of future time intervals preempted by already-scheduled interrogations is up-dated by the new interrogation,

2. the first available interval of time, following the most recent interrogation, which is larger than τ in length is found,

3. the list of targets not yet called is searched for targets which could be interrogated sometime during the time interval found in step (2), and whose reply would not overlap any expected ones (the target which can be called with the least delay is chosen),

4. if such a target is found, revert to step (1), otherwise the available time interval is not used and step (2) is repeated to find the next available time, and so on.

This algorithm has a theoretical interest only because it does not require a preliminary ordering of the target list but there is no guarantee that an efficient schedule results. The scheme also suffers from "truncation loss", which refers to the time wasted while waiting for replies from the last few targets interrogated, when there are no more targets to fill the available time gap with interrogations.

This scheme works well, however, in the extreme case where all targets are at the same range, R. Targets can be interrogated in any order in groups, $1 + [2R/c\tau]$ in number, when $[x]$ stands for the integer part of x, i.e., the largest integer which does not exceed x. Each group of interrogations is followed, perhaps after a small delay, by the corresponding group of replies. The set of interrogations in such a group, together with

their replies, forms a self-contained "block" in the interrogation schedule, and the scheduling sequence for such a list of targets at constant range breaks up into a number of blocks which can be rearranged at will. Algorithms with this property of breaking up schedules into self-contained blocks will be called "block algorithms", and they are useful because of the flexibility they provide to rearrange blocks and interrupt them, inserting other uses of the channel (such as ATCRBS interrogation cycles) between blocks.

It is easy to schedule interrogations so that they do not interfere with other interrogations or the replies not yet received from previous interrogations, hence the only real problem is to guarantee that replies will not overlap each other. It is obvious that the replies from two targets at the same range cannot overlap, since the corresponding interrogations do not overlap, and it is also clear that the replies from an arbitrary pair will be prevented from overlapping if the closer target is interrogated first. Thus, if targets are first arranged in order of increasing range, the simple scheme of addressing the next target on the list as soon as the channel is free will always work. With this scheme there need never be a gap preceding an interrogation, or a gap longer than τ before a reply, hence the channel efficiency exceeds $2/3$. The only disadvantages of this algorithm are (1) the truncation loss suffered at the end of the list and (2) the inefficiency which results if the schedule must be interrupted. A strict increasing-range-ordered (IRO) algorithm does not usually result in a schedule made up of blocks, so that interruptions cannot easily be admitted without additional truncation loss. The truncation loss which occurs at the end of the target list can seriously reduce the efficiency of an otherwise acceptable algorithm, and its dependence on the detailed distribution of targets in range, makes it difficult to predict the total time required to handle calls to a given number of targets, or to compute the number of calls that can be made in a given interval of time.

We have noted that the replies to a given pair of interrogations cannot overlap if the range of the second target is not less than that of the first.

However, if the second target's range is less than that of the first by at least the distance $c\tau$, then the two targets may be addressed in immediate sequence (back-to-back interrogations) without reply overlap. This is the basis of a special class of decreasing-range-ordered algorithms, one of which is discussed below.

We now give a description of several specific and practical algorithms, all based on some kind of preliminary range-ordering, and then discuss the implications of a requirement for multiple calling (monopulse) and for ATCRBS/DABS interlacing.

a. The Fixed-Ring Algorithm

We have observed the convenience and efficiency that would result if all targets were found at the same slant range. This provides the motivation for the "fixed-ring algorithm", in which the coverage circle of the sensor is divided up into annular rings by a series of concentric circles. If the rings are relatively narrow and the targets in each ring are addressed as a group, a fairly simple and efficient block algorithm results. Consider a ring whose inner radius is R, and whose outer radius is $R + \Delta R$. If we begin a block of calls with targets in this ring, we have a time interval of length $\tau + (2R/c)$ available for interrogations before any reply can arrive. This time will accommodate $1 + [2R/c\tau]$ interrogations, hence it is desirable to choose R to be a multiple of $c\tau/2$. If this interrogation period is immediately followed by a listening period $\tau + (2/c)(R + \Delta R)$ in duration, all replies will automatically be received. The interrogations within a block must be increasing-range-ordered to prevent reply garble, but the targets may otherwise be chosen arbitrarily from those available in the ring. Let

$$n \equiv 1 + 2R/c\tau \qquad (5.1)$$

be the "calling capacity" of the ring, assumed to be an integer, and let the time required for a single block of calls to this ring be S. Then,

$$S = \frac{2R}{c} + \frac{2(R + \Delta R)}{c} + 2\tau$$

$$= 4R/c + 2\tau + 2\Delta R/c$$

$$= 2\tau n + \frac{2\Delta R}{c} \qquad (5.2)$$

When (5.2) is compared to the general relation, (3.3), we see that $a = 2\tau$, the minimum possible time per call, and the wasted time for the block is

$$b = W(n) = 2\Delta R/c \qquad (5.3)$$

This wasted time is fixed by the ring structure, since no attempt is made in this algorithm to tailor the listening period to the actual set of reply delays expected for a given block. According to (3.5), the efficiency, for the duration of a block is

$$\eta = 1 - \frac{b}{S} = \left(1 + \frac{\Delta R}{R}\right)^{-1} \qquad (5.4)$$

As expected, the efficiency is high if ΔR is small compared to R.

In order to make up a complete scheduling algorithm, based on this idea, the total target list is broken up into separate lists, containing the targets in each of the fixed rings. A given ring is then called, using as many blocks as necessary to exhaust the list for that ring. The blocks of calls may be rearranged in any way, and interrupted without incurring truncation loss. Each time a ring is called, a time given by (5.3) is wasted, so long as the ring is full. In general, the final block devoted to a ring will not find the ring fully occupied, so an addition "quantization loss" is incurred. This loss can be reduced in two ways: 1) The scheduler can arrange to save,

for the last block of calls to a ring, those targets at minimum range. If the listening interval is curtailed, so that the scheduler waits only for those replies expected, a considerable time may be saved. 2) The ring width can be increased, at some sacrifice of block efficiency, in order to increase the number of targets expected in the ring, since this can reduce the quantization losses.

To get some idea of how many times a ring may be called, we note that the area of the ring bounded by R and R + ΔR is

$$A = \pi(R + \Delta R)^2 - \pi R^2$$
$$= 2\pi \Delta R \left(R + \frac{\Delta R}{2}\right) \qquad (5.5)$$

Since the calling capacity is proportional to R, the target density, D, required to fill the ring to capacity (for one block) is nearly independent of R; it is given by

$$D = n/A = \left\{\pi c\tau \ \Delta R \ \frac{1 + \Delta R/2R}{1 + c\tau/2R}\right\}^{-1} \qquad (5.6)$$

targets per unit area. Thus, a fixed-ring algorithm, with rings of equal width is nearly matched to a uniform density of targets, in the sense that a uniform target distribution would result in an equal number of scheduling blocks being devoted to each ring. More precisely, the algorithm is "matched" to the distribution of targets in range which corresponds to uniform areal density (target density linearly proportional to range). Since rings can be repeatedly called, the algorithm adapts to any range distribution of targets, and the only significance which attaches to the match to a uniform distribution is the implication that all rings will be called with roughly the same frequency over long periods of time.

The advantage of the algorithm is its simplicity and block schedule

character, which lends itself to interruption, a point discussed below. The disadvantage is its modest efficiency due to quantization losses and the repeated wasting of time with each block devoted to a ring. To get an estimate of the overall efficiency of the algorithm, suppose that N targets are placed within the coverage circle (of radius R_o) in such a way that the target density in range is a linear function of range (as would be the case if the areal distribution were uniform). Then the number of targets in a ring bounded by R and R + ΔR is approximately

$$\frac{2NR\Delta R}{R_o^2} , \qquad (5.7)$$

hence this ring will have to be called K times, where K is given (roughly) by (5.7), divided by (5.1):

$$K = \frac{1}{n} \frac{2NR\Delta R}{R_o^2} = \frac{Nc\tau\Delta R}{R_o^2} \cdot \frac{2R}{2R + c\tau} \approx \frac{Nc\tau\Delta R}{R_o^2}$$

The wasted time for one call to this ring is given by (5.3), and the total time wasted if each ring is called once is $2R_o/c$, since the sum of the ring widths is R_o, regardless of the actual choice of rings. If each ring is called K times, the resultant wasted time is given by

$$K \cdot \frac{2R_o}{c} \approx 2N\tau \frac{\Delta R}{R_o} \qquad (5.8)$$

In addition to this wasted time, due to the fundamental inefficiency of the algorithm with full rings, we must allow for the quantization loss. In the worst case, each ring would be called once (i.e. a schedule block devoted to the ring), with only one target to be addressed. We would then lose nearly the full time, (5.2), required for the block, or roughly $4R/c$ seconds, assuming no curtailment of the listening period. If the rings all have nearly

the same width, ΔR, then they will be $R_o/\Delta R$ in number, and the total worst-case time loss for quantization is roughly

$$\frac{R_o}{\Delta R}\,\frac{4(R_o/2)}{c} = \frac{2R_o^2}{c\Delta R}$$

The actual quantization loss may be taken equal to this amount, multiplied by a factor, f, which lies between zero and unity, and probably averages near one-half hence the total time required by the algorithm is approximately given by

$$T = 2N\tau\left(1 + \frac{\Delta R}{R_o}\right) + 2f\,\frac{R_o^2}{c\Delta R} \qquad (5.9)$$

To obtain this expression, we have added the estimates for the two sources of wasted time to the total "useful time", $2N\tau$. We note that (5.9) has the general form given previously as (3.3), with

$$a = \left(1 + \frac{\Delta R}{R_o}\right) 2\tau$$

$$b = 2f\,\frac{R_o^2}{c\Delta R} \qquad (5.10)$$

Equation (5.9) implies that the ring width, ΔR, can be optimized, to minimize T by striking a balance between the two loss terms, as suggested previously. The optimum value, however, depends on the target load:

$$(\Delta R)_{opt} \approx R_o\left(\frac{fR_o}{NcT}\right)^{1/2} \qquad (5.11)$$

Since N is roughly equal to $T/2\tau$, we can express the optimum width in terms of the available time:

$$(\Delta R)_{opt} \approx R_o\left(f\,\frac{2R_o}{cT}\right)^{1/2} \qquad (5.12)$$

(note that $2R_o/c$ is the pulse repetition period of a radar whose maximum unambiguous range is R_o). Formulas (5.11) and (5.12) are not to be taken too seriously in the design of a fixed-ring algorithm, since a special class

of target distributions was assumed in their derivation. They do, however, indicate the existence of a design tradeoff and identify the parameters involved.

b. The Full-Ring Algorithm

The fixed-ring algorithm attains its greatest efficiency for the special case in which each ring is full and need be interrogated only once. In this case the wasted time for each ring is incurred only once, and there is no quantization loss. It is natural to inquire if the ring geometry can be adapted to an arbitrary range distribution to bring about this condition, and, if so, whether the resulting algorithm (where the rings change with every interrogation cycle) is practical. It turns out this can easily be done with an elegant algorithm, due to Russell Johnson, which we call the full-ring algorithm.

The first step in this algorithm is to determine the rings, or groups of targets which will be called in each block. Then the targets are increasing range-ordered within each ring, and the interrogation sequence of the rings determined, as in the fixed-ring algorithm. The rings are found by working inward from the last ring, whose outer radius is the operating range of the sensor. The inner radius of this outer ring is chosen so the calling capacity of the ring just equals the number of targets whose range exceeds this radius. Taking the radius found in this way as the outer radius of the next largest ring, and considering only targets not yet assigned to a ring, the process continues inward to the center. There is no truncation loss in this scheme, as there would have been had we started at the center and worked outward to define the rings.

In order to evaluate the efficiency of this algorithm, we must consider the procedure in a little more detail, and in the process we will obtain a complete and precise specification of the algorithm. Let the given target list be characterized by a cumulative histogram, H(R), which denotes the number of aircraft on the list whose range exceeds R. The list includes only targets within the range of the sensor, hence we have $H(R_o) = 0$ and $H(0) = N$, the total number of targets. We define the function

$$L(R) = 1 + (2R/c\tau) \tag{5.13}$$

and note that $[L(R)]$ is the calling capacity of a ring whose inner radius is R. Since $L(R)$ increases while $H(R)$ decreases to zero with increasing range, the curves must cross at some value of R, say R_1, and a ring bounded by R_1 and R_o will accommodate all the targets whose range exceeds R_1, which are $H(R_1)$ in number. We have to be careful because of the step character of $H(R)$, and distinguish between two cases: 1) $L(R)$ intercepts $H(R)$ on a horizontal portion of the histogram, and 2) $L(R)$ intercepts $H(R)$ on a vertical portion. In case 1), $L(R_1) = H(R_1)$ is an integer, and the ring exactly holds $L(R_1)$ targets, i.e. the interrogation portion of the scheduling block devoted to this ring is completely filled with interrogations. On the other hand, if R_1 equals the range of a particular target, then $L(R)$ will cross a vertical portion of $H(R)$ and

$$H(R_1) < L(R_1) < H(R_1-)$$

Thus, in case 2), $L(R_1)$ is not an integer, and the ring will not accommodate the target at R_1. It is convenient to choose a ring radius, R_2, so that $L(R_2)$ is integral, and we define R_2 in this case so that

$$L(R_2) = [L(R_1)] \tag{5.14}$$

The details are understood more easily in graphical form, and Figure (5.1) illustrates case 1), while Figure (5.2) illustrates case 2).

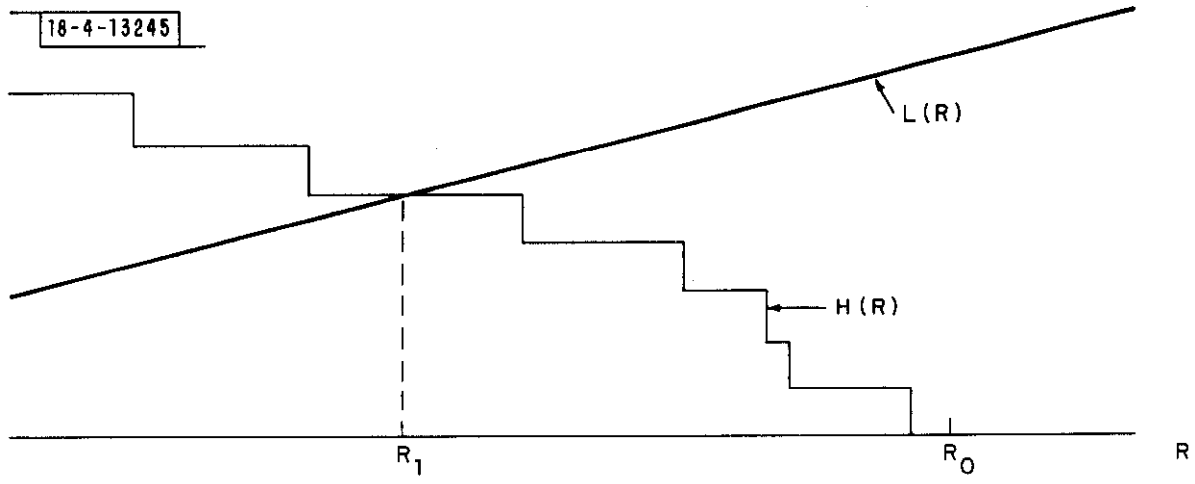


Fig. 5.1

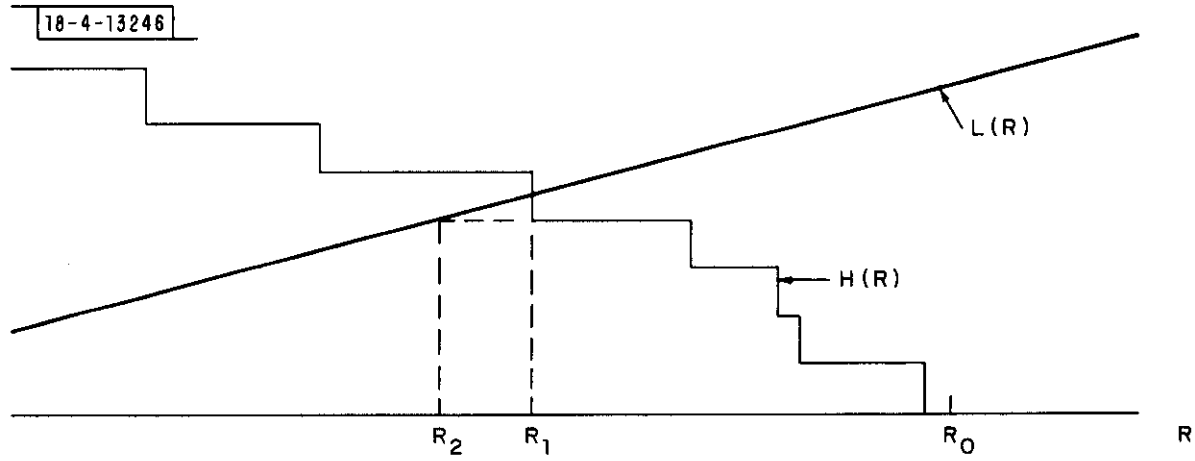


Fig. 5.2

The process continues by deleting the targets assigned to the outer ring from the list and forming a new histogram. The new histogram, H' (R), denotes the number of targets whose range exceeds R but does not exceed R_1, hence

$$H'(R) = H(R) - H(R_1) \tag{5.15}$$

where R now ranges from zero to R_1. The intersection of H' (R) with L (R) defines the next outermost ring, and so on.

In all cases the inner ring radius is chosen so that L (R) is integral, and no time is wasted in the interrogation period. Any lost time between the last interrogation and the first reply is ascribed to the listening period (an arbitrary choice). The listening period is terminated at the end of the last scheduled reply, so that no time is wasted "hearing out" the remaining empty portion of the ring.

It should be noted that a situation can arise in which there are several targets beyond range R_2 which must be called in the following ring. This is illustrated in Figure (5. 3), in which case four targets must await the next ring for interrogation.

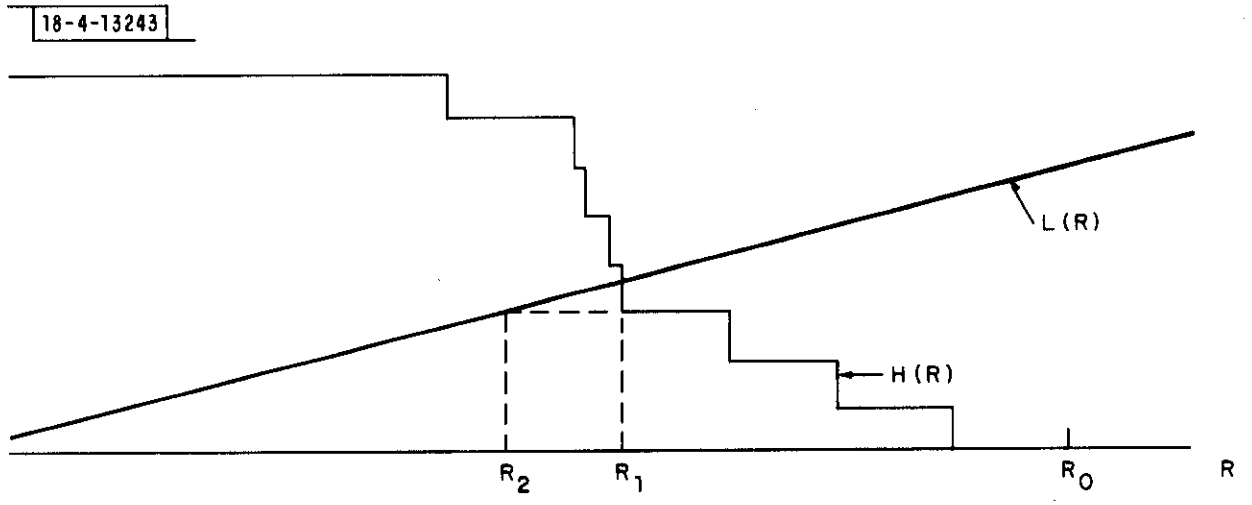


Fig. 5.3

In this algorithm the rings can actually overlap since the largest ring starts at R_2, while the next largest ring extends to R_1. This should cause no confusion since the definition of ring radii is in any case arbitrary, and the function of the algorithm is to break the target list into separate blocks for interrogation.

The efficiency of the algorithm is very high, since neither truncation loss nor quantization loss is incurred, and rings are not called repeatedly. By our definitions, all wasted time occurs during the listening period, due to the range increments, δR, between targets. This increment between two targets causes a lost time $2\delta R/c$. However no time is lost after the last target, and in case 1), shown in Figure (5.1), only a portion of the range increment preceding the first target causes wasted time. In case 2) (Figures (5.2) and (5.3)), the full range increment preceding the first target causes loss, and an additional loss, equal to $(2/c)\ (R_1 - R_2)$ occurs. Since

$$L\ (R_1)\ -\ L\ (R_2)\ <\ 1\ ,$$

it is easily shown that the extra time loss is bounded by

$$(2/c)\ (R_1 - R_2)\ <\ \tau \tag{5.16}$$

Indeed, this must be so or we could have squeezed in one more interrogation. Therefore the total wasted time is bounded by the sum of the range-gap losses, which adds up to $2\ R_o/c$, regardless of the target number and distribution, and the sum of the extra loss, each bounded by (5.16). The latter loss will not exceed $M\ \tau$, where M is the number of rings, hence the total wasted time, W(n), is bounded by

$$W(N)\ \leq\ (2\,R_o/c)\ +\ M\,\tau \tag{5.17}$$

We can obtain a bound on M by returning to the model used for the fixed-ring algorithm, namely N targets with a linear distribution in range. For this case, the range histogram is approximated by

$$H\ (R)\ =\ N\ \left(1\ -\ \frac{R^2}{R_o^2}\right) \tag{5.18}$$

and the outermost ring has a width, ΔR, approximately equal to

$$\Delta R = \frac{R_o^2}{Nc\tau} \tag{5.19}$$

The rings gradually widen towards the center, hence they are fewer than $(R_o/\Delta R)$ in number:

$$M \leq \frac{Nc\tau}{R_o} \tag{5.20}$$

We would obtain (5.20) as an equality if we put N equal to the product of the number of rings, M, and the calling capacity of the ring halfway out (i.e. whose inner radius is $R_o/2$). For this target distribution, then, we obtain the bound

$$\begin{aligned} T(N) &\leq 2N\tau + (2R_o/c) + Nc\tau^2/R_o \\ &= 2N\tau\left(1 + \frac{\tau}{t_o}\right) + t_o , \end{aligned} \tag{5.21}$$

where t_o is the natural radar pulse repetition period:

$$t_o = 2R_o/c \tag{5.22}$$

Equation (5.21) represents a high efficiency, since t_o will be very small compared to the available scan time (or interrogation cycle time), and yet long compared to the message length, τ.

c. A Decreasing-Range-Ordered Algorithm

We have pointed out that a series of targets can be addressed in order of decreasing range, if the interrogations take place in immediate sequence and each target is closer to the sensor, by at least the amount $c\tau$, than its predecessor. A sequence of this type terminates automatically,

forming a single block of scheduling. In order to schedule an arbitrary target list, one begins a block with a target near maximum range. A second target must be found, at least $c\tau$ closer, and so on, ending with a close-in target. Each block lasts $t_o + 2\tau$ seconds (or less, if the first target is closer than R_o), and channel efficiency depends entirely on the availability of targets at or near the range separation $c\tau$. The efficiency of the blocks will clearly decrease as the end of the target list appears, since less choice will be available, and no analytic expression can be derived for the resulting channel efficiency. It is clear that the algorithm will be most efficient with a target distribution whose density is uniform in range, which corresponds to an inverse R-law clustering around the origin in two dimensions.

A perfectly filled block with this algorithm would begin with a target at range R_o, and contain targets at ranges $R_o - c\tau$, $R_o - 2c\tau$, etc.; altogether $1 + [R_o/c\tau]$ in number. If $R_o/c\tau$ is not an integer, a small gap will appear in the center of the block. An interesting feature of such a block is the fact that all the aircraft addressed will start to reply at or close to the same instant. If the block is less perfectly filled (i.e. the range increments between targets exceed $c\tau$), the interrogations can be delayed slightly to maintain the symmetry of interrogations and replies about the midpoint of the block, thus preserving the simultaneity of reply of the set of aircraft addressed. It has been suggested by T. Amlie, of the FAA, that this scheme be employed, with interrogation cycles synchronized among sensors, in order to minimize interference between sensors (and also to provide a basis for air-derived CAS and PWI). We cannot comment here on the viability of this system concept in the large, but the basic interrogation scheduling algorithm appears to be inefficient and difficult to analyze.

A simple-minded method of implementing a decreasing-range-ordered algorithm is to fix a series of range rings, all ΔR in width, where

$$\Delta R = c\tau/K$$

and where K is an integer. A set of these rings is arranged with their outer radii in the following order:

$$R_o,\ R_o - (K+1)\,\Delta R,\ R_o - 2(K+1)\,\Delta R,\ R_o - 3(K+1)\,\Delta R, \ldots \text{etc.}$$

Targets are now addressed by choosing one from each ring, in sequence, until this list of rings is exhausted. A single block will result, in which all successive range differences exceed $c\tau$, but not $c\tau + \Delta R$. The next block proceeds through another set of rings, arranged as follows:

$$R_o - \Delta R,\ R_o - (K+2)\,\Delta R,\ R_o - (2K+3)\,\Delta R,\ R_o - (3K+4)\,\Delta R, \ldots$$

After K + 1 such blocks have been called, each of the original rings will have been cleared of one target, and the process repeats. As rings are exhausted, gaps appear in the schedule, in a way dependent on the given target distribution in range. The time required to make one pass through all the rings, denoted by S, is given by

$$S = (K+1)\,(t_o + 2\tau)\ ,$$

and the total time required equals S times the number of targets in the most fully filled ring.

We can get an idea of the capacity of this algorithm by assuming the most favorable range distribution, i.e. uniform in range. If there are N targets all together, the number in any ring is

$$\frac{N\,\Delta R}{R_o}$$

Therefore the total schedule time is

$$T(N) = \frac{N \Delta R}{R_o} S$$

$$= \frac{Nc\tau}{KR_o} (K + 1) (t_o + 2\tau)$$

$$= 2N\tau \cdot \frac{K+1}{K} \cdot (1 + \frac{2\tau}{t_o})$$

The chief source of inefficiency is the factor $(K+1)/K$, which results from the finite width, ΔR, devoted to each target. If K is large, the rings are narrow and they must still be nearly equally occupied to avoid wasted time. A distribution uniform in two dimensions puts $2N\Delta R/R_o$ targets in the outer ring, thus reducing efficiency by a factor of two. These low efficiencies are due to the rigidity of the algorithm and its demands on the range distribution of the targets, but it is hard to see how a scheme based on decreasing-range ordering can be made much more flexible and adaptable, so as to keep reasonable efficiency for all target distributions. By contrast, the fixed-ring and full-ring algorithms adapt easily to any range distribution and, although our capacity evaluations were made for a special distribution, the efficiency should not be much different for any other distribution with either of these algorithms.

Many variations of these three basic schemes are possible and, no doubt, other classes of algorithms can be devised. Our analysis of the scheduling problem is not sufficiently deep to permit us to claim that our algorithms exhaust the basic possibilities. It is likely, however, that these examples illustrate most of the problems involved in scheduling, and on this basis we proceed to discuss the complications introduced by 1) monopulse, 2) variable data rate and/or rapid access interrogation cycling, 3) a requirement to interrupt the surveillance schedule for urgent message delivery, and 4) the needs of ATCRBS/DABS interlacing. These complications will be further specified as we proceed (to the extent that they have not been already defined), and the problems they cause will be discussed in the context of

phased-array scheduling using any of the three specific algorithms just treated, as well as the general increasing-range-ordered (IRO) algorithm mentioned at the beginning of this chapter.

1) Monopulse

In our discussion of monopulse in Chapter 3 we introduced four classes of requirements:

i. m calls per scan, no restrictions;
ii. m calls per scan, spaced in time by some minimum amount, say τ_m;
iii. m calls per scan, spaced in angle relative to boresight so that one call falls in each of m equal parts of the beam; and
iv. variable number of calls per scan, using a sequential technique in which the need for another call (and possibly the corresponding antenna pointing angle) depends on the processed results of previous calls and their replies in the current scan.

As already mentioned, the beam agility of the array makes cases i and iii equivalent, and the multiple calls can be provided for by entering each target on the roll call m times, since in these two cases the timing of the calls is unimportant. With the IRO algorithm, or either of the ring algorithms, this will usually result in m consecutive interrogations, back-to-back, to each target, and the efficiencies of these algorithms should not be affected, to first order, by the fact that the effective histogram of the target range distribution now increases in steps of m, instead of in unit steps. The expressions (5.9) and (5.21), for time required to handle N targets, should remain valid with N replaced by mN in the right-hand members. In a decreasing-range-ordered (DRO) algorithm, the mutliple interrogations to a given target cannot be back-to-back, but the same effect can be achieved by repeating each DRO scheduling block m times. This separates the interrogations in

time by at least t_o (see (5.22)), and thus case ii can easily be handled by separating the repeated blocks as much as need be. The inefficiency of each block is then also repeated, so that the required time to handle N targets is directly increased by the factor m.

Case ii can, in a sense, be met by the ring algorithms in the same way as with the DRO schemes, by repeating blocks of scheduling m times. If the blocks are repeated back-to-back,a time separation is achieved which is larger for distant rings, in approximate proportion to ring radius. This may or may not be an acceptable solution, but it is reasonable in the sense that angle rates also decrease with range, so that the longer intervals between measurements on distant targets will not cause trouble on this account. Multiple calls to close-in rings can be spaced out, if necessary to attain the minimum spacing, τ_m. As with the DRO scheme in cases i and iii, the scheduling time for a ring algorithm meeting case ii is increased by a factor of m. The IRO algorithm is seriously compromised by the case ii monopulse requirement unless τ_m can be relatively large. For example, if the DABS scan time, T_d is divided into m parts, and the entire target list is scheduled, by means of the IRO algorithm, during each part, then case ii is met with $\tau_m = T_d/m$ and little loss in efficiency. However, if τ_m must be much smaller, then it is apparently necessary to schedule the scan in T_d/τ_m pieces, each target appearing in m pieces. If the IRO algorithm obeys a time-capacity relation like (3.3), then the fragmentation of the total target load into many lists will lead to inefficiency (see (3.6) and related discussion).

Case iv is awkward for all the algorithms and is probably prohibitively costly, in terms of channel efficiency, unless the variable call number is actually a fixed number for all targets occasionally supplemented by an extra call or two to a few aircraft to resolve an angle measurement problem. The extra call can be handled the same way as an urgent message interrupt, and will be discussed below.

2) Variable Data Rate and/or Rapid Access Interrogation Cycling

Rapid access cycling refers to the scheme, mentioned in Chapter 3, of breaking the DABS scan time, T_d, into a small number of cycles and normally scheduling different portions of the target list in each cycle. Targets can, however, be addressed in more than one cycle if necessary to insure message delivery. It may prove desirable to further divide each cycle into two sub-cycles, so that the scheduling computation can be performed a half-cycle in advance. Rapid access can be combined in a natural way with variable data rate scheduling.

An example is shown in Figure 5.4, in which each DABS period is broken into six cycles.

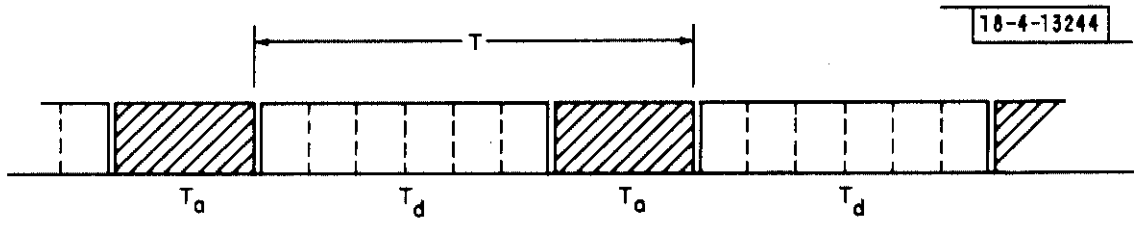


Fig. 5.4

The basic data update interval (unchangeable for ATCRBS) is T, divided into two long intervals: T_a for an entire ATCRBS scan, and T_d (subdivided into six cycles) for a nominal DABS scan. Data update intervals on DABS targets may be any multiple of T, or submultiples up to one-sixth, although the higher-than-nominal data rates do not provide periodic position reporting (not necessarily a disadvantage). Communications access is roughly $T_d/6$ seconds.

Each cycle is treated as a complete scheduling problem, with small loss in efficiency for the IRO and ring algorithms. The effect on a DRO algorithm is hard to assess, although efficiency will certainly go down, since the schedule is "running out of targets" at the end of each list, and this usually is the chief cause of wasted time in the DRO case.

3) Schedule Interrupt

If the normal communications access fails to deliver an urgent message and the access delay until the next opportunity is too large, an interrupt may be necessary. As mentioned, a sequential monopulse technique could also lead to a requirement for schedule interruption. We assume that interrupts are uncommon enough that they must be treated individually, although the modifications are fairly obvious if they can be delayed slightly and handled in groups.

Any block algorithm can easily be interrupted between blocks and the fixed block length, t_o, of the DRO algorithm guarantees channel availability in a fixed time. In ring algorithms the blocks are of variable length, but in no case longer than $2t_o$, the duration of a thin ring at maximum range. A delay as small as this should be acceptable, hence the block itself need not be broken. If a simple call to one target is scheduled during the interrupt, the propagation time is wasted (less is wasted if multiple contiguous interrogations are used). However, an alert schedulercould probably squeeze in a short block of scheduling during the waiting time.

The straight IRO algorithm can also be interrupted, at a cost in channel efficiency which depends on the complexity one is willing to devote to handling this problem. The simplest and least efficient method is to cease interrogation when the interrupt occurs, wait out the propagation delay for this last call, insert the special call (wasting its propagation delay), and then resuming IRO scheduling on the remaining targets. The best, and most complex way would be to cease interrogation and then search for the earliest moment at which the interrupting call can be scheduled so that neither its interrogation nor its reply will interfere with replies yet to be received from targets interrogated before the interrupt. One could resume IRO scheduling as soon as it is clear that the next regular target reply will not interfere with the reply from the interrupting target. As long as interrupts are relatively infrequent, this procedure could be implemented without much difficulty. The logic required is similar to that involved in the first algorithm described in

this chapter, for scheduling an arbitrary target list without preliminary ordering.

It would probably be a good compromise to design the DABS interrogation scheduling program with an interrupt capability, but to assure, through choice of system parameters and modulation/coding technique, that interrupts will be infrequent.

4) ATCRBS/DABS Interlacing

The general principles of ATCRBS/DABS interlacing were discussed in Chapter 4, where we have seen that the portion of the basic data update interval devoted to DABS scheduling may be available in one block (this has been assumed so far in the present chapter) or as the sum of many disjoint blocks of time. In the extreme case, DABS scheduling is performed during the dead time periods between ATCRBS interrogations, in which case the DABS time is broken into as many pieces as there are ATCRBS interrogations per scan. It seems clear that the choice of ATCRBS/DABS interlace scheme will be affected by the complications thereby introduced into the DABS scheduling problem.

If DABS time is broken into short blocks, then the IRO algorithm can be used only if the resulting truncation losses are acceptable. On the other hand, the DRO algorithms, having blocks of fixed length, are readily adaptable to rapid interlacing, and the minimum block time required is simply t_o, which is also the minimum ATCRBS pulse repetition period. If longer intervals are available for DABS, several DRO blocks can be grouped together in a predetermined way.

For the ring algorithms, the problem is more complicated since the rings require varying lengths of time (load-dependent in the full-ring case) and they may be called (in the fixed-ring case) a load-dependent number of times. Thus, a secondary scheduling problem arises, to fit the total set of rings (including multiple ring calls) into the available fixed blocks of time. This problem could cause the loss of considerable channel time to these otherwise efficient algorithms. In the fixed-ring case, with rings of

roughly equal width, the times required for the successive rings increase nearly linearly with ring inner radius. Thus the combinations first-plus-last, second-plus-next-to-last, etc., require equal time. Thus, by some suitable pairing of rings, one can achieve a fixed block length, at a cost in efficiency which results from the inflexible calling of rings in pairs. An extreme case would have a rigid schedule for calling the rings in a fixed-ring scheme. Each ring would be called the same number of times per scan, and the problem of fitting ring blocks into the DABS/ATCRBS interlace pattern would be solved once and for all. The resulting scheme is inflexible, like the DRO schemes, and would prove inefficient unless the target range distribution were nearly linear in range.

This problem of secondary scheduling has not been given much study, but it could be a crucial area in terms of overall channel efficiency. If the time blocks available for DABS are several times longer than 2 t_o (the time required for the farthest ring), then it should not prove too difficult to find algorithms for efficient secondary scheduling for either of the ring algorithms.

6. DISCRETE ADDRESS INTERROGATION SCHEDULING FOR THE ROTATOR

The basic constraint imposed on the interrogation scheduling problem by the use of a rotator is the need to address targets during the fraction of a scan in which they are accessible to the antenna. In effect, this means that the total target list must be reorganized into a number of smaller lists, and this aspect of the problem was discussed in some detail in Chapter 3.

As in Chapter 5, we begin with the simplest case, in which a single block of time, of length T_d, is available for the DABS scan, and only one call per target per scan is required. With a rotator, this situation could occur in a back-to-back antenna configuration with one antenna dedicated to DABS targets only, while the single-hit monopulse capability is at least conceivable if multiple receiver beams are formed.

In ATCRBS, the target run on each target is automatically centered on the target azimuth (barring drop-outs), and it would be desirable to direct

the single DABS call to each target as that target passed the nominal antenna boresight direction. Except for very sparse target distributions, this will, of course, be impossible, hence we will in any case have to settle for calling each target when it lies within an angular bound, $\pm \Theta/2$, of boresight. The logical way to schedule interrogations is then the one described in Chapter 3; the azimuth circle is divided into $N_b = 2\pi/\Theta$ wedges and the target list is broken into N_b corresponding sublists. Each sublist is then scheduled during the time, T_d/N_b, when the antenna dwells in the associated angular wedge.

The scheduling of each sublist can be accomplished with any of the algorithms introduced in Chapter 5, with the penalty that, in general, the fixed component of wasted time is then magnified by the factor N_b (see the discussion at the end of Chapter 3). It is likely that Θ will be of the order of one degree, so that N_b might range from roughly 100 to 500, and hence the increase in wasted time can be considerable. In addition to this possibility of degraded efficiency, the rotator imposes a capacity limit on the number of targets in any wedge of angular width Θ, rather than on the total number of targets, as pointed out earlier.

We turn now to a brief discussion of the effects of the four complications discussed in Chapter 5, this time in the context of scheduling for the rotator.

1) Monopulse

The monopulse requirement in case i (see Chapter 3) can easily be met by scheduling m consecutive interrogations, as with the phased array. The target capacity is reduced by the factor m, but efficiency should not be seriously affected. Case iv is really not practical for a rotator and case ii can be implemented only if the time interval τ_m, demanded between calls is short enough to allow the total string of m calls to fall within the dwell time of the antenna beam. If this limitation is met, case ii effectively reduces to case iii, in which the m calls must be spaced across the beam.

To handle case iii, the basic beamwidth, Θ, is divided into m equal parts and the azimuth circle is divided into mN_b wedges. The target list is now decomposed into mN_b sublists, corresponding to the azimuth wedges. The targets on a given sublist must be addressed once while they lie within the wedge $[-\Theta/2m, +\Theta/2m]$ about boresight, again in each of the wedges $[-3\Theta/2m, -\Theta/2m]$, $[+\Theta/2m, +3\Theta/2m]$, and so on, m times in all. The scheduling can be accomplished as follows. Each of the mN_b target sublists is organized and scheduled for interrogation using some range-ordered algorithm, but only T_d/m^2N_b seconds is allocated to this list. During the T_d/mN_b seconds that the antenna beam dwells in one of the small wedges, the target sublists for this wedge and its (m - 1) nearest neighbors must be called. The schedule for a given wedge is used repeatedly until the wedge has been called m times.

The repeated roll-calling will certainly cause scheduling inefficiency, and a very rough idea of the cost in capacity can be obtained by the following argument. Suppose that the scheduling algorithm used (e.g. the full-ring algorithm) can handle N targets in a time, T, given by (3.3):

$$T = a N + b \ , \tag{3.3}$$

and that this relation holds for large and small values of the available time, T, and is independent of the distribution of targets in range. These assumptions oversimplify the situation, but they allow us to make our point in a direct way. If the full time, T_d, were used with a phased array and the assumed algorithm, the target capacity would simply be

$$N = \frac{T_d - b}{a} \tag{6.1}$$

If the algorithm is used with a rotator, having a beamwidth of $2\pi/N_b$ radians, and a single hit monopulse capability (m = 1), then we can apply (3.3) to each of the N_b target sublists:

$$\frac{T_d}{N_b} = a \frac{N}{N_b} + b$$

The resulting total capacity is

$$N = \frac{T_d - N_b \cdot b}{a} \quad , \tag{6.2}$$

although the effective capacity bound limits the number of targets to N/N_b (with N given by (6.2)) in any wedge of angular width $2\pi/N_b$.

If we change the monopulse requirement to m calls with no restrictions (case i), then, roughly, the capacity is reduced by m:

$$N = \frac{T_d - N_b \cdot b}{ma} \tag{6.3}$$

which is equivalent to increasing the "time-per-target coefficient", a, by the factor m. Finally, with case iii monopulse, we must schedule N/mN_b targets in T_d/m^2N_b seconds, hence

$$\frac{T_d}{m^2 N_b} = a \frac{N}{mN_b} + b \quad ,$$

and the total capacity is

$$N = \frac{T_d - m^2 N_b \cdot b}{ma} \tag{6.4}$$

Again, the real limitation is expressed by the bound of N/mN_b targets in each wedge $2\pi/mN_b$ in width. The succession of formulas (6.1) through (6.4) shows clearly the cost in efficiency due to repeated roll-calling, and suggests that a compromise might be made, in algorithm choice, in favor of a small value of b at the cost of a larger value of a. Formula (5.23) implies that the DRO algorithm might satisfy the requirements for a rotator rather well, although the derivation of this relation made use of a stringent assumption on target distribution. By the time a total target list is divided into mN_b parts, each list could be quite modest in length, hence special algorithms, perhaps of the DRO type, should be devised for this case, with

emphasis on attaining a small value of the parameter b. An interesting possibility is to use the general algorithm for scheduling an arbitrary target list (described at the beginning of this chapter), combined with a preliminary decreasing-range ordering of the targets.

2) Variable Data Rate and/or Rapid Access Interrogation Cycling

It is always possible to reduce the data rate with a rotator to submultiples of the rate of rotation, but the rate can only be increased by increasing the number of calls per scan, and this leads to a burst of interrogations which may not be as useful as an increase in periodic rate. In the same way, communications access can only be increased during the dwell time, but this could quite likely meet the basic need of a readdress capability. If m = 1, the standard rotator scheme can be modified by dividing each of the N_b target sublists into K separate parts, each containing the targets in a small wedge Θ/K in width. The related time intervals, T_d/N_b in length, are also divided into K parts, each devoted to one of the K parts of the sublist. The method is precisely analogous to that discussed for the array, but applied to a sublist instead of the entire list. Efficiency is reduced, as usual, by magnification of the constant term in the wasted time per roll calls. In normal operation, as the antenna rotates, each target is addressed as it is illuminated by the first wedge of the beam, Θ/K in width. If necessary, targets can be readdressed (at a cost in total capacity) during any or all of the remaining (K - 1) wedges.

If the monopulse technique employed is characterized by a value of m greater then unity and with case iii restrictions, it will be very awkward to provide the possibility of additional access.

3) Schedule Interrupt

Any of the techniques for schedule interrupt discussed for the array can be applied to the scheduling of sublists for a rotator. It might be useful to combine an interrupt capability with rapid-access, with a small value of K, say K = 2 (and m = 1!). Targets are normally addressed in the

leading half of the beam, but can be added to the list for the second half to increase access. Interrupts can be handled during the first half-beam, but can be postponed to the second half for targets requiring interrupt calling which occur near the end of the first half-beam. Again, the problem is much more difficult if m > 1.

4) ATCRBS/DABS Interlace

If a rotator must time-share ATCRBS and DABS interrogations, the limitations of angular access will probably force the design towards a rapid interlace between the two modes. The typical case is the one where DABS scheduling is performed during dead time intervals between ATCRBS interrogations, assuming that the ATCRBS interrogation rate is less than the "nominal rate", discussed in Chapter 4. The result is simply that the time intervals, T/N_b or T/mN_b, during which targets are accessible for DABS interrogation, are shared between ATCRBS and DABS (T now stands for the rotation period of the antenna). If m = 1, the time interval T/N_b during which a particular DABS target sublist must be called, will contain N_r ATCRBS interrogations, and an equal number of dead time blocks for DABS. If m > 1 each sublist will be fitted into N_r/m^2 blocks, a situation which may begin to constrain the scheduling somewhat. As with the array, block algorithms will be preferable, and fixed-length blocks will be easiest to adapt to the intermittently available blocks of time.

7. THE EFFECTS OF TARGET MOTION ON THE INTERROGATION SCHEDULING PROBLEM

With any algorithm which could be used for DABS interrogation scheduling, the final schedule produced for a given scan is directly dependent upon the ranges of the targets to be handled. In addition, for the rotator, the schedule depends upon the azimuths of the targets. Throughout this entire discussion we have treated target range and azimuth as fixed quantities when in fact they will change, due to target motion, during the course of each scan for which an interrogation schedule is being determined. Target motion affects interrogation scheduling in two related ways: 1) the scheduler must

cope with the fact that target motion will occur, and 2) the scheduler must operate with imperfect knowledge of present and future target position. These two aspects of the problem will be discussed briefly in this chapter.

It is conceivable that all tracking will be done at a central computation or control facility, and that track information will be sent back to the interrogators, along with the target list for each scan, in order to permit the interrogation scheduler to predict position. Although this configuration would allow sophisticated tracking, based on multisensor data, it seems much more likely that adequate tracking can be performed at each interrogator site, using only data obtained by that site.

The interrogator will provide position measurements with independent errors in range and azimuth, hence it will be logical to perform (ρ, θ) tracking and prediction. Since it is only necessary to predict ahead for the duration of one scan, it should be sufficient to assume unaccelerated target motion for this time. If, moreover, the distance traveled during a scan is small compared to target range, then the (ρ, θ)-prediction equations simplify to linear independent prediction of ρ and θ;

$$\rho(\tau) \approx \rho + \dot{\rho}\,\tau$$
$$\theta(\tau) \approx \theta + \dot{\theta}\,\tau \qquad (7.1)$$

In equations (7.1), ρ and θ refer to position at the start of a scan, and $\dot{\rho}$, $\dot{\theta}$ refer to range and azimuth rate at the same time. These four parameters are derived from the track, based on the previous several scans. By way of example, we note that an aircraft traveling at 600 knots travels one nautical mile in 6 seconds. This is a reasonable upper bound on speed, and 6 seconds is probably a conservative (i.e. long) estimate of data update interval for DABS, hence equations (7.1) will be valid for targets at ranges large compared to one mile, i.e. most targets. The errors in the four track parameters will cause errors in the predictions given by (7.1), and those will be most serious at the end of the scan interval, T.

a. The Effect of Changing and Imperfectly Known Target Range

In the ring algorithms defined in Chapter 5, the time of interrogation of a particular target is usually not known until the entire target range distribution has been considered, a ring assignment has been made, and the "secondary scheduling" of rings has been accomplished. One can then determine what the target ranges will be at the times they are actually scheduled to be called and make necessary adjustments in the schedule. One iteration should be enough, assuming the first attempt at scheduling was based on the target ranges predicted for the mid-point of the scan interval and recognizing that the "message time", τ, will contain buffer intervals before and after the actual message to ease this and other problems. The necessary schedule adjustments will probably be simple interchanges of target pairs whose ranges cross during the scan interval.

In the IRO algorithm, the general scheduler for an arbitrary list, and in some versions of the DRO principle, the scheduling proceeds sequentially through the scan interval being processed. In this case, a simple range update of those targets not yet scheduled can be made periodically during the scheduling process. A range rate of 600 knots implies a propagation delay change of about two microseconds per second of elapsed time, hence an update every few hundred milliseconds of scan time would be adequate to keep this error under control.

With any algorithm, the effect of range prediction error can only be handled by the use of buffer intervals surrounding the actual messages, which introduces a direct trade-off between channel efficiency and tracking accuracy, which is discussed below.

b. The Effect of Changing and Imperfectly Known Target Azimuth

With a phased array, the schedule timing is independent of azimuth, hence it is only necessary to compute azimuth, using (7.1), for the time at which each interrogation has been scheduled. The error in

predicted azimuth will have to be small compared to the pointing demands of the monopulse system, whatever they may be.

In the case of the rotator, it will be recalled that scheduling required a preliminary sorting of targets into azimuth wedges whose width depends on the character of the monopulse scheme employed. Azimuth change is accounted for by simpling computing the time at which aircraft azimuth and antenna boresight direction will coincide,again using (7. 1), and basing the wedge assignment of each target on the azimuth at the time of this coincidence. If the azimuth prediction error is not very small, compared to the width of an azimuth wedge, then a smaller wedge can be used to assure meeting the pointing requirements of the monopulse system.

c. Tracking Requirements for Interrogation Scheduling

It is clear from the preceding discusssion that the accuracy in predicted range must be small compared to the message length and the corresponding azimuth accuracy must be small compared to the angular wedge within which the antenna must be pointed (or the interrogation timed, for the rotator). The message length will likely be tens of microseconds and the pointing wedges will be on the order of one or two degrees in width, hence the tracking accuracy requirements are not great. For definiteness, we may assume that a range prediction error corresponding to a propagation delay uncertainty of one or two microseconds will be acceptable, and an azimuth prediction accuracy of about one-third to one-half degree will suffice.

Suppose the range error for a single measurement (i. e. one scan) has a standard deviation given by σ_R, while σ_θ is the standard deviation of a single azimuth measurement. The predicted position will be worst a full scan in advance, and we denote these worst-case range and azimuth prediction errors by σ'_R and σ'_θ. It is convenient to express these errors in predicted position in terms of the basic system measurement accuracy

by means of the equations:

$$\sigma'_R = K_R \sigma_R$$

$$\sigma'_R = K_\theta \sigma_\theta \qquad (7.2)$$

The constants, K_R and K_θ, depend upon the nature of the tracking algorithm and the duration of the tracking window. The DABS system is expected to provide measurement accuracies on the order of 150 feet in range, and 0.2° in azimuth [2]. A range error of 150 feet corresponds to a propagation delay error of 0.3 microseconds, hence K - values as large as two or three in equations (7.2) would appear to suffice.

Values of the K - constants of this order can be obtained by relatively short tracking windows, covering only a few scans. For short times such as these, the linear approximations (7.1) can be applied for most targets, although close-in targets may require special treatment. For the purpose of a rough estimate of required tracking performance, let us assume the validity of (7.1) over the track smoothing and prediction intervals, so that range and azimuth are tracked separately, each with a model of constant-velocity motion. A simple least-squares tracker is optimum in this case, and for this tracker, smoothing on the data from N scans, equations (7.2) are valid with

$$K^2_R = K^2_\theta = \frac{2\,(2N+1)}{N\,(N-1)} = K^2 \qquad (7.3)$$

For small numbers of scans in the tracking window, we obtain the following results:

N	K
2	2.24
3	1.53
4	1.23
5	1.15

Therefore, a very few scans should be adequate for tracking, and this has the further advantage of keeping small the prediction error bias due to non-linearities in the motion (due either to the breakdown of (7.1) or actual aircraft maneuver).

References

[1] Report of the DOT Air Traffic Control Advisory Committee, December 1969.

[2] Technical Development Plan for a Discrete Address Beacon System, DOT, October 1971.